aLIGO SUS Operations Manual

Mark Barton and the SUS Team

LIGO-E1200633-v2, 05/27/14

This document is a snapshot of the wiki at https://awiki.ligo-wa.caltech.edu/aLIGO/Suspensions/OpsManual .

See there for the latest information.

Table of Contents

aLIGO SUS Operations Manual

Introduction

This wiki page is the working version of the aLIGO SUS Operation Manual. DCC entry E1200633 is reserved for archiving it periodically. All pages in the list at /ArchiveList will be archived. The system being documented is taken to be all SUS-group suspensions (QUAD, BSFM, HLTS, HSTS and OMCS) as installed in vacuum and interacted with primarily via CDS hardware and software. Non-SUS suspensions (HAUX, HTTS, OFIS, TMTS) are addressed as time, enthusiasm and help from the responsible groups permit. Anything not flagged as site-specific applies to both sites equally.

Before editing this page, see the recommendations and hints at /StyleGuide.

Table of Contents

Sections mandated by M1200366-v1:

Description and Purpose

The SUS subsystem comprises 5 major types of suspension (QUAD, BSFM, HLTS, HSTS, OFMC) for various optical payloads. Each suspension provides addition vibration isolation (relative to that provided by the SEI subsystem), and allows for sensing of the position of the optics and actuation on them.

The SUS subsystem does *not* include 4 other types of suspension (Tip-Tilt, HAUX, TransMon and OFI Suspension), which are the scope of other groups. However the overall purpose (isolation, sensing, actuation) is the same and much of the infrastructure was designed by the same people, so they are covered here as time permits.

SUS suspensions

aLIGO, SUS DCC Document Tree, E1200482

A general overview of the SUS suspensions can be found at G1100866 Overview of Advanced LIGO Suspensions. More background information on the five major types is at the following subpages:

- QUAD - Really a combination of two quadruple pendulums side by side, the main chain supporting an ITM or ETM, and the reaction chain both supporting a TCP or ERM and doubling as a quiet platform for applying actuation to the main chain.
- BSFM - A triple pendulum supporting the beamsplitter, or in the original H2 design, a folding mirror.
- HLTS - HAM Large Triple: a triple pendulum supporting a PR3 or SR3, the largest of the optics used in HAM chambers.
- HSTS - HAM Small Triple: a triple pendulum supporting optics such as MC1, MC2, MC3, smaller optics used in HAM chambers.
- OMCS - A double pendulum supporting the output mode cleaner bench.

For information on individual suspensions, see the Master List Of Suspensions By Chamber, derived from T1100073 List of Suspensions in Chambers.

Non-SUS suspensions (don't hassle SUS about these!)

These suspensions are the responsibility of other groups but have to varying extents been incorporated as honorary SUS.

- HAUX - HAM Auxiliary suspension. A single pendulum similar to the iLIGO SOS but with blade springs for extra vertical isolation, supporting IO optics IM1 through IM4. Responsibility of IOO.
- HTTS - Tip-tilt. A single pendulum with blade springs similar to HAUX but with more powerful BOSEM actuators for high pitch and yaw actuation bandwidth. Supports IO optics RM1, RM2, OM1, OM2 and OM3. Responsibility of ISC.
- TMTS - TransMon suspension. A double pendulum somewhat similar to OMCS but with two rather than four blade springs on the top mass, supporting the Transmission Monitor behind the ETM/ERM QUAD. Responsibility of ISC.
- OFIS - Output Faraday Isolator Suspension. This is a totally passive suspension with no electronic damping or CDS interface. Responsibility of AOS.

Related Systems

- Optical Levers Signals from these are used by or plumbed through SUS front-end models.

Subsystems

- OSEMs are a combination of shadow sensor and voice coil, and are used extensively across the suspensions for global and local control. There are two types: the AOSEM is a refined iLIGO OSEM and the BOSEM is a larger version accommodating a much larger magnet.
- Electrostatic Drive (ESD) is used on the bottom stage of the QUAD as feeble but very low noise actuation.
- Electronics Suspensions are interacted with through electronics supplied by CDS.
- Watchdogs are modules inserted at three different levels in the software (at the sensor-actuator group level, the user model level and the IOP model level) that shut down outputs in the event of inputs or outputs exceeding defined thresholds. (A hardware watchdog is planned that will absorb some of this functionality.) Since the SEI system is the primary potential cause of damage to SUS, the SUS watchdogs also take inputs from the SEI watchdogs.

Context within aLIGO

aLIGO Seismic Isolation and Suspension Cartoon, G1200071

SUS physically supports optical payloads belonging to COS (ITM, ETM, BS, FM), IOO (MC1, MC2, MC3, PRM, PR2, PR3, SRM, SR2, SR3), TCS (the TCP) and ISC (the output modecleaner bench). Honorary SUS suspensions support payloads for IOO (IM1-4) and ISC (TMS, RM1-2, OM1-3).

SUS is physically supported by vibration isolation platforms provided by SEI (BSC-ISI, HAM-ISI, HEPI).

SUS is electrically connected to the outside world by cables provided by CDS to designs negotiated with SUS and SEI. The SUS group is responsible for routing the cable segments running up the suspension structure, SEI is responsible for routing the segments that weave through the ISI system to the vacuum feedthrough, and CDS is responsible for the remainder.

SUS is electrically integrated into the CDS computing system via cables, satellite amps, coil drivers, AA/AI/whitening/dewhitening boards, ADC/DAC boards, and computers supplied by Birmingham and CDS and maintained by CDS.

SUS is integrated into the CDS computing system via "models" (in Matlab/Simulink) compiled by the CDS-supplied RTCDS software package to run on CDS front-end computers and process high-bandwidth data to/from the sensors and actuators. A system of slow controls connecting different models and the user interfaces for this are constructed with EPICS/MEDM.

The SUS control system implements local control for each suspension (taking as input just the OSEM sensor signals giving the position of the suspended masses relative to the suspension structure and attempting to minimize those motions without regard to the state of other optics). It also channels and processes various signals that will be used to construct a global control system, including actuation requests from ISC and QPD signals from optical levers impinging on the optics.

Return to Contents

aLIGO SUS Operations Manual - Nomenclature

Back to Operation Manual main page

Groups	
SUS	SUSpensions group
CDS	Control and Data Systems group
SEI	SEIsmic isolation group
COS	Core Optics Subsystem
AOS	Auxiliary Optics Subsystem
IOO	Input/Output Optics
ISC	Interferometer Sensing and Control
LSC	Length Sensing and Control
ASC	Angular Sensing and Control
Suspension Types	
QUAD	Quadruple suspension
BSFM	Beam-Splitter and/or Folding Mirror suspension
HLTS	HAM Large Triple Suspension (for PR3 and SR3 optics)
HSTS	HAM Small Triple Suspension (for MC1, MC2, MC3, PR2, PRM, SR2 and SRM optics)
OMCS	Output Mode-Cleaner Suspension
HAUX	HAM Auxiliary Suspension
HTTS	HAM Tip-Tilt Suspension
TMTS	TransMon Telescope Suspension
OFIS	Output Faraday Isolator Suspension
Suspension Components/Concepts	
OSEM	Optical Sensor and Electro-Magnetic actuator
BOSEM	"Birmingham"-style large OSEM
AOSEM	"Advanced"-style small OSEM
SAG	Sensor-Actuator Group
QUAD Mass Names	
Top mass	Top mass (no abbreviation) of the main or reaction chains in a quad
UIM	Upper Intermediate Mass - second top mass of the main or reaction chains in a quad
PUM	PenUltimate Mass - second bottom mass of the main chain in a quad
PRM	Penultimate Reaction Mass - second bottom mass of the reaction chain in a quad
ITM	Input Test Mass - one type of bottom mass of the main chain in a quad
ETM	End Test Mass - second type of bottom mass of the main chain in a quad
TCP or CP	Thermal Compensation Plate - one type of bottom mass of the reaction chain in a quad, paired with ITM
ERM	End Reaction Mass - second type of bottom mass of the reaction chain in a quad, paired with ETM
Triple Suspension Mass Names	
UM	Upper Mass of a BSFM, HLTS or HSTS triple
IM	Intermediate Mass of a BSFM, HLTS or HSTS triple
MC1, MC2, MC3	Mode Cleaner mirrors 1, 2 and 3, payload for HSTS

Software	
MEDM	Motif Editor and Display Manager
EPICS	Experimental Physics and Industrial Control System
BURT	BackUp and Restore Tool
DTT	Diagnostic Test Tool

aLIGO: Suspensions/OpsManual/Nomenclature (last edited 2013-09-13 13:17:43 by MarkBarton)

aLIGO SUS Operations Manual - Safety, General Warnings and Cautions

Back to Operation Manual main page

When the suspensions are installed in vacuum, as assumed for the primary scope of this manual, they present negligible danger to personnel, being totally contained by the vacuum system. Hazard analyses for installation are linked at the end for reference in case they are useful in maintenance/retrofit/repair scenarios.

The main potential sources of damage to SUS and interacting systems during operation are:

- Gross external disturbance as from earthquake or crane accident.
- Runaway control system on SUS, ISI or HEPI, due to hardware/software/user error.
- Damage to ESD system if left energized through pumpdown or vent.

The SUS structures have earthquake stops fitted at many points, so as to contain the optics in the event of a disturbance or wire/fibre failure. The nominal gap between each EQ stop tip and the optic is 0.75 mm, so an optic should never move more than a few mm within the structure. However the stops have been designed purely as emergency measures and should never contact the optic in normal operations. To prevent recurrence of iLIGO issues with electrostatic charging due to Viton/Fluorel-tipped stops, many of the stops are tipped with silica pieces in Viton/Fluorel mounts. Temporary Teflon shields are used to prevent them touching the optic during installation. If worst comes to worst and the optic is pushed into the stops forcibly or repeatedly there is some risk of damaging the coating and/or generating glass dust.

Ideally, the fibres in the lower stage of the QUAD will be be well-protected by the earthquake stops and their own compliance (they are about three times more compliant than steel of the same breaking strength, and stretch by 6 mm under normal load, so the additional stretch before the earthquake stops are contacted should be less than a 15% overload). However the fibres are extremely sensitive to surface damage and there has already been an incident where the fibres on ITMy broke after about 1 hour of pounding from a runaway HEPI control system, so there may be vulnerabilities to be discovered.

The nominal clearance of the stops of 0.75 mm was chosen to protect magnets, OSEM flags and other items stuck to the optic from contacting OSEMs and other items on the structure. However some of the gaps are very difficult to set due to poor visibility and so some stops may not be close enough to prevent all damage. Also, some degree of twist is possible, so laser beams reflecting off optics may be diverted to sensitive items such as suspension wires or electrical cables.

For all these reasons it is important to be extremely careful about avoidable gross disturbances and runaway controls. There is a system of software watchdogs and a planned hardware watchdog to protect against runaway controls, and it is vital to make sure the software matches the installed hardware and that it is never disabled or defeated without appropriate high-level approvals.

The ESD subsystem has not yet been fully installed but will use high voltages (up to 500 V). It can be energized in air at atmospheric pressure or high vacuum, but there is an intermediate range of pressures (per Paschen's law) where the conductivity of air is high and the system will be destroyed if not powered off. A hardware lockout is being prepared to make this impossible but the underlying danger should be kept in mind.

References

DCC Search for SUS safety docs

BSC

- E1000030 QUAD Suspension Metal Assembly Hazard Analysis
- E1100814 Cartridge Assembly Hazard Analysis
- E1000489 Hazard Analysis for Silica Fiber Pulling and Welding In the LHO Fiber Lab and In the LHO and LLO LVEAs and VEAs
- E1000558 aLIGO SUS Quad Suspension Installation/Initial Alignment Hazard Analysis
- M1000334 LHO_CO2 Silica Fiber Welding Machine Operating in the LVEAs and END Station, Standard Operating Procedure
- E0900163 Beamsplitter/Folding Mirror Suspension Assembly Hazard Analysis
- E1200925 BSC Cartridge Installation (Universal) Hazard Analysis, aLIGO

HAM Triple

- E1000043 HLTS Assembly and Installation Hazard Analysis
- E0900332 HSTS Assembly and Installation Hazard Analysis

Other SUS suspensions

- E0900042 AdvLIGO Output Mode Cleaner Hazard Analysis
- E1300537 Addendum to E0900042 aLIGO OMC Hazard Analysis

Non-SUS suspensions

- T1000311 aLIGO TransMonSUS Assembly & Installation Hazards Analysis
- E1101232 HAM Auxiliary Suspension Hazard Analysis
- (HTTS Hazard Analysis?)
- E1300283 OFI Assembly & Installation Hazard Analysis

Related

- M1100297 SOP – 5 Watt 808nm Autocollimator Lasers Operating in the LVEA under the Test Stand
- E1100070 aLIGO COS/IAS Laser Autocollimator Hazard Analysis
- T080231 ERGO Arm Hazard Analysis

aLIGO: Suspensions/OpsManual/Safety (last edited 2013-10-08 13:08:18 by MarkBarton)

aLIGO SUS Operations Manual - Quick Start Guide

Back to Operation Manual main page

- M1200366-v1 mandates a quick start guide but the SUS system is of such complexity that such a document is not feasible. Instead, see Suspensions/OpsManual/OperatingInstructions.

References

- E1300860, Janeen's Quick Start Guide.

aLIGO: Suspensions/OpsManual/QuickStart (last edited 2014-04-25 09:46:33 by MarkBarton)

aLIGO Operations Manual - Setup

Back to Operation Manual main page

By the time the suspension are installed in vacuum, per the assumed scope of this manual, pretty much all setup will have been done. However references to installation procedures and the like are given below.

The part of the setup that is most likely to be relevant to operators and scimons concerns the OSEMs. The OSEM sensors are set up by measuring their output in the "open light" condition (flag fully retracted), calculating and setting gains and offsets that map each OSEM's range to a standard ±15K counts, and then mechanically adjusting each OSEM to the midpoint of its range. The magnets associated with the OSEMs need to be set in a certain pattern of N and S poles. See the OSEMs page.

References

- E1100290 aLIGO SUS QUAD Assembly and Installation Documentation
- E1100599 aLIGO SUS BSFM Assembly and Installation Documentation
- E1100472 aLIGO SUS HLTS Assembly and Installation Documentation
- E1100471 aLIGO SUS HSTS Assembly and Installation Documentation
- E1300429 aLIGO SUS OMCS Assembly and Installation Documentation
- T1100304 TMS ASSEMBLY and ALIGNMENT DOCUMENTATION
- T1300029 IO HAUX Material and Assembly Documents
- E1100440 aLIGO HAM Tip-Tilt Suspension Assembly Procedure
- E1201074 Output Faraday Isolator Assembly and Alignment Procedure
- E1300056 Output Faraday Isolator Installation and Final Alignment Procedure

aLIGO Operations Manual - Checkout/Testing

Back to Operation Manual main page

The overall testing procedure for SUS is documented in G1100693 (QUAD) and G1200070 (BSFM/HLTS/HSTS).

An index of alogs with key measurements used for installation acceptance is maintained at E1400187.

Key procedures are documented on the following subpages:

- Suspensions/OpsManual/TFs - Transfer functions
- Suspensions/OpsManual/Spectra - Spectra of sensors without actuation (besides damping)
- TransferFunctionColoringBook - Rogues' gallery of bad TFs with diagnoses
- Suspensions/OpsManual/B&K - B&K Hammer Tests on structures
- Suspensions/OpsManual/Diagonalization - Coil Balancing/Diagonalization

aLIGO: Suspensions/OpsManual/Testing (last edited 2014-05-23 12:19:10 by MarkBarton)

aLIGO SUS Operations Manual - Taking and Plotting SUS Transfer Functions

Back to Operation Manual main page

There are two ways to take data for transfer functions: with Diagnostic Test Tools (DTT) templates, and with a Matlab script. DTT is quicker overall and gives partial results immediately. However it requires a lot of operator time. The Matlab script is very quick to set up but requires about five times as long to run.

Data taking

1. Choose either DTT or Matlab, follow the instructions on the corresponding page (which will involve taking data and doing Stage 1 plots), and then return here for the Stage 2 plotting and cleanup steps.

DTT	Matlab

Plotting - Stage 2

2. If Matlab is not already open from having used it for data taking, open it from the terminal and navigate to the `^/trunk/${susType}/Common/MatlabTools` directory, e.g, for QUAD,

```
$ matlab&
>> cd /ligo.svncommon/SusSVN/sus/trunk/QUAD/Common/MatlabTools
```

3. In the Matlab command window, do *another* svn update. (Unlike the Stage 1 script, where edits are mostly disposable, the Stage 2 script accumulates important information from both sites, and it's very painful to recover if you edit a stale version because the other site has committed a new one while you were working on Stage 1.)
4. Still in the `MatlabTools` directory for the suspension type, open the `plotall`...`tfs.m` script for the suspension, e.g., `plotallquad_dtttfs.m` (QUAD), `plotallbsfm_tfs.m` (BSFM), `plotallhlts_tfs.m` (HLTS), etc. Note that the name of the QUAD one is a misnomer - it's not specific to DTT.
5. Identify the sections near the top that look like this (from `plotallquad_dtttfs.m`):

```
printFigs = false;
```

and

```
useMeasts.M0 = [85 95 97];
useMeasts.R0 = [86 96 98];

figFileTag = '130125_H1SUSITMY_Phase3a';

            %                                 M0          R0           M0            R0
Damping  Accept
            % ifo,    quadID,    buildNum,  susType,    susType,     measDate,    measDate,
measType      measNum    quadID         chainType  measPhase  isiState  loopState  level
aLOG #     Notes
measList = { 'X2', 'QUAD11',  'BUILD01',   'wire',       'thincp','2011-07-31','2011-07-
31',''     ;... % (1,2)   | QUAD11, BUILD01 |  (ITM)  |    1B     |  S.S.   |   OFF   |
PASS |         | (No Lacing Cables)
             'L1',   'ITMX',          '',   'wire',       'thincp','2012-10-31','2012-10-
31','DTTTF';... % (3,4)   |    L1SUSITMX    |    |    |    2A     |  S.S.   |   OFF   |
FAIL | LLO5142 | M0 good, R0 bad, weak R0-F2 BOSEM?

…
```

6. Edit the assignment to `printFigs` to `false` for now.
7. Edit the assignment for `figFileTag` to include the date, the suspension and the phase of testing for the new measurement.
8. Copy one of the lines of the cell array being assigned to `measData`, paste it into a logical spot in the array, and edit it to reflect the newest measurement. The columns on the left are active code; the columns on the right are Matlab comments but should still be updated. The QUAD case illustrated above is more complicated than the others because it allows for both an M0 set and an R0 set but most of the columns for it and the simpler BSFM/HLTS/etc cases should be obvious. `measType` is `DTTTF` for DTT TFs (i.e., according to these instructions) or `MATTF` for Matlab

TFs.

9. `measNum` is one of the columns in the commented-out half but needs to equal the index of each result in the array. Edit the `measNum` of the added line *and all subsequent lines* so that the entire column forms a smoothly incrementing sequence. (This is unbelievably inelegant and tedious, but that's the way it is.)
10. When the `measNum` column has been updated, identify approximately 2-4 measurements including the current one that would be of interest to compare, and copy their `measNum` values to the assignment to `useMeasts`. For QUAD you need to do `useMeasts.M0` and `useMeasts.R0` separately; for other suspension types there will be just `useMeasts.M1`.
11. Click the green run button and check the command window for log messages that all `.mat` files from Stage 1 have been successfully located and read. If there have ever been multiple measurements on a single day, you may be asked to choose one or other of them.
12. As with Stage 1, when everything is set up correctly, a series of plots will appear. When success is achieved, change the assignment to `printFigs` to `true` and rerun the script. This will write PDF versions to `^/trunk/${susType}/Common/Data`, e.g., `^/trunk/QUAD/Common/Data`, i.e., a *different* `Data` directory than above, common to the whole suspension type.
13. Do a comment to the alog from above. If you didn't do an earlier comment announcing the end of data taking, mention that it has finished and give the data file names now. Upload the PDF from the previous step as an attachment (the `figFileTag` you specified will be in the filename but unfortunately not at the beginning so you may have to browse through most of the `^/trunk/${susType}/Common/Data` directory to find it). Mention whether the plots are good and the suspension passes, or whether there are issues, and if so, what.

Typical set of Stage 2 plots, for H1 ITMy:
attachment:allquads_130124_H1SUSITMY_Phase3a_ALLM0_TFs.pdf

Cleanup

Cleanup involves submitting all the new stuff to the SVN. The following example assumes QUAD (H1:ITMy) but other suspension types will be similar. The example assumes the cleanup is done within the Matlab command window but can just as easily be done from a terminal window - just omit the `!`'s.

1. Commit the modified scripts:

```
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/Common/MatlabTools  # (or navigate there with the Matlab file browser)
>> !svn commit -m "New data for H1 ITMy"
```

2. Commit the comparison plots:

```
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/Common/Data  # (or navigate there with the Matlab file browser)
>> !svn add * # harmless warnings will be generated for previously-added files
>> !svn commit -m "New data for H1 ITMy"
```

3. Commit the raw data and exported text files:

```
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/H1/ITMY/SAGM0/Data # (or navigate there with the Matlab file browser)
>> !svn add * # harmless warnings will be generated for previously-added files
>> !svn commit -m "New data for H1 ITMy"
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/H1/ITMY/SAGR0/Data  # (or navigate there with the Matlab file browser)
>> !svn add * # harmless warnings will be generated for previously-added files
>> !svn commit -m "New data for H1 ITMy"
```

4. Commit the single measurement plots.

```
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/H1/ITMY/SAGM0/Results # (or navigate there with the Matlab file browser)
>> !svn add * # harmless warnings will be generated for previously-added files
>> !svn commit -m "New data for H1 ITMy"
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/H1/ITMY/SAGR0/Results  # (or navigate there with the Matlab file browser)
>> !svn add * # harmless warnings will be generated for previously-added files
>> !svn commit -m "New data for H1 ITMy"
```

5. Check you haven't missed anything.

```
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/ # (or navigate there with the Matlab file browser)
```

```
>> !svn status
```

aLIGO: Suspensions/OpsManual/TFs (last edited 2014-05-27 09:09:43 by MarkBarton)

aLIGO SUS Operations Manual - Taking TF Data with Matlab

Back to Operations Manual main page.
Back to TFs main page.

You should be here if you want to take the data for your transfer functions using Matlab. If you wanted to do it with DTT, go to Suspensions/OpsManual/TFs/DTT.

Data Taking

1. Check with users of the suspension that it's available for testing, which will involve applying excitations and usually involve changing the damping state.
2. Do an alog announcing that you're commencing TFs. Make a note of the alog entry number.
3. Open the MEDM screen for the suspension.
    1. Check that the TEST FILTERS for M0 (QUAD) or M1 (all others) are all ON (green).
    2. For QUAD, check the RO TEST FILTERS and DAMP FILTERS as well (these are on a subscreen accessed by a button).
    3. Check that the Master Switch is ON.
    4. Check that the M0 (QUAD) or M1 (all others) watchdog is not tripped, and reset it if necessary.
    5. For QUAD, check that the R0 watchdog is not tripped, and reset it if necessary.
    6. Check that the USER DACKILL watchdog is not tripped, and reset it if necessary.
    7. Check that the IOP watchdog is not tripped, and reset it if necessary.
4. Open Matlab:

    ```
    $ matlab &
    ```

5. Navigate to `^trunk/Common/MatlabTools/SchroederPhaseTools`:

    ```
    > cd /ligo/svncommon/SusSVN/sus/trunk/Common/MatlabTools/SchroederPhaseTools
    ```

6. Open `LHO_Matlab_TFs.m` (at LHO) or `LHO_Matlab_TFs.m` (at LLO) in the Matlab editor.
7. Check that all sections not relevant to the optic you are working with are commented out.
8. Find an existing section that does what you need and uncomment it, or create a new one based on the following pattern for H1 ITMY.

    ```
    % ITMY
    switch_dampcomm('H1','ITMY','M0OFF','ON')
    Matlab_TFs('H1','ITMY','','','3A','M0','wire_rehang','TFSettingsLHOQUADPhase3a'); % H1 ITMY main chain, damping off
    switch_dampcomm('H1','ITMY','M0ON','NC')
    Matlab_TFs('H1','ITMY','','','3A','M0','wire_rehang','TFSettingsLHOQUADPhase3a'); % H1 ITMY main chain, damping on
    switch_dampcomm('H1','ITMY','R0OFF','NC')
    Matlab_TFs('H1','ITMY','','','3A','R0','thincp','TFSettingsLHOQUADPhase3a'); % H1 ITMY reaction chain, damping off
    switch_dampcomm('H1','ITMY','R0ON','NC')
    Matlab_TFs('H1','ITMY','','','3A','R0','thincp','TFSettingsLHOQUADPhase3a'); % H1 ITMY reaction chain, damping on
    switch_dampcomm('H1','ITMY','NC','OFF')
    ```

    The `switch_dampcomm()` function allows convenient switching of the damping (3rd argument) and the measurement status or commissioning switch (4th argument). The `Matlab_TFs()` function makes the measurement. Both functions have documentation in internal comments and will print more info if run without arguments.
9. When the script has been edited, click the green run button.
10. Wait about a minute and then check that excitations are being applied, as indicated by messages in the Matlab Command Window, and changing numbers emerging from the TEST FILTERS section on the MEDM screen and making their way past the IOP DACKILL block.
11. Settle down for a long wait (about 6 hours per call to Matlab_TFs()). Note that the estimated total time peris typically off by about 1.5 hours, so towards the end you may see estimated times to completion such as -70 minutes.
12. Check from time to time that nobody has decided to work on or near the suspension and that the watchdog is not tripped. The watchdog is reset automatically at the end of each frequency band for each actuation DOF but missed

tripped. The watchdog is reset automatically at the end of each frequency band for each actuation DOF but missed bands are not repeated and will show up as dropouts in the final data. If the watchdog trips consistently or without external cause, the amplitude values defined in the settings script (`'TFSettingsLHOHSTSPhase3a'` or the like are probably too high.)

13. If it's not convenient to proceed immediately to generating plots, add a comment to the alog entry from above announcing that the suspension is free and that plots are pending. Mention the location of the data files in the SVN, e.g., `^/trunk/QUAD/H1/ITMY/SAGM0/2012-12-10_1600_H1SUSITMY_M0_*_WhiteNoise.xml`.

Plotting - Stage 1

This stage generates a set of plots with data from just the most recent set of measurements. In Stage 2 later, a common script can generate comparison plots involving measurements that were taken with DTT and ones that were taking with Matlab.

14. Navigate to the `^/trunk/${susType}/Common/MatlabTools` directory for the suspension type, e.g., `^/trunk/QUAD/Common/MatlabTools` for QUAD.

```
>> cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/Common/MatlabTools
```

15. In the Matlab command window, do an svn update to make sure all the software is up to date (`!` is used to execute shell commands from Matlab):

```
> !svn update
```

16. In the Matlab editor, open the script for Stage 1 plotting: `plotquad_dtttfs.m` (QUAD), `plotBSFM_dtttfs.m` (BSFM), `plotHLTS_dtttfs.m` (HLTS), `plotHSTS_dtttfs.m` (HSTS) etc. Identify the section near the top that looks like the following:

```
printFigs = true;
saveData = true;

ifo             = 'H1';            % ['H1' 'H2' 'L1' 'M1' 'X1' 'X2']
quadID          = 'ITMY';          % Optic Name or "QUAD" if on a test stand
['QUAD','ETMX','ITMY']
quadNum         = '';              % Must be two digits, or empty string ['02','03','']
buildNum        = '';              % Must be two digits, or empty string ['01','02']
sagLevel        = 'M0';            % ['M0', 'R0', 'L1', 'L2', 'L3'] -- no damping anywhere
else!
buildType       = 'wirerehang';        % ['wire','fiber','erm','thincp','wirerehang']
meas.yyyymmdd   = '2013-01-25';    % e.g. 2012-04-23 for April 23rd 2012
meas.hhmm       = '1330';          % e.g. 1003 for 10:03AM
meas.sensCalib  = false;           % true or false -- are sensors calibrated?
meas.author     = 'M. Barton';      % Let us know who did all the work!
```

17. Edit the assignments to `printFigs` and `saveData` to `false` (or 0) to begin with. Edit the rest of the assignments to match the measurements just taken. `buildType` is the name of the model parameter set that will be used for model comparisons. `meas.sensCalib` records whether the filter in the OSEMINF block that converts from counts to μm had been enabled. (As of 1/28/13, LHO was running with this off but LLO had enabled it for some or all suspensions.)
18. Click the green run button and check the command window for log messages that all exported files have been successfully read. A failure at this point is most likely due to not having visited all three pages in the Export dialog so that the exported file doesn't have the full complement of columns, normally 25: frequency plus real and imaginary parts of 12 signals. If this is suspected, reopen the `.xml` file for the affected excitation channel and re-do the export.
19. When the script runs without errors, review the main plots, `L->L`, `P->P` etc and see if any have symptoms that suggest a problem with the measurement rather than the suspension, e.g., small magnitude and high noise level as from a tripped watchdog. If so, consider rerunning the measurement for that excitation channel and re-exporting the data.
20. When any measurement issues have been fixed, edit the assignments to `printFigs` and `saveData` to `true` and re-run the script. This generates the same plots as before but writes them to the `Results` directory, sibling to the `Data` directory, e.g., `^/trunk/QUAD/H1/ITMy/SAGM0/Results` for M0 of the H1 ITMy. A failure at this point probably means the `Results` directory does not exist, in which case it should be created.

Typical set of Stage 1 plots, for H1 ITMy:
📎📎attachment:2013-01-24_H1SUSITMY_M0_ALL_TFs.pdf

aLIGO SUS Operations Manual - Taking TF Data with DTT

Back to Operations Manual main page.
Back to TFs main page.

You should be here if you want to take the data for your transfer functions using DTT. If you wanted to do it with Matlab, go to Suspensions/OpsManual/TFs/Matlab.

Data Taking

1. Check with users of the suspension that it's available for testing, which will involve applying excitations and usually involve changing the damping state.
2. Do an alog announcing that you're commencing TFs. Make a note of the alog entry number.
3. Open the MEDM screen for the suspension.
   1. Turn the Measurement Status (QUAD/BSFM/HxTS) or Commissioning (HAUX) switch ON.
   2. Check that the TEST FILTERS for M0 (QUAD) or M1 (all others) are all ON (green).
   3. Check that DAMP FILTERS for M0 (QUAD) or M1 (all others) are all in the desired state (ON for damped, OFF for undamped).
   4. For QUAD, check the RO TEST FILTERS and DAMP FILTERS as well (these are on a subscreen accessed by a button).
   5. Check that the Master Switch is ON.
   6. Check that the M0 (QUAD) or M1 (all others) watchdog is not tripped, and reset it if necessary.
   7. For QUAD, check that the R0 watchdog is not tripped, and reset it if necessary.
   8. Check that the USER DACKILL watchdog is not tripped, and reset it if necessary.
   9. Check that the IOP watchdog is not tripped, and reset it if necessary.
4. Open DTT. For QUAD you will normally want to get TFs for both M0 and R0, in which case it is convenient to open two instances of DTT and work in parallel.

   ```
   $ diaggui &
   ```

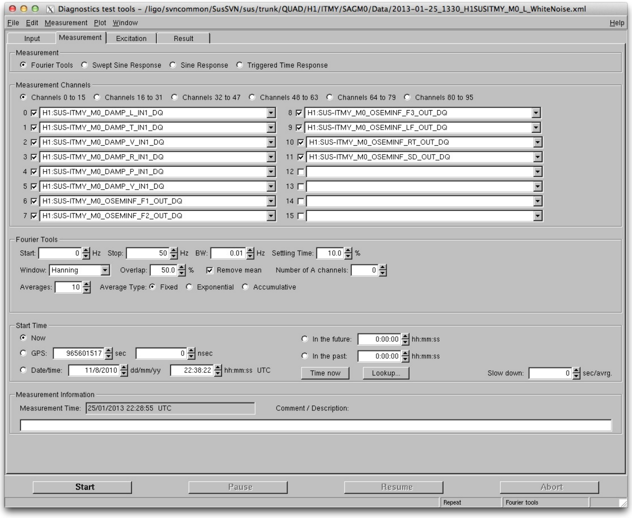
5. Choose Open from the File menu and navigate to the `Data` directory for the suspension and sensor-actuator group: `^/trunk/${susType}/${ifo}/${optic}/SAG${sag}/Data`, e.g., `^/trunk/QUAD/H1/ITMy/SAGM0/Data` for M0 of the H1 ITMy.
6. Identify the measurement template files, which will have names like `yyyy-mm-dd_hhmm_H1SUSITMY_M0_*_WhiteNoise.xml` for `*` = `L`,`P`,`R`,`T`,`V` and `Y`. (If there aren't any templates, create them by copying a set for another suspension of the same type and doing a search and replace on the IFO, suspension, and sensor-actuator group names with a text editor.)
7. For each of these files, in any convenient order, repeat the following steps. (If time is not pressing, then alphabetical order is easiest: `LPRTVY`. For a quick go/no-go decision when a mechanical issue is suspected, the most informative order is something like `PRVYLT`.)
   1. Open the template and immediately resave it with the `yyyy` etc replaced by the year, month, date and start time, e.g., `2012-12-10_1600_H1SUSITMY_M0_L_WhiteNoise.xml`. (The start time need only be approximate, e.g., `1200`, and should be the same for all files in a group.)
   2. Click Start.
   3. Check the MEDM screen to see that excitation is appearing (as changing numbers) at the output to the test filters and is making its way across the screen past the Master Switch and watchdogs all the way to the output.
   4. When the measurement is done, choose File->Save.
   5. Choose File->Export.
   6. In the resulting dialog, choose Transfer Function from the Data Type popup menu.
   7. In the Column Selection area, for the first 12 columns (extending over pages 0 to 4, 5 to 9 and 10 to 14) enable the checkbox and set the A popup to the excitation channel, `H1:SUS-ITMY_M0_TEST_L_EXC` or the like.
   8. In the B popups, select the `OSEMINF` channels (`H1:SUS-ITMY_OSEMINF_F1_OUT_DQ` or the like) followed by the `DAMP` channels (`H1:SUS-ITMY_DAMP_L_IN1_DQ` or the like). For QUAD AND BSFM, the `OSEMINF` channels should be in the order `F1`, `F2`, `F3`, `LF`, `RT` and `SD`, and for HxTS, `T1`, `T2`, `T3`, `LF`, `RT` and `SD`. The DAMP channel order should be `L`, `T`, `V`, `R`, `P` and `Y`. For OMCS it is TBD. For HAUX it is `UL`, `LL`, `UR`, `LR`, `L`, `P` and `Y`. (Unfortunately, DTT doesn't encode export settings in its files, so it's not possible to have all this set up in the templates. **If you reuse the same instance of DTT for all templates**

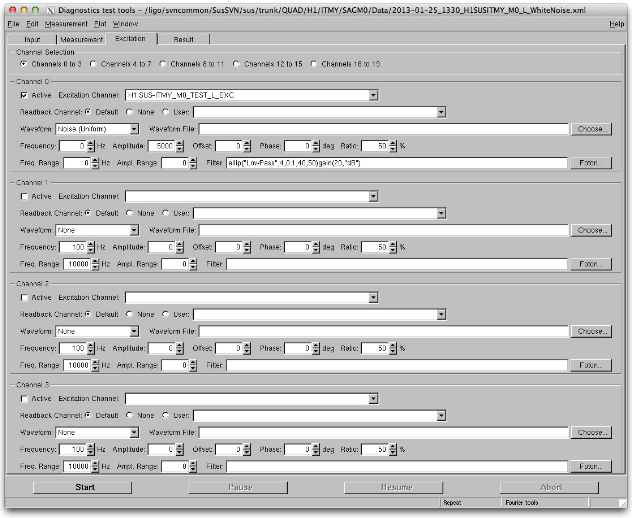
in a set, DTT will *usually* update the A channels appropriately and remember the B channels. However for this to happen it is crucial to visit all three pages (0 to 4, 5 to 9 and 10 to 14) in the Column Selection area. If you don't force the second and third pages to be displayed, those channels will not be exported and the plotting will fail.**) When everything is set up, press Export.

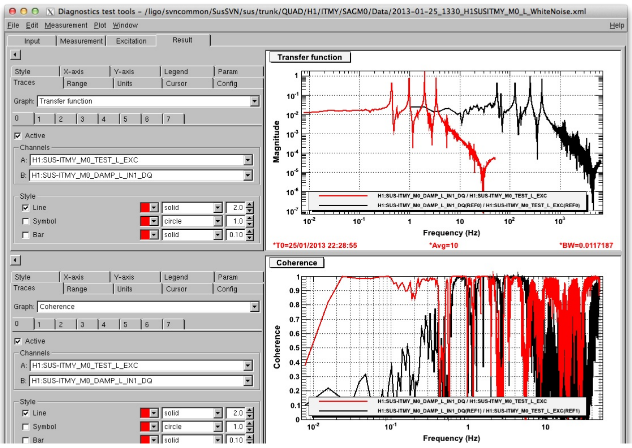
   9. Save the exported data in the same `Data` directory as the raw data, with the same name file name except for `_tf.txt` at the end instead of `.xml`. (For the first file in a set it's most convenient and least error-prone to use the "Files of type:" popup to show `.xml` files, then click on the data file to copy its name into the text field where it can be edited as above. For subsequent files it's easier to leave the "Files of type:" set to `.txt` then click on the first text file and edit the `_L_` part of the name to match the current excitation DOF.)

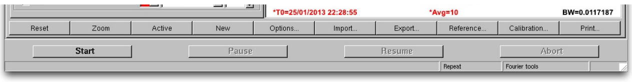
8. For the QUAD, repeat the above steps for the other chain (R0 or M0).
9. When all TFs have been taken and exported, set the MEASUREMENT STATUS or COMMISSIONING switches back to OFF, and restore the suspension to the working state, e.g., by reenabling damping.
10. If it's not convenient to proceed immediately to generating plots, add a comment to the alog entry from above announcing that the suspension is free and that plots are pending. Mention the location of the data files in the SVN, e.g., `^/trunk/QUAD/H1/ITMY/SAGM0/2012-12-10_1600_H1SUSITMY_M0_*_WhiteNoise.xml`.

Typical Measurement, Excitation and Result panes in DTT:

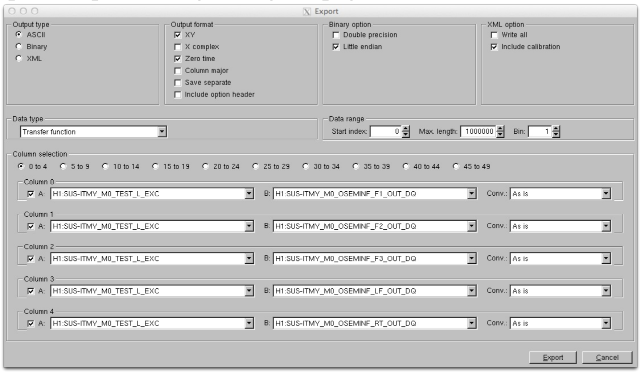


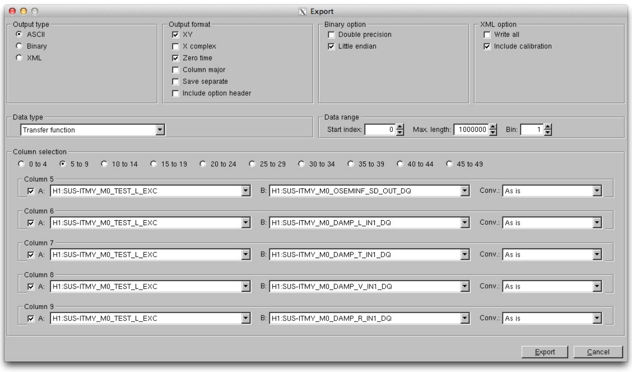


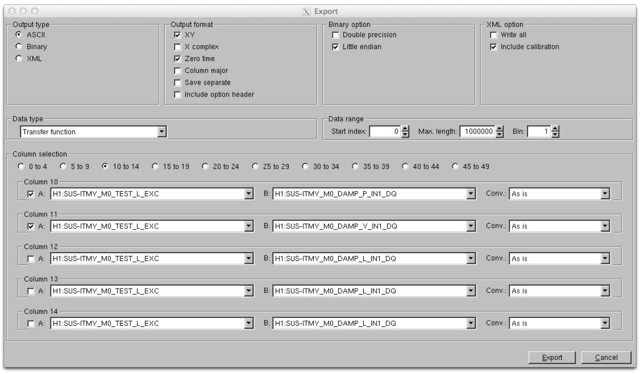




Typical Export Dialog settings (3 pages for columns 1-4, 5-9 ad 10-14)







Plotting - Stage 1

This stage generates a set of plots with data from just the most recent set of measurements. The plotting script for DTT data reexports the data in a more convenient form, so that in Stage 2 later, a common script can generate comparison plots involving measurements that were taken with DTT and one that were taking with Matlab.

11. Open a copy of Matlab:

```
$ matlab &
```

12. Navigate to the `^/trunk/${susType}/Common/MatlabTools` directory for the suspension type, e.g., `^/trunk/QUAD/Common/MatlabTools` for QUAD.
13. In the Matlab command window, do an svn update to make sure all the software is up to date (`!` is used to execute shell commands from Matlab):

```
> !svn update
```

14. In the Matlab editor, open the script for Stage 1 plotting: `plotquad_dtttfs.m` (QUAD), `plotBSFM_dtttfs.m` (BSFM), `plotHLTS_dtttfs.m` (HLTS), `plotHSTS_dtttfs.m` (HSTS) etc. Identify the section near the top that looks like the following:

```
printFigs = true;
saveData = true;

ifo             = 'H1';           % ['H1' 'H2' 'L1' 'M1' 'X1' 'X2']
quadID          = 'ITMY';         % Optic Name or "QUAD" if on a test stand
```

```
['QUAD','ETMX','ITMY']
quadNum         = '';             % Must be two digits, or empty string ['02','03','']
buildNum        = '';             % Must be two digits, or empty string ['01','02']
sagLevel        = 'M0';           % ['M0', 'R0', 'L1', 'L2', 'L3'] -- no damping anywhere
else!
buildType       = 'wirerehang';       % ['wire','fiber','erm','thincp','wirerehang']
meas.yyyymmdd   = '2013-01-25';   % e.g. 2012-04-23 for April 23rd 2012
meas.hhmm       = '1330';         % e.g. 1003 for 10:03AM
meas.sensCalib  = false;          % true or false -- are sensors calibrated?
meas.author     = 'M. Barton';     % Let us know who did all the work!
```

15. Edit the assignments to `printFigs` and `saveData` to `false` (or 0) to begin with. Edit the rest of the assignments to match the measurements just taken. `buildType` is the name of the model parameter set that will be used for model comparisons. `meas.sensCalib` records whether the filter in the OSEMINF block that converts from counts to µm had been enabled. (As of 1/28/13, LHO was running with this off but LLO had enabled it for some or all suspensions.)
16. Click the green run button and check the command window for log messages that all exported files have been successfully read. A failure at this point is most likely due to not having visited all three pages in the Export dialog so that the exported file doesn't have the full complement of columns, normally 25: frequency plus real and imaginary parts of 12 signals. If this is suspected, reopen the `.xml` file for the affected excitation channel and re-do the export.
17. When the script runs without errors, review the main plots, `L->L`, `P->P` etc and see if any have symptoms that suggest a problem with the measurement rather than the suspension, e.g., small magnitude and high noise level as from a tripped watchdog. If so, consider rerunning the measurement for that excitation channel and re-exporting the data.
18. When any measurement issues have been fixed, edit the assignments to `printFigs` and `saveData` to `true` and re-run the script. This generates the same plots as before but writes them to the `Results` directory, sibling to the `Data` directory, e.g., `^/trunk/QUAD/H1/ITMy/SAGM0/Results` for M0 of the H1 ITMy. A failure at this point probably means the `Results` directory does not exist, in which case it should be created.

Typical set of Stage 1 plots, for H1 ITMy:
2013-01-24_H1SUSITMY_M0_ALL_TFs.pdf

1. Continue with Stage 2 Plotting on the TFs main page.

aLIGO SUS Operations Manual - Taking Spectra with Matlab

Back to Operation Manual main page

Taking damped and undamped spectra of all the raw OSEM signals and derived DOFs is a standard part of testing per G1100693 (QUAD) and G1200070 (BSFM/HLTS/HSTS), as well as official testing on OMCS and the unofficial testing done on non-SUS suspensions (TMTS/HAUX/HTTS). It takes much less time than a set of transfer functions and is convenient for diagnosing gross mechanical and electrical problems.

Measurement and Initial Plotting

Before taking spectra, be sure to inform people who may be using the suspension and do an alog announcing that spectra are about to be taken. Disable any feedback or excitation other than local damping.

For each suspension type there is a Matlab script which grabs the damped and undamped time series:

- `^/trunk/QUAD/Common/MatlabTools/plotquad_spectra.m`
- `^/trunk/BSFM/Common/MatlabTools/plotbsfm_spectra.m`
- `^/trunk/HLTS/Common/MatlabTools/plothlts_spectra.m`
- `^/trunk/HSTS/Common/MatlabTools/plothsts_spectra.m`
- `^/trunk/OMCS/Common/MatlabTools/plotomcs_spectra.m`
- `^/trunk/TMTS/Common/MatlabTools/plottmts_spectra.m`
- `^/trunk/HAUX/Common/MatlabTools/plothaux_spectra.m`
- `^/trunk/HTTS/Common/MatlabTools/plothtts_spectra.m`

Before using any of these scripts it is prudent to do an SVN update of the appropriate directory, e.g.:

```
cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/Common/MatlabTools
svn up
}}
Typically the only changes will be to the configuration section described below, and of no long-
term significance, but occasionally there will be improvements to the body of the script.

Settings near the top of the appropriate script need to be edited for the particular IFO and
suspension, e.g., for QUAD:
{{{
%% Configuration parameters
printFigs = 1; % Save PDF plots?
saveData  = 1; % Save data file (*.mat)?
plotOsemComparison = 1; % If you want to plot individual osems ASD in one figure (comparing
osems/level), instead of the 38 individual figures ! It does not save the pdf though.

ifo            = 'H1';          % Interferometer name ('X1','X2','L1','H1','H2')
opticID        = 'ETMY';        % Optic name or "BSFM" if on a test stand
['BSFM','ETMX','ITMY','MC2']
susType        = 'QUAD';        % Type of suspension ('HSTS')
meas.sensCalib = true;          % true or false -- are sensors calibrated?
meas.author    = 'A.P';    % Let us know who did all the work!
automatic_gps_saving = true;    % true/false -> "true" if you want to automate the process

svnDir = '/ligo/svncommon/SusSVN/sus/trunk/'; % LLO and everything except LHO triple test stand
% svnDir = '/ligo3/svncommon/SusSVN/sus/trunk/'; % ONLY LHO triple test stand
% svnDir = 'C:\svn\sus\trunk\'; % SMA local working directory

resultsDirName = 'Results';     % Results directory name
figFileTag = '2014-03-29_1900'; % Output filename, use to differentiate plots of different
collections of data (date_time)

%% Measurement parameters

numAvgs = 5;
freqRes = 0.01;
measDuration = 500;
freqRange = [0.01 900];
damping_on_gps=tconvert('02/06/2014 21:23:24');          % GPS Start Time Damping ON  = RES
damping_off_gps=tconvert('02/06/2014 21:23:24');         % GPS Start Time Damping OFF = REF
```

There are two modes of operation:

1. `automatic_gps_saving = true` turns on the damping, waits, turns off the damping, waits again and then reenables the damping and grabs the data.
2. `automatic_gps_saving = false` expects `damping_on_gps` and `damping_off_gps` to be the GPS times of previously identified time intervals with the damping on and off respectively.

`figFileTag` should be set to a date/time string in the form `yyyy-mm-dd_hhmm` (it is just used as a unique identifier for the data set and doesn't have to be exact).

`printFigs` and `saveData` can be set to `0` if there is any question about the validity of the data, to avoid cluttering up the SVN with broken .pdf and .mat files. The script *must* be run with `saveData=1` before proceeding to generating comparison plots as described below.

When the data has been taken and the first-stage plots have been generated, restore the suspension to its original state and post a comment to the alog saying that data-taking is over. Then do comparison plots as described in the next section.

Comparison Plots

For each suspension type there is a Matlab script which is used to collect historical measurements and generate comparison plots:

- `^/trunk/QUAD/Common/MatlabTools/plotallquad_spectra.m`
- `^/trunk/BSFM/Common/MatlabTools/plotallbsfm_spectra.m`
- `^/trunk/HLTS/Common/MatlabTools/plotallhlts_spectra.m`
- `^/trunk/HSTS/Common/MatlabTools/plotallhsts_spectra.m`
- `^/trunk/OMCS/Common/MatlabTools/plotallomcs_spectra.m`
- `^/trunk/TMTS/Common/MatlabTools/plotalltmts_spectra.m`
- `^/trunk/HAUX/Common/MatlabTools/plotallhaux_spectra.m`
- `^/trunk/HTTS/Common/MatlabTools/plotallhtts_spectra.m`

Before using any of these scripts it is ***vital*** to do an SVN update of the appropriate directory, e.g.:

```
cd /ligo/svncommon/SusSVN/sus/trunk/QUAD/Common/MatlabTools
svn up
}}
The script contains a large array called `measList` which, as explained below, contains
historical data added by SUS personnel at both sites. If the local copy is modified while there
are independent changes that have been committed to the SVN by someone else but not downloaded
via an update, there will be SVN conflicts at the next update which will be a pain to sort out.

As with the first-stage script there is a configuration section that needs to be edited:

 {{{
%% Configuration parameters
printFigs = true;
sensorNoise = true; % Include expected Sensor noise in plots

% ...

figFileTag = '2014-04-15_Phase3b_L1ETMY'; % Use to differentiate plots of different collections
of data

            %ifo  susID  measDate                 | Ref# |Phase| Site |Damping| ISI  |Approved?|
Damp ON GPSs | Damp OFF GPSs | aLOG #  | Notes
measList = {'L1','ITMY','2012-11-05_1400'; ... % | (1)  | 2c  | LLO  | BOTH  | S.S. |   FAIL  |
1036179387  |  1036179904   |LLO 5170 | Noisey OSEM channels present, R0-F3 and L1-UR.
            'L1','ITMY','2012-11-06_1400'; ... % | (2)  | 2c  | LLO  | BOTH  | S.S. |   PASS  |
1036269000  |  1036269682   |LLO 5186 | No noisey channels.

            ...

            };

useMeasts = [9 15 20 27];
```

Data for the new measurement should be added as a new line in `measList`. Only the first three items on each line are data visible to Matlab, but the columns in the comments should be filled in as well. In particular new lines should be added at a logical spot (keeping all measurements for a particular suspension together), and all subsequent lines should have value in the `Ref#` column incremented, so that the column as a whole is a uniformly increasing sequence from 1 to the number of entries. (This is tedious but no one has thought of a better solution.) The `measDate` column should be set to the date/time label that was used in the original measurement.

Set `useMeasts` to be a vector of `Ref#` values including the index of the new data and a few other data sets that make for an interesting comparison, such as earlier measurements on the same suspension and measurements on a similar suspension (if possible from the other site).

Set `figFileTag` to a string like `'2014-04-15_Phase3b_L1ETMY'` that includes the date of the latest data, the testing phase and the suspension.

Set `printFigs = 0` to begin with until the plots look good, then rerun with `printFigs = 1`.

The comparison plots will be generated in `^/trunk/XXXX/Common/Data`. Do a second comment to the original alog with the plot as an attachment, explaining the phase of testing, the choice of data sets, the colour code for the legend, and whether the data is a pass or has some particular problem.

Finally, commit all changes to the SVN. This will normally involve checking the appropriate scripts directory (`^/trunk/QUAD/Common/MatlabTools/`) and the top-level `Data` directory (`^/trunk/QUAD/Common/Data/`). This is most easily done by going to the suspension directory, doing an `svn status` command, doing `svn add` commands to add anything that shows up as modified, and then doing an `svn commit`. (It's best to be moderately selective with `svn add` - it's possible to use the `*` wildcard operator with entire directories, but then large numbers of informational messages are generated for files that have already been added, making it hard to see what has successfully been freshly added.)

```
$ cd /ligo/svncommon/SusSVN/sus/trunk/QUAD
$ svn status
? Common/Data/2014-04-15_0900_L1SUSETMY_R0M0L1L2_Spectra.mat
? Common/Data/allquads_2014-04-15_Phase3b_L1ETMY_ALL_Spectra_Doff.pdf
? Common/Data/allquads_2014-04-15_Phase3b_L1ETMY_ALL_Spectra_Don.pdf
M Common/MatlabTools/plotquad_spectra.m
M Common/MatlabTools/plotallquad_spectra.m
$ svn add Common/Data/*2014-04-15*
$ svn commit -m "New spectra data and plots"
```

aLIGO SUS Operations Manual - Operating Instructions

Back to Operation Manual main page

The front-end processes that control the suspensions are compiled from Simulink models. See the Suspensions/OpsManual/Startup page for information on (re)starting the models.

Expert-level control of the suspensions is via a large number of user-interface screens created with MEDM. See Suspensions/OpsManual/Screens for links to collections of screenshots and descriptions for controls screens for each suspension type.

Day to day operation of the suspensions will be via Guardian. See the Suspensions/OpsManual/Guardian page for background information, programming details and usage instructions.

The `safe.snap` file (see BURT) brings the suspension up in a safe state with the master switch off and the watchdogs tripped. ***By design the Guardian will not attempt to reset the watchdogs, and this must be done manually.*** See the Suspensions/OpsManual/Watchdogs page for instructions on resetting the watchdogs.

For troubleshooting advice, see the Suspensions/OpsManual/Troubleshooting page.

aLIGO: Suspensions/OpsManual/OperatingInstructions (last edited 2014-04-22 10:20:19 by MarkBarton)

aLIGO SUS Operations Manual - Calibration

Back to Operation Manual main page

The main SUS-scope items requiring some calibration are the OSEM shadow sensors and coil/magnet actuators. (When the ESD actuators are brought into service, they may need calibration as well.) Calibration is done during installation and should not need revisiting, except for some minor improvements to the coil balancing.

OSEM Calibration

Basic calibration on the OSEMs consists of measuring the "Open Light" counts value of each OSEM shadow sensor and setting the gains and offsets in the `OSEMINF` block to normalize the output to ±15000. The gains in the `COILOUTF` blocks should be set to compensate for the magnet signs.

See Suspensions/OpsManual/OSEMs for information and procedures.

Where a optical lever or the like is available, the actuators in a sensor-actuator group (e.g., QUAD L1) can be tuned to reduce unwanted coupling from longitudinal to pitch or yaw. See Suspensions/OpsManual/Diagonalization.

ESD Calibration

See Suspensions/OpsManual/ESD for information and procedures.

Electronics calibration

Jeff Kissel has done an extensive characterization of the electronics for all suspensions. Together with the OSEM calibration we understand the closed-loop transfer function to much better than a factor 2. See:

- G1100986 Suspensions Control Design Summary Table (covers all SUS and non-SUS suspensions)
- T1100378 QUAD Controls Design Description
- T1100479 BSFM Electronics Design
- T1000061 HAM Triple Suspension Controls Design Description (covers HLTS and HSTS)
- T1300535 OMCS Controls Design Description
- T1300537 TMTS Controls Design Description
- T1400030 HTTS Controls Design Description
- T1400029 HAUX Controls Design Description

aLIGO Operations Manual - Maintenance

Back to Operation Manual main page

Once SUS components are installed in vacuum, no routine maintenance is expected to be required. Failure of components will have to be addressed on a case-by-case basis drawing on experience from assembly and installation. Installation procedures are linked below for reference:

- Add link to SUS spares policy if possible.
- E1100290 aLIGO SUS QUAD Assembly and Installation Documentation
- E1100599 aLIGO SUS BSFM Assembly and Installation Documentation
- E1100472 aLIGO SUS HLTS Assembly and Installation Documentation
- E1100471 aLIGO SUS HSTS Assembly and Installation Documentation
- E1300429 aLIGO SUS OMCS Assembly and Installation Documentation
- T1100304 TMS ASSEMBLY and ALIGNMENT DOCUMENTATION
- T1300029 IO HAUX Material and Assembly Documents
- E1100440 aLIGO HAM Tip-Tilt Suspension Assembly Procedure
- E1201074 Output Faraday Isolator Assembly and Alignment Procedure
- E1300056 Output Faraday Isolator Installation and Final Alignment Procedure

aLIGO: Suspensions/OpsManual/Maintenance (last edited 2013-09-13 15:49:05 by MarkBarton)

aLIGO Operations Manual - Storage

Back to Operation Manual main page

Once SUS components are installed in vacuum per the currently assumed scope of this document, no further storage procedures are required. Storage of H2 components being diverted for LIGO-India will be addressed elsewhere.

Except for wires and blades, spares for H1/L1 are cleaned to Class A and packed per E960022 LIGO Clean and Bake Methods and Procedures.

Blades are cleaned as usual but then stored in a desiccant cabinet unpackaged but interleaved with Contec wipes.

Wires are stored uncleaned on the manufacturer's spool in a desiccant cabinet and then cleaned with a three-stage solvent wipe procedure immediately prior to use in clamp-wire-clamp assemblies.

aLIGO: Suspensions/OpsManual/Storage (last edited 2013-09-13 15:39:39 by MarkBarton)

aLIGO Operations Manual - Troubleshooting

Back to Operation Manual main page

Alarms

The suspension and seismic IOP watchdogs are wired into the Alarm Handler system. If you are the operator, see the Alarms page for help on interpreting/silencing/acknowledging alarms. However if you're just trying to troubleshoot a tripped watchdog you're probably better starting with the IOP Watchdog Overview Screen below.

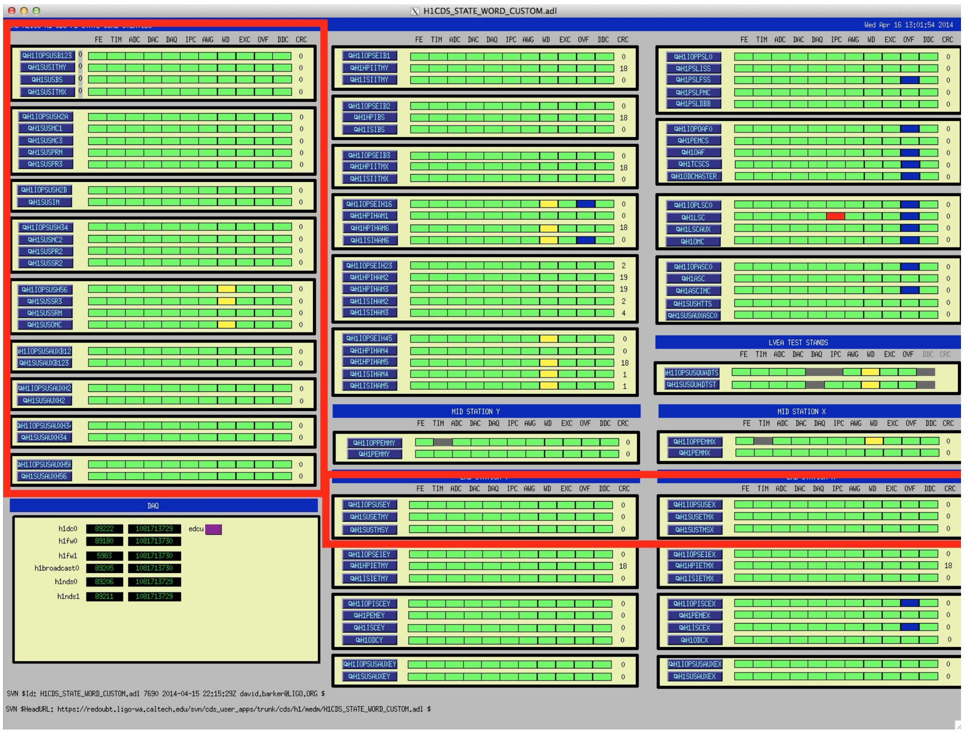
Overview Screens

To tell quickly whether one or more of the suspensions needs troubleshooting, the following screens are helpful (*some are probably LHO specific - need to check*):

- CDS Overview Screen - are the computers running
- OSEM Overview Screen - are the OSEMs giving sensible readings
- IOP Watchdog Overview Screen - are the watchdogs tripped
- Guardian Overview Screen - is the Guardian reporting a problem

CDS Overview Screen

There is an overview screen for all the computers, including those running the suspensions, which is permanently displayed on one of the wall monitors or can be summoned from the `CDS` popup menu in the `SITEMAP`. Glance at this screen every so often and confirm that all the IOP and user processes for suspensions are running normally.

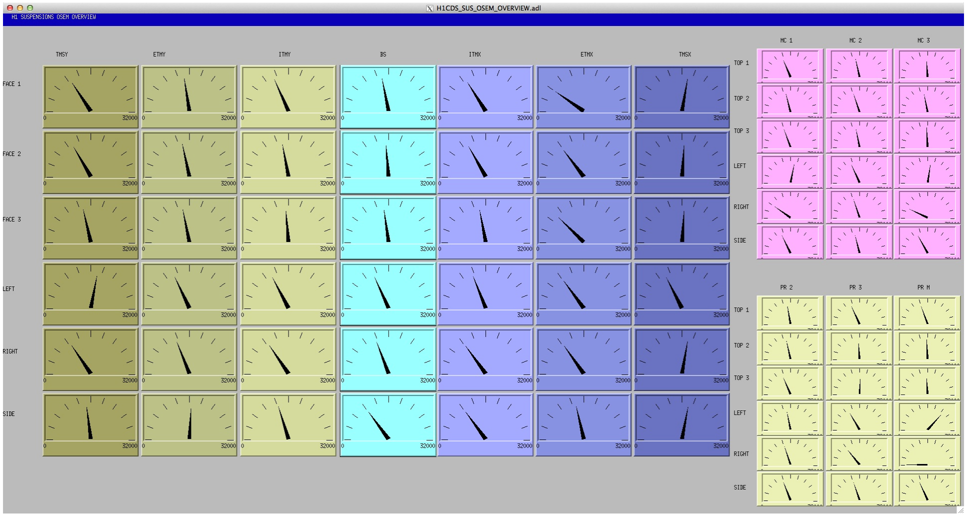


OSEM Overview Screen

There is an overview screen for the top-level OSEMs of all SUS suspensions plus the TMTSs, which is permanently displayed on one of the wall monitors or can be summoned from the `CDS` popup menu in the `SITEMAP`. Glance at this screen every so often and confirm that the indicators are roughly mid-range and not waving or jittering.

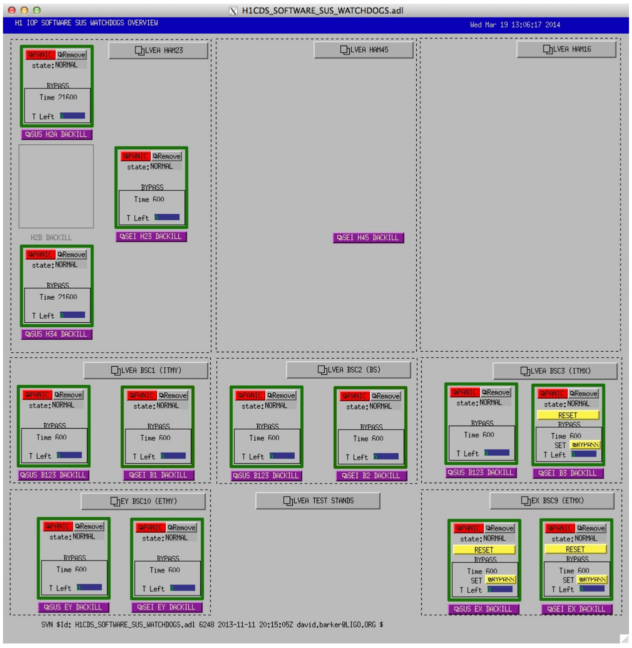
- Indicators at zero can be due to OSEMs unplugged or electronics powered down.
- Indicators near max can be due to OSEMs not yet mechanically adjusted to half open light (i.e., half-maximum) output.
- Indicators at max can be due to overload in the analog electronics.

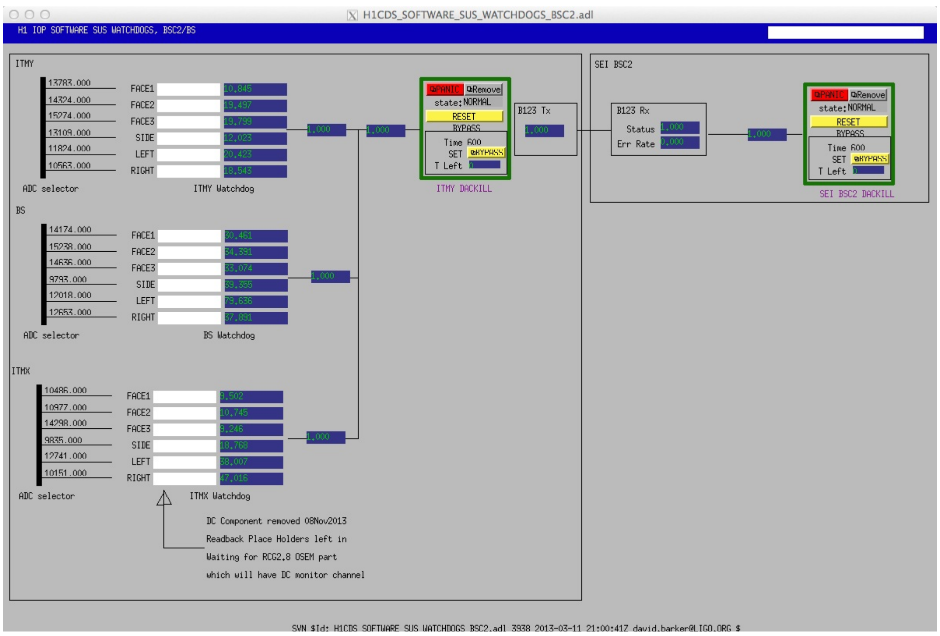
- Indicators waving can be due to air currents (if the chamber is open), people working nearby, crane activity, or ISI/HEPI excitations.
- Indicators jittering can be due to high frequency oscillations in the electronics or problems in the DAC software.



IOP Watchdog Overview Screen

There is an overview screen (below left) for all the SEI/SUS watchdogs which can be summoned from the `WD` popup menu in the `SITEMAP`. Clicking on one of the buttons for specific chambers gives an overview of that chamber (below right).





If any of the SUS IOP watchdogs, lower-level watchdogs are probably tripped as well. See the Suspensions/OpsManual/Watchdogs page for more information on resetting watchdogs.

Guardian Overview Screen

There is an overview screen for the top-level OSEMs of all SUS suspensions plus the TMTSs, which is permanently displayed on one of the wall monitors or can be summoned from the `GRD` popup menu in the `SITEMAP`. Glance at this screen every so often and confirm that none of the SUS Guardian tasks are in states other than `EXEC`, or reporting errors.

aLIGO SUS Operations Manual - Screens

Back to Operation Manual main page

See the MEDM page for general information on the location and editing of screens.

Each suspension type (except of course OFIS, which is purely passive and has no CDS interface) has its own page describing its MEDM control screens:

- QUAD
- BSFM
- HLTS
- HSTS
- OMCS
- HAUX
- HTTS
- TMTS

(These pages use a lot of common text fragments inlined from Suspensions/OpsManual/Boilerplate.)

aLIGO: Suspensions/OpsManual/Screens (last edited 2014-04-25 10:16:59 by MarkBarton)

aLIGO SUS Operations Manual - Alarms

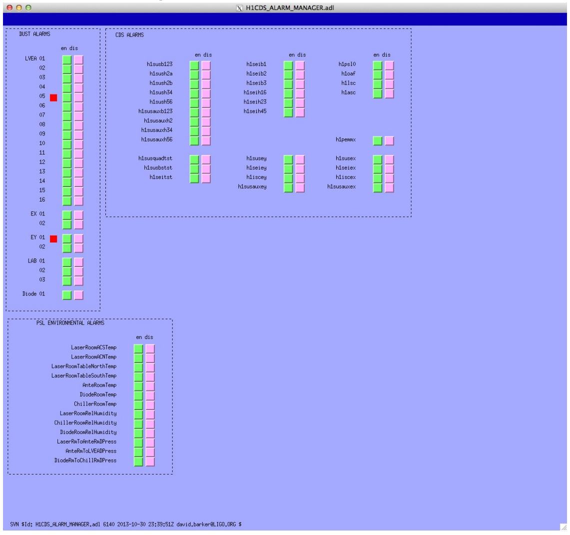
Back to Operation Manual main page

Currently this page describes alarms for LHO. LLO is probably somewhat different and a section for it will need to be written.

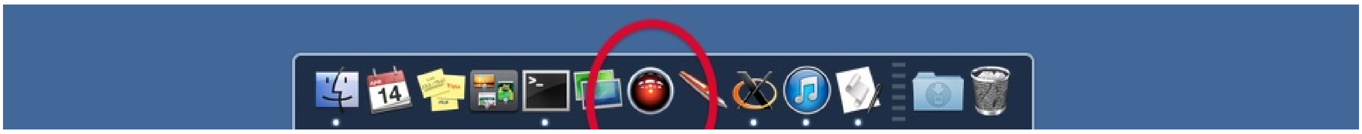
Contents

LHO Alarms Overview

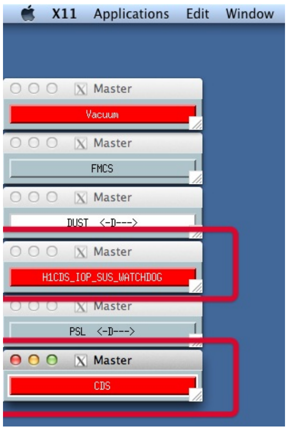
The alarm system monitors run on the iMac `alarm0` in the operator station of the control room. Alarms can only be silenced or acknowledged from this machine, but there is an overview screen that can be accessed from any machine via the `CDS->Alarm Mgr` item in the `SITEMAP`:



Alarm handlers normally run all the time but if necessary can be started by clicking on the indicated icon in the Dock at the bottom of the screen on `alarm0`:



This runs an AppleScript application called `ALH.app` which asks the operator which handlers should be (re)started and opens a cluster of small windows all called `MASTER`, one for each group of alarms. Restarting the handlers is done internally by calls to scripts `/opt/rtcds/userapps/release/cds/h1/scripts/restart_*_alarm_handler.bsh` (`*=ve/fmcs/dust/iopwatchdogs/psl/cds`), which in turn call the EPICS `alh` utility (alh manual) with config file `/opt/rtcds/lho/h1/alh/cds/h1*.alhConfig`. The two with buttons `CDS` and `H1CDS_IOP_SUS_WATCHDOG` are relevant to SUS:

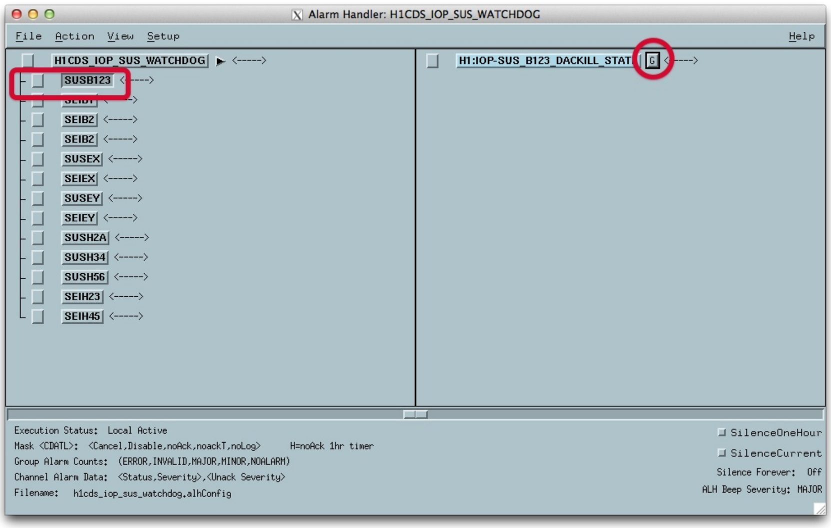


If there is an alarm, there will be a beep and the corresponding button will highlight red.

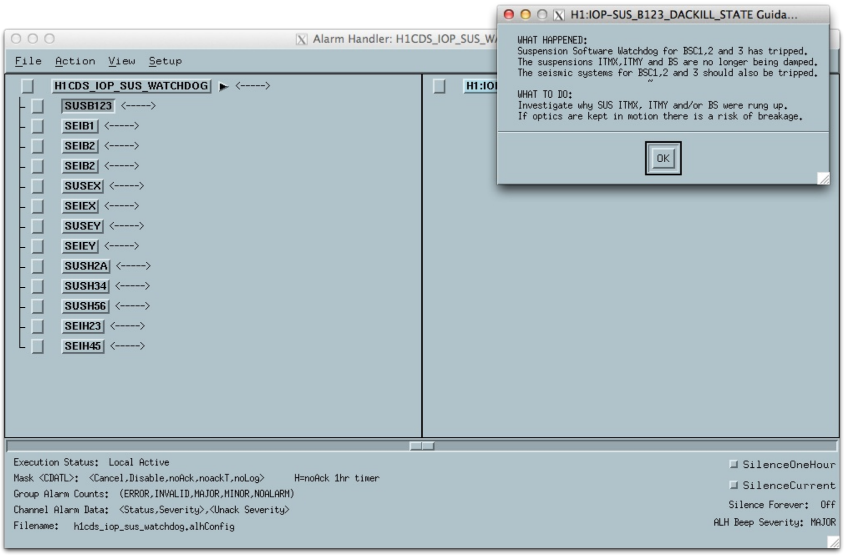
H1CDS_IOP_SUS_WATCHDOG

(See the CDS Alarm handler section below for examples of alarms being silenced and acknowledged.)

Clicking on the `H1CDS_IOP_SUS_WATCHDOG` button brings up the SUS watchdog alarm screen. Further clicking on the front end name ((H1)`SUSB123` in the example) displays further information in the pane on the right:



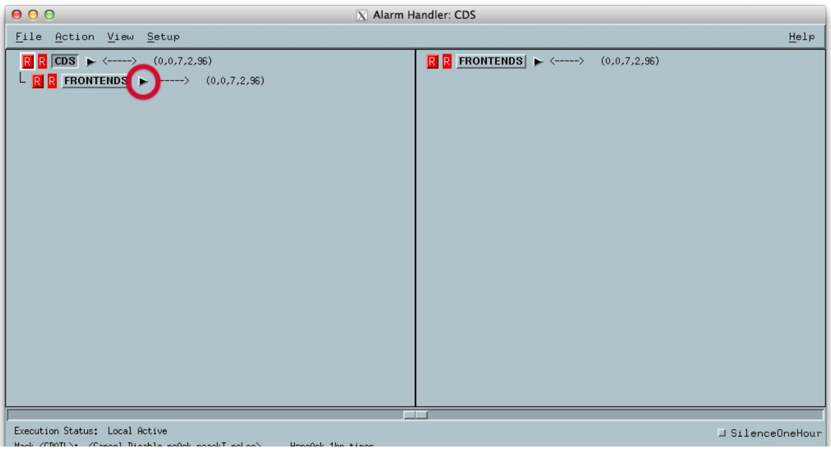
Clicking the `G` button gives guidance information in an alert window:



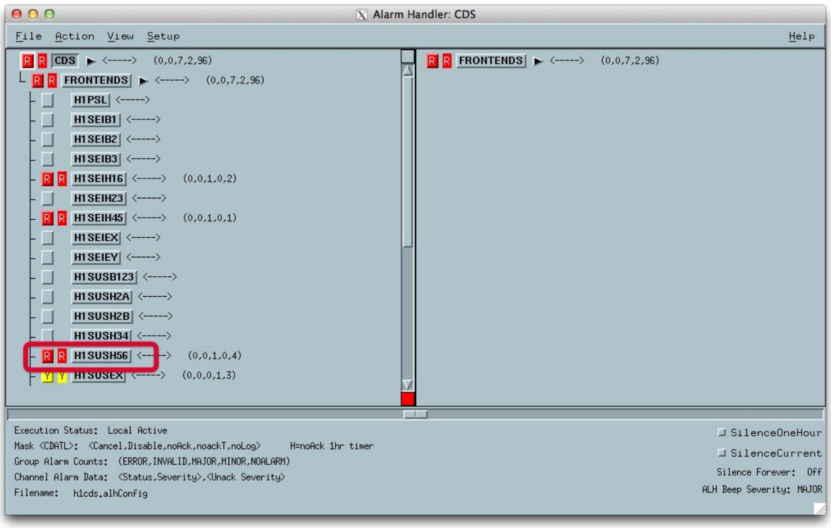
Currently all the alarms concern chamber-level watchdogs. For instructions on diagnosing and resetting watchdogs see the Suspensions/OpsManual/Watchdogs page.

CDS

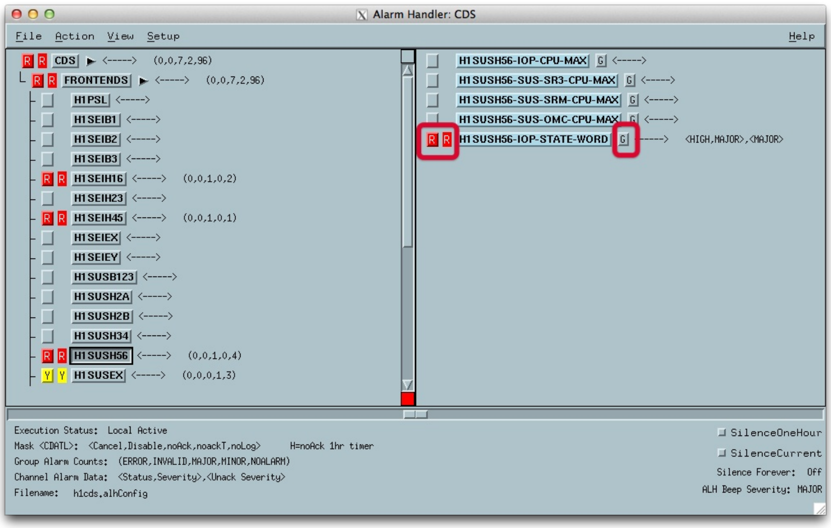
Clicking on the `CDS` button brings up the alarm screen for the front end computers:



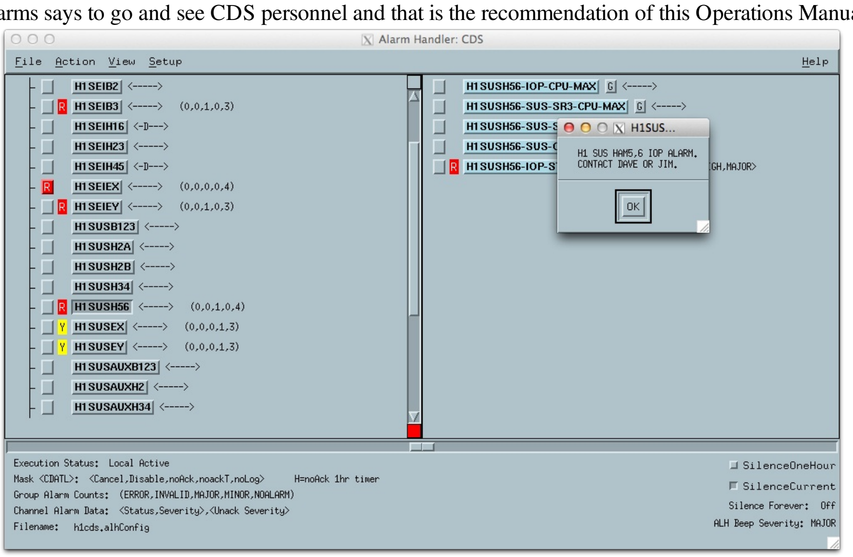
Clicking on the triangle next to `FRONTENDS` opens a view with more detail:



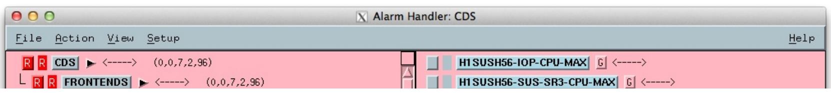
The list in the pane at the left is arranged by front end computer. Clicking on the triangle next to the alarming front end ("SUS56A" in the example) shows more detail in the pane at the right:

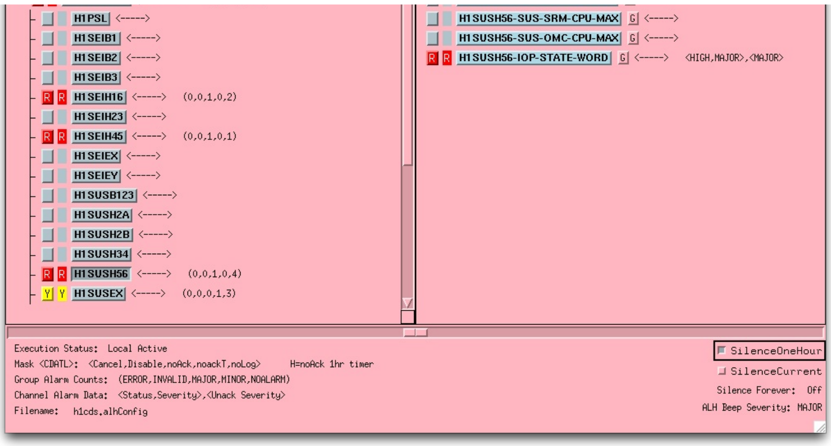


Clicking on the `G` button next to a channel shows guidance information in an alert dialog. Most/all of the guidance for CDS alarms says to go and see CDS personnel and that is the recommendation of this Operations Manual as well:

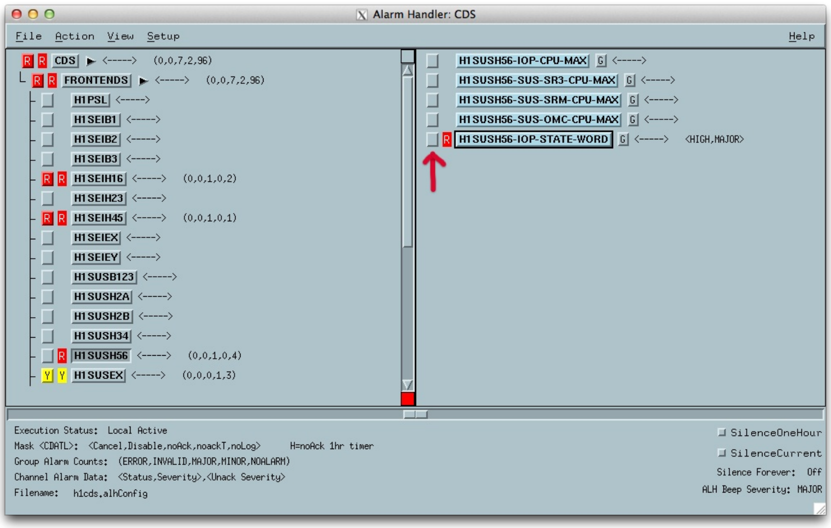


All alarms can be temporarily or indefinitely silenced by clicking on the control in the lower right in which case the background turns pink:





An individual alarm can be permanently acknowledged by clicking on the red R button (the corresponding R indicator next to it remains while the alarm is still retriggering):



LLO Alarms Overview

Needs content.

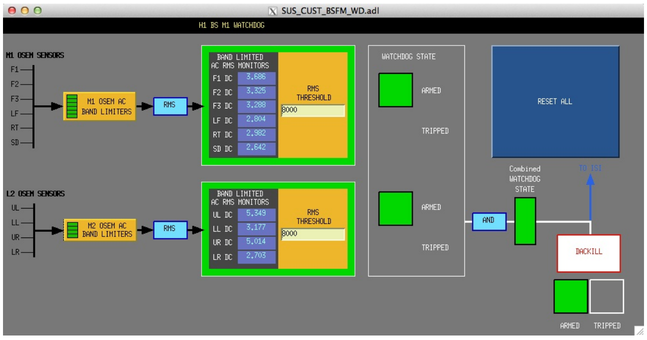
aLIGO SUS Operations Manual - Watchdogs

Back to Operation Manual main page

As of 4/20/14 at LHO, the user interface to the watchdogs for QUAD/BSFM/HxTS/OMCS/TMTS was drastically simplified, as well as some of the underlying logic. This change will be rolled out to HAUX/HTTS and LLO soon. The description below is for the new, simpler version. (It will probably change further when the forthcoming hardware watchdog is introduced.) Each suspension is protected by a master switch and two levels of watchdog, and these feed into further watchdogs for the SEI. These can be seen in the output section of the main MEDM screen for the suspension, e.g., for BS:

The master switch can simply be switched on or off at any time. The suspension watchdog feeds into the IOP watchdog, so in the case of a trip the suspension watchdog must be reset first.

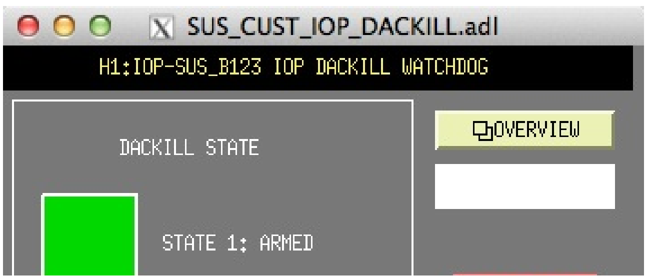
The suspension watchdog screen (accessible via the `WATCHDOGS` button) merges what were originally separate watchdogs for each of the sensor-actuator groups (M1 and M2 for BS) plus one for the user model as a whole. However, the SAG watchdogs no longer have independent screens or independent reset buttons. Each SAG watchdog checks for excess AC band-limited RMS excitation at any of its sensors. The top-level watchdog triggers if ALL the SAG watchdogs are tripped (formerly the logic was ANY, but this proved too sensitive):

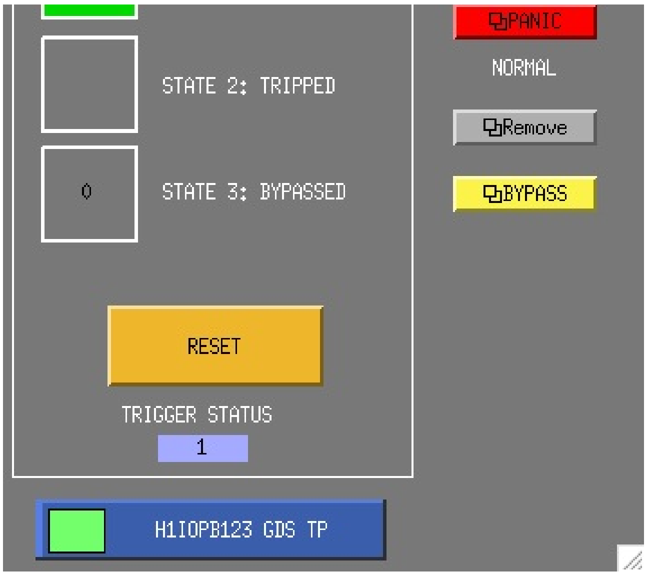


Threshold values around 8000 allow for typical excursions when the SEI group are doing ISI transfer functions.

To reset, check that none of the triggers are active (diagnose/correct as necessary) and then click `RESET ALL`, which resets all the SAG watchdogs and the top-level watchdog in one operation.

Last is the IOP watchdog. If triggered, this disables actuation on all user models served by the SUS IOP process, which is to say all suspensions controlled by the same front end (e.g., BS, ITMX and ITMY for H1SUSB123). The control screen for it is accessible via the `IOP DACKILL` button:

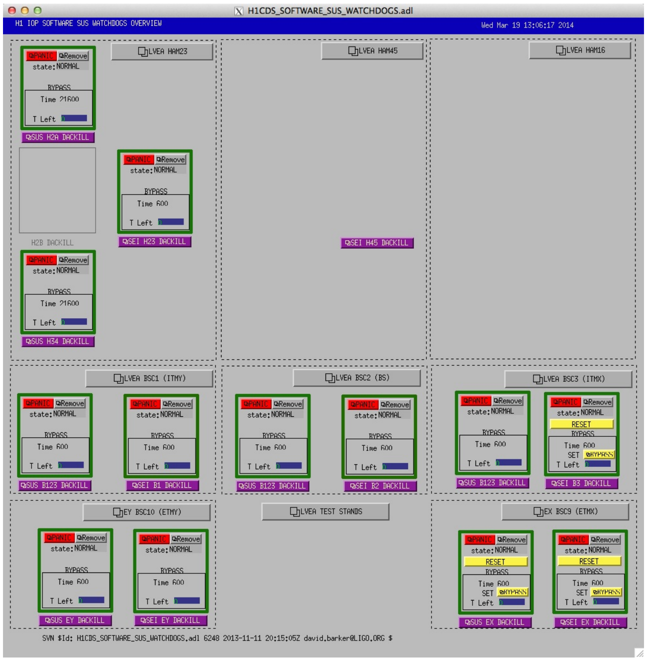




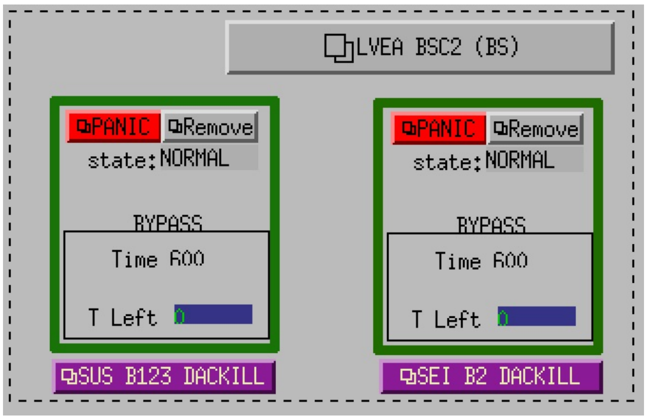
To reset, click RESET.

(*Details past here may well be different for LLO - need to check.*)

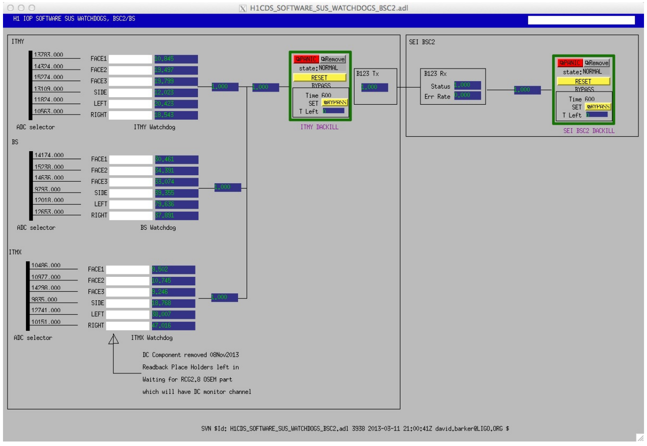
Currently at LHO, none of the SUS IOP watchdogs take trigger inputs from SEI, but they have done in the past and in principle they could again, in which case any SEI triggers would have to be resolved before the SUS IOP watchdog could be reset. (Conversely, the SEI systems do normally take triggers from SUS, so SEI personnel will need the SUS watchdogs to be reset before they can continue work.) To see related watchdogs, click on the OVERVIEW button to see the IOP watchdog overview screen :



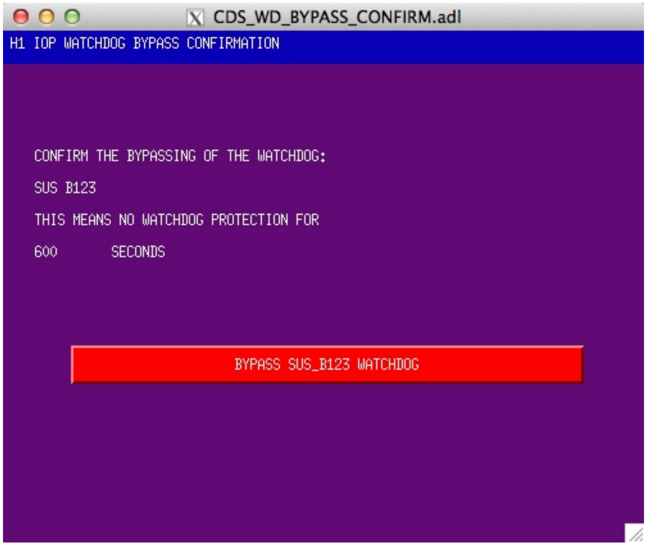
Click on the button for the chamber of interest:



The resulting screen shows all the potential triggers:



If necessary, diagnose/resolve any triggers then try clicking the RESET button in the original `SUS_CUST_IOP_DACKILL` window again. If one of the triggers cannot be resolved but is known to be harmless (e.g., a SEI system cannot disturb the SUS because it is powered down or mechanically locked) the watchdog can be by passed by clicking the `BYPASS` button:



aLIGO: Suspensions/OpsManual/Watchdogs (last edited 2014-04-22 12:03:21 by MarkBarton)

aLIGO SUS Operations Manual - Guardian

Back to Operation Manual main page

Contents

Guardian is a suite of Python utilities by Jamie Rollins for managing the interferometer, including the suspensions. Each suspension has its own Guardian process, as do many other IFO subsystems, and there is a system of higher-level manager Guardians. A recent overview of the system is G1400016. Each Guardian process is a state machine that attempts to navigate to a requested state via defined state transitions. See the #Programming section below for details of how these states and transitions are programmed.

Note: ***By design, the Guardian will warn if there are any tripped watchdogs but does not attempt to reset them.*** See Suspensions/OpsManual/Watchdogs for instructions on resetting watchdogs. As soon as the watchdogs are manually reset, Guardian detects this and resumes trying to transition to the requested state.

Operation

See the #ControlScreens section below for more information on the Individual Guardian Screens for individual suspensions and the Guardian Overview Screen.

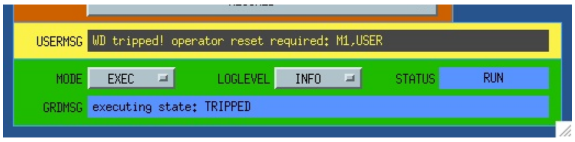
Normal Operation

Normal operation is very simple: summon the Guardian for the particular suspension using the `!Guardian` button on the main control screen for the suspension, then just make sure that the `MODE` pop-up is set to `EXEC` and select the desired state (e.g., `ALIGNED`) from the `REQUEST` pop-up.

Error conditions

Watchdog trip

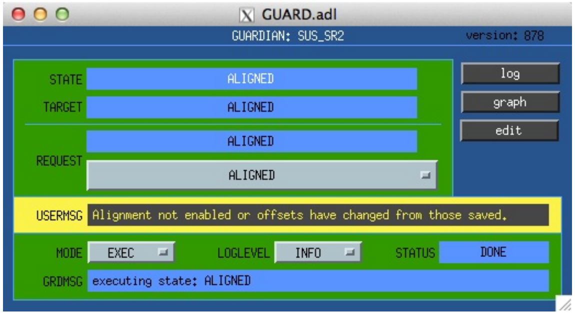
If the watchdogs trip, the Guardian will post a notification on the individual Guardian screen and transition to the `TRIPPED` state. It will also change two of the indicators on the summary screen to indicate that a notification has been posted (and should be read!) and that suspension is no longer in the requested state:

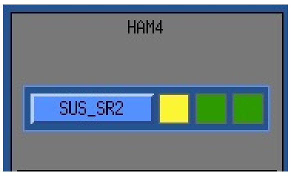


To fix this, investigate why the watchdogs have tripped and, if safe, reset them according to the instructions at Suspensions/OpsManual/Watchdogs. The Guardian will then automatically attempt to transition back to the `REQUEST` state.

Manual override

If a commissioner manually overrides one of the settings that is considered definitional of a state, especially the alignment settings that define the `ALIGNED` and `MISALIGNED` states, the Guardian will detect this and post a notification:



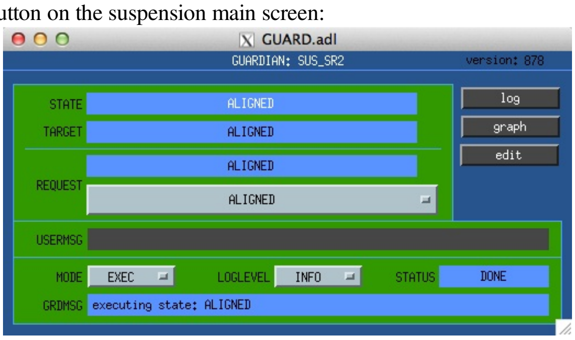


To fix this, either return the settings to the defined values, or, if the new settings are to be permanent, redefine the state to match. To change the alignment offsets that define the `ALIGNED` and `MISALIGNED` states, see the #AlignmentOffsets section below. For other state redefinitions, see the #Programming section.

Control Screens

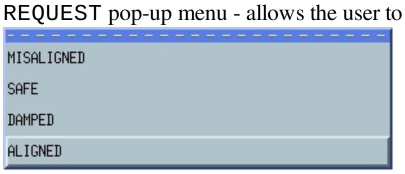
Individual Guardian Screens

Each suspension (H1:SR2 in the examples) has a Guardian control screen that can be accessed from the `!Guardian` button on the suspension main screen:



The key controls/displays are as follows:

- `STATE` display field - shows the current state.
- `TARGET` display field - shows the immediate goal state that Guardian is trying to transition to on its way to the `REQUEST` state.
- `REQUEST` pop-up menu - allows the user to select the desired state to transition to.

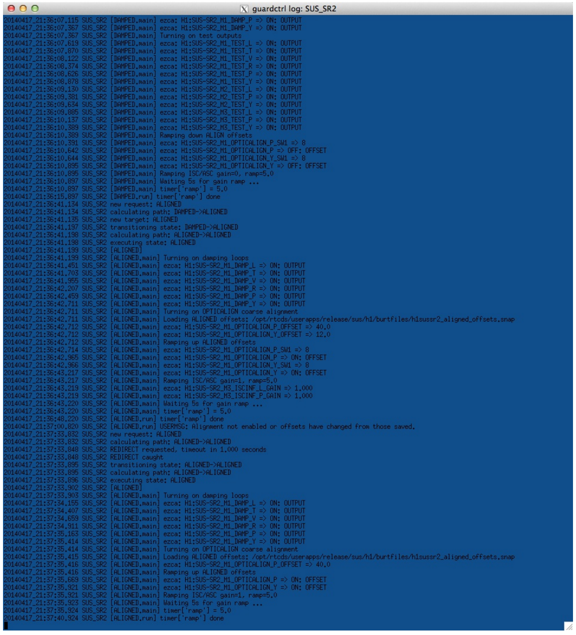
  

- `REQUEST` display field (immediately above the pop-up menu) - shows the most recent request. The default set of states defined in SUS.py is as follows. Note however that states only appear in the popup if they are accessible from the current state via explicitly and implicitly defined transitions.
  - `INIT` - state representing condition immediately after startup
  - `TRIPPED` - state representing condition after a watchdog trip
  - `SAFE` - a state in which watchdogs can safely be reset without initiating drive signals, with ISC input, local damping, and alignment offsets all off.
  - `DAMPED` - damping on, alignment offsets off
  - `MISALIGNED` - damping on, alignment offsets to parking values
  - `ALIGNED` - damping on, alignment offsets to nominal working values
- `USERMSG` display - highlights yellow and displays text if any notification has been posted by the Guardian.

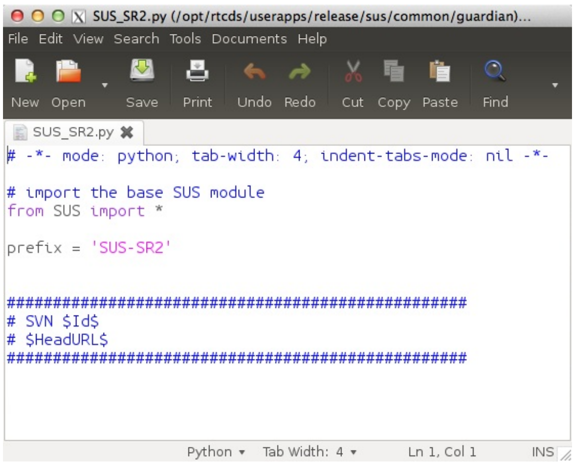
- The MODE popup allows the execution mode to be set. 

The options are:
    - LOAD - the Python code defining the states is (re)loaded
    - PAUSE - do nothing
    - EXEC - normal execution
    - MANAGED - under the control of a higher-level Guardian process
- The LOGLEVEL popup defines the verbosity of logging. 

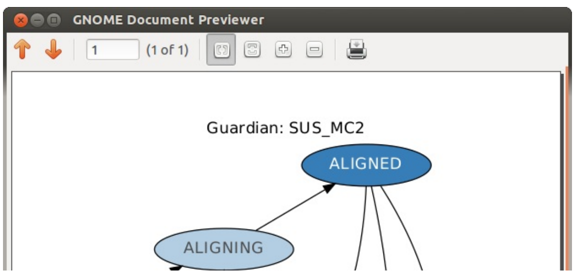
The options are
    - INFO - normal
    - DEBUG - ???
    - LOOP - ???
- The STATUS display field shows the progress of attempted transitions. If the Guardian has reached the REQUEST state it shows DONE.
- The GRDMSG shows informational messages from the Guardian.
- The log button brings up a display of log messages:

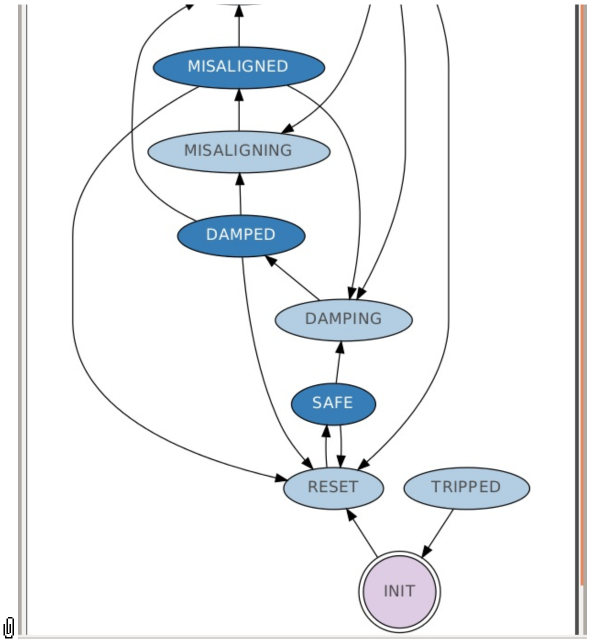


- The edit button brings the Guardian source code in an editor window (this is typically not very helpful because currently all individual suspension Guardians do nothing but inherit from the generic SUS.py):



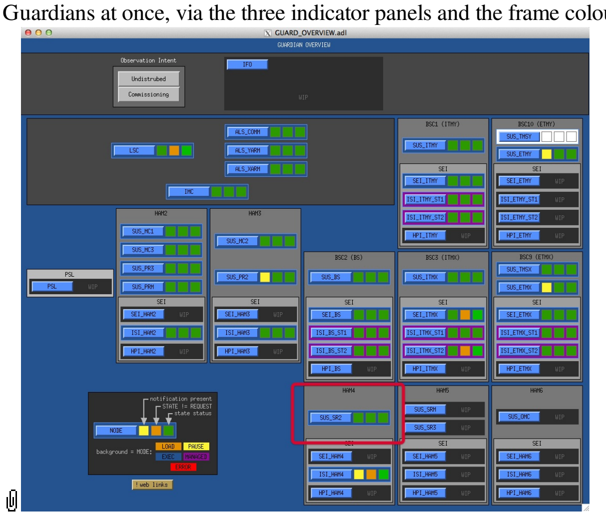
- The graph button brings up a graphical representation of the states, with requestable states colour-coded:





Guardian Overview Screen

The `GUARD_OVERVIEW` screen (accessible from the `GRD` popup in the `SITEMAP`) gives key status details for all Guardians at once, via the three indicator panels and the frame colour:



The frame should normally be dark blue meaning `EXEC` mode. The three indicators should normally be green and have the following interpretations:

- Notification present - changes to yellow if the Guardian has posted a notification to the user.
- `STATE != REQUEST` - changes to orange if the Guardian has not yet transitioned to the `REQUEST` state.
- State status - *need to find out what this does*

Programming

The main Guardian code lives at

```
/ligo/svncommon/CdsSVN/guardian
```

The Guardian for an individual suspension is defined in terms of states and edges (terminology from graph theory). Guardian accepts commands in the form of a request. Guardian then looks at the current state and calculates the shortest path to reach the request. The code for the current state is executed to completion. Once done, Guardian transitions to the next state in the path and executes its code. Once it reaches the requested state it holds there. States can also specify jump transitions that bypass the normal dynamics of the graph. Jump transitions are used for recovery from conditions that are in some sense undesirable (e.g. lock loss, watchdog trip, etc).

Each suspension Guardian is defined by a Python module that lives in

```
/opt/rtcds/userapps/release/sus/common/guardian/
```

There is a generic Guardian module `SUS.py` that is not directly usable but can be inherited from if convenient, and a bunch of suspension-specific Guardian modules, `SUS_BS.py`, `SUS_ETMX.py` etc.

Each module should import (directly or via `SUS.py`) the `GuardState` class and other useful utilities (i.e. sustools.py):

```
from guardian import GuardState
from guardian.ligopath import userapps_path
import sustools
```

Each module needs to define `prefix` as a string which is used as the base for constructing channel names:

```
prefix = 'SUS-MC2'
```

It should also define a list `edges` of (from,to) tuples of state names, representing allowed transitions:

```
edges = [
   ('DAMPED', 'ALIGNED')
]
```

IT may define `request` to be the name of a state that Guardian should transition to immediately on startup.

```
request = 'SAFE'
```

Each state is defined as a class inheriting from `GuardState`, with methods `main` and/or `run`. The `main` method is executed when the state is first reached; the `run` method is executed repeatedly thereafter. It may define the property `goto=True` to signal that the user can request Guardian to try to go to that state. The methods have access to `self.ezca` which is an object initialized with `prefix` that provides methods for channel access. However it is most conveniently used to initialize a `sustools.Sus` object which then has more suspension-specific methods:

```
class SAFE(GuardState):
    goto = True
    def main(self):
        self.optic = sustools.Sus(self.ezca)

        self.log('Turning off LOCK outputs')
        self.optic.lockOutputSwitchWrite('OFF')
        self.log('Turning off TEST outputs')
        self.optic.testOutputSwitchWrite('OFF')
        self.log('Turning off DAMP outputs')
        self.optic.dampOutputSwitchWrite('OFF')
        self.log('Ramping down ALIGN offsets')
        self.optic.alignRampWrite(rampTime)
        self.optic.alignOffsetSwitchWrite('OFF')
        self.log('Setting LOCK filter bank''s ramp time')
        self.optic.lockRampWrite(rampTime)
        # [FIXME] Add turning off locking excitation switches

        # [FIXME] Charles wrote a function to watch the yellow
        # light on the ramp. Let's watch that instead of sleeping.
        self.log('Waiting %ds for gain ramp ... ' % rampTime)
        self.timer['ramp'] = rampTime

    def run(self):
        # wait for ramp timer to complete
        if not self.timer['ramp']:
            return

        # Turn off Master Switch
        self.optic.masterSwitchWrite('OFF')

        if self.ezca.read('DACKILL_STATE') == 1:
            return True
```

Alignment Offsets

Settings that define the `ALIGNED` and `MISALIGNED` and other states with offsets are currently stored in files in BURT syntax in the `userapps` repository, e.g.:

`/opt/rtcds/userapps/release/sus/${ifo}/burtfiles/h1sussr2_aligned_offsets.snap`

(This system of settings files is being re-considered and may be partly or wholly replaced by settings stored in EPICS channels.)

Standard States

The following standard states are defined in `SUS.py`.

General purpose states

INIT State

This is the initial state when a Guardian process is being started. It tries to determine if the suspension was previously in the `ALIGNED, or MISALIGNED` states by looking at the alignment offsets that are being applied. Otherwise it brings the suspension to the

RESET state and then transitions to the "INITIAL REQUEST" state (defined for now as the ALIGNED state):

1. Is the masterswitch on/off (if off go to the INIT REQUEST)?
2. Is the suspension aligned/misaligned (if not go to the INIT REQUEST)?
3. Are the output switches of alignment offsets filter banks on (if not go to the INIT REQUEST)?
4. Are both Pitch and Yaw alignment offsets on (if not go to the INIT REQUEST)?
    1. If the current offsets are equal to the aligned offsets, go to the `ALIGNED` state.
    2. If the current offsets are equal to the misaligned offsets, go to the `MISALIGNED` state.
    3. Otherwise go to the `DAMPED` state.

TRIPPED State

This state sits and waits for operator/user to reset the watchdogs. When watchdogs are reset, Guardian brings the suspension back to the `REQUEST` state.

Requestable States

The following states are "requestable" from the menu:

SAFE State

No signal sent to the DAC. ISC inputs off / damping output off / test filters off / masterswitch off. Constantly checks for WD trips.

UNDAMPED State

No damping engaged / "Test" filter banks on / master Switch ON. Constantly checks for WD trips

DAMPED State

Damping loops on / test filters on / master Switch ON. Constantly checks for WD trips

ALIGNED State

Suspension damped and test mass aligned with respect to its cavity. Constantly checks for WD trips & compares the alignment offset value with the one saved last.

MISALIGNED State

Suspension damped and test mass misaligned with respect to its cavity. Constantly checks for WD trips & compares the "mis"alignment offset value with the one saved last.

Transitioning States

The following states represent intermediate states on the way to the requested state:

RESET State : Transitions to SAFE

1. Checks if WD is tripped (if tripped, goes to `TRIPPED` state).
2. Ramps down ISC control signals inputs.
3. Turns off optical lever damping (for suspensions that have it).
4. Ramps down/turn off alignment offsets.
5. Turns off TEST output switches
6. Turns off DAMPING output switches.
7. Turns off the master switch after ramping down the gains.
8. Constantly checks for WD trips.

UNDAMPING State : Transitions to UNDAMPED

1. Checks if the WD is tripped (if tripped, it goes to the `TRIPPED` state)
2. Turns on Master Switch
3. Turns on test filters
4. Turns off damping outputs

DAMPING State : Transitions to DAMPED

This state means the suspension is damped and is sitting in its undriven equilibrium position (no alignment offsets or other DC actuation):

1. Checks if the WD is tripped (if tripped, it goes to the `TRIPPED` state).

2. Turns DAMPING output switches.
3. Turns off optical lever damping (for suspensions that have it).
4. Ramps down/turn off alignment offsets.
5. Ramps down ISC control signals inputs.

ALIGNING State : Transitions to ALIGNED

In this state the suspension is damped and the stored `ALIGNED`-state offset values are being applied:

1. Checks if the WD is tripped.
2. Turns on Master Switch (redundant since should be ON from DAMPED).
3. Turns on DAMPING output switches (redundant since should be ON from DAMPED).
4. Ramps up ISC input.
5. Loads the alignment offsets from the BURT file.
6. Ramps up/turn on alignment offsets.
7. Constantly checks if the current offsets are matching the saved ones. (If not it sends an error message on the Guardian control screen. You you all the)

MISALIGNING State : Transitions to MISALIGNED

Similar to the `ALIGNING` state except the values of the offsets are different, the other DC alignments are turned of, and the optical lever damping is turned off for the suspension.

TMSX and TMSY Special States

These are defined in `SUS_TMS.py` which inherits the standard states from `SUS.py` and is inherited by `SUS_TMSX.py` and `SUS_TMSY.py`.

ALIGNED _TO_PD1

Suspension damped and test mass aligned to "photodiode 1". The pd1 is located on the lower left of itm baffle. Constantly checks for WD trips and alignment values.

ALIGNED_TO_PD4

Suspension damped and test mass aligned to "photodiode 4". The pd4 is located on the upper right of itm baffle. Constantly checks for WD trips and alignment values.

ALIGNING_TO_PD1 : Transitions to ALIGNED_TO_PD1

Similar to the ALIGNING state except the values of the offsets are different

ALIGNING_TO_PD4 : Transitions to ALIGNED_TO_PD4

Similar to the ALIGNING state except the values of the offsets are different

aLIGO: Suspensions/OpsManual/Guardian (last edited 2014-05-22 18:41:02 by ArnaudPele)

aLIGO SUS Operations Manual - Suspensions By Chamber

Back to Operation Manual main page

This index is derived from T1100073 List of Suspensions in Chambers. Old H2 links have been moved to a subpage and have been replaced with I1 Most of the linked subpages on this wiki have not been created, but can be used if convenient for suspension-specific notes.

The DCC links are to acceptance reports. See the following DCC file cards for up-to-date lists of accepted suspensions with documentation:

Links to sections below:

QUAD	BSFM	HLTS	HSTS	OMCS	HAUX	HTTS	TMTS

Quad (ITM, ETM, CP and ERM)

- Info on Quad
- E1201044: aLIGO SUS QUAD Individual Acceptance Reports

H1

- H1:ITMx (BSC3)
- H1:ETMx (BSC9)
- H1:ITMy (BSC1)
- H1:ETMy (BSC10)

L1

- L1:ITMx (BSC3)
- L1:ETMx (BSC4)
- L1:ITMy (BSC1)
- L1:ETMy (BSC5)

I1

- I1:ITMx (BSC3)
- I1:ETMx (BSC4)
- I1:ITMy (BSC1)
- I1:ETMy (BSC5)

BSFM

Info on BSFM

E1201045: aLIGO SUS BSFM Individual Acceptance Reports

H1

- H1:BS (BSC2) E1300712

L1

- L1:BS (BSC2) E1300699

I1

- I1:BS (BSC2)

HLTS (HAM Large Triple Suspension - PR3 and SR3)

Background Info on HLTS

E1201046: aLIGO SUS HLTS Individual Acceptance Reports

H1

- H1:PR3 (HAM2)
- H1:SR3 (HAM5)

L1

- L1:PR3 (HAM2)
- L1:SR3 (HAM5)

I1

- I1:PR3 (HAM2)
- I1:SR3 (HAM5)

HSTS (HAM Small Triple Suspension - MC1, MC2, MC3, PR2, SR2, PRM and SRM)

- Info on HSTS
- E1201047: aLIGO SUS HSTS Individual Acceptance Reports

H1

- H1:MC1 (HAM2)
- H1:MC3 (HAM2)
- H1:PRM (HAM2)
- H1:MC2 (HAM3)
- H1:PR2 (HAM3)
- H1:SR2 (HAM4)
- H1:SRM (HAM5)

L1

- L1:MC1 (HAM2)
- L1:MC3 (HAM2)
- L1:PRM (HAM2)
- L1:MC2 (HAM3) E1201042
- L1:PR2 (HAM3) E1300513
- L1:SR2 (HAM4)
- L1:SRM (HAM5)

I1

- I1:MC1 (HAM2) E1300514
- I1:MC3 (HAM2)
- I1:PRM (HAM2) E1300497
- I1:MC2 (HAM3)
- I1:PR2 (HAM3)
- I1:SR2 (HAM4)
- I1:SRM (HAM5)

OMCS (Output Mode Cleaner Suspension)

- Info on OMCS
- E1201048: aLIGO SUS OMCS Individual Acceptance Reports

H1

- H2:OMC (HAM6)

L1

- L1:OMC (HAM6)

I1

- I1:OMC (HAM6)

TMTS

- Background Info on TransMon
- TMTS (non-SUS suspension, but link if possible)

H1

- H1:TMSx (BSC9)
- H1:TMSy (BSC10)

L1

- L1:TMSx (BSC4)
- L1:TMSy (BSC5)

I1

- I1:TMSx (BSC4)
- I1:TMSy (BSC5)

HAUX (HAM Auxiliary Suspension)

- Info on HAUX
- HAUX (non-SUS suspension, but link if possible)

H1

- H1:IM1 (HAM2)
- H1:IM2 (HAM2)
- H1:IM3 (HAM2)
- H1:IM4 (HAM2)

L1

- L1:IM1 (HAM2)
- L1:IM2 (HAM2)
- L1:IM3 (HAM2)
- L1:IM4 (HAM2)

I1

- I1:IM1 (HAM2)

- I1:IM2 (HAM2)
- I1:IM3 (HAM2)
- I1:IM4 (HAM2)

HTTS (Tip-Tilt Suspension)

- Info on HTTS
- HTTS (non-SUS suspension, but link if possible)

H1

- H1:RM1 (HAM1)
- H1:RM2 (HAM1)
- H1:OM1 (HAM6)
- H1:OM2 (HAM6)
- H1:OM3 (HAM6)

L1

- L1:RM1 (HAM1)
- L1:RM2 (HAM1)
- L1:OM1 (HAM6)
- L1:OM2 (HAM6)
- L1:OM3 (HAM6)

I1

- I1:RM1 (HAM1)
- I1:RM2 (HAM1)
- I1:OM1 (HAM6)
- I1:OM2 (HAM6)
- I1:OM3 (HAM6)

aLIGO: Suspensions/OpsManual/ByChamber (last edited 2014-04-25 10:20:12 by MarkBarton)

aLIGO SUS Operations Manual - OSEM Info and Setup

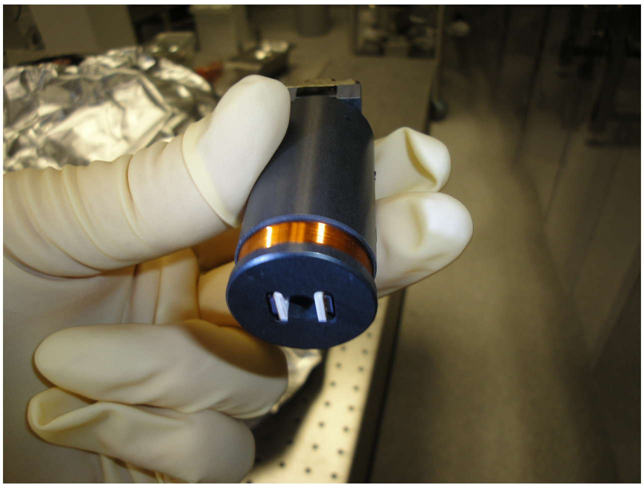
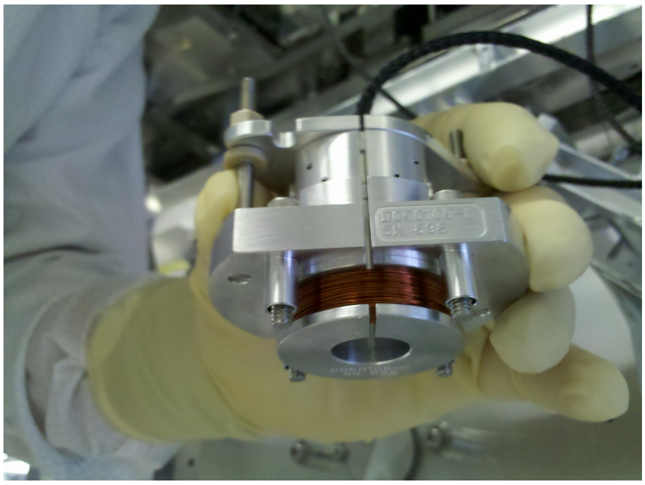
Back to Operation Manual main page

Contents

Background

OSEMs are Optical Sensors, with ElectroMagnetic actuators. That is, they're a combination of a shadow sensor that detects the position of a flag via how much it blocks a light beam, and a coil which acts on a magnet.

There are two varieties of OSEMs used in the suspensions: AOSEMs (A="advanced" version of iLIGO design) and BOSEMs (B="Birmingham University", accommodating a much larger magnet).

AOSEM	BOSEM
	

See the #References section below for more background.

Coil/Magnet Calibration

There are a variety of different combinations of coils and magnets - see M0900034.

The force per current for each of these combinations has been calculated in T1000164 Calculation and measurement of the OSEM actuator sweet spot position.

For each type of suspension, the magnets are set in a standard orientation pattern which tries to give a zero net dipole moment for each mass as near as practical. See the following:

- E1000617: Quad Suspension Controls Arrangement Poster
- E1100108: Beamsplitter-Folding Mirror Controls Arrangement
- E1100109: HAM Suspensions Controls Arrangement Poster (also covers OMCS)
- E1200045: TMTS Suspension Controls Arrangement Poster
- E1200215: HAM Auxiliary Suspensions Controls Arrangment

Since AOSEMs and BOSEMs are wound oppositely, the correspondence between the outward facing pole specified on the posters and COILOUTF gain is that N=+1 for AOSEM and S=+1 for BOSEM.

HAUX and HTTS have magnets as for the lower stages of an HLTS or HSTS (but HAUX have AOSEMs whereas HTTS

HAUX and HTTS have magnets as for the lower stages of all HLTS or HSTS (but HAUX have AOSEMs whereas HTTS have BOSEMs).

The coils are very consistent in manufacture, and the magnets fairly so, so it was not considered worthwhile to individually calibrate each coil/magnet pair. Information on magnet variation can be found in T1000618 Information on Magnets for BSC Suspensions.

Shadow Sensor Calibration

The response of the AOSEM and BOSEM sensors is a sigmoidal shape which has been characterized in T1200468 and T1100455. The typical sensitivities to motion are best expressed as 1.6 OL/mm for AOSEMs and 1.220 OL/mm for BOSEMs, where OL is the "open light" output (i.e., maximum output, in whatever units). Scaling by the OL value removes most of the variation due to different LED brightnesses and PD sensitivities, leaving only a small scatter thought to be due to geometric variations. However in the software a common value of 1/0.7 = 1.429 OL/mm has been used for simplicity. The net calibration is applied in two parts:

- During installation, the OL counts at the DAC are measured for each OSEM (a typical value is 22-29K) and the gain fields in the OSEMINF blocks are set to 30000/OL and the offset fields to -OL/2, so as to normalize the input to a standard ±15000.
- The FM5 filter in the OSEMINF block then applies a gain of 0.0233333 (=30000*0.7/1000) to convert to µm.

Procedure

Measuring the OL values of OSEMs and calculating the gains and offsets for the `OSEMINF` blocks is more part of Assembly/Installation than Operations, but operators might well be called to assist because it involves control room tools. The work is naturally a two-person job, one person in the cleanroom or vacuum chamber, and one near a workstation.

1. In the cleanroom or chamber, note the serial numbers of the OSEMs in each position for recording in ICS per the applicable assembly procedure.
2. Wind in the OSEMs to be set up until the flag entirely blocks the light beam and confirm that the ADC values are within a few counts of zero. If the OSEM runs out of range without the count going to zero, it usually means that the flag or magnet is not well centered. (This test may not be practical for the lower stage OSEMs on a quad, because adjusting the OSEMs in and out perturbs the pitch of the reaction chain.)
3. Back out the OSEMs to be set up until they are well clear of the flags. (Again this test may not be practical for the lower stage OSEMs on a quad in situ, in which case it needs to be done before they are installed.)
4. At the workstation, open Matlab and navigate to `^/trunk/Common/MatlabTools`:

```
$ matlab&
>> cd /ligo/svncommon/SusSVN/sus/trunk/Common/MatlabTools
```

5. The `logOLs()` function takes a 10 s average of all the OSEMs on a suspension (at the `_IN1` test point of the corresponding OSEMINF block, i.e., before any gains or offsets), and `prettyOSEMgains()` uses this to generate a table with the gains and offsets per the above formulae. Use them as per the following example:

```
>> prettyOSEMgains('H1','PR3')

M1T1 25835 1.161 -12917
M1T2 30072 0.998 -15036
M1T3 28666 1.047 -14333
M1LF 25623 1.171 -12812
M1RT 25798 1.163 -12899
M1SD 28238 1.062 -14119
M2UL 17706 1.694  -8853
M2LL 20285 1.479 -10142
M2UR 18746 1.600  -9373
M2LR 17714 1.694  -8857
M3UL 17313 1.733  -8656
M3LL 22891 1.311 -11445
M3UR 24376 1.231 -12188
M3LR 17159 1.748  -8580
```

6. Open up the MEDM screen for the suspension, click on the appropriate OSEM INPUT FILTERS button(s) and enter the gains and offsets in the fields of the OSEMINF screens. If the OSEMs are still physically at the OL position, the outputs on the right of the OSEMINF screen should be almost exactly 15000. Ignore any rows that are invalid because the corresponding OSEMs were not fully backed out (e.g., because they had already been set up earlier).
7. In the cleanroom or chamber, wind the OSEMs in until the output from the OSEMINF screen is near zero (try for ±600). (For vertically mounted OSEMs, aim off by the buoyancy correction from T1100616 so that the value will be near zero after the suspension sags during pump-down.) It may be convenient to use the panel meter displays accessed from the light blue CENTERING buttons on the suspension main screen.

8. At the workstation, update the `safe.snap` file per the instructions at ../BURT.
9. Do an alog with the table from Matlab. If any rows are invalid because those OSEMs were not fully backed out be sure to delete them before posting. Mention, if true, that the gains and offsets were entered and the `safe.snap` updated.
10. Download the spreadsheet from E1200343 OSEM Chart, enter the new OL values, and upload the modified spreadsheet as a new version.

References

- E1100337: Suspension Electronics Drawing Tree
- E1000617: QUAD input/output electronics summary
- E1100108: BSFM input/output electronics summary
- E1100109: HxTS input/output electronics summary
- T1200468: OSEM calibration
- T1100616: Buoyancy correction (for vertically mounted OSEMs)
- M0900034: Magnet sizes and types and OSEM types in Adv. LIGO suspensions
- T1200015: SUS Sensor/Actuator Sign Conventions
- E1000617: Quad Suspension Controls Arrangement Poster
- E1100108: Beamsplitter-Folding Mirror Controls Arrangement Poster
- E1100109: HAM Suspensions Controls Arrangement Poster

aLIGO: Suspensions/OpsManual/OSEMs (last edited 2014-04-25 13:23:18 by MarkBarton)

aLIGO SUS Operations Manual - Optical Levers

Back to Operation Manual main page

References

- D1201069: Optical Lever System Wiring
- D1100022: aLIGO SUS ITM and BS Wiring Diagrams (for H1/L1)
- D1001725: aLIGO SUS ITM, BS, and FM System Wiring Diagrams (for H2)
- D1002741: aLIGO SUS ETM System Wiring Diagrams
- D0902810: aLIGO HAM2 Suspension Controls Wiring Diagrams
- D1000599: aLIGO HAM 3-4 Suspension Controls Wiring Diagram
- D1002740: aLIGO SUS HAM 5-6 Wiring Diagrams

Overview

The signals from the optical levers are plumbed through SUS front ends and models and used variously by SUS and SEI models.

The quadrant signals for each optical lever come in through one of the SUS controls front ends, whereas the associated laser power monitor signal comes in through one of the SUS "aux" front ends.

The signals destined for other suspensions or SEI are sent across IPC to the model that needs them.

The following tables show the intended scheme for H1. (L1 should be identical, except with h1->l1 and H1->L1.) A "*" denotes items that (as of 4/24/14 at LHO) need to be updated to provide the necessary support.

There doesn't appear to be support for the SR3 oplev in D1002740 (HAM5-6) or anywhere else in the SUS wiring diagrams. Jeff Kissel and Mohana Mageswaran are working on this. Some software support has appeared, presumably based on prerelease versions of D1002740.

Optical Lever Segment ADC Info

This table shows which IOP and user models read the optical lever quadrant signals. The ADC and channel numbers are for quadrant #1, but in all cases #2, #3 and #4 follow in sequence. Where an IPC channel is given, the model that ultimately uses the signals is different - see next section.

Opt Lev	Quadrant IOP Model	Quadrant User Model	Quadrant ADC Num	Quadrant #1 Chan Num	Quadrant #1 IPC Name	Wiring Diagram
ITMx	h1iopsusb123	h1susitmx	0	0	N/A	D1100022
ITMy	h1iopsusb123	h1susitmy	0	4	N/A	D1100022
ETMx	h1iopsusb9	h1susetmx	1	12	N/A	D1002741
ETMy	h1iopsusb10	h1susetmy	1	12	N/A	D1002741
BS	h1iopsusb123	h1susbs	0	8	N/A	D1100022
PR3	h1iopsus2b	h1susim	0	20	H1:SUS-IM_PR3_OPLEVSIG_QUAD1	D0902810
SR3	h1iopsush56	h1sussrm	1	12	H1:SUS-SRM_2_SR3_OPLEVSIG_QUAD1	D1002740??
HAM2	h1iopsush2b	h1susim	0	16	H1:SUS-IM_2_ISIHAM2_OPLEVSIG_QUAD1	D0902810
HAM3	h1iopsush34	h1susmc2	1	16	H1:SUS-MC2_2_ISIHAM3_OPLEVSIG_QUAD1	D1000599
HAM4	h1iopsush34	h1susmc2	1	20	H1:SUS-MC2_2_ISIHAM4_OPLEVSIG_QUAD1	D1000599
HAM5	h1iopsush56	[illegible]	1	8	H1:SUS-	D1002740

HAM5	h1iopsush56	h1sussrm	1	8	SRM_2_ISIHAM5_OPLEVSIG_QUAD1	D1002740

Optical Lever Client Model Info

The following models use the optical lever signals.

Opt Lev	Quadrant #1 Client Model	Quadrant #1 EPICS Name
ITMx	h1susitmx	H1:SUS-ITMX_L3_OPLEV_SEG1_IN1
ITMy	h1susitmy	H1:SUS-ITMY_L3_OPLEV_SEG1_IN1
ETMx	h1susetmx	H1:SUS-ETMX_L3_OPLEV_SEG1_IN1
ETMy	h1susetmy	H1:SUS-ETMY_L3_OPLEV_SEG1_IN1
BS	h1susbs	H1:SUS-BS_M3_OPLEV_SEG1_IN1
PR3	h1suspr3	H1:SUS-PR3_M3_OPLEV_SEG1_IN1
SR3	h1sussr3	H1:SUS-SR3_M3_OPLEV_SEG1_IN1
HAM2	h1isiham2	H1:ISI-HAM2_OPLEV_QUAD1_IN1
HAM3	h1isiham3	H1:ISI-HAM3_OPLEV_QUAD1_IN1
HAM4	*h1isiham4*	*H1:ISI-HAM4_OPLEV_QUAD1_IN1*
HAM5	*h1isiham5*	*H1:ISI-HAM5_OPLEV_QUAD1_IN1*

Optical Lever Power Monitor ADC Info

There is provision in the hardware for laser power monitor signals. The ADC for these signals is sometimes done by different front ends from those that read the quadrant signals (physically convenient to the laser rather than the QPD). At LHO almost none of this has been wired up in either hardware or software yet.

Opt Lev	Pwr Mon IOP Model	Pwr Mon User Model	Pwr Mon ADC Num	Pwr Mon Channel Num	Pwr Mon EPICS Name	Wiring Diagram
ITMx	h1iopsusauxb123	h1susauxb123*	6	20	??	D1100022
ITMy	h1iopsusauxb123	h1susauxb123*	6	21	??	D1100022
ETMx	h1iopsusauxex	h1susauxex*	1	16	??	D1002741
ETMy	h1iopsusauxey	h1susauxey*	1	16	??	D1002741
BS	h1iopsusauxb123	h1susauxb123*	6	22	??	D1100022
PR3	h1iopsusauxh2	h1susauxh2*	0	20	??	D0902810
SR3	??	??	??	??	??	??
HAM2	h1iopsusauxh2	h1susauxh2*	0	16	??	D0902810
HAM3	h1iopsusauxh3	h1susauxh3*	2	4	??	D1000599
HAM4	h1iopsusauxh3	h1susauxh3*	2	5	??	D1000599
HAM5	h1iopsusauxh56	h1susauxh56*	0	20	??	D1002740

aLIGO SUS Operations Manual - Electrostatic Drive

Back to Operation Manual main page

See

- E1300848: aLIGO SUS QUAD ESD Design Documentation

aLIGO: Suspensions/OpsManual/ESD (last edited 2014-04-29 15:40:57 by MarkBarton)

aLIGO SUS Operations Manual - Diagonalization

Back to Operation Manual main page

Add more stuff on procedures for diagonalizing actuators.

The actuator electronics, coil, and magnet chains for a given isolation stage are specified to be precise to within 5% in hardware. However, to minimize unwanted Longitudinal to angular (Pitch and Yaw) coupling, we desire a balance of better than 1%, such that the longitudinal coupling to angle is less than 0.1 [rad / m] at all frequencies.

Various approaches to tweaking the coils for 4-osem isolation stages can be found in LHO aLOG 9453, and LHO alog 11393.

For TOP masses (i.e. 6 OSEM configurations), we do not yet have a method for balancing the coils a select frequency. Instead, we measure the frequency dependent cross-coupling, say TOP L to TEST P or TOP L to TOP Y transfer functions, and design filters to stick in off-diagonal elements of the L P Y drive matrix that compensate for this frequency dependence.

aLIGO: Suspensions/OpsManual/Diagonalization (last edited 2014-04-25 10:16:11 by MarkBarton)

aLIGO SUS Operations Manual - Damping

Back to Operation Manual main page

The damping design is described in the following documents:

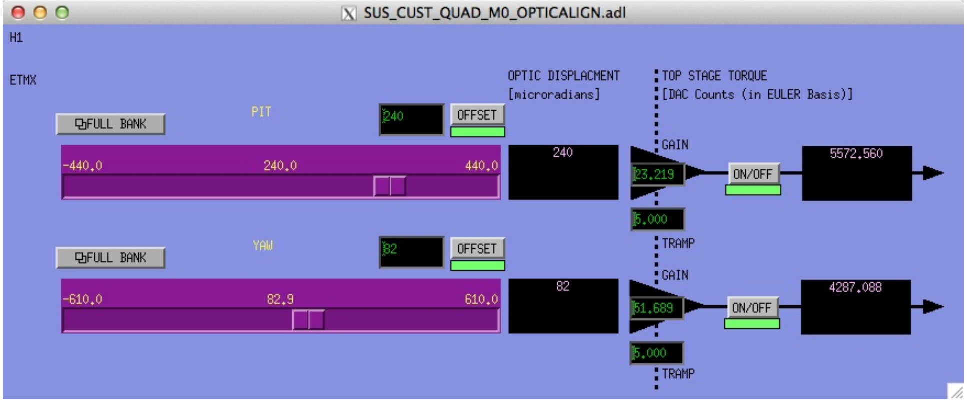
- T1100232: aLIGO Damping Loop Design and Performance
- G1300537: aLIGO QUAD "Level 2" Damping Loop Design
- G1300561: aLIGO BSFM "Level 2" Damping Loop Design
- G1300621: aLIGO TMTS "Level 2" Damping Loop Design
- G1400151: aLIGO HLTS "Level 2" Damping Loop Design

aLIGO: Suspensions/OpsManual/Damping (last edited 2014-04-29 15:38:03 by MarkBarton)

aLIGO SUS Ops Manual - Alignments

Back to Operation Manual main page

Each suspension has a filter block and associated screen which is used to apply DC offsets in pitch and yaw via the top level OSEMs. The QUAD main chain alignment screen is representative:



The sliders and the associated offset field allow actuation requests in microradians at the optic. (The optic tracks the top mass exactly in yaw, but in pitch the optic moves less than the optic by a factor of order 2 depending on the suspension type.)

The `GAIN` fields contain appropriate values to convert microradians to counts of actuation and should not normally be touched.

The slider ranges have been set to match the maximum actuation that can be applied given the drive capability of the electronics and the actuators, and the pendulum compliance.

The alignments should normally be turned on and off with the `OFFSET` buttons rather than the `ON/OFF` buttons, because this takes advantage of the ramping. A good `TRAMP` value that avoids triggering watchdogs is 5 seconds.

See T1400293 for information on the calculation of the default gains and offsets.

aLIGO: Suspensions/OpsManual/Alignments (last edited 2014-05-27 09:08:35 by MarkBarton)

aLIGO SUS Ops Manual - Projections

Back to Operation Manual main page

This page gives information on the numbers used in various transformation matrices in the MEDM screens.

OSEM2EUL, EUL2OSEM

Information on the appropriate lever arms and averaging to use when converting OSEM signals to and from the Euler basis (e.g., F1/F2/F3/LF/RT/SD <-> L/T/V/R/P/Y for quad M0) is currently scattered about. For now see, the Matlab scripts in the SUS SVN that are used in the various suspension-specific plotting commands:

- QUAD: `^/trunk/QUAD/Common/MatlabTools/make_susquad_projections.m`
- BSFM: `^/trunk/HLTS/Common/MatlabTools/make_susbsfm_projections.m`
- HLTS: `^/trunk/HLTS/Common/MatlabTools/make_sushlts_projections.m`
- HSTS: `^/trunk/HSTS/Common/MatlabTools/make_sushsts_projections.m`
- OMCS: `^/trunk/OMCS/Common/MatlabTools/make_susomcs_projections.m`
- TMTS: `^/trunk/TMTS/Common/MatlabTools/make_sustmts_projections.m`
- HAUX: `^/trunk/HAUX/Common/MatlabTools/make_sushaux_projections.m`
- HTTS: `^/trunk/HTTS/Common/MatlabTools/make_sushtts_projections.m`

Note that the appropriate EUL2OSEM matrix for the actuation is always the transpose (not inverse) of the corresponding OSEM2EUL matrix.

The same numbers have also been added to the sustools.py module in the userapps SVN:

- `^/trunk/sus/common/scripts/sustools.py`

The numbers in the above were mostly derived from the top assembly drawings:

- QUAD: D0901346
- BSFM: D1000392
- HLTS: D070447
- HSTS: D020700
- OMCS: D0900295
- TMTS: D0901880
- HAUX: D1000120
- HTTS: D1001396

CART2EUL

The matrices for transforming from the associated SEI coordinate system to the SUS one have been calculated by Jeff Kissel:

- T1100617 Transformation Matrices Between SEI/SUS Coordinates

aLIGO SUS Operations Manual - Computing Tools

Back to Operation Manual main page

Mathematica/Matlab Models	Dynamical models of the mode shapes and frequencies of the suspensions
Matlab TF scripts	to measure and plot transfer functions
Projection info and scripts	to set OSEM2EUL and other matrices
Alignment slider info and scripts	to set OSEMALIGN gains
sustools.py	a Python module and command line tool for manipulating suspensions
Guardian	a suite of Python programs to manage the IFO generally and SUS specifically
cdscfg	Tools to set up the computing environment
BURT	Backup and restore MEDM/EPICS settings
FOTON	Edit filters
MEDM	Edit user interface screens
Simulink	Edit/build RTCDS Simulink models
ODC	Stuff on ODC
SVN Repository	SUS code/data storage and version management
DTT	Diagnostic Test Tools
dataviewer	Graph signals in real time or retrospectively
diag	All sorts of stuff, esp. monitor/clear test points
Network	Network topology and remote login information

aLIGO: Suspensions/OpsManual/Computing (last edited 2014-04-25 10:14:59 by MarkBarton)

aLIGO SUS Operations Manual - Python Tools

Back to Operation Manual main page

sustools.py

A Python module/script `/opt/rtcds/userapps/release/sus/common/scripts/sustools.py` provides many functions for reading and writing EPICS settings for the suspensions.

It has been written to support three modes of operation:

- As a standalone utility callable from the command-line or via the system() function in Matlab.
- As a Python module that can be imported into any Python script.
- As a Python module for use with Jamie Rollins' Guardian system.

The master copy of `sustools.py` lives at `/opt/rtcds/userapps/release/sus/common/scripts/`. The file `sustoolsdev.py` in the same directory is a development version which may have more advanced features but may not compile either. The shell script `sustools_unittest` is a set of unit tests for it (in its usage mode as a command line utility).

To make it easy for Guardian to find, there should be a symlink to it at `/opt/rtcds/userapps/release/sus/common/guardian/`. It uses the Python `ezca` module for channel access. Guardian also uses this so it should be available if Guardian has been installed, but if not it will have to be installed manually and the directory where it lives added to the shell variable `$PYTHONPATH`. If run under OS X it uses `/opt/rtcds/userapps/release/sus/common/scripts/fakeezca.py`, which prints diagnostics but does not attempt to do actual channel access.

To make `sustools.py` available as a module within Python, `/opt/rtcds/userapps/release/sus/common/scripts/` needs to be added to the shell variable `$PYTHONPATH` before running Python, or to `sys.path` from within Python before importing. To make `sustools.py` available as a shell command, `/opt/rtcds/userapps/release/sus/common/scripts/` should be added to `$PATH`, and the execute bit should be set with `chmod +x sustools.py` (in the unlikely event that it ever comes unset). These directories will probably be made a standard part of the `cdscfg` system before long.

Usage within a standalone Python script

The module defines a class `Sus` which should be instantiated with a string giving the name of a suspension to create an object for accessing that suspension:

```
import sustools
itmx = sustools.Sus('ITMX')
result = itmx.read('M0_DAMP_L_GAIN')
```

By default the interferometer name is read from the shell variable `$IFO`. After initialization, methods that take channel names expect them to be specified without the initial prefix, `H1:SUS-ITMX_` or the like.

Usage within a Guardian script

Guardian scripts for suspensions are structured as Python class definitions that inherit from Jamie's base class `GuardState`. In this usage, the `Sus` class should instead be initialized with the ezca object:

```
from guardian import GuardState
import sustools

class SAFE(GuardState):
    goto = True
    def main(self):
        self.optic = sustools.Sus(ezca)
        self.optic.dampOutputSwitchWrite('OFF')
        ...
```

Usage as a command-line tool

As a shell tool, it takes one positional argument, which represents a command and is typically the name of one of the methods of the `Sus` class object. (To be invoked from the command line, a method needs to be declared in one the lists with names like `callableGlobals`, `callableSusFunctions` etc in the `sustools.py` code.) Arguments to the method are supplied as switches. It's not required but saves accidents to put the command argument first, lest it get eaten by one of the switches that accept multiple arguments.

Switch	example	meaning
`-o`	`-o ITMX`	optic - required for nearly all commands
`-i`	`-i H1`	ifo - defaults to value of shell variable IFO
`-l`	`-l M0 R0`	level (SAG) - defaults to `[]` meaning all
`-c`	`-c F1 F2 F3`	(input) channel (or DOF) - defaults to `[]` meaning all
`-C`	`-C L T V`	output channel (or DOF), for matrix blocks
`-f`	`-f 1 3 5`	filters (in cdsFilt blocks)
`-e`	`-e`	enable (a switch or filter)
`-d`	`-d`	disable (a switch or filter)
`-n`	`-n`	no change (a switch or filer)
`-v`	`-v 1.0`	value
`-a`	`-a 1.0 2.0 3.0`	array of values
`-k`	`-k QUAD M0 damp`	key - list of dictionary keys for data lookup commands
`-s`	`-s _EXC`	suffix to be appended by commands that return PV lists)
`-m`	`-m`	Matlab friendly output (everything as nested cell arrays)
`-x`	`-x "2+2"`	execute arbitrary Python code before main command
`-X`	`-X "2+2"`	execute arbitrary Python code after main command; result will be in `result`
`-b`	`-b`	bare - leave off the `H1:SUS-ITMX_` or similar prefix in PV list commands
`-B`	`-B`	half-bare - leave off the `H1:SUS-` but include the `ITMX_` in PV list commands
`--verbose`	`--verbose`	enable debugging output

Examples:

Print the data structure `susData` which has the reference information about different suspension types:

```
controls@opsws1:scripts $ ./sustools.py susData
{'TMTSR': {'levelorder': ['M1'], 'levels': {'M1': {'osemConfig': {'magnet': {'diameter': 10,
'length': 10, 'material': 'NdFeB', 'force': 1.694},
...
```

Return the suspension type of an ITMx

```
controls@opsws1:scripts $ ./sustools.py suspensionType -o ITMx
QUAD
```

Return DAMP block channels for ITMx:

```
controls@opsws:scripts $ ./sustools.py dampPvs -o ITMx
['H1:SUS-ITMX_M0_DAMP_L', 'H1:SUS-ITMX_M0_DAMP_T', 'H1:SUS-ITMX_M0_DAMP_V', 'H1:SUS-
ITMX_M0_DAMP_R', 'H1:SUS-ITMX_M0_DAMP_P', 'H1:SUS-ITMX_M0_DAMP_Y', 'H1:SUS-ITMX_R0_DAMP_L',
'H1:SUS-ITMX_R0_DAMP_T', 'H1:SUS-ITMX_R0_DAMP_V', 'H1:SUS-ITMX_R0_DAMP_R', 'H1:SUS-
ITMX_R0_DAMP_P', 'H1:SUS-ITMX_R0_DAMP_Y']
```

Return DAMP block channels for just level M0 and DOFs L, P and Y of ITMx:

```
controls@opsws1:scripts 0$ ./sustools.py dampPvs -o ITMx -l M0 -c L P Y
['H1:SUS-ITMX_M0_DAMP_L', 'H1:SUS-ITMX_M0_DAMP_P', 'H1:SUS-ITMX_M0_DAMP_Y']
```

Return DAMP block channels for ITMx with no `H1:SUS-ITMX_` prefix but a _EXC suffix:

```
controls@opsws1:scripts $ ./sustools.py dampPvs -o ITMx -b -s _EXC
['M0_DAMP_L_EXC', 'M0_DAMP_T_EXC', 'M0_DAMP_V_EXC', 'M0_DAMP_R_EXC', 'M0_DAMP_P_EXC',
'M0_DAMP_Y_EXC', 'R0_DAMP_L_EXC', 'R0_DAMP_T_EXC', 'R0_DAMP_V_EXC', 'R0_DAMP_R_EXC',
'R0_DAMP_P_EXC', 'R0_DAMP_Y_EXC']
```

Return "witness" channels (from DAMP and WIT blocks) for ITMx with a _DQ suffix in Matlab-friendly format:

```
controls@opsws1:scripts $ ./sustools.py witPvs -o ITMx -m -s _DQ
{'H1:SUS-ITMX_M0_DAMP_L_IN1_DQ' 'H1:SUS-ITMX_M0_DAMP_T_IN1_DQ' 'H1:SUS-ITMX_M0_DAMP_V_IN1_DQ'
'H1:SUS-ITMX_M0_DAMP_R_IN1_DQ' 'H1:SUS-ITMX_M0_DAMP_P_IN1_DQ' 'H1:SUS-ITMX_M0_DAMP_Y_IN1_DQ'
'H1:SUS-ITMX_R0_DAMP_L_IN1_DQ' 'H1:SUS-ITMX_R0_DAMP_T_IN1_DQ' 'H1:SUS-ITMX_R0_DAMP_V_IN1_DQ'
'H1:SUS-ITMX_R0_DAMP_R_IN1_DQ' 'H1:SUS-ITMX_R0_DAMP_P_IN1_DQ' 'H1:SUS-ITMX_R0_DAMP_Y_IN1_DQ'
'H1:SUS-ITMX_L1_WIT_L_DQ' 'H1:SUS-ITMX_L1_WIT_P_DQ' 'H1:SUS-ITMX_L1_WIT_Y_DQ' 'H1:SUS-
ITMX_L2_WIT_L_DQ' 'H1:SUS-ITMX_L2_WIT_P_DQ' 'H1:SUS-ITMX_L2_WIT_Y_DQ'}
```

aLIGO: Suspensions/OpsManual/PythonTools (last edited 2014-05-27 09:07:41 by MarkBarton)

aLIGO SUS Operations Manual - Info on cdscfg utilities

Back to Operation Manual main page

The directory `/ligo/cdscfg` contains utilities to set up a standard environment in several shells and programming environments including bash, csh, Perl and TCL. These are documented further in T1000379

Site/IFO Values

The directories `/ligo/cdscfg/site` and `/ligo/cdscfg/ifo` (on some of the front ends it is `/opt/cdscfg/site` etc) should each contain single files whose names specify the site and interferometer according to the following table:

Location	${site}	${ifo}
Caltech – 40m	cit	c1
GEO	geo	g1
LHO Interferometers	lho	h1
	lho	h2
LLO Interferometer	llo	l1
MIT - LASTI	mit	m1
Stanford – Engr. Test Facility	stn	s1
Scratch test stands	tst	x0
LHO Test Stands	tst	x1
LLO Test Stands	tst	x2
Caltech Test Stands	tst	x3

Shell Variables

cdscfg makes available the following shell variables which should be used in scripts.

Variable	Definition
SCRIPTDIR	Directory for IFO scripts
MEDMDIR	Top-level directory for MEDM screens
TARGETDIR	Top-level directory for front-end executables
CHANSDIR	Top-level directory for filters, channel lists
SYSBIN	PATH to system executables
SYSLIB	LD_LIBRARY_PATH to system libraries
EPICSBIN	PATH to EPICS executables
EPICSLIB	LD_LIBRARY_PATH to EPICS libraries
GDSDIR	Top-level GDS directory
GDSBIN	PATH for GDS executables
GDSLIB	LD_LIBRARY_PATH to GDS libraries
MINPATH	Minimal PATH (PATH is set to this)
MINLIB	Minimal LD_LIBRARY_PATH (LD_LIBRARY_PATH is set to this)
MA TDIR	Top-level MATLAB directory
LIGONDSIP	local NDS server for ifo
SITE	LIGO site (upper-case)

site	ligo site (lower-case)
IFO	IFO label (upper-case)
ifo	ifo label (lower-case)
IFOLET	IFO letter from label (upper-case)
IFONUM	IFO number from label
ASC, GDS, HPI, IOO, IMC, ISC, ISI, LSC, OMC, PEM, PSL, SEI, SUS, SYS, TCS, VE	IFO-specific channel prefix for EPICS (i.e. L1:ASC)

aLIGO: Suspensions/OpsManual/cdscfg (last edited 2014-05-27 09:07:13 by MarkBarton)

aLIGO SUS Operation Manual - Info on BURT

Back to Operation Manual main page

Contents

Background

BURT is a tool for backing up and restoring some or all of the EPICS state of one or more models. There is a GUI interface, `burtgooey` which invokes two command line tools, `burtwb` and `burtrb`. The `burtrb` tool needs a BURT request file (`.req`) specifying a list of channels to be backed up and generates a BURT snapshot file (`.snap`) containing a list of channel names and saved values. The `burtwb` tool then loads the values back into the model. Each model should have a `autoBurt.req` file specifying a default list of channels to be backed up (not all will be of interest). A background task backs up all models according to their `autoBurt.req` files at regular intervals (currently hourly), but additional snapshots can be taken manually. The kill scripts which shut down models in an orderly fashion (e.g., `killh1susmc2` for H1:MC2) do a backup first. (Startup scripts, e.g., `starth1susmc2` for H1:MC2, invoke the corresponding kill script first and thus also trigger a backup.)

Each model should have a `safe.snap` file representing a safe state in which the suspension or other system is least likely to be a danger to itself or the rest of the system. Each model is automatically restored to the `safe.snap` state after a reboot of the front-end computer. The "safe" state is currently defined as having the damping filters ON but the master switch and most other sources of excitation OFF.

The Guardian system will use additional files similar to `safe.snap` to define standard states. The `lock.snap` files define the desired state for locking the cavity of which the suspension is a part, with the damping, master switch, alignment offsets and locking filters all on. **Need to check this!!!**

Directory structure

Many BURT related utilities put or expect to find BURT-related files at

`/opt/rtcds/${site}/${ifo}/target/` , e.g., `/opt/rtcds/lho/h1/target/` .

These are organized by the name of the Simulink model controlling the suspension:

`/opt/rtcds/${site}/${ifo}/target/${model}/${model}epics/`, e.g., for H1:MC2, `/opt/rtcds/lho/h1/target/h1susmc2/h1susmc2epics/` .

The `autoBurt.req` file is

`/opt/rtcds/${site}/${ifo}/target/${model}/${model}epics/autoBurt.req` , e.g. for H1:MC2, `/opt/rtcds/lho/h1/target/h1susmc2/h1susmc2epics/autoBurt.req`.

The `safe.snap` file used when a model is started is

`/opt/rtcds/${site}/${ifo}/target/${model}/${model}epics/burt/safe.snap` e.g., for H1:MC2, `/opt/rtcds/lho/h1/target/h1susmc2/h1susmc2epics/burt/safe.snap` .

However by convention, all `safe.snap` files are now symbolic links into the SVN-controlled area at `/opt/rtcds/userapps/release/sus/${ifo}/burtfiles/`, e.g,

`/opt/rtcds/lho/h1/target/h1susmc2/h1susmc2epics/burt/safe.snap` -> `/opt/rtcds/userapps/release/sus/burtfiles/h1susmc2_safe.snap`

Because BURT (`burtgooey` etc) is not smart enough to follow a symbolic link and write to the location pointed to (rather, it replaces the link with a plain data file), it is important always to give it a explicit path into the `userapps` area so as not to break the link.

Snapshot files generated by kill scripts are at

`/opt/rtcds/${site}/${ifo}/target/${model}/${model}epics/burt` , e.g., for H1:MC2, `/opt/rtcds/lho/h1/target/h1susmc2/h1susmc2epics/burt/` .

with names like `h1susmc2_burt_121204_124643.snap` (i.e., with a yymmdd_hhmmss date/time stamp).

Hourly backups live at `/ligo/cds/${site}/${ifo}/burt` organized by year, month, day, time, and model, e.g.,

`/ligo/cds/lho/h1/burt/2013/01/04/09:00/h1susmc2epics.snap`.

Saving snapshots

The following sections explain several different tools for making snapshots.

Updating a safe.snap file via Matlab

There is a Matlab function `save_safe_snap()` to make new `safe.snap` files conveniently. (Currently there is no equivalent `save_lock_snap()`, but stay tuned.) It saves the current state of a suspension, puts it into the safe state, saves the state again with the appropriate name in the `userapps` working copy area, restores the original state and prints an svn command that can be used to commit the new file to the `userapps` repository.

1. Run Matlab and navigate to the `^trunk/Common/MatlabTools directory.

   ```
   $ matlab&
   > cd /ligo/svncommon/SusSVN/sus/trunk/Common/MatlabTools
   ```

2. Invoke save_safe_snap() with the desired ifo and suspension.

   ```
   > save_safe_snap('H1','ITMY')
   ```

3. Copy and paste the svn command that is printed by the function and paste it into the Matlab window. (Note the use of `!` to execute shell commands within Matlab - omit it if you prefer to use a regular terminal window.)

   ```
   > !svn commit -m "New safe.snap file for H1:ITMY"
   /opt/rtcds/userapps/release/sus/h1/burtfiles/h1susitmy_safe.snap
   ```

Creating a snapshot manually

1. Put the suspension in the desired state. For `safe.snap` files, check the following:
   - [**DAMP**] filter outputs should be **ON**.
   - [**MASTER_SWITCH**] should be **OFF**.
   - [**TEST**] filter outputs should be **OFF**.
   - [**LOCK**] filter outputs should be **OFF**.
   - [**OPTICALIGN**] filter outputs should be **OFF**.

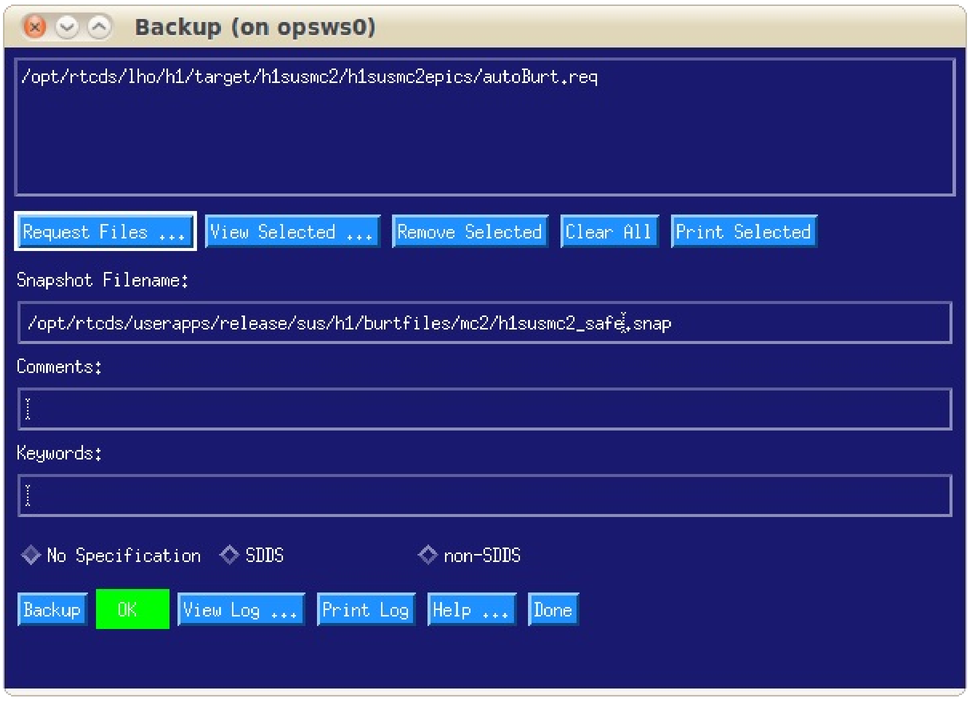
   For `lock.snap` files check the following [**Need to check this!**]:
   - [**DAMP**] filter outputs should be **ON**.
   - [**MASTER_SWITCH**] should be **ON**.
   - [**TEST**] filter outputs should be **OFF**.
   - [**LOCK**] filter outputs should be **???**.
   - [**OPTICALIGN**] filter outputs should be **ON**.
2. Open `burtgooey` in the command window:

   ```
   $ burtgooey &
   ```

3. Click [**Backup**]. In the new window, click [**Request Files**]. The Request File Selector dialog appears.
4. Navigate to the `/opt/rtcds/${site}/${ifo}/target/` area for your site and ifo, and go down two levels to locate the `autoBurt.req` file for your suspension, e.g.

levels to locate the `autoBurt.req` file for your suspension, e.g., `/opt/rtcds/lho/h1/target/h1susmc2/h1susmc2epics/` for H1 MC2. Click on `autoBurt.req`, then [**OK**] and [**Cancel**] (the UI allows for adding more than one file, hence the need for [**Cancel**]).

5. In the [**Snapshot Filename**] field, enter in **THE ENTIRE PATH** and desired filename for the snapshot. For `safe.snap` and `lock.snap` files, this should be in the part of the `userapps` directory containing the suspension's individual `safe.snap` file, e.g., `/opt/rtcds/userapps/release/sus/h1/burtfiles/h1susmc2_safe.snap`.



6. Click the [**Backup**] then the [**Done**] buttons.
7. If the snapshot is in the `userapps` SVN directory sure to commit the newly-written file:

```
$ cd /opt/rtcds/userapps/release/sus/h1/burtfiles/
$ svn commit -m "New safe.snap for MC2"
```

Restoring from snapshots

Restoring to a snap.snap file with the GDS_TP screen

1. Open the SITEMAP.
2. Open the screen for the suspension.
3. Click on the button which opens the GDS TP screen for the user model, e.g., for H1:MC2, [H1SUSMC2 GDS TP].
4. Click on the [BURT] button.

Restoring from an arbitrary saved state with the BURT GUI

1. Open `burtgooey` in the command window:

```
$ burtgooey &
```

2. Click on [**Restore**]. The Restore window appears.
3. Click on [**Snapshot Files ...**]. The Snapshot File Selector dialog appears.
4. Navigate to the directory with the backup file
   - For kill script backups, navigate to `/opt/rtcds/${site}/${ifo}/target/${model}/${model}epics/burt/` e.g., for H1 MC2, `/opt/rtcds/lho/h1/target/h1susmc2/h1susmc2epics/burt/`.
   - For hourly backups, navigate to `/ligo/cds/${site}/${ifo}/burt/${YYYY}/${MM}/${DD}/${hh}:{mm}/`
5. Identify and a select a file from a known good time, then click [**OK**] and [**Cancel**] (the UI allows for adding more than one file, hence the need for [**Cancel**]).
   - Kill script snapshots have a date/time in yymmdd_hhmmss format at the end of their name, e.g.,

h1susmc2_burt_121204_124643.snap.
    - Hourly backups are called ${model}epics.snap, e.g, for H1:MC2, h1susmcsepics.snap.

6. Check that the file selected in the previous step is the only one in the list. If not clear unwanted files from the list with [**Remove Selected**].
7. Click [**Restore**]. After a few seconds the indicator immediately to the right should turn green and display "OK". If there is a problem (the indicator turns red and says "Not OK"), in which clear the file from the list and choose a better one.

Restoring from an hourly backup

The burtrestore utility is a convenient way of restoring to an hourly backup. It takes the hourly snapshot file name for a particular model (i.e., the name of the model plus "epics") and as many of time, day, month and year as are necessary:

```
$ burtrestore h1susmc2epics 09:00            % restore to 9 o'clock this morning
$ burtrestore h1susmc2epics 09:00 02         % restore to 9 o'clock on the 2nd of this month
$ burtrestore h1susmc2epics 09:00 02 01      % restore to 9 o'clock on the 2nd of January this
year
$ burtrestore h1susmc2epics 09:00 02 01 2012 % restore to 9 o'clock on 1/2/12
```

aLIGO: Suspensions/OpsManual/BURT (last edited 2013-11-13 15:52:13 by ArnaudPele)

aLIGO SUS Ops Manual - Info on Foton and filter files

Back to Operation Manual main page

The `CDSFilt` part can used in a model to represent a bank of ten filters. When the model is compiled and run, filter definitions for all the `CDSFilt` blocks are looked for in `/opt/rtcds/${site}/${ifo}/chans/${MODEL}.txt`, e.g., `/opt/rtcds/lho/h1/chans/H1SUSMC2.txt`.

Currently there is no consistent system for the use of symbolic links. Some of the `.txt` files are symlinks to files of the same name in `/opt/rtcds/release/sus/${ifo}/filterfiles/`, whereas others are plain files.

Foton follows symbolic links consistently, so there should be no problems with wiping out a link in the process of editing it.

aLIGO: Suspensions/OpsManual/FOTON (last edited 2014-05-27 09:06:53 by MarkBarton)

Information on MEDM screens for Suspensions

Back to Operation Manual main page

MEDM is a system for creating user interface screens that can display and set EPICS channels. It provides for many standard UI elements including read-only and read-write text fields, dials, sliders, buttons, etc. See the MEDM documentation for detailed info. Screens are defined by text files with extension `.adl`. The `medm` program allows screens to be edited in a GUI which is approximately WYSIWYG (what-you-see-is-what-you-get). (However due to limited support for cut and paste, some changes are more easily made with a plain text editor such as `gedit`.)

MEDM supports generic screens, which are customized on the fly by arguments passed in by the screen which invokes them. See [[#CallingConventions]|Calling Conventions] below for information on the recommended argument list for SUS screens. The `medm` main window has an Execute button which is meant to make a newly created/edited screen live and start it displaying EPICS data. However because of the extensive use of generic screens, it is rarely much use. Rather it is necessary to start at the so-called SITEMAP screen for a site in a separate, execute-only instance of `medm` and click through to the screen being edited. See below.

Screens of Importance

SITEMAP

The SITEMAP is the master MEDM screen for an entire site. It can be accessed with the `sitemap` command, or via various GUI shortcuts. It has a large number of buttons linking to screens for individual suspensions and other subsystems of interest.

The traditional location for the SITEMAP at `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. The terminal command `sitemap` is defined to open the screen at that location. However it should not be edited directly because it is a symlink to a version-controlled file at `/opt/rtcds/userapps/release/cds/${ifo}/medm/SITEMAP.adl`.

The version-controlled copy should only be edited with extreme care and following consultation with CDS personnel. Most other screens cannot usefully be opened *except* directly or indirectly from the SITEMAP, because they rely on macro arguments passed in, so breaking the SITEMAP breaks MEDM for everyone.

As of 4/4/13 at LLO and 4/30/13 at LHO, most of the macro arguments for SUS screens were offloaded from the SITEMAP.adl file to separate per-suspension files (see below). (However the location of those files is still specified in the SITEMAP.)

For more, see HowToEditTheSitemapMedmScreen

Custom Screens

Most SUS MEDM screens are hand-made custom ones which are generic to all suspensions of the same type (QUAD/BSFM/etc) and are particularized to an individual suspension by macro arguments. The common screens for most suspension types live at `/opt/rtcds/userapps/release/sus/common/medm/` in subdirectories `quad`, `bsfm`, `hxts`, `omcs`, `tmts` and `haux`.

Tip-Tilts (HTTS) are owned by ASC and files for them live at `/opt/rtcds/userapps/release/asc/common/medm/asctt` (note: `asc`).

HSTS and HLTS are similar enough to be handled by a single set of "HXTS" screens.

The macro files live in the top-level directory `/opt/rtcds/userapps/trunk/sus/common/medm` and have names of the form `sus${susid}_overview_macro.txt`, e.g., `susetmx_overview_macro.txt`.

Automatically Generated Screens (especially *_GDS_TP and *_IOP_TP)

When a model is compiled, a large number of MEDM screens are automatically generated by RCG and placed in directories

`/opt/rtcds/${site}/${ifo}/medm/${site}sus${susid}`

e.g.,

`/opt/rtcds/lho/h1/medm/h1susmc2/`

Most of the custom screens were created originally by cutting and pasting elements from the automatically generated screens into more convenient arrangements.

Of the automatically generated screens about the only one that is normally used as-is is the GDS_TP screen, which gives an overview of housekeeping info like the CPU ID of the computer running the model, the loop time, the GPS time (as reported by the model), the ADC, DAC and test point status, etc:

`/opt/rtcds/${site}/${ifo}/medm/${site}sus${susid}/${SITE}SUS${SUSID}_GDS_TP.adl`

`/opt/rtcds/lho/h1/medm/h1susmc2/H1SUSMC2_GDS_TP.adl`

The IOP_TP screen gives similar information for the IOP process running on the same computer, and can be found in the automatically generated screens for the IOP model, e.g.,

`/opt/rtcds/lho/h1/medm/h1iopsush34/H1IOPSUSH34_GDS_TP.adl` for `h1susmc2`.

See the `[Detailed Overview]` link on the SITEMAP for info on which IOP model corresponds to which suspension model.

Calling Conventions

An MEDM screen can allow for the opening of a second one by having a "Related Display" button (or a popup menu of them). In the dialog that allows setting the properties of the button/popup, there are fields "Display File" and "Arguments". These show up in the underlying .adl file as a pair of lines that look like

```
name="$(USERAPPS)/sus/common/medm/hxts/SUS_CUST_HXTS_MONITOR_OVERVIEW.adl"
args="%(read
$(USERAPPS)/sus/common/medm/susmc1_overview_macro.txt),USERAPPS=$(USERAPPS),SITE=$(SITE),si
te=$(site),IFO=$(IFO),ifo=$(ifo)"
```

Note:

- Here and in the sections below, we use the text-editor view.
- The `$(macro)` syntax represents substitution of the value of the macro variable `macro`.
- The `macro=value` syntax represents macro values being passed to a related screen.
- The related screen inherits no macros by default, and any that need to be passed unchanged need to be specified explicitly with `IFO=$(IFO)` or the like.
- The macro names and values are case sensitive, and often come in pairs with different capitalization, e.g., `IFO=H1` and `ifo=h1`.
- $(USERAPPS) is a macro defined as the path to the `release` section of the checkout of the cds_user_apps SVN, `/opt/rtcds/userapps/release`.
- Most screens are suspension-type-specific and live in the following directories (some chamber-specific screens, e.g., `SUS_AUX_EX_OVERVIEW.adl` live one level above):
  - `$(USERAPPS)/sus/common/medm/quad`
  - `$(USERAPPS)/sus/common/medm/bsfm`
  - `$(USERAPPS)/sus/common/medm/hxts`
  - `$(USERAPPS)/sus/common/medm/omcs`
  - `$(USERAPPS)/sus/common/medm/tmts`
  - `$(USERAPPS)/sus/common/medm/haux`
  - `$(USERAPPS)/asc/common/medm/asctt` (note: `asc`)
- Most screens are generic to a whole suspension type and are particularized for individual suspensions via a combination of macro text files and `macros=value`.
- The `%(read ...)` syntax in the `args="..."` line reads argument values from a text file.
- The macro files live at:
  - `$(USERAPPS)/sus/common/medm/` for SUS except TipTilt
  - `$(USERAPPS)/asc/common/medm/` for TipTilt
- The macro files have names like
  - `sus*_overview_macro.txt` for SUS except TipTilt, *=name of suspension, e.g., `itmx`
  - `asc*_overview_macro.txt` for TipTilt, *=name of suspension, e.g., `rm1`
- The macro files are site-independent and do not contain values for `SITE`/`site`/`IFO`/`ifo`. These need to be passed in explicitly.

in explicitly.

- Currently `USERAPPS` is the only macro guaranteed to be defined in SITEMAP.adl (it is passed in with the `-macro` argument of `medm` by the `sitemap` alias). `SITE`/`site`/`IFO`/`ifo` have to be given explicit values (`LHO`/`lho`/`H1`/`h1` or the like). There is a proposal afoot to add more macros to the standard set, but since the SITEMAPs are highly site-specific in any case, it's not a big deal.

From the SITEMAP to a generic screen particularized for a single suspension

SUS screens in the SITEMAP should be called per the following example (the main screen for ITMx):

```
name="$(USERAPPS)/sus/common/medm/quad/SUS_CUST_QUAD_OVERVIEW.adl"
args="%(read
$(USERAPPS)/sus/common/medm/susitmx_overview_macro.txt),USERAPPS=$(USERAPPS),SITE=LHO,site=
lho,IFO=H1,ifo=h1"
```

From the SITEMAP to a summary screen for all suspensions of the same type

Some suspension types like HAUX have summary screens showing an overview of all suspensions of that type. These have special macro text files with only the common information:

```
name="$(USERAPPS)/sus/common/medm/haux/SUS_CUST_HAUX_OVERVIEW_all.adl"
args="%(read
$(USERAPPS)/sus/common/medm/susimall_overview_macro.txt),USERAPPS=$(USERAPPS),SITE=LHO,site
=lho,IFO=H1,ifo=h1"
```

From one generic suspension screen to another

Copy the following example (`SUS_CUST_HSTS_OVERVIEW.adl`, with say, `optic=mc1`, is calling `SUS_CUST_HXTS_MONITOR_OVERVIEW.adl`):

```
name="SUS_CUST_HXTS_MONITOR_OVERVIEW.adl"
args="%(read
$(USERAPPS)/sus/common/medm/sus$(optic)_overview_macro.txt),USERAPPS=$(USERAPPS),SITE=$(SIT
E),site=$(site),IFO=$(IFO),ifo=$(ifo)"
```

Note:

- Since the called screen lives in the same directory, the path information can be omitted.
- In addition to the suspension-specific macros in the macro file, `USERAPPS` and the four standard site-specific macros need to be passed in unchanged.

From a generic suspension screen to a summary screen

Some suspension types like HAUX have summary screens showing an overview of all suspensions of that type. These have special macro text files with only the common information:

```
name="SUS_CUST_HAUX_OVERVIEW_all.adl"
args="%(read
$(USERAPPS)/sus/common/medm/susmc1_overview_macro.txt),USERAPPS=$(USERAPPS),SITE=$(SITE),si
te=$(site),IFO=$(IFO),ifo=$(ifo)"
```

Note:

- In the usual case that the called screen lives in the same directory, the path information can be omitted.
- In addition to the suspension-specific macros in the macro file, `USERAPPS` and the four standard site-specific macros need to be passed in unchanged.

aLIGO SUS Operations Manual - Developing models in Simulink

Back to Operation Manual main page

Software running on front end computers is written in Simulink per T080135 AdvLigo CDS Realtime Code Generator (RCG) Application Developer's Guide.

Simulink models live in the `userapps` repository, which by convention is checked out as `/opt/rtcds/userapps`. SUS models can be found at

> `/opt/rtcds/userapps/release/sus/${ifo}/models`, e.g,
> `/opt/rtcds/userapps/release/sus/h1/models` (`sus->asc` for Tip-Tilt/HTTS)

Only certain computers (`h1build`, `l1build` etc) are configured for building models from the Simulink source - they also have the appropriate release of the `rtbuild` system linked to as

> `/opt/rtcds/${site}/${ifo}/release`

and `rtbuild` knows to look in `userapps` for models.

Models currently running

The most recent rebuild of each Simulink model is archived at one of the following locations and can be accessed from outside the CDS network with ligo.org credentials:

- Snapshots of LLO models
- Snapshots of LHO models

Building models

1. Get advice from the CDS system manager (Dave Barker or Keith Thorne) on what other parts of the CDS system are likely to be affected. For example:
   1. Restarting a SUS model may interfere with commissioning of the physical suspension, or the ISI that it is on, or the beam path that it is a part of.
   2. Restarting a SUS model may interfere with SEI or ISC models that receive IPC signals from it.
   3. If new channels are to be added, it may be necessary to restart the DAC, which is disruptive to people needing trend data.
2. Check with users who may be working on the suspension or may be otherwise affected and confirm that it is OK to proceed.
3. Open a work permit and get all necessary signoffs.
4. Open a fresh terminal window and log into a build machine (`h1build`, `l1build` etc).

   ```
   $ ssh h1build
   ```

5. Go to the build directory for the most recent release of rtcds.

   ```
   $ cd  /opt/rtcds/lho/h1/release % or the like
   ```

6. Do a `make` with the model name less `.mdl`:

   ```
   $ make h1susbs
   ```

7. Diagnose and correct any errors. If the model depends on IPC from other models that have not yet been compiled, compile those at this point and try again.
8. Do a `make install-` with the model name less `.mdl`:

   ```
   $ make install-h1susbs
   ```

9. Open the CDS Overview screen (SITEMAP->CDS->Overview at LHO, ??? at LLO), and identify the computer that runs the model.

10. Click on the associated button to open the GDS TP screen for the model.
11. Also open the main control screen for the suspension.
12. Let potentially affected users know that the model is about to be restarted and check that this is still OK.
13. Alert the operator that you are about to restart a model, and which one, so that the alarm generated in the next step does not come as a surprise.
14. Open a second fresh terminal window and log into the front end that is to run the model and (re)start the model from the newly compiled version.

```
$ ssh h1susb123
$ starth1susbs
```

15. Check that the GDS TP screen and the main MEDM screen for the suspension go white briefly and then recovers after a few tens of seconds.
16. **As soon as the GDS TP screen recovers, click the `BURT` button**.
17. Check that the line for the model on the CDS Detailed Overview screen goes white briefly and then recovers (this normally takes a few tens of seconds longer to come back than the MEDM screen for the suspension).
18. Check that the terminal window says that the `safe.snap` file has been loaded.
19. If the `safe.snap` file is not known to be up to date, open a third fresh terminal window (on one of the ops workstations) and, without logging onto anywhere else, load a more recent BURT snapshot, e.g., the most recent hourly backup:

```
$ burtrestore h1susbsepics 09:00
```

20. If everything looks OK on the MEDM screen for the suspension, reenable the master switch, the watchdogs and the damping.
21. On the CDS Detailed Overview screen, press the `[DR]` button to clear any transitory errors, and debug any that persist.
22. When all looks OK, close the work permit and do an alog describing the changes.

Inserting SVN revision numbers

Note that Simulink models can have a macro string in a text box which inserts the SVN revision number when the model is compiled, so it can be displayed later and serve as documentation of which revision was most recently compiled (this is *usually* the one running). See InsertingSvnVersionStringIntoSimulinkModels.

aLIGO SUS Operations Manual - Network Info

Back to Operation Manual main page

References

- T1000588 AdvLigo CDS Computer/Networking Rack Layouts And Real-Time Code Assignments
- T080135 Real-time Code Generator (RCG) Application Developer's Guide
- T1000248 aLIGO CDS File System Directories
- D1001173 LHO Test Stands Network Overview
- D1001614 LLO Test Stands Network Overview
- D1200664 LIGO Hanford Observatory aLIGO CDS Networks
- D1102217 LLO CDS Network Diagram

Network layout

The various SUS (and SEI) test stands are on individual private networks described in D1001173 for LHO, (D1001614 for LLO. These require a single stage login for most operations.

The main interferometer computer systems are on more complicated private networks which require a multi-stage login for added security.

Logging in from outside

See separate wiki page.

H1 System

To access the H1 system from outside requires a two stage login. The first stage requires `ligo.org` credentials (`albert.einstein` or the like, with associated password), plus a one-time password generated by a YubiKey dongle.

To do a first-stage login from outside the private network, use the following terminal command:

```
ssh -Y albert.einstein@cdsssh.ligo-wa.caltech.edu # your ligo.org name in place of
albert.einstein
```

when prompted supply first your `ligo.org` password and then the passcode from your YubiKey.

At LHO the gateway computer `cdsssh` is not usable for commissioning work and you need to do a second-stage login to some other computer, typically one of the control room workstations, `opsws0` through `opsws7`. At LHO it has been usual to login as user `controls` using the open-secret password from https://secrets.ligo.org/secrets/, but it is now recommended to login with your `ligo.org` credentials.

To do a second-stage login:

```
ssh -Y albert.einstein@opsws0  # pick opsws0 through opsws7 or another workstation
```

Note: it is important to use the `-Y` switch (trusted X11 forwarding) rather than the more familiar `-X` (regular X11 forwarding) because the security settings cause `-X` to time out after 20 minutes, so that existing X11 windows (e.g., MEDM screens) continue to work but new ones can't be opened.

L1 System

To access the L1 system from outside requires `ligo.org` credentials (`albert.einstein` or the like, with associated password), plus a one-time password generated by a YubiKey dongle. Use the following terminal command:

```
ssh -Y albert.einstein@cdsssh.ligo-la.caltech.edu # your ligo.org name in place of
albert.einstein
```

when prompted supply first your `ligo.org` password and then the passcode from your YubiKey.

At LLO the gateway computer `cdsssh` is usable for commissioning work, but you can also login to another workstation with your `ligo.org` credentials.

Note: it is important to use the `-Y` switch (trusted X11 forwarding) rather than the more familiar `-X` (regular X11 forwarding) because the security settings cause `-X` to time out after 20 minutes, so that existing X11 windows (e.g., MEDM screens) continue to work but new ones can't be opened.

X1 Test Stands (LHO)

The X1 test stand topology is given D1001173. However note that -v4 is out of date:

1. The active front end is `bscteststand2` with IP number 10.11.0.26. This is the number shown as reserved for `hamteststand` in the SUS Triple Test Stand. The old front end (`bscteststand`/10.11.0.25) is still on the network - don't use it by accident!
2. The "laptop", with IP number 10.11.0.96, is actually the SUS workstation in the office area.
3. Port 22 on the router now goes to the workstation. That is, if you log into `bisbee.ligo-wa.caltech.edu` from the outside, you get to the workstation, not the front end as formerly.
4. There is no separate workstation for the triples test stand. Rather, login as user `controls3` to the quad workstation.

To log in from outside the private network to the workstation, use the following terminal command:

```
ssh -Y controls@bisbee.ligo-wa.caltech.edu # use controls3 for access to the triple
```

To get to the front end from the workstation or iMac, do

```
ssh -Y controls@bscteststand2
```

To get to the iMac from the workstation or front end, do

```
ssh -X cleanroom@10.11.0.95
```

or

```
ssh -X controls@10.11.0.95
```

To get to the workstation from the iMac or front end, do

```
ssh -X controls@10.11.0.96
```

X2 Test Stands (LLO)

The X2 network topology is given in D1001614. The various

To log in from outside the private network to the workstation, use the following terminal command:

```
ssh -Y controls@llosus1.ligo-la.caltech.edu # or llosus2, llosus3, llosus4
```

aLIGO: Suspensions/OpsManual/Network (last edited 2014-05-27 09:41:11 by MarkBarton)

aLIGO SUS Operations Manual - Stuff on ODC

Back to Operation Manual main page

- G1300270 Data Quality For aLIGO: ODC Channels

aLIGO: Suspensions/OpsManual/ODC (last edited 2014-05-27 09:05:44 by MarkBarton)

aLIGO SUS Operations Manual - DTT

Back to Operation Manual main page

DTT is a standard CDS utility for taking time series and FFT data on EPICS channels. The underlying command line utility is `diag` (see Suspensions/OpsManual/diag), but the GUI version used most of the time is `diaggui`. DTT is one of two tools that can be used for taking transfer functions for testing (see Suspensions/OpsManual/TFs, Suspensions/OpsManual/TFs/DTT). (It requires rather more operator involvement but much less total time than the alternative involving Matlab scripts.)

The DTT user manual is T990013.

aLIGO: Suspensions/OpsManual/DTT (last edited 2014-05-27 09:05:08 by MarkBarton)

aLIGO SUS Operations Manual - dataviewer

Back to Operation Manual main page

aLIGO: Suspensions/OpsManual/dataviewer (last edited 2014-05-27 09:04:07 by MarkBarton)

aLIGO SUS Operations Manual - diag

Back to Operation Manual main page

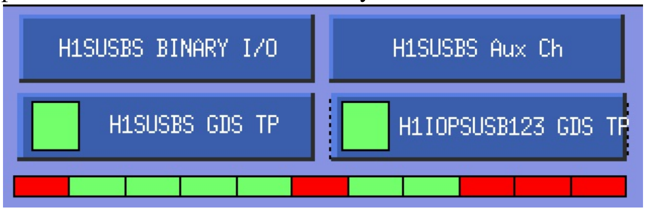
`diag` is the command-line CDS utility underlying DTT. See T990013 for a full description.

Clearing Excitation Testpoints

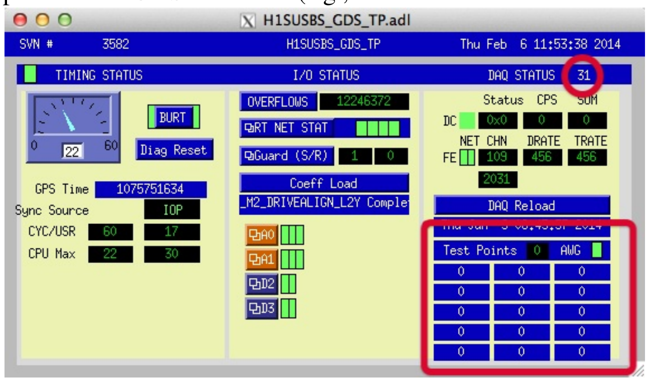
A mundane but important use for it is clearing excitation test points when AWG (or the excitation function of DTT) has become confused.

If this is suspected,

1. Open the overview screen and identify the user and IOP GDS TP screen buttons in the top right:



2. Open the user GDS TP screen (e.g., `H1SUSBS GDS TP` for H1:BS):



3. Note the processor number (next to `DAQ STATUS`; 31 for `H1SUSBS`) and check if there are any non-zero entries in the `Test Points` area.
4. If there are test points active when they shouldn't be, open a terminal window and enter `diag` in local (`-l`) mode and use the `tpclear` command:

```
controls@opsws1:~ 0$ diag -l
supported capabilities: testing  testpoints  awg
diag> tpclear * 31
test point cleared
diag> quit
EXIT KERNEL
```

The number `31` in the example should be replaced by the processor number identified in step 3. Note that it comes *after* the `*` meaning all test points.

aLIGO: Suspensions/OpsManual/diag (last edited 2014-05-27 09:04:47 by MarkBarton)

aLIGO SUS Operators Manual - Information on SVN

Contents

Subversion Overview

Subversion, a.k.a. SVN, is a client/server system for version control of files, especially software but also data. See http://subversion.apache.org/ for software and http://svnbook.red-bean.com/ for documentation. Dave Barker has a nice summary on this wiki at CDSSubVersion and a shorter one on the LHO CDS wiki. The server hosts one or more repositories, each of which contains the current reference copy of a set of directories and files, as well as all past versions. Clients can "checkout" a repository or selected directories to create "working copies" on other machines, make modifications, and then "commit" changed files back to the server. They can also "update" to get files that may have changed on the server since the last checkout or update, and they can "revert" files back to the state at the last checkout.

The server is a specialized HTTP server and can in a pinch be browsed with an ordinary web browser, but advanced features such as commit and update can only be accessed with the special purpose client. The server may also support a separate `websvn` interface, which optimized for read-only access with a web browser and doesn't support client access at all.

The main repositories of interest to SUS are:

Description	Client Access URL	WebSVN URL
SUS SVN (SUS data and analysis software)	https://redoubt.ligo-wa.caltech.edu/svn/sus	https://redoubt.ligo-wa.caltech.edu/websvn/listing.php?repname=sus&
`userapps` (RTCDS models for SUS and other clients)	https://redoubt.ligo-wa.caltech.edu/svn/cds_user_apps	https://redoubt.ligo-wa.caltech.edu/websvn/listing.php?repname=cds_user_apps&
Lots of others	https://redoubt.ligo-wa.caltech.edu/svn/?	https://redoubt.ligo-wa.caltech.edu/websvn/?

The repositories hosted at `redoubt.ligo-wa.caltech.edu` can be accessed in read-only mode with `ligo.org` (`albert.einstein`) credentials. Users needing write access must apply to David Barker.

ON CDS machines it is standard to check these repositories out from their roots at the following locations (on GC or personal machines, the user is free to check out any subtree of interest to anywhere convenient on their local disk):

SUS	`/ligo/svncommon/SusSVN/sus/` (or `/ligo3`... on the X1 triples test stand)
`userapps`	`/opt/rtcds/userapps/`

The root of the repository or the checked-out section is often denoted `^` in documentation and commands. It is common (but not at all required) for the root to have three top-level folders `branches`, `tags` and `trunk`, with the main files in `^/trunk/`. The `tags` directory is for frozen copies of part or all of `trunk` at key milestone points (one might create `v3.0` to define, well, v3.0), and `branches` is for copies that are being worked on independently and may possibly be "merged" back into the trunk. (SVN uses lazy copies, so it's very cheap in time and disk space to copy an entire directory within the repository - cost is only incurred if/when files in a branch are modified.) For the SUS repository, everything is in `^/trunk`, but `userapps` makes use of `tags` as well as `trunk`. See #sussvnlayout for further information on the SUS SVN Layout.

Note that Simulink models and other files can have a macro string in a text box which inserts the SVN revision number when the model is committed, so it can be displayed later and serve as documentation of which revision was most recently committed (this is *usually* the one running). See InsertingSvnVersionStringIntoSimulinkModels.

SUS SVN Layout

As noted above, the SUS svn is checked out at `/ligo/svncommon/SusSVN/sus/`, hereafter denoted `^`. The main set of files is contained in `^/trunk`. Within that are directories for the various suspension types along with a `/Common` directory. Each suspension-specific directory is organized according to first IFO, then optic, then sensor-actuator level. Some of the key directories are as follows:

```
trunk
    Common
        MatlabTools
            DoubleModel_Production
            SingleModel_Production
            TripleModel_Production
            SchroederPhaseTools
    QUAD
        Common
            MatlabTools
                QuadModel_Production
            MathematicaModels
        Data
        H1
            ITMX
                SAGM0
                    Data
                    Results
                SAGR0
                    Data
                    Results
                SAGL1
                    Data
                    Results
                SAGL2
                    Data
                    Results
                SAGL3
                    Data
                    Results
    HSTS
        H1
            MC1
                SAGM1
    etc
```

Notes:

- Many useful Matlab scripts are contained in `^/trunk/Common/MatlabTools`, or in the similar areas for specific suspension types, e.g., `^/trunk/QUAD/MatlabTools`, `^/trunk/BSFM/MatlabTools` etc.
- The scripts for taking transfer data via Matlab (see Suspensions/OpsManual/TFs/Matlab) live at `^/trunk/Common/MatlabTools/SchroederPhaseTools`.
- The Mathematica pendulum dynamics models live at `^/trunk/Common/MathematicaModels`. Instructions for using them are at Suspensions/MathematicaModels. Relative to early versions of the models which used many hard-coded paths, the file structure for the models has ben optimized for use with the SVN, but is still tricky. Therefore see especially Suspensions/MathematicaModels/SVNInstructions.
- The Matlab pendulum dynamics model for the QUAD lives in the `QUAD/Common` area at `^/trunk/QUAD/Common/MatlabTools/QuadModel_Production/` (QUAD) whereas those for the other suspensions live in the top-level `Common` directory: `^/trunk/Common/MatlabTools/XxxxxxModel_Production/`.
- Measured data for suspensions is filed by IFO or test stand (H1/L1/X1/X1), suspension (ITMX etc), sensor-actuator group (SAGM0/SAGR0 etc), e.g., `^/trunk/QUAD/H1/ITMX/SAGM0/Data`. Semi-processed .mat file versions and plots of raw data live in the sibling `Results` directory, e.g.

versions and plots of raw data live in the sibling Results directory, e.g., ^/trunk/QUAD/H1/ITMX/SAGM0/Results. Some very old data is filed by build number instead of suspension name but this system is obsolete.

- Comparison plots between various suspensions of a particular type live in the Data directory for that type, e.g., ^/trunk/QUAD/Data.

aLIGO SUS Operations Manual - Dynamical Models in Mathematica and Matlab

Back to Operation Manual main page

Modeling Software

There are two families of models of suspensions, in Mathematica and Matlab. These have been maintained in parallel, take parameters of almost identical names, and should give the same results to 3-4 decimal places for symmetrical suspension configurations. (In fact the Matlab uses state-space matrices originally exported from Mathematica.) Both have advantages for certain tasks. The Mathematica models the damping more accurately and can calculate thermal noise. It can also handle more types of asymmetry, such as wires of uneven length. The Matlab runs more quickly and interfaces to Simulink as well as many control room utilities in Matlab.

Within each family there are versions for quad (ETM, ITM and their reaction chains), triple (BSFM, HLTS and HSTS), dual (OMCS) dual-blade dual-mass (TMTS), single (HAUX) and single with blades (HTTS).

The Mathematica software is described in T020205. Guidance on installing and running it is at Suspensions/MathematicaModels. Cases of the models representing suspensions of interest live in the SUS SVN at `^/trunk/Common/MathematicaModels` (web interface, SVN client interface). See also the list of links below.

The Matlab software is described in T080188. A version of most of the key models can be downloaded from the same link. Adapted versions live in the SUS SVN - see below.

Mathematica Models of Specific Suspension Types

- QUAD: Suspensions/OpsManual/QUAD/Models
- BSFM: Suspensions/OpsManual/BSFM/Models
- HLTS: Suspensions/OpsManual/HLTS/Models
- HSTS: Suspensions/OpsManual/HSTS/Models
- OMCS: Suspensions/OpsManual/OMCS/Models
- TMTS: Suspensions/OpsManual/TMTS/Models
- HAUX: Suspensions/OpsManual/HAUX/Models
- HTTS: Suspensions/OpsManual/HTTS/Models

Matlab Models of Specific Suspension Types

The models for the various suspensions live at the following locations in the SUS SVN (web interface, SVN client interface). For more on usage see OpsManual/MatlabModels.

- QUAD: `^/trunk/QUAD/Common/MatlabTools/QuadModel_Production`
- BSFM: `^/trunk/Common/MatlabTools/TripleModel_Production`
- HLTS: `^/trunk/Common/MatlabTools/TripleModel_Production`
- HSTS: `^/trunk/Common/MatlabTools/TripleModel_Production`
- OMCS: `^/trunk/Common/MatlabTools/DoubleModel_Production`
- TMTS: `^/trunk/Common/MatlabTools/DoubleModel_Production`
- HAUX: `^/trunk/Common/MatlabTools/SingleModel_Production`
- HTTS: `^/trunk/Common/MatlabTools/SingleModel_Production`

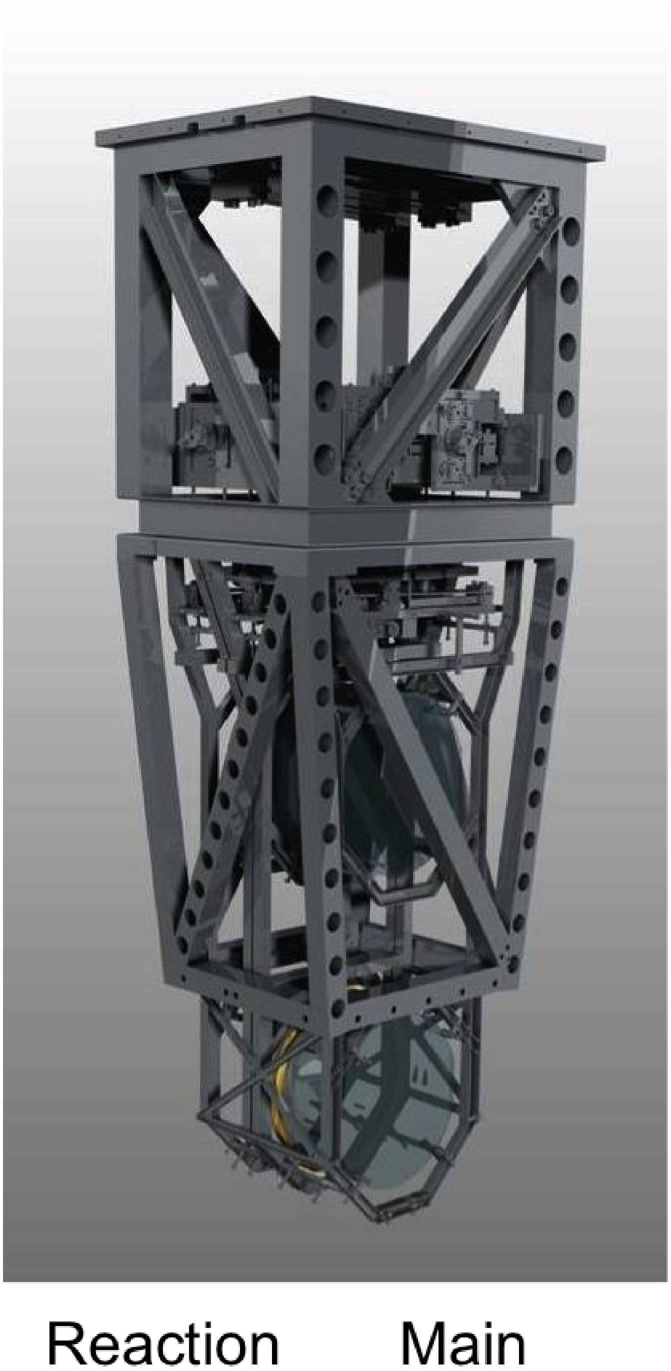
aLIGO SUS Operation Manual - Info on QUAD Suspensions

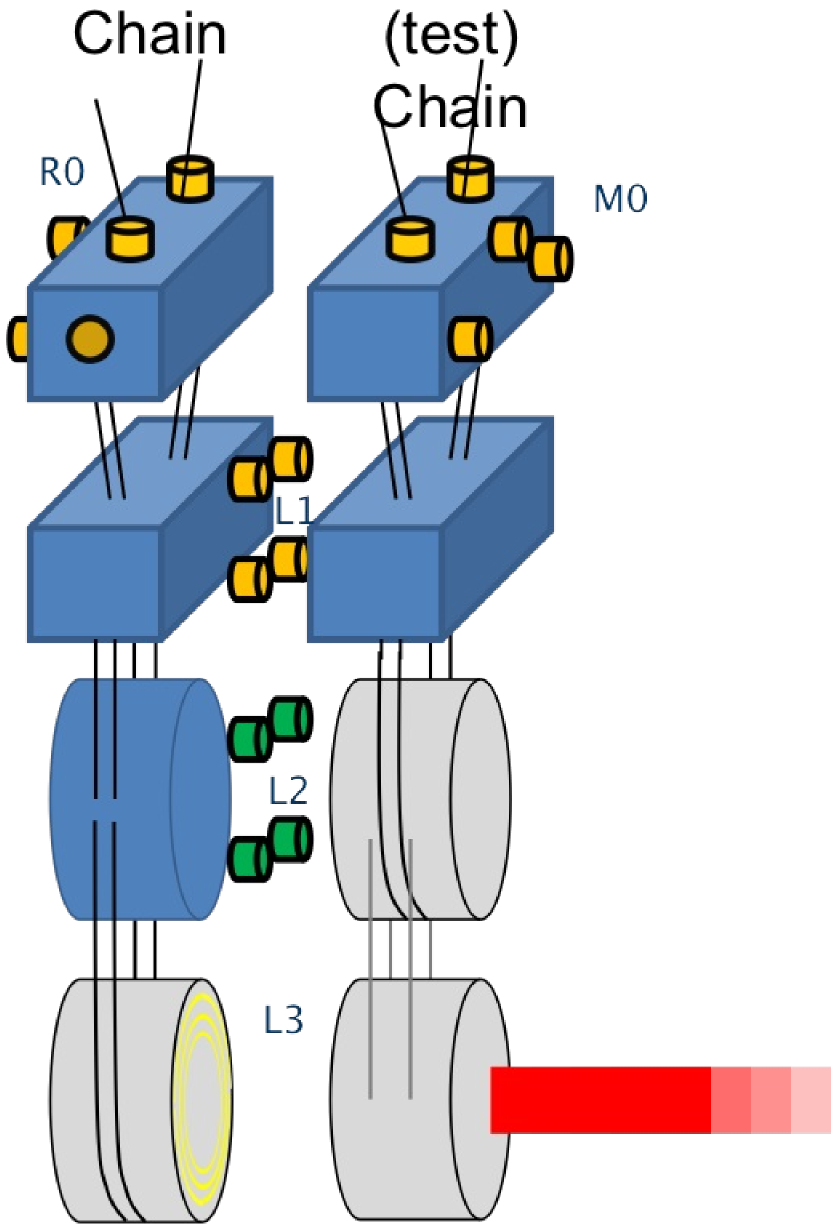
Back to Operation Manual main page

Overview

Each "quad" suspension actually consists of two four-mass chains, the main chain and the reaction chain. The main chain supports a test mass (ITM or ETM) as its lowest level, and above that are in order the penultimate mass (PUM), upper intermediate mass (UIM), and the top mass (no abbreviated form; TM = test mass). The PUM and the ITM/ETM are fused silica and are connected by fused silica fibres for low thermal noise - this subassembly is called the monolithic stage. The upper masses and all other wires are metal.

The reaction chain has non-contacting actuators on the bottom three levels that can apply force to the corresponding masses of the main chain. The lowest reaction mass in the ITM configuration is also the compensation plate (CP); the lowest reaction mass in the ETM configuration is purely a reaction mass (the end reaction mass or ERM).





To do: get better diagram, with mass labels (UIM etc), front/back labels, coordinate system marker.

From the point of view of an operator/scimon, the quad appears as 5 sensor/actuator groups (SAGs): M0, R0, L1, L2, and L3 (see figure).

M0 consists of 6 BOSEMs mounted on the structure, which sense and actuate all 6 DOFs of the top mass of the main chain.

R0 is the same as M0 except for the reaction chain.

M0 and R0 are specific to a particular chain, as are the control loops that use them. They provide local control, whereby each chain is damped purely relative to the support structure, without reference to either the other chain in the same quad or the other quad in the same arm. The M0 and R0 actuators are also where static pitch and yaw corrections are added.

L1 consists of 4 BOSEMs mounted on the second top mass (a.k.a. upper intermediate mass, UIM) of the reaction chain. These sense the relative longitudinal, pitch and yaw of the UIM of the main chain.

L2 consists of 4 BOSEMs mounted on the PRM (2nd bottom mass of the reaction chain). These sense the relative longitudinal, pitch and yaw of the PUM of the main chain.

L3 consists 4 electrostatic drive (ESD) electrodes, which can actuate in longitudinal, pitch and yaw on the test mass. The optical lever is grouped with L3 on the control screens, although it's not officially part of SUS.

L1, L2 and L3 are joint to the two chains. They sense differential displacements and apply equal and opposite forces/torques. The cabling for L1/L2/L3 runs down the reaction chain, making it a little stiff, but leaving the main chain unencumbered. The actuators towards the bottom of the chain are chosen to be weaker but less noisy, so as to enable a hierarchical control strategy where control is applied as remotely as feasible from the payload, so actuator noise will be filtered by the intermediate stages.

References

- Quad Suspensions Wiki Page (old)
- T010103: aLIGO Suspension System Conceptual Design
- P020001: Quadruple Suspension Design for Advanced LIGO
- P1200056: Update on Quadruple Suspension Design for Advanced LIGO
- D0901346: Advanced LIGO Quadruple Suspension
- E1000617: Quad Suspension Controls Arrangement Poster
- T1100595: ETM/ITM Quad Suspension Control Ranges
- T1100378: aLIGO QUAD Controls Design Description
- D1100022: aLIGO SUS ITM and BS Wiring Diagrams (for H1/L1)
- D1001725: aLIGO SUS ITM, BS, and FM System Wiring Diagrams (for H2)
- D1002741: aLIGO SUS ETM System Wiring Diagrams

Models

The quad suspension has been extensively modelled. Key results are at Suspensions/OpsManual/QUAD/Models .

Screens

Quad MEDM screens are documented at /Screens.

aLIGO: Suspensions/OpsManual/QUAD (last edited 2013-09-11 15:11:16 by MarkBarton)

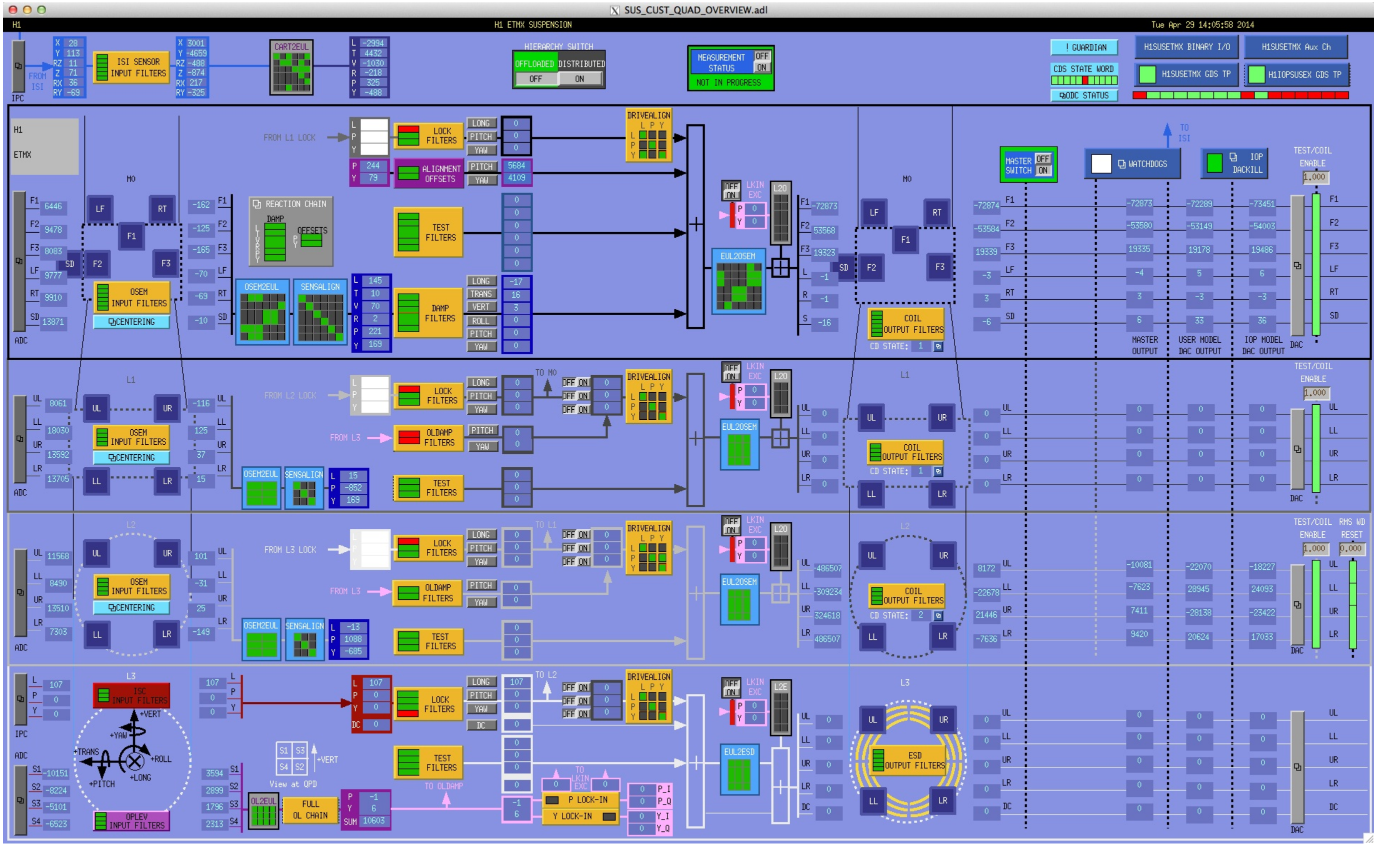
aLIGO SUS Operations Manual - Overview of QUAD MEDM screens

Back to Operations Manual main page
Back to QUAD main page

Except where noted, the quad screens described below live at `/opt/rtcds/userapps/release/sus/common/medm/quad/`. They are all generic screens which rely for their execution on site-, interferometer- and suspension-specific arguments passed in when they are called. They can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_QUAD_OVERVIEW

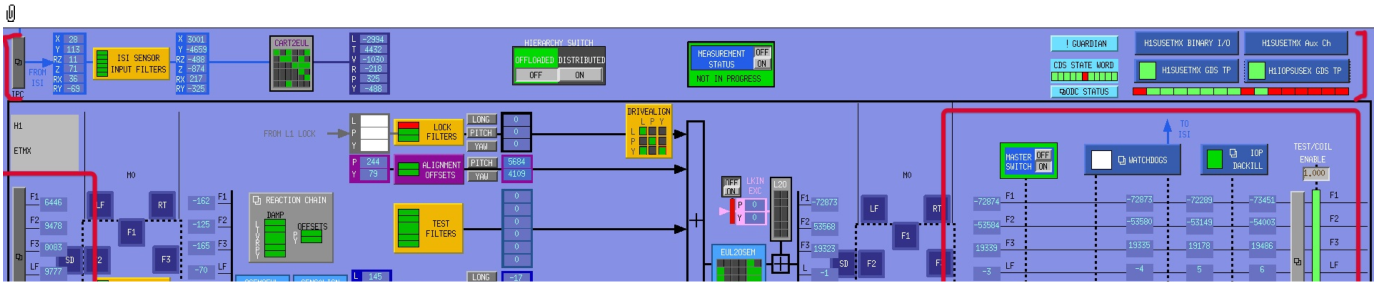


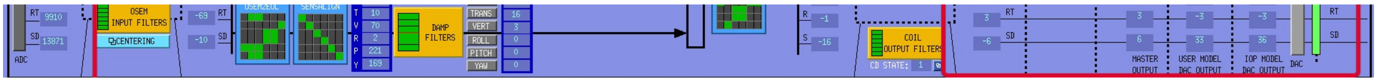
This is the overview screen. The functionality is divided up according to the five sensor-actuator groups, plus an odds-and-ends area at the top:

- Other - Placeholders for ISI feedforward stuff, links to CDS utility screens.
- M0 - 6 BOSEMs on the structure engaging the main chain top mass
- R0 - 6 BOSEMs on the structure engaging the reaction chain top mass (most of the UI for this is hidden in a subscreen accessed by a button)
- L1 - 4 BOSEMs on the reaction chain upper intermediate mass engaging the main chain upper intermediate mass
- L2 - 4 AOSEMs on the reaction chain penultimate mass engaging the main chain penultimate mass
- L3 - 4 ESD quadrants on the CP or ERM engaging the ITM or ETM, plus optical lever channels.

M0 and R0 are used for local damping (relative to the structure) and the control loops are already functional. DC pitch and yaw offsets are also injected at M0 or R0. L1, L2 and L3 are intended for implementing global control (relative to other optics) and have placeholder inputs for actuation requests from ISC. The L3 level also processes optical lever signals.

Other Screens



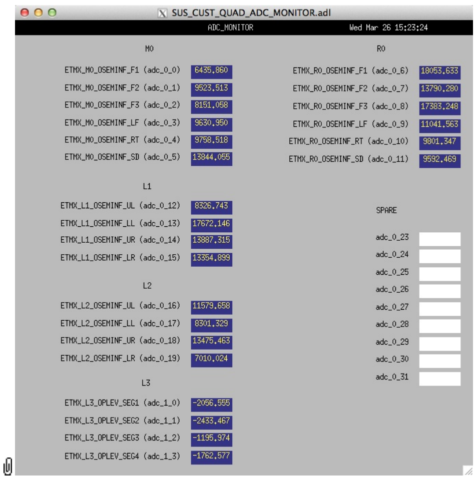


- MEASUREMENT STATUS - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- HIERARCHY SWITCH - enables various ISC inputs and outputs.
- MASTER SWITCH - switches all actuation
- ADC - a monitor screen (common to all sensor-actuator groups) for the semi-raw ADC values as reported by the IOP process after downsampling.
- DAC - a popup allowing access to all the DAC outputs as reported by the IOP process.
- H1SUSETMX Aux Ch - monitors for the voltage, current, etc readbacks as reported by the AUX process.
- WATCHDOGS - a block implementing the watchdog on the various sensor actuator groups.

groups).

- IOP DACKILL - a screen for the watchdog on the IOP process serving all suspensions on the same front end.
- OVERVIEW of SUS WATCHDOGS - accessed via IOP DACKILL; a screen for watchdogs on all IOP processes serving SUS models.
- EX BSC9 (ETMX) (or the like) - accessed via above OVERVIEW; a screen for watchdogs for the current chamber (including SEI).
- H1SUSETMX BINARY I/O (or the like) - the binary input-output controls
- H1SUSETMX GDS TP (or the like) - the monitor screen for the user model process
- H1IOPSUSEX GDS TP (or the like) - the monitor screen for the IOP model process
- IPC - interprocess control signals monitor (diagnostic only)
- CART2EUL - matrix converting IPC signals from SEI into SUS basis (diagnostic only)
- ODC STATUS - ODC status
- !GUARDIAN Guardian controls

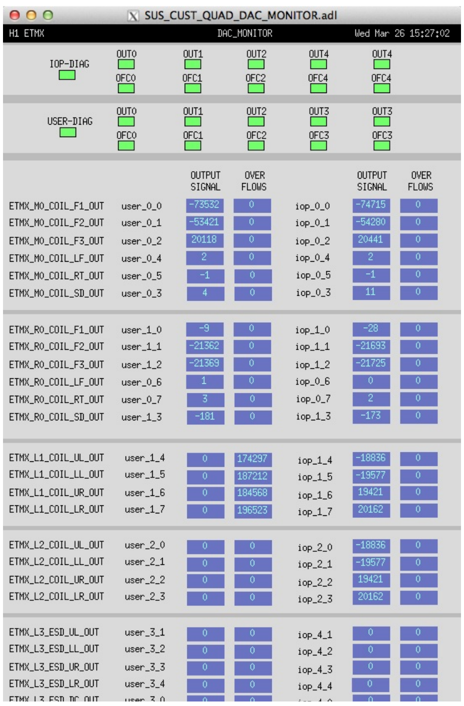
Screen SUS_CUST_QUAD_ADC_MONITOR



Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen SUS_CUST_QUAD_DAC_MONITOR



Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

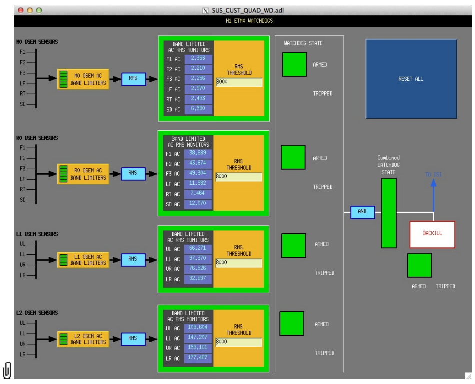
Screen SUS_CUST_QUAD_MONITOR_OVERVIEW

Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

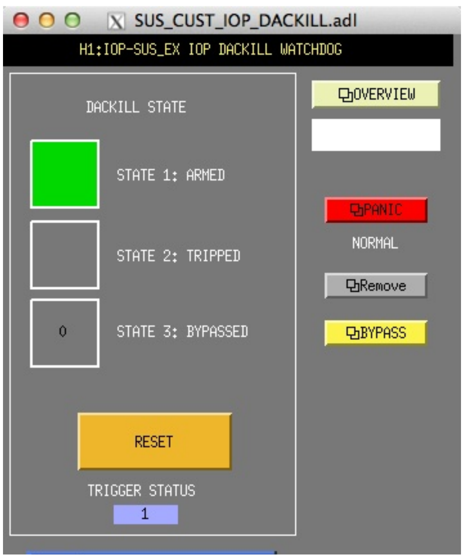
Screen SUS_CUST_QUAD_WD



Suspensions/OpsManual/Boilerplate/WD:

This screen allows for setting the watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

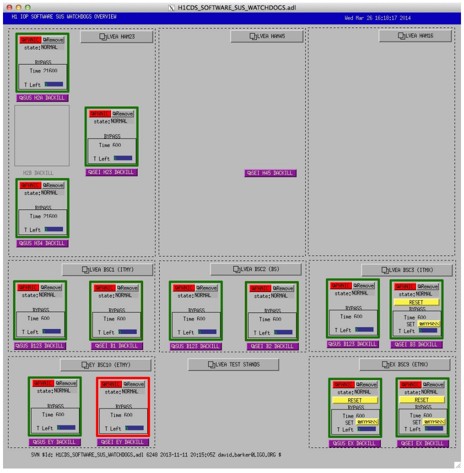
Screen SUS_CUST_IOP_DACKILL



Suspensions/OpsManual/Boilerplate/IOP_DACKILL:

The watchdog controls for the IOP process. If this WD is tripped it disables IO to/from all user models on the front end. RESET resets from the tripped state if there are no outstanding triggers. BYPASS is a manual override that allows for operation of the suspension if there is a trigger condition that is known by the operator to be harmless.

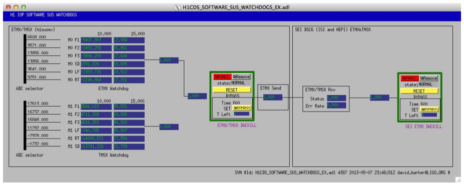
Screen H1CDS_SOFTWARE_SUS_WATCHDOGS



Suspensions/OpsManual/Boilerplate/SUS_WATCHDOGS:

The overview screen for all SUS/SEI watchdogs, organized by chamber. (LHO version shown - LLO may be very different - need to check.)

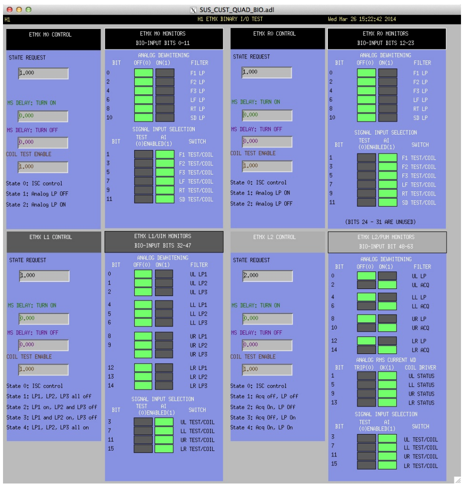
Screen H1CDS_SOFTWARE_SUS_WATCHDOGS_EX



Suspensions/OpsManual/Boilerplate/SUS_WATCHDOGS_CHAMBER:

The overview screen for all the SUS/SEI watchdogs in the same chamber.

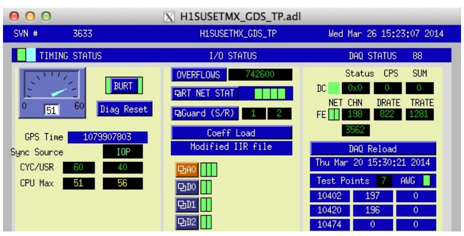
Screen SUS_CUST_QUAD_BIO

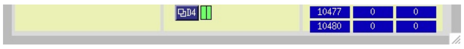


Suspensions/OpsManual/Boilerplate/QUAD_BIO:

The Binary Input-Output control screen. State Requests for the various sensor-actuator groups can be input. Positive state values lock out manual control of the filters on the COILOUTF screen and set the digital dewhitening filters to match the analog whitening filters selected via the binary outputs. Negative state values allow the dewhitening filters to be manually overridden.

Screen H1SUSETMX_GDS_TP





Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

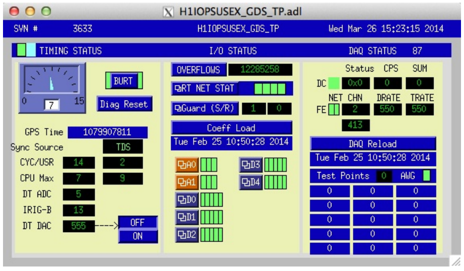
This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1IOPSUSEX_GDS_TP



Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

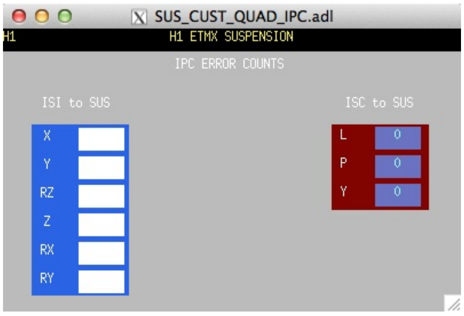
This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen SUS_CUST_QUAD_IPC



Suspensions/OpsManual/Boilerplate/IPC:

Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

Screen SUS_CUST_QUAD_M0_CART2EUL

Suspensions/OpsManual/Boilerplate/M0_CART2EUL:

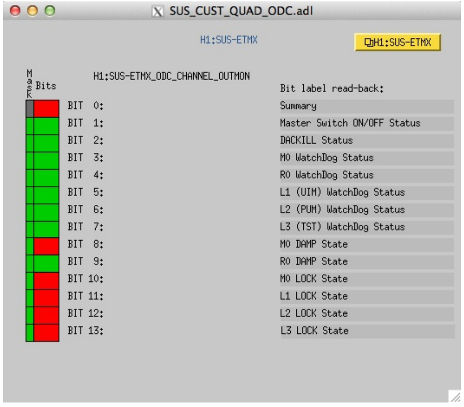
A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

Screen GUARD.adl

Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

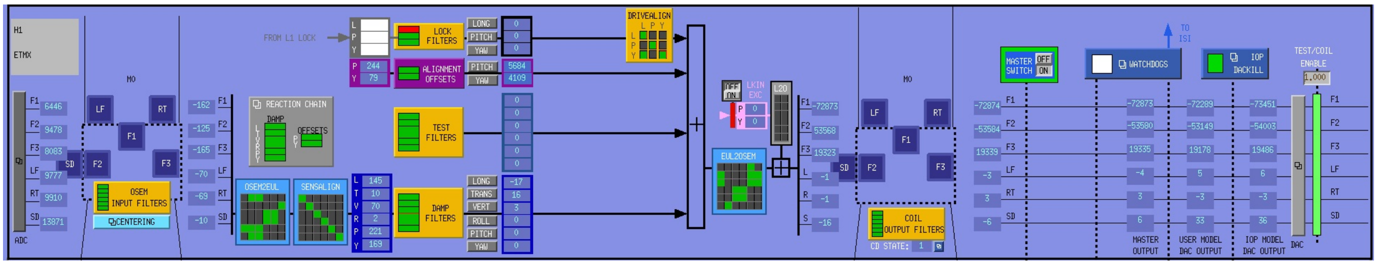
Screen SUS_CUST_QUAD_ODC.adl



Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

Sensor Actuator Group M0 - Main Chain Top Mass



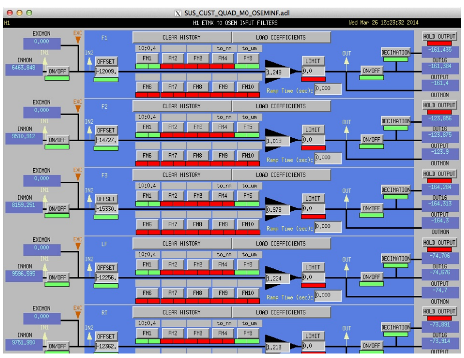
The M0 section of the main QUAD screen is excerpted above. Most of the items on it correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the main chain local control loop, which involves the following elements:

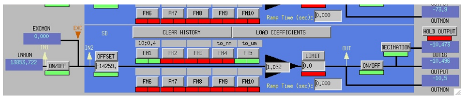
- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a * OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- DAMP FILTERS - a filter bank applying the damping filters.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- L2O - ????.
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.
- MASTER SWITCH - a manual on/off switch for all actuation (common to all sensor-actuator groups).

There are the following auxiliary inputs:

- TEST - a filter bank (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.
- LOCK FILTERS - filters for global control signals.
- REACTION CHAIN - the reaction chain controls

Screen SUS_CUST_QUAD_M0_OSEMINF





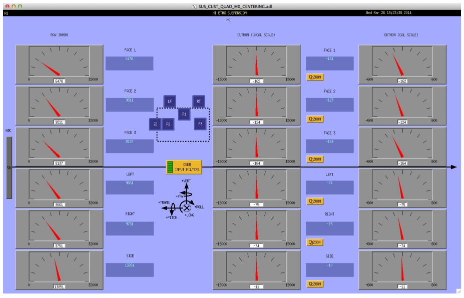
Suspensions/OpsManual/Boilerplate/M0_OSEMINF:

This block has 6 filter groups corresponding to the 6 M0 BOSEMs, F1/F2/F3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_QUAD_M0_CENTERING



Suspensions/OpsManual/Boilerplate/M0_CENTERING:

This screen gives various views of the M0 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_QUAD_M0_OSEM2EUL

Suspensions/OpsManual/Boilerplate/M0_OSEM2EUL:

This screen allows entry of the matrix which converts from the M0 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

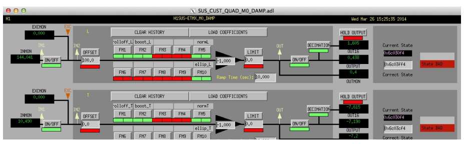
The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

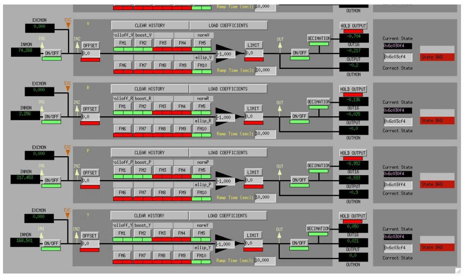
Screen SUS_CUST_QUAD_M0_SENSALIGN

Suspensions/OpsManual/Boilerplate/M0_SENSALIGN:

This screen is reserved for tweaking the M0 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_QUAD_M0_DAMP

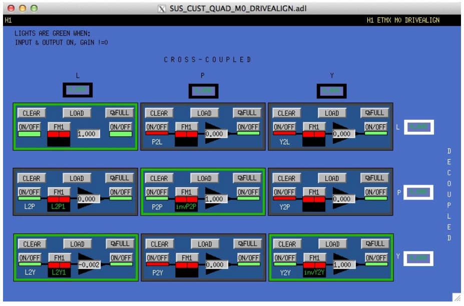




Suspensions/OpsManual/Boilerplate/M0_DAMP:

These filters implement the local damping. See Suspensions/OpsManual/Damping.

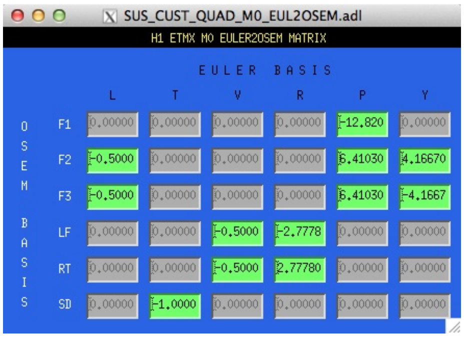
Screen SUS_CUST_QUAD_M0_DRIVEALIGN



Suspensions/OpsManual/Boilerplate/M0_DRIVEALIGN:

This screen is reserved for tweaking the M0 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

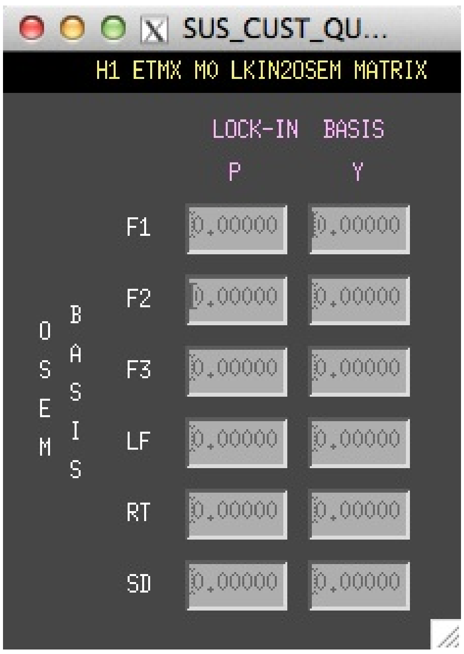
Screen SUS_CUST_QUAD_M0_EUL2OSEM



Suspensions/OpsManual/Boilerplate/M0_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the M0 OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

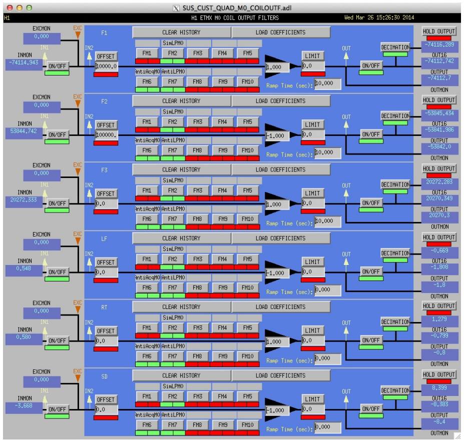
Screen SUS_CUST_QUAD_M0_LKIN2OSEM.adl



Suspensions/OpsManual/Boilerplate/M0_LKIN2OSEM:

This screen displays the transformation matrix from the lock-in request signals to the M0 OSEM actuation basis. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_QUAD_M0_COILOUTF



Suspensions/OpsManual/Boilerplate/M0_COILOUTF:

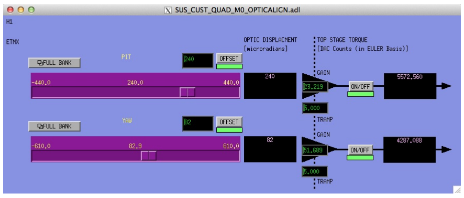
This screen applies compensation for the hardware filters in the M0 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

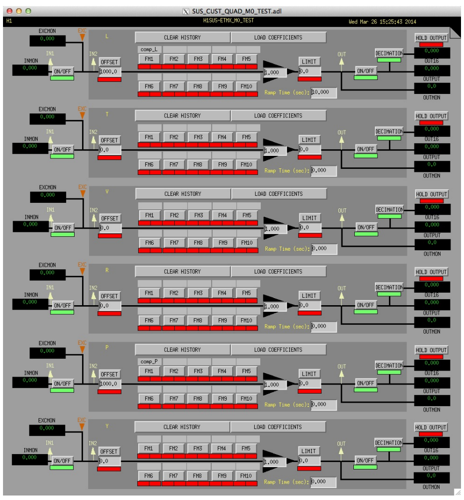
Screen SUS_CUST_QUAD_M0_OPTICALIGN



Suspensions/OpsManual/Boilerplate/M0_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets for the quad main chain. See Suspensions/OpsManual/Alignments for more info.

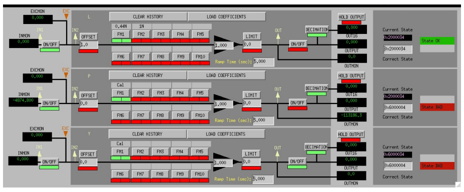
Screen SUS_CUST_QUAD_M0_TEST



Suspensions/OpsManual/Boilerplate/M0_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the M0 Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

Screen SUS_CUST_QUAD_M0_LOCK

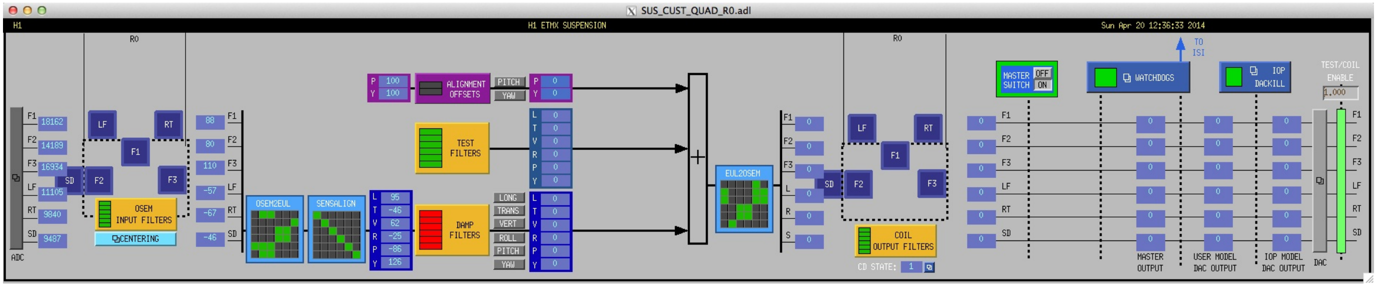


Suspensions/OpsManual/Boilerplate/M0_LOCK:

Filters for the locking signals to be applied at M0.

Sensor Actuator Group R0 - Reaction Chain Top Mass

Screen SUS_CUST_QUAD_R0



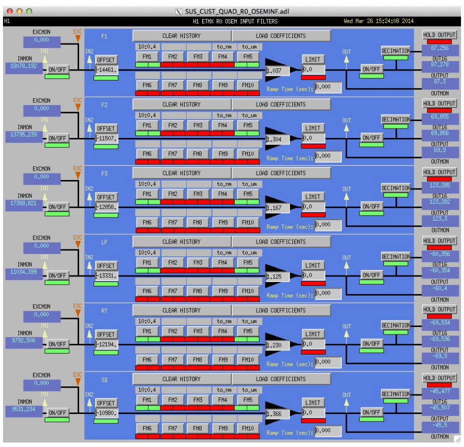
The R0 controls are accessed via a button in the M0 section of the main QUAD screen above, and appear on a separate screen very similar to the M0. Most of the items on it correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the reaction chain local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- DAMP FILTERS - a filter bank applying the damping filters.
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.
- MASTER SWITCH - a manual on/off switch for all actuation (common to all sensor-actuator groups).
- WATCHDOGS - same watchdog screen for all watchdogs as accessible from the #Other section of the main overview screen.
- IOP DACKILL - same watchdog screen for the IOP process as accessible from the #Other section of the main overview screen.

There are the following auxiliary inputs:

- TEST FILTERS - a filter bank (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.

Screen SUS_CUST_QUAD_R0_OSEMINF



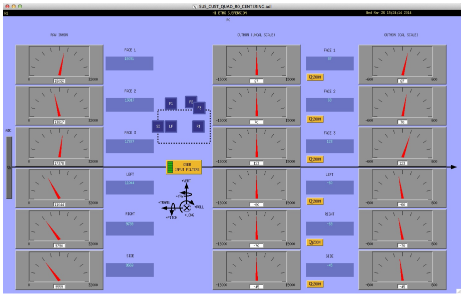
Suspensions/OpsManual/Boilerplate/R0_OSEMINF:

This block has 6 filter groups corresponding to the 6 R0 BOSEMs, F1/F2/F3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_QUAD_R0_CENTERING



Suspensions/OpsManual/Boilerplate/R0_CENTERING:

This screen gives various views of the R0 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_QUAD_R0_OSEM2EUL

Suspensions/OpsManual/Boilerplate/R0_OSEM2EUL:

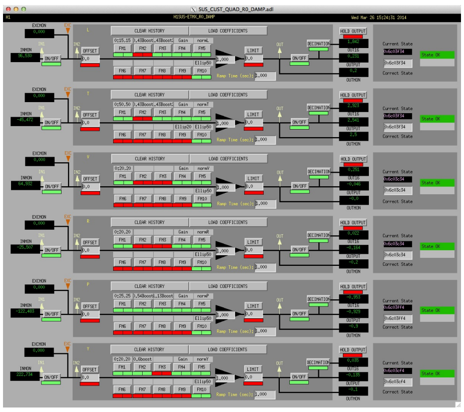
This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_QUAD_R0_SENSALIGN

Suspensions/OpsManual/Boilerplate/R0_SENSALIGN:

This screen is reserved for tweaking the R0 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_QUAD_R0_DAMP



Suspensions/OpsManual/Boilerplate/R0_DAMP:

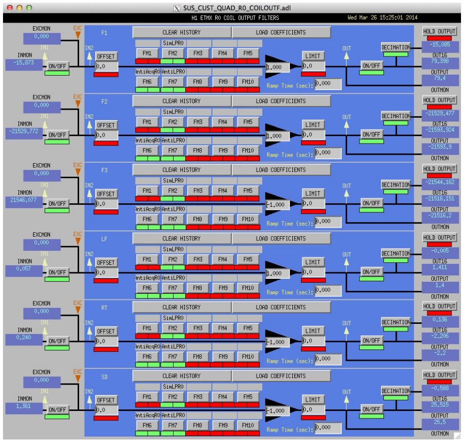
These filters implement the local damping for the reaction chain.

Screen SUS_CUST_QUAD_R0_EUL2OSEM

Suspensions/OpsManual/Boilerplate/R0_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the R0 OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_QUAD_R0_COILOUTF


Suspensions/OpsManual/Boilerplate/R0_COILOUTF:

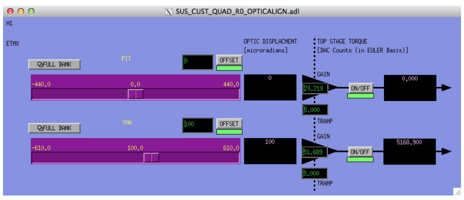
This screen applies compensation for the hardware filters in the R0 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

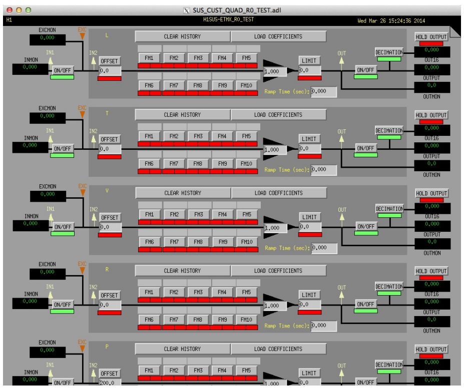
The sign convention at the output is positive for a positive current (rather than force).

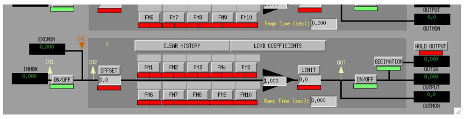
Screen SUS_CUST_QUAD_R0_OPTICALIGN


Suspensions/OpsManual/Boilerplate/R0_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets for the quad reaction chain. See Suspensions/OpsManual/Alignments for more info.

Screen SUS_CUST_QUAD_R0_TEST

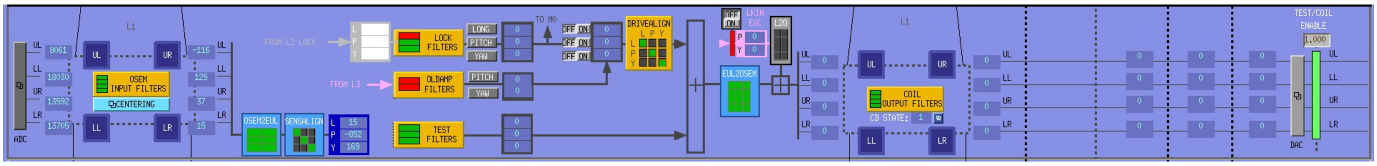




Suspensions/OpsManual/Boilerplate/R0_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the R0 Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

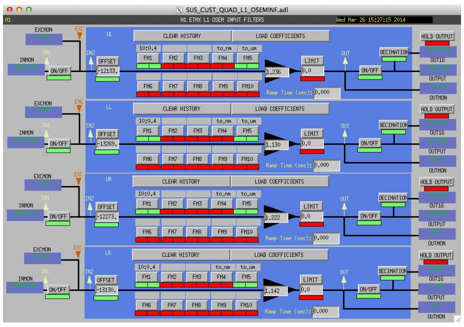
Sensor Actuator Group L1 - Between Upper Intermediate Masses



- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- TEST - a filter bank (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- LOCK - filters for global control signals.
- OLDAMP FILTERS - filters for optical lever damping.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ????.
- [[#SUS_CUST_QUAD_L1_COILOUTF| COIL OUTPUT FILTERS]) - a filter bank that corrects for hardware run/acquisition mode filters and

for magnet polarity.

Screen SUS_CUST_QUAD_L1_OSEMINF



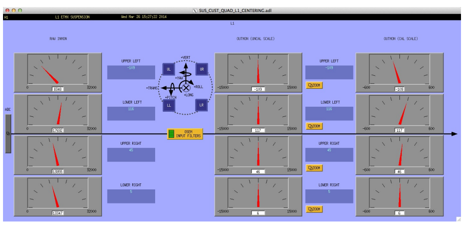
Suspensions/OpsManual/Boilerplate/L1_OSEMINF:

This block has 4 filter groups corresponding to the 4 L1 BOSEMs, UL/LL/UR/LR. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_QUAD_L1_CENTERING

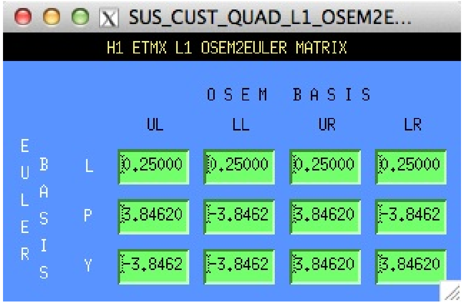


Suspensions/OpsManual/Boilerplate/L1_CENTERING:

This screen gives various views of the L1 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_QUAD_L1_OSEM2EUL

Screen SUS_CUST_QUAD_L1_OSEM2EUL



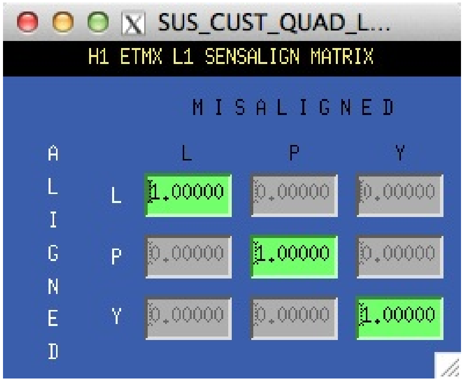
Suspensions/OpsManual/Boilerplate/L1_OSEM2EUL:

This screen allows entry of the matrix which converts from the L1 OSEM basis to the L1 Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in μm the outputs are then in μrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

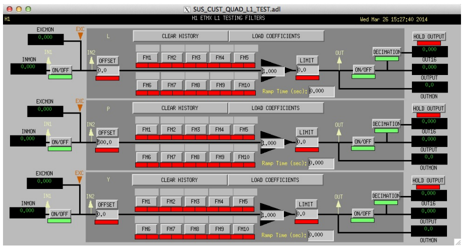
Screen SUS_CUST_QUAD_L1_SENSALIGN



Suspensions/OpsManual/Boilerplate/L1_SENSALIGN:

This screen is reserved for tweaking the L1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

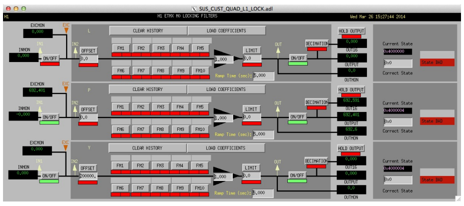
Screen SUS_CUST_QUAD_L1_TEST



Suspensions/OpsManual/Boilerplate/L1_TEST:

This is a deliberately empty set of 3 filter banks corresponding to the L1 L, P and Y DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

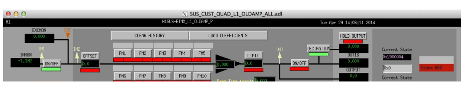
Screen SUS_CUST_QUAD_L1_LOCK

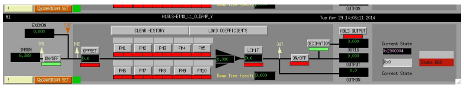


Suspensions/OpsManual/Boilerplate/L1_LOCK:

Filters for the locking signals to be applied at L1.

Screen SUS_CUST_QUAD_L1_OLDAMP_ALL.adl

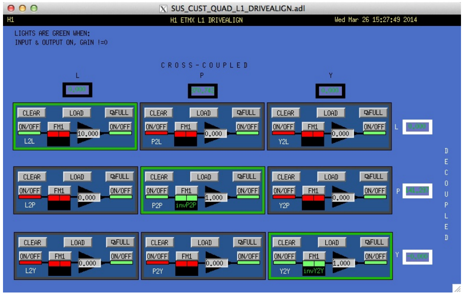




Suspensions/OpsManual/Boilerplate/L1_OLDAMP:

Filters for the optical lever locking signals.

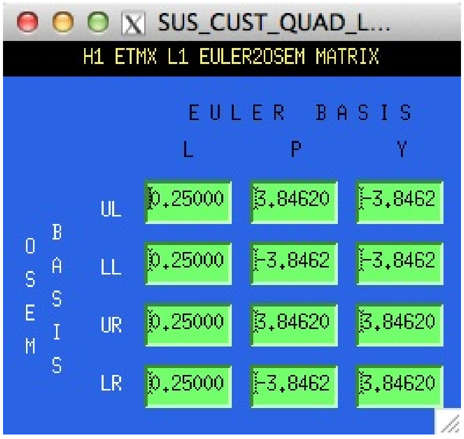
Screen SUS_CUST_QUAD_L1_DRIVEALIGN



Suspensions/OpsManual/Boilerplate/L1_DRIVEALIGN:

This screen is reserved for tweaking the L1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

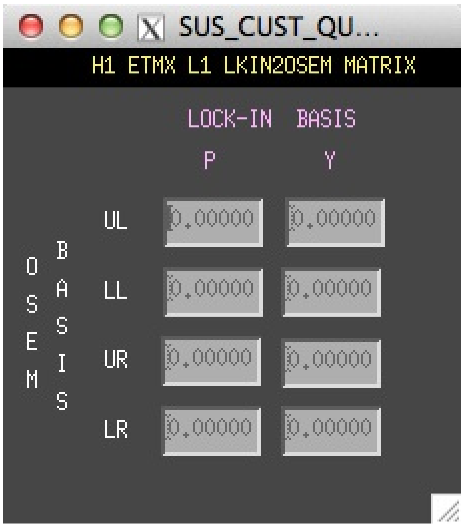
Screen SUS_CUST_QUAD_L1_EUL2OSEM



Suspensions/OpsManual/Boilerplate/L1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the L1 Euler basis to the L1 OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

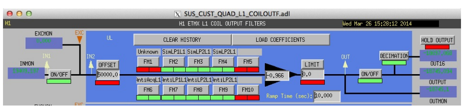
Screen SUS_CUST_QUAD_L1_LKIN2OSEM.adl

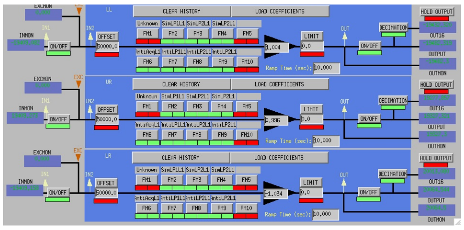


Suspensions/OpsManual/Boilerplate/L1_LKIN2OSEM:

This screen displays the transformation matrix from the lock-in request signals to the L1 OSEM actuation basis. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_QUAD_L1_COILOUTF




Suspensions/OpsManual/Boilerplate/L1_COILOUTF:

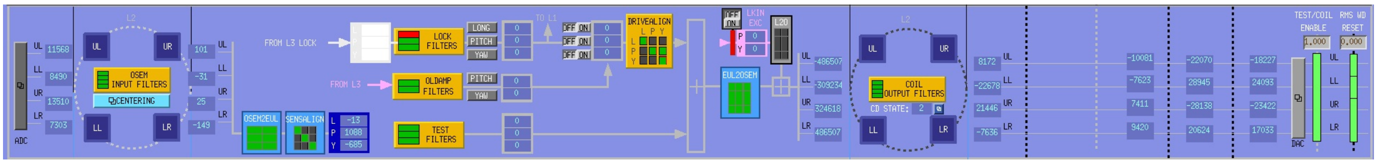
This screen applies compensation for the hardware filters in the L1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

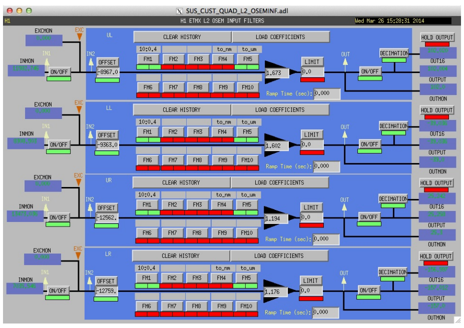
Sensor Actuator Group L2 - Between Penultimate Masses



- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signals to signals in a longitudinal/pitch/yaw (L/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- TEST - a filter bank (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- LOCK - filters for global control signals.
- OLDAMP FILTERS - filters for optical lever damping.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ????.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and

for magnet polarity.

Screen SUS_CUST_QUAD_L2_OSEMINF

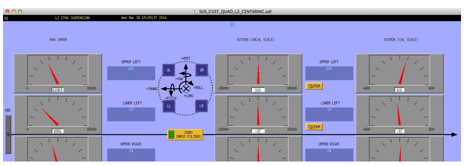

Suspensions/OpsManual/Boilerplate/L2_OSEMINF:

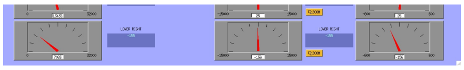
This block has 4 filter groups corresponding to the 4 L2 AOSEMs, UL/LL/UR/LR. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_QUAD_L2_CENTERING

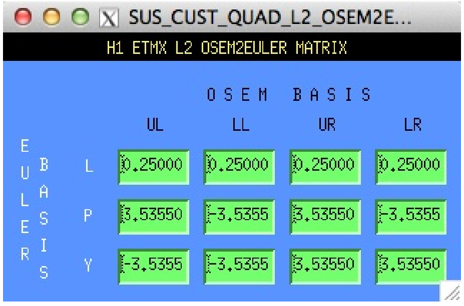




Suspensions/OpsManual/Boilerplate/L2_CENTERING:

This screen gives various views of the L2 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_QUAD_L2_OSEM2EUL



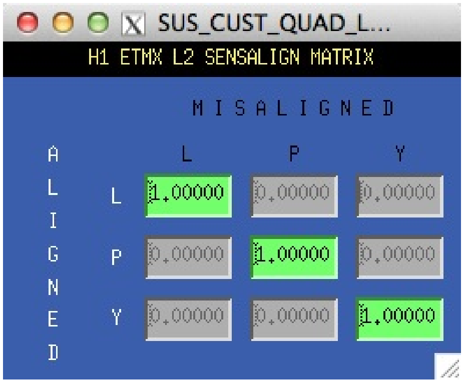
Suspensions/OpsManual/Boilerplate/L2_OSEM2EUL:

This screen allows entry of the matrix which converts from the L2 OSEM basis to the L2 Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

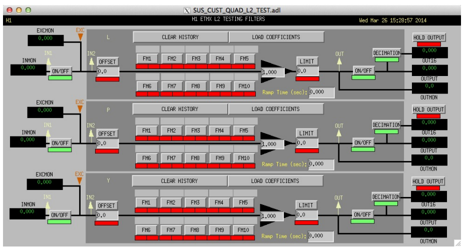
Screen SUS_CUST_QUAD_L2_SENSALIGN



Suspensions/OpsManual/Boilerplate/L2_SENSALIGN:

This screen is reserved for tweaking the L2 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

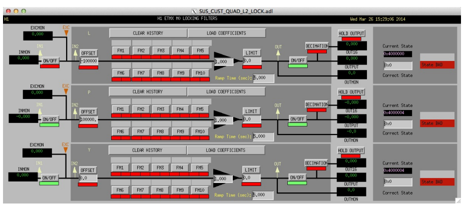
Screen SUS_CUST_QUAD_L2_TEST



Suspensions/OpsManual/Boilerplate/L2_TEST:

This is a deliberately empty set of 3 filter banks corresponding to the L2 L, P and Y DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

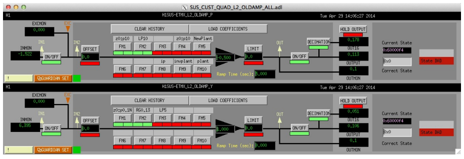
Screen SUS_CUST_QUAD_L2_LOCK



Suspensions/OpsManual/Boilerplate/L2_LOCK:

Filters for the locking signals to be applied at L2.

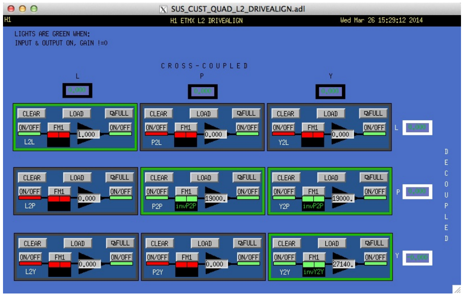
Screen SUS_CUST_QUAD_L2_OLDAMP_ALL.adl



Suspensions/OpsManual/Boilerplate/L2_OLDAMP:

Filters for the optical lever locking signals.

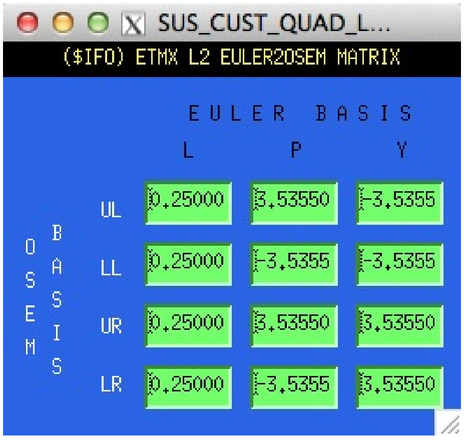
Screen SUS_CUST_QUAD_L2_DRIVEALIGN



Suspensions/OpsManual/Boilerplate/L2_DRIVEALIGN:

This screen is reserved for tweaking the L2 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

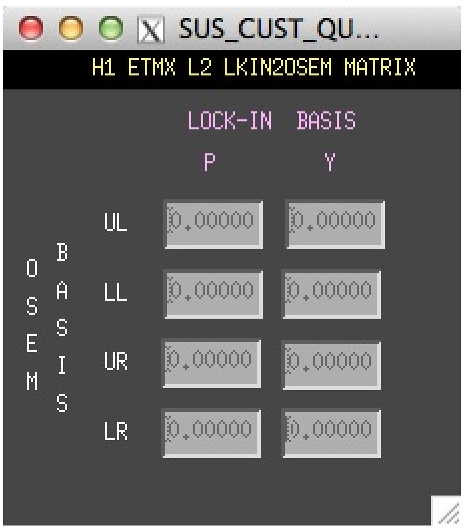
Screen SUS_CUST_QUAD_L2_EUL2OSEM



Suspensions/OpsManual/Boilerplate/L2_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the L2 Euler basis to the L2 OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_QUAD_L2_LKIN2OSEM.adl

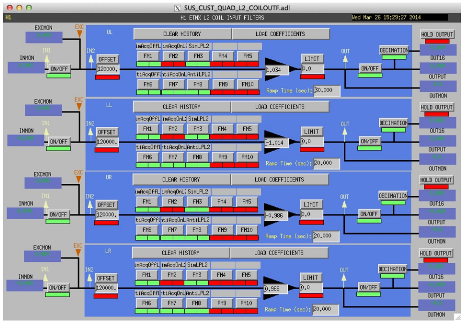


Suspensions/OpsManual/Boilerplate/L2_LKIN2OSEM:

This screen displays the transformation matrix from the lock-in request signals to the L2 OSEM actuation basis. See

Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_QUAD_L2_COILOUTF



Suspensions/OpsManual/Boilerplate/L2_COILOUTF:

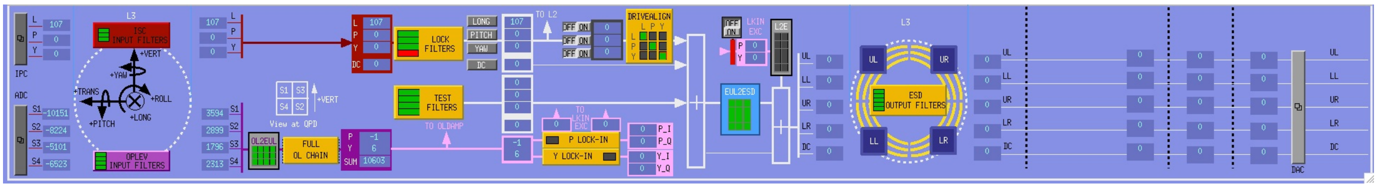
This screen applies compensation for the hardware filters in the L2 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

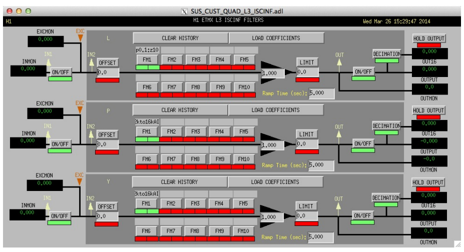
The sign convention at the output is positive for a positive current (rather than force).

Sensor Actuator Group L3 - Between ITM and CP or ETM and ERM



- ISCINF - input filters for ISC signals.
- OPLEV INPUT FILTERS - OL QPD segment signal filters
- OL2EUL - transformation from OL segment basis to P/Y/SUM
- FULL OL CHAIN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- LOCKIN_DEMOD - ????
- TEST - a filter bank (with no filters!) serving as a place to apply AWG actuation in the L/P/Y basis without it having to go through the damping filters.
- LOCK - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2ESD - a block that de-diagonalizes the damping signals back to per-ESD-quadrant signals.
- L2O - ????.
- ESD OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters.

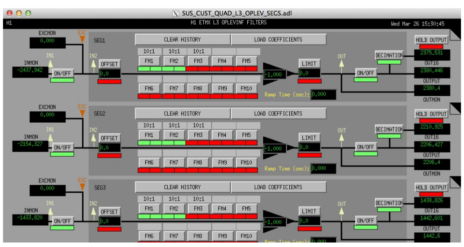
Screen SUS_CUST_QUAD_L3_ISCINF



Suspensions/OpsManual/Boilerplate/L3_ISCINF:

Filters for the locking signals from ISC.

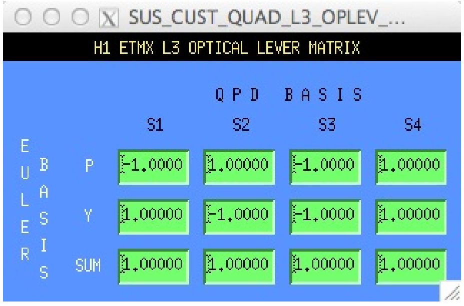
Screen SUS_CUST_QUAD_L3_OPLEV_SEGS



Suspensions/OpsManual/Boilerplate/L3_OPLEV_SEGS:

This block has 4 filter groups for the 4 L3 optical lever segments.

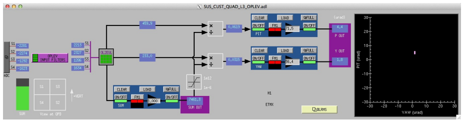
Screen SUS_CUST_QUAD_L3_OPLEV_MTRX



Suspensions/OpsManual/Boilerplate/L3_OPLEV_MTRX:

This screen displays the transformation matrix from optical lever segments to the pitch/yaw/sum basis. See Suspensions/OpsManual/Projections for more info.

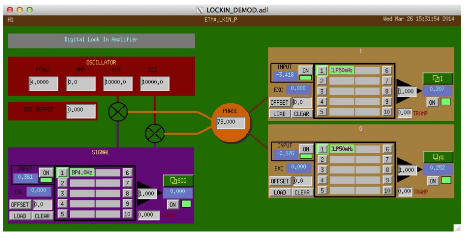
Screen SUS_CUST_QUAD_L3_OPLEV

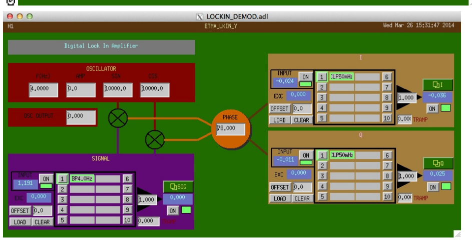


Suspensions/OpsManual/Boilerplate/L3_OPLEV:

This screen has the optical lever overview. It has some of the same buttons (for OPLEV_SEGS and OPLEV_MTRX) as on the main quad screen, and also shows the filters for the pitch, yaw and sum signals.

Screen LOCKIN_DEMOD.adl (two versions: P and Y)

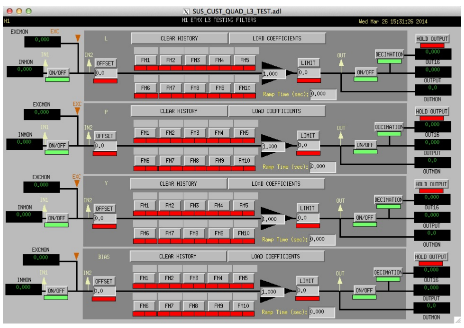




Suspensions/OpsManual/Boilerplate/LOCKIN_DEMOD:

Controls for the lockin demodulation. Comes in P and Y versions for pitch and yaw.

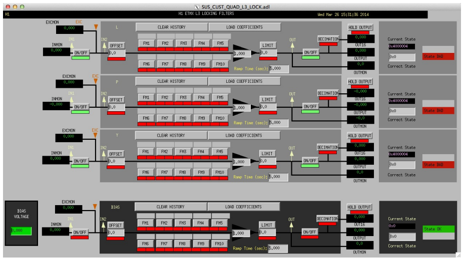
Screen SUS_CUST_QUAD_L3_TEST



Suspensions/OpsManual/Boilerplate/L3_TEST:

This is a deliberately empty set of 4 filter banks corresponding to the ESD L, P, Y and BIAS channels. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

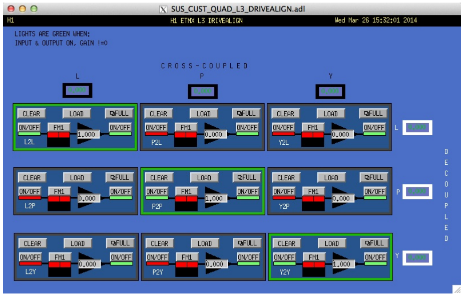
Screen SUS_CUST_QUAD_L3_LOCK



Suspensions/OpsManual/Boilerplate/L3_LOCK:

Filters for just the components of the locking signals to be applied at L3.

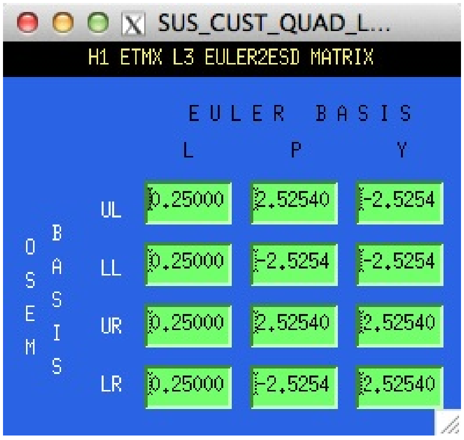
Screen SUS_CUST_QUAD_L3_DRIVEALIGN



Suspensions/OpsManual/Boilerplate/L3_DRIVEALIGN:

This screen is reserved for tweaking the L3 ESD actuator diagonalization and frequency response. There is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

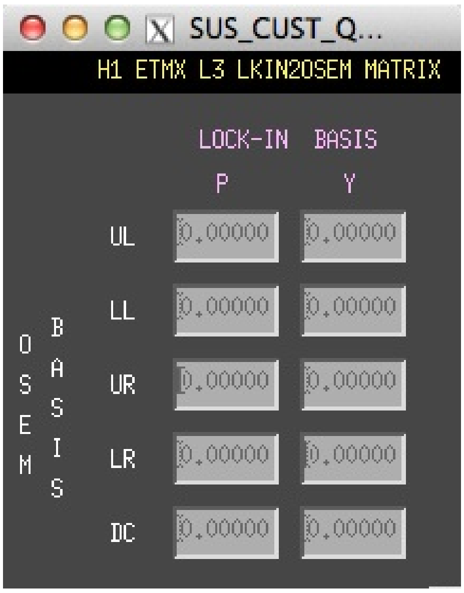
Screen SUS_CUST_QUAD_L3_EUL2ESD



Suspensions/OpsManual/Boilerplate/L3_EUL2ESD:

This screen allows entry of the matrix which converts force/torque requests from the L3 Euler basis to the L3 ESD quadrant basis. See Suspensions/OpsManual/Projections for more info. Lever arms have been taken into account but not any any calibration of the actuation train.

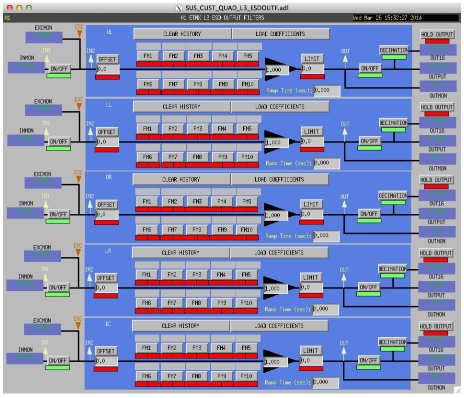
Screen SUS_CUST_QUAD_L3_LKIN2ESD.adl



Suspensions/OpsManual/Boilerplate/L3_LKIN2ESD:

This screen displays the transformation matrix from the lock-in request signals to the L3 ESD actuation basis. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_QUAD_L3_ESDOUTF



Suspensions/OpsManual/Boilerplate/L3_ESDOUTF:

This screen applies compensation for the hardware filters in the L3 ESD actuation electronics. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the QUAD_BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets can be applied temporarily to check that actuation is taking effect (and with the right sign).

aLIGO SUS Operations Manual - Models for QUAD Suspensions

Back to Operation Manual main page

The quad suspensions have been extensively modelled. There are Mathematica (T020205) and Matlab (T080188) programs that take a parameter set describing a single chain and calculate the dynamics. The parameter set needs to be formatted differently for Mathematica vs Matlab, but the parameter names are the same as far as was practical.

The quad Mathematica model and parameter sets for it live in the SUS SVN at `^/trunk/Common/MathematicaModels/QuadLite2Lateral/`. Parameter sets for Matlab live at `^/trunk/QUAD/Common/MatlabTools/QuadModel_Production`. Mark Barton maintains the Mathematica, Jeff Kissel maintains the Matlab. Unfortunately two different naming conventions are in use and it's difficult to tell what corresponds to what - if in doubt, ask. References given below are to Mathematica versions as subdirectories of `^/trunk/Common/MathematicaModels/QuadLite2Lateral/`.

Key cases of the quad model are given below, with results such as mode frequencies and mode shapes. T1200404 has transfer functions for many of the same models.

Explanation of the standard result set is at .../Key. Instructions for generating wiki pages for new cases of Mathematica models is at .../HowTo.

Main Chain (Monolithic)

.../mark.barton/20140304TMproductionTM is based on 20120601TMproductionTM but has many fitted numbers by Brett.

Main Chain (Wire Rehang of Broken Monolithic)

.../mark.barton/20120831TMproductionTMrehang is the LHO ITMy as rehung on wires after the fibre break incident.

Reaction Chain CP Version

.../mark.barton/20120831TMproductionCP is the CP-type reaction chain.

Reaction Chain ERM Version

.../mark.barton/20120831TMproductionERM is the ERM-type reaction chain.

Main Chain Metal Build

.../20121115TMproductionWire is the old-style metal build of the main chain.

.../20130523TMproductionWireLoop is the new-style metal build of the main chain, with prisms on the PUM and a wire loop under it.

Old Models

.../mark.barton/20120601TMproductionTM is the current production monolithic main chain.

/Old

Case 20140304TMproductionTM of Mathematica model QuadLite2Lateral

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
{"mark.barton", "20140304TMproductionTM"}
```

```
20140304TMproductionTM, equivalent to
^trunk/QUAD/Common/MatlabTools/QuadModel_Production/quadopt_fiber.m r6374 of 3/14/14, with
mostly Brett's fitted numbers.
```

Links to Result Sections

Pendulum Mode Summary	Violin Mode Summary	Pendulum Mode Shapes	Parameters

modeL1	modeT1	modeP1	modeV1	modeY1	modeR1
modeL2	modeT2	modeP2	modeY2	modeP3	modeL3
modeT3	modeV2	modeY3	modeR2	modeP4	modeY4
modeR3	modeL4	modeV3	modeT4	modeV4	modeR4

Pendulum Mode Summary

Back to Top

```
N    f          name     type

1    0.432096   modeL1   pitch3   pitch2   pitch1

2    0.461778   modeT1   y3       roll3    roll2

3    0.522439   modeP1   pitch3   pitch2

4    0.552861   modeV1   z3       z2

5    0.599185   modeY1   yaw3     yaw2

6    0.86847    modeR1   roll1    roll3    roll2

7    0.99234    modeL2   pitch0   pitch1   x2

8    1.04328    modeT2   y2       y1       y3

9    1.33906    modeP2   pitch0   pitch1

10   1.34913    modeY2   yaw3     yaw1

11   1.59901    modeP3   pitch0   pitch2

12   1.98325    modeL3   pitch0   x0       x1

13   2.10062    modeT3   roll1    y0       y1

14   2.22968    modeV2   z0       z1

15   2.39081    modeY3   yaw0     yaw2

16   2.64322    modeR2   pitch1   roll1    roll0

17   2.74352    modeP4   pitch1   pitch0

18   3.0388     modeY4   yaw1     yaw0
```

```
19   3.30053    modeR3   pitch0   roll0    roll1

20   3.39905    modeL4   x0       x1       pitch0

21   3.56066    modeV3   z1       z0

22   5.08788    modeT4   roll0    pitch0

23   9.68987    modeV4   z2       z3

24   13.8069    modeR4   roll2    roll3
```

Violin Mode Summary

Back to Top

Violin Modes	Top	UIM	PUM/PRM	TM/CP/ERM
f (Hz), n=1-4	335.963 672.964 1012.05 1354.29	495.47 992.132 1491.19 1993.85	421.319 843.578 1267.73 1694.72	268.347 537.034 806.401 1076.79

||Q, n=1-4||145974. 190003. 187191. 166336.||112538. 158582. 165672. 153803.||97703.3 145771. 160909. 156453.|| 9 1.20513 10 9 1.10271 10 8 9.65693 10 8 8.22564 10||

Mode Shapes

Mode #1 - modeL1

Back to Top

```
0.432096 Hz
```

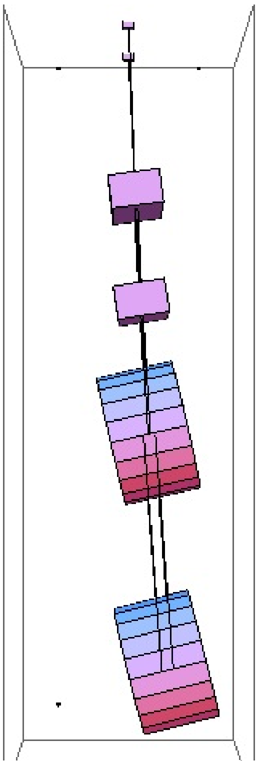
```
         x           y              z    yaw    pitch        roll

Mass N   0.0446664   0              0    0      -0.324194    0.00109208

Mass U   0.0791596   0.000232412    0    0      -0.380551    0.00221296

Mass 2   0.12786     0.000408359    0    0      -0.54992     0.00186767

optic    0.244087    0.000781052    0    0      -0.602881    0.00187027
```

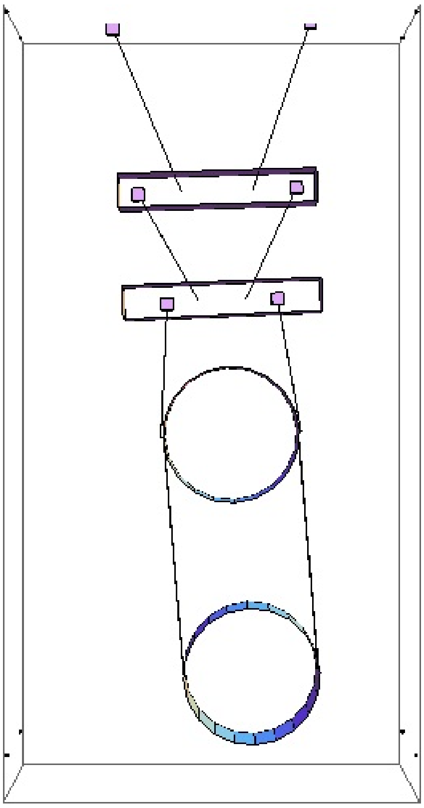


Mode #2 - modeT1

Back to Top

```
0.461778 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000632666	-0.0968249	0	0	0.0167644	-0.212624
Mass U	0.00107719	-0.183064	0	0	0.0138264	-0.292702
Mass 2	0.00169942	-0.329919	0	0.000113724	0.0135456	-0.360056
optic	0.00300257	-0.675626	0	0.000138669	0.0143016	-0.361108

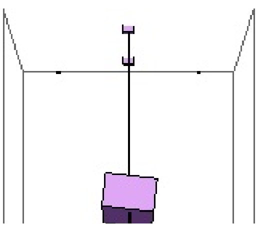


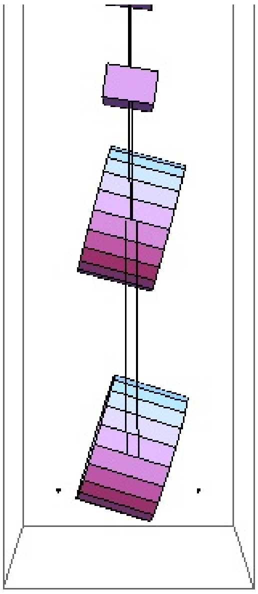
Mode #3 - modeP1

Back to Top

```
0.522439 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00693332	0.000119618	0	0	0.262043	-0.000983804
Mass U	0.0114391	0.000112371	0	0	0.341725	-0.00252828
Mass 2	0.0171067	0.000207678	0	0	0.593497	-0.00192483
optic	0.0216106	0.000502315	0	0	0.679238	-0.00192823



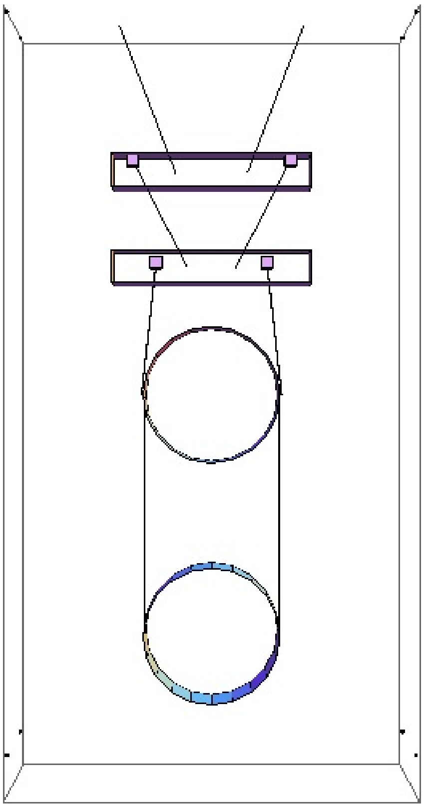


Mode #4 - modeV1

Back to Top

```
0.552861 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.265486	0	0	0
Mass U	0	0	0.47436	0	0	0
Mass 2	0	0	0.59153	0	0	0
optic	0	0	0.595477	0	0	0



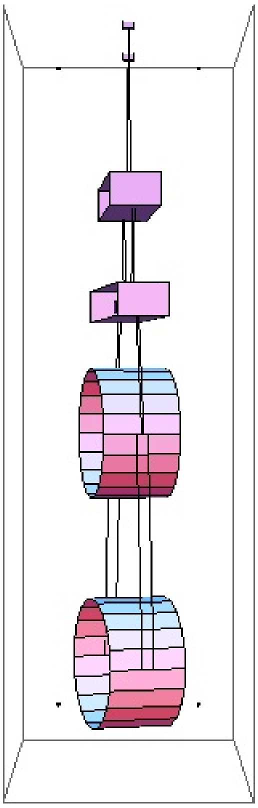
Mode #5 - modeY1

Back to Top

```
0.599185 Hz
```

0.599105 Hz

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0.170268	0	0
Mass U	0	0	0	0.388959	0	0
Mass 2	0	0	0	0.517768	0	0
optic	0	0	0	0.742723	0	0

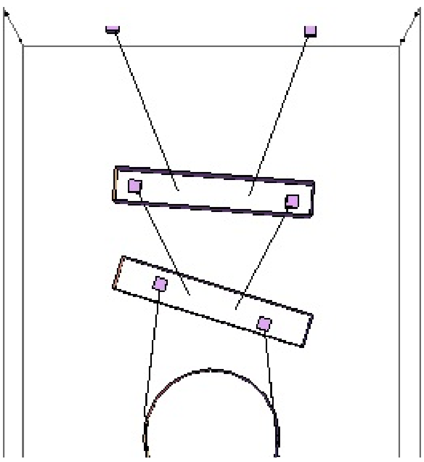


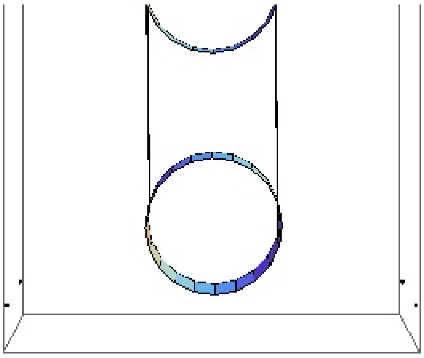
Mode #6 - modeR1

Back to Top

0.86847 Hz

	x	y	z	yaw	pitch	roll
Mass N	0.00025117	-0.0100391	0	0	-0.0513238	0.150309
Mass U	0.000491543	-0.00633961	0	0	-0.0345004	0.582232
Mass 2	0.000662311	-0.000190294	0	0	0.00920784	0.560864
optic	-0.000582965	-0.0106522	0	0.000231782	0.0223267	0.564962



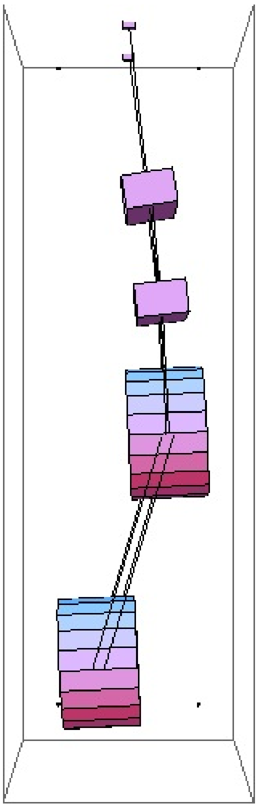


Mode #7 - modeL2

Back to Top

```
0.99234 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.201785	0.000831986	0	0	-0.60539	-0.00314325
Mass U	0.316891	0.00137866	0	0	-0.442235	-0.0290051
Mass 2	0.379499	0.00108046	0	0	-0.179399	-0.0310807
optic	-0.288347	-0.000458858	0	0	-0.186116	-0.0313911



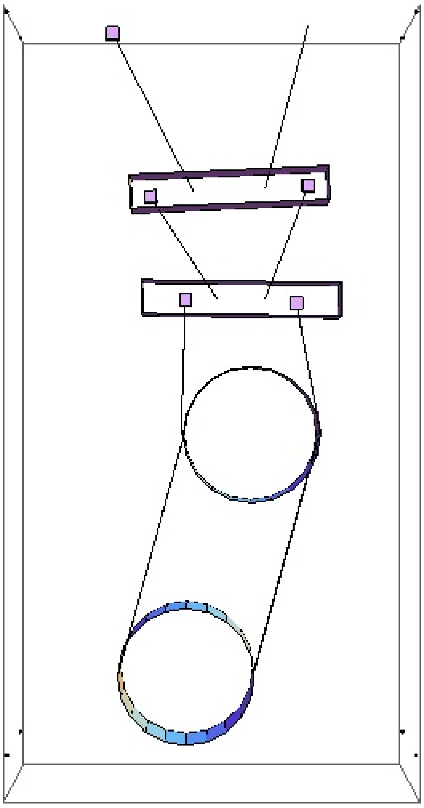
Mode #8 - modeT2

Back to Top

```
1.04328 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00131543	-0.288727	0	0.000293014	0.11462	-0.287145
Mass U	0.00186187	-0.477464	0	0.000136709	0.0528336	0.0912048
Mass 2	0.00215679	-0.640716	0	0	-0.00372088	0.0472575

optic	-0.00148265	0.407495	0	-0.000538818	-0.0261775	0.0495065

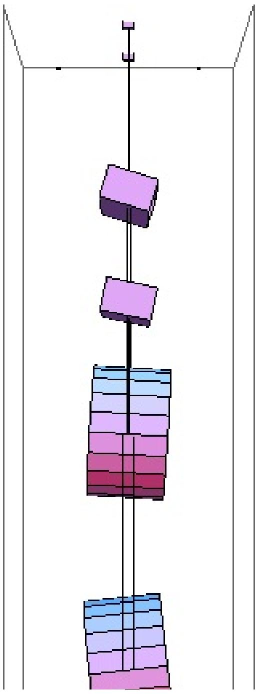


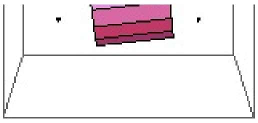
Mode #9 - modeP2

Back to Top

```
1.33906 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00261482	0.000549308	0	-0.000188425	0.774792	-0.00681833
Mass U	0.00228783	-0.000309567	0	-0.000286895	0.543672	0.0103221
Mass 2	0.00110235	0.000668843	0	-0.000192884	0.153168	0.0186825
optic	-0.000316125	-0.000294776	0	0.000376224	-0.28245	0.0190362



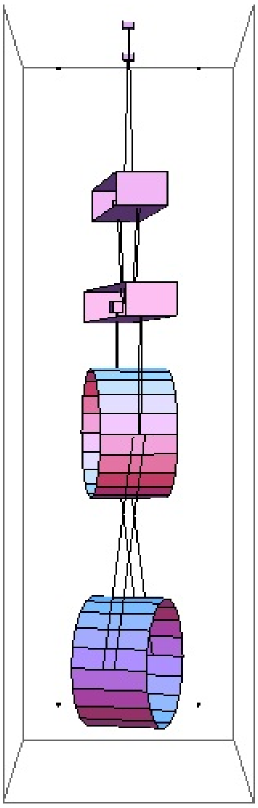


Mode #10 - modeY2

Back to Top

```
1.34913 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0.314305	0.00157893	-0.000168972
Mass U	0	0	0	0.53664	0.00117278	0.000121751
Mass 2	0	0	0	0.36969	0.000356302	0.000230763
optic	0	0	0	-0.690329	-0.00062761	0.000235194

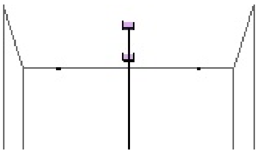


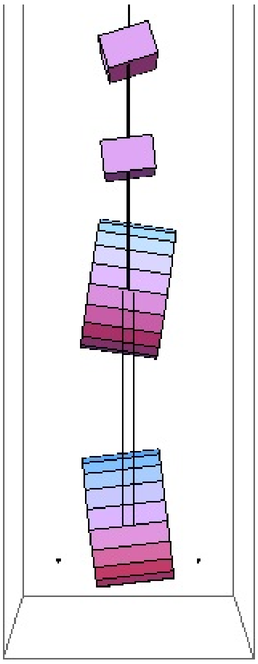
Mode #11 - modeP3

Back to Top

```
1.59901 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00234652	-0.00105381	0	0	-0.797996	0.0149651
Mass U	-0.00168769	0.000669621	0	0	-0.32356	-0.00298686
Mass 2	-0.00063393	-0.00058753	0	0	0.397123	-0.0156049
optic	0.000591277	0.000163941	0	0	-0.316323	-0.0160371



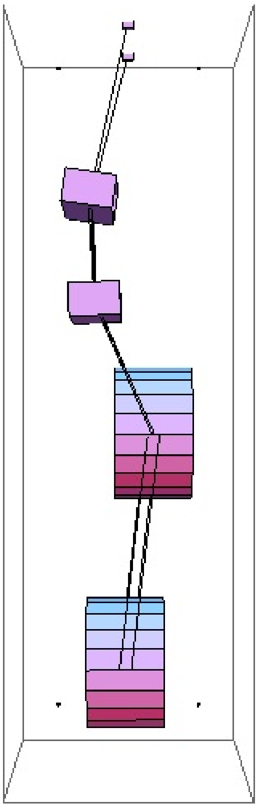


Mode #12 - modeL3

Back to Top

```
1.98325 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.480711	0.00294573	0	0	0.674011	-0.0269927
Mass U	-0.413098	-0.000345081	0	0	-0.215982	-0.0036095
Mass 2	0.306682	0	0	0	0.00538404	0.00948682
optic	-0.0370101	0	0	0	-0.0306104	0.00990393



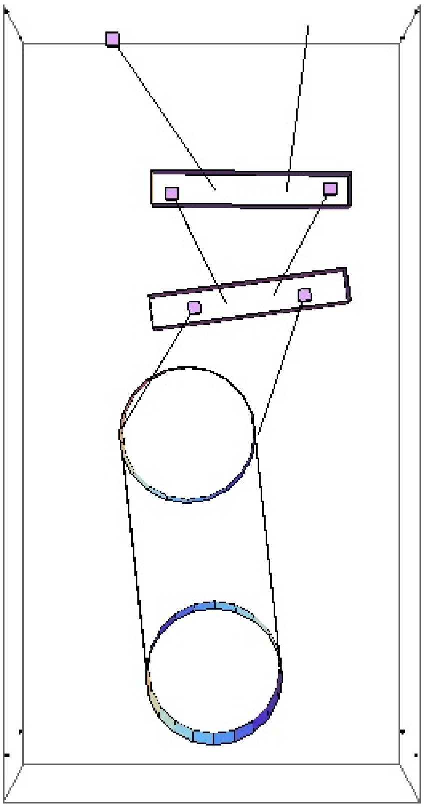
Mode #13 - modeT3

Back to Top

Back to Top

```
2.10062 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00161698	-0.480414	0	-0.000435999	-0.129032	0.0616061
Mass U	-0.00116536	-0.456548	0	0	0.0888022	-0.668225
Mass 2	0.000919667	0.287952	0	0.000353871	-0.013452	0.0299615
optic	-0.000109474	-0.0306128	0	-0.000129977	0.00443716	0.0309064

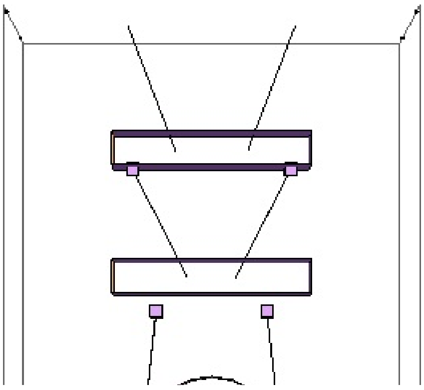


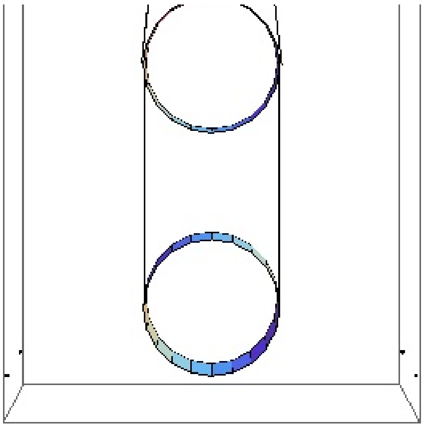
Mode #14 - modeV2

Back to Top

```
2.22968 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.831597	0	0	0
Mass U	0	0	0.471054	0	0	0
Mass 2	0	0	-0.19586	0	0	0
optic	0	0	-0.21953	0	0	0



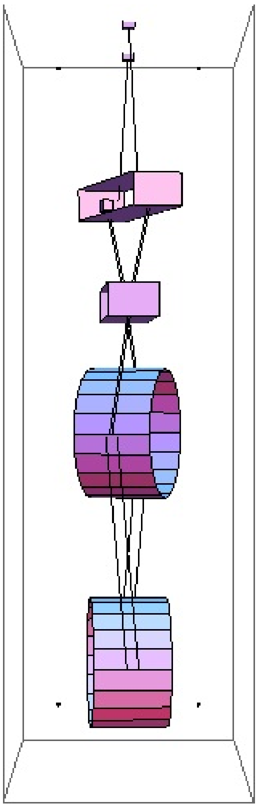


Mode #15 - modeY3

Back to Top

```
2.39081 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0.000307432	0	0.743546	-0.00130682	-0.00386038
Mass U	0	-0.000224069	0	0.125576	-0.00216639	-0.00239203
Mass 2	0	0	0	-0.635374	0.000212478	0.00185445
optic	0	0	0	0.166237	0	0.00197618



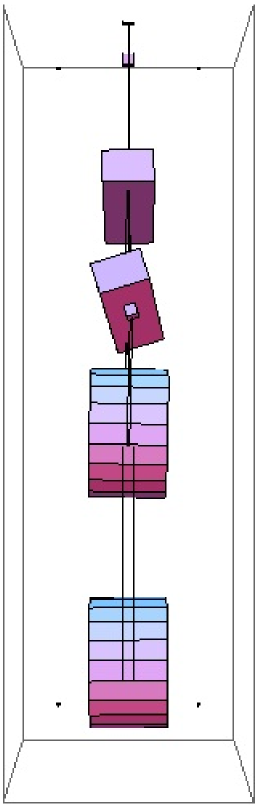
Mode #16 - modeR2

Back to Top

```
2.64322 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00125698	0.0423531	0	-0.0041309	-0.0373226	-0.452283

Mass U	0.000748585	-0.0251221	0	-0.00321687	-0.553322	-0.541333
Mass 2	-0.000869167	0.0015148	0	0.00556569	0.0407389	0.296386
optic	0	-0.000395266	0	-0.00113724	-0.00753511	0.320609

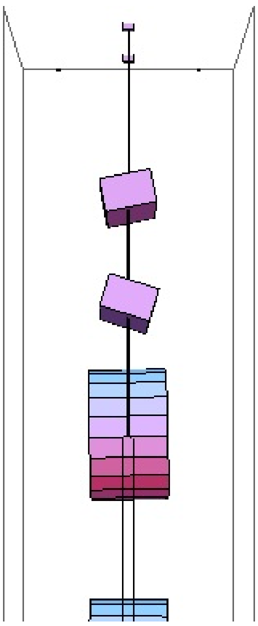


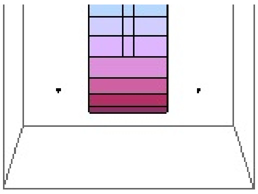
Mode #17 - modeP4

Back to Top

2.74352 Hz

	x	y	z	yaw	pitch	roll
Mass N	-0.00272613	0.000801139	0	-0.000108206	-0.554211	0.00470834
Mass U	-0.000274325	0.00261765	0	0	0.822741	-0.0896412
Mass 2	0.000942841	-0.00165665	0	0	-0.0548208	0.0467624
optic	0	0	0	0	0.00928134	0.0509137



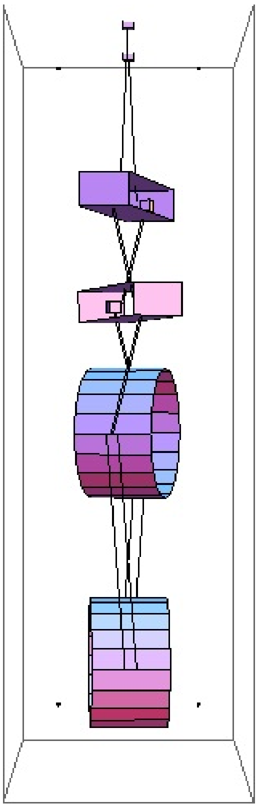


Mode #18 - modeY4

Back to Top

```
3.0388 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	-0.000127681	0	-0.535567	0.0026218	0.00126435
Mass U	0	0.000223349	0	0.653663	-0.00450946	-0.00484177
Mass 2	0	0	0	-0.528929	0.000231535	0.00182134
optic	0	0	0	0.0778953	0	0.00202428

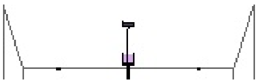


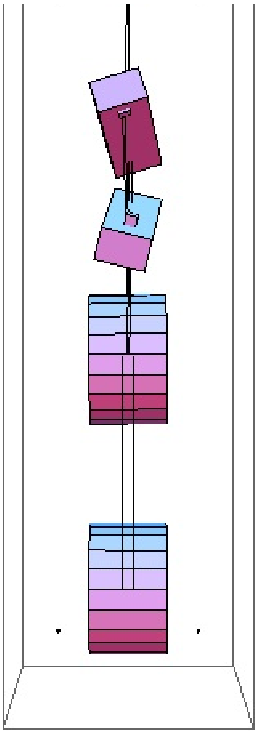
Mode #19 - modeR3

Back to Top

```
3.30053 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00613579	0.0729485	0	-0.00608276	-0.528465	-0.497251
Mass U	0.00505715	-0.0762194	0	0.00332943	0.429175	0.472648
Mass 2	-0.000551833	0.0157517	0	-0.00194057	-0.0180342	-0.153739
optic	0	-0.000538587	0	0.000236945	0.0020129	-0.17441



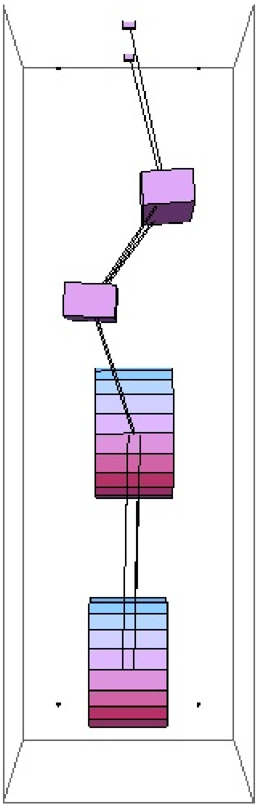


Mode #20 - modeL4

Back to Top

```
3.39905 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.587773	0.0204377	0	-0.00131728	-0.439788	-0.120964
Mass U	-0.569142	-0.0205561	0	0.000588286	0.317492	0.103643
Mass 2	0.087309	0.00382919	0	-0.000308675	-0.0130804	-0.0318959
optic	-0.00332659	-0.000126405	0	0	-0.000620275	-0.0364847

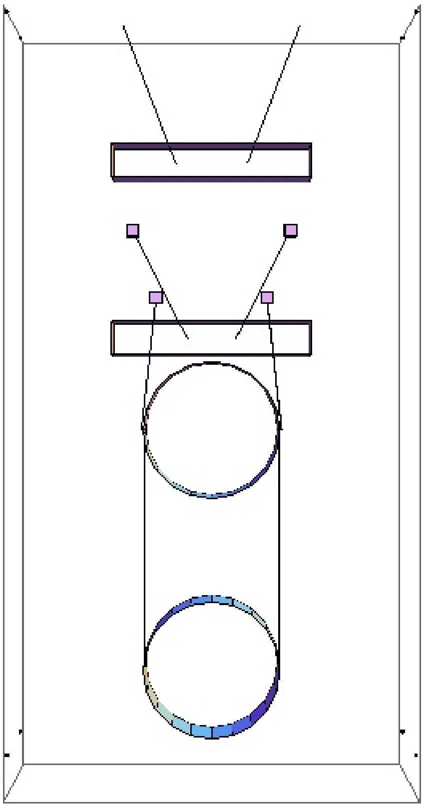


Mode #21 - modeV3

Back to Top

```
3.56066 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.560768	0	0	0
Mass U	0	0	-0.812684	0	0	0
Mass 2	0	0	0.0929675	0	0	0
optic	0	0	0.128226	0	0	0

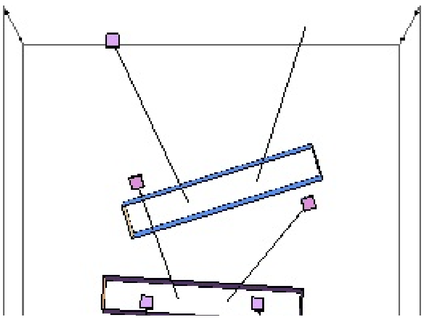


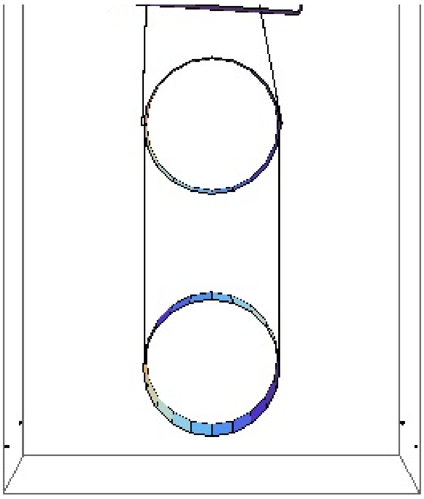
Mode #22 - modeT4

Back to Top

```
5.08788 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000147808	-0.0744417	0	-0.00414778	-0.499286	-0.835612
Mass U	-0.000263313	0.0654167	0	0.000467317	0.0986643	0.180529
Mass 2	0	-0.00401371	0	0	-0.00158016	-0.00982513
optic	0	0	0	0	0	-0.0136866



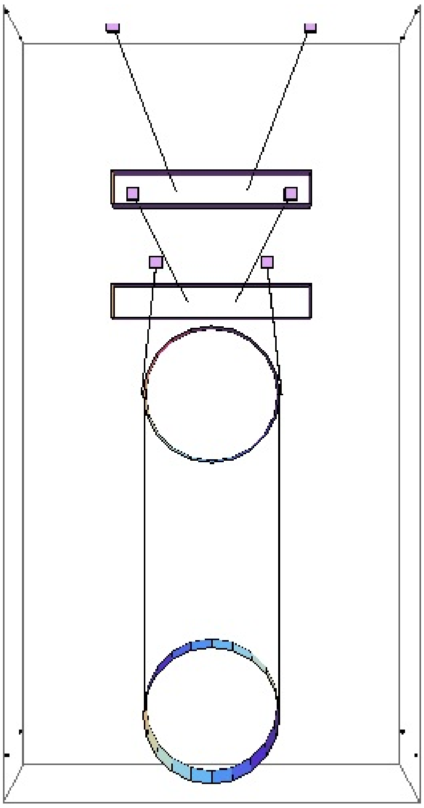


Mode #23 - modeV4

Back to Top

```
9.68987 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.00209572	0	0	0
Mass U	0	0	-0.0475533	0	0	0
Mass 2	0	0	0.718812	0	0	0
optic	0	0	-0.693574	0	0	0



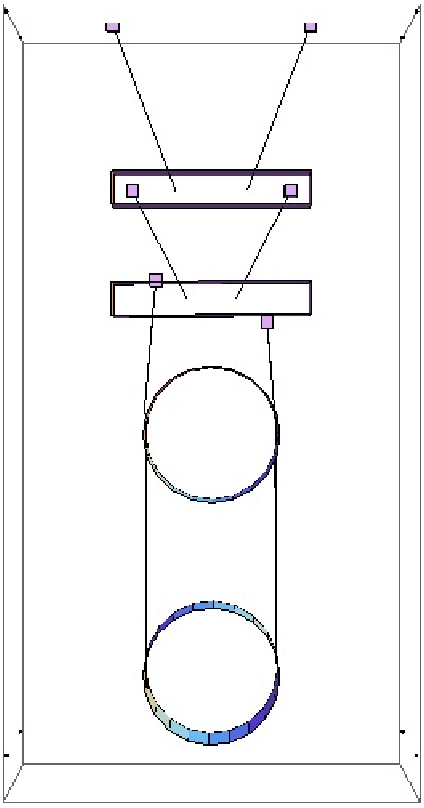
Mode #24 - modeR4

Back to Top

```
13.8069 Hz
```

	x	y	z	yaw	pitch	roll

Mass N	0	0	0	0	0.000251475	0.00039854
Mass U	0	0.000442164	0	0	-0.0033907	-0.0202281
Mass 2	0	-0.000236631	0	0	0	0.735944
optic	0	0	0	0	0	-0.676731



Parameters

Back to Top

```
g -> 9.81

nx -> 0.13                                                          T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

ny -> 0.5                                                           T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

nz -> 0.084                                                         T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

denn -> 683.94                                                      T1000286-v5 (info only)

mn -> 21.999                                                        measured value listed in
E1000186_v18_QUAD-2_Process_Traveler

Inx -> 0.454897                                                     T1000286-v9, correction from
Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

Iny -> 0.0749375                                                    T1000286-v9, correction from
Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

Inz -> 0.468032                                                     T1000286-v9, correction from
Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

ux -> 0.13                                                          T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

uy -> 0.5                                                           T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

uz -> 0.084                                                         T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)
```

```
den1 -> 740.934                                                   T1000286-v5 (info only)

m1 -> 21.999                                                      targeted - same as mn per
Betsy

I1x -> 0.524969                                                   T1000286-v9, 11.5 Main UI
Mass, After Addition of Pitch Adjuster (*1e-6 to get into kg/m^2, rotated into right-handed
coordinates (rotate 180deg about Z axis)), updated with Gauss-Newton Fit by Brett Shapiro on 14
Feb 2013

I1y -> 0.0813713                                                  T1000286-v9, 11.5 Main UI
Mass, After Addition of Pitch Adjuster (*1e-6 to get into kg/m^2, rotated into right-handed
coordinates (rotate 180deg about Z axis)), updated with Gauss-Newton Fit by Brett Shapiro on 14
Feb 2013

I1z -> 0.51776                                                    T1000286-v9, 11.5 Main UI
Mass, After Addition of Pitch Adjuster (*1e-6 to get into kg/m^2, rotated into right-handed
coordinates (rotate 180deg about Z axis)), updated with Gauss-Newton Fit by Brett Shapiro on 14
Feb 2013

ix -> 0.2                                                         T1000405-v1, BNS 5/15/09

ir -> 0.17                                                        T1000405-v1, BNS 5/15/09

den2 -> 2201
http://www.sciner.com/Opticsland/FS.htm (T1000405-v1 says 2200)

m2 -> 39.633                                                      E1000186_v18_QUAD-
2_Process_Traveler - metal mass

I2x -> 0.552453                                                   Gauss-Newton Fit by Brett
Shapiro on 14 Feb 2013

I2y -> 0.456591                                                   Gauss-Newton Fit by Brett
Shapiro on 14 Feb 2013

I2z -> 0.412188                                                   Gauss-Newton Fit by Brett
Shapiro on 14 Feb 2013

tx -> 0.2                                                         T1000405-v1

tr -> 0.17                                                        T1000405-v1

den3 -> 2201
http://www.sciner.com/Opticsland/FS.htm

m3 -> 39.631                                                      E1000186_v18_QUAD-
2_Process_Traveler - metal mass

I3x -> 0.580252                                                   Gauss-Newton Fit by Brett
Shapiro on 14 Feb 2013

I3y -> 0.382096                                                   Gauss-Newton Fit by Brett
Shapiro on 14 Feb 2013
calculated from shape, density

I3z -> 0.413514                                                   Gauss-Newton Fit by Brett
Shapiro on 14 Feb 2013

ln -> 0.449009                                                    derived from tlnspec, d's,
previously 0.449192

l1 -> 0.308373                                                    derived from tl1spec, d's,
previously 0.308585

l2 -> 0.339908                                                    derived from tl2spec, d's,
previously 0.330787

l3 -> 0.594578                                                    derived from tl3spec, d's,
previously 0.5820

nwn -> 2

nw1 -> 4

nw2 -> 4

nw3 -> 4

rn -> 0.00055                                                     T1000428-v2

r1 -> 0.0003555                                                   T1000428-v2
```

r2 -> 0.0003175 T1000428-v2

r3 -> 0.0002 T1000428-v2, midsection

Yn -> $2.12\ 10^{11}$ measured, MB, 11/18/05, via IFOModel v4.1; cf. 2.2 in T010103-05

Y1 -> $2.12\ 10^{11}$ measured, MB, 11/18/05, via IFOModel v4.1; cf. 2.2 in T010103-05

Y2 -> $2.12\ 10^{11}$ measured, MB, 11/18/05, via IFOModel v4.1; cf. 2.2 in T010103-05

Y3 -> $7.2\ 10^{10}$ IFOModel v4.1, unchanged, different from new Ysilica

dm -> -0.00353391 effective value 0.001

dn -> 0.00399036 effective value 0.002 after blade correction, Jeff's increase of 0.001; % (1-0.22438)/1000 correction from Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

d0 -> -0.00175221 effective value 0.001

d1 -> 0.0057902 effective value 0.002 after blade correction, Jeff's increase of 0.001; (1+1.8536)/1000 correction from Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

d2 -> -0.00262327 0.0003 effective per M080134-v2

d3 -> 0.00614069 0.010 effective per M080134-v2 (equivalent to Brett's d2=0.010, flex3=0)

d4 -> 0.00128109 0.010 effective per M080134-v2 (equivalent to Brett's d3=0.010, flex3=0)

sn -> 0 T1000405-v1 (unused)

su -> 0.003 T1000405-v1

si -> 0.003 T1000405-v1

sl -> 0.015 T1000405-v1

nn0 -> 0.25 T1000405-v1

nn1 -> 0.09 T1000405-v1

n0 -> 0.2 T1000405-v1

n1 -> 0.06 T1000405-v1

n2 -> 0.14 T1000405-v1

n3 -> 0.1775 from drawings, BNS 22 June 2010

n4 -> 0.17025 Email from Marielle to BNS 4/27/10

n5 -> 0.17025 Email from Marielle to BNS 4/27/10

tln -> 0.416 derived from l's, d's, should match tlnspec

tl1 -> 0.277 derived from l's, d's, should match tl1spec

tl2 -> 0.341 derived from l's, d's, should match tl2spec

tl3 -> 0.602 derived from l's, d's, should match tl3spec

ltotal -> 1.636

```
bd -> 0.

unstretched -> False

vertblades -> True

matlabcompat -> False

uln -> 0.447566

ul1 -> 0.307352

ul2 -> 0.338918

ul3 -> 0.589189

sln -> 0.449009

sl1 -> 0.308373

sl2 -> 0.339908

sl3 -> 0.594578

sin -> -0.356341

si1 -> -0.453995

si2 -> 0.110324

si3 -> 0.

cn -> 0.934356

c1 -> 0.891004

c2 -> 0.993896

c3 -> 1.

pitchbul -> 0

pitchbur -> 0

pitchbil -> 0

pitchbir -> 0

pitchbll -> 0

pitchblr -> 0

rollbul -> 0

rollbur -> 0

rollbil -> 0

rollbir -> 0

rollbll -> 0

rollblr -> 0

                -7
An -> 9.50332 10

                -7
A1 -> 3.97035 10

                -7
A2 -> 3.16692 10

                -7
A3 -> 1.25664 10

kwn -> 448700.

kw1 -> 272953.
```

kw2 -> 197520.

kw3 -> 18034.7 Brett: grad_descent_fit_z_bfgs_28July2010_part1 divided by 2 wires per side, updated by Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

flexn -> 0.00453391 calculated as usual

flex1 -> 0.00275221 calculated as usual

flex2 -> 0.00292327 calculated as usual

flex3 -> 0.00385931 calculated as usual

kbuz -> 1444.29 Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

kbiz -> 1666.68 Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

kblz -> 2437.53 Gauss-Newton Fit by Brett Shapiro on 14 Feb 2013

kbux -> 100000. as for middle

kbix -> 100000. Justin 11/29/05

kblx -> 80000. Ian 12/09/05

bdu -> 0.418615

bdi -> 0.298015

bdl -> 0.159502

mn3 -> 123.262 calculated as usual

m13 -> 101.263 calculated as usual

m23 -> 79.264 calculated as usual

Inxy -> -0.0375599 T1000286-v9, 11.1 Main Top Mass, (*1e-6 to get into kg/m^2, rotated into right-handed coordinates (rotate 180deg about Z axis))

Inyz -> -0.0000465463 T1000286-v9, 11.1 Main Top Mass, (*1e-6 to get into kg/m^2, rotated into right-handed coordinates (rotate 180deg about Z axis))

Inzx -> -0.00171841 T1000286-v9, 11.1 Main Top Mass, (*1e-6 to get into kg/m^2, rotated into right-handed coordinates (rotate 180deg about Z axis))

COM0x -> 0

COM0y -> 0

COM0z -> 0

FRP0x -> 0

FRP0y -> 0

FRP0z -> 0

I1xy -> -0.0132064 T1000286-v9, 11.5 Main UI Mass, After Addition of Pitch Adjuster (*1e-6 to get into kg/m^2, rotated into right-handed coordinates (rotate 180deg about Z axis))

I1yz -> 0.0000137417 T1000286-v9, 11.5 Main UI Mass, After Addition of Pitch Adjuster (*1e-6 to get into kg/m^2, rotated into right-handed coordinates (rotate 180deg about Z axis))

I1zx -> $-8.08401\ 10^{-6}$ T1000286-v9, 11.5 Main UI Mass, After Addition of Pitch Adjuster (*1e-6 to get into kg/m^2, rotated into right-handed coordinates (rotate 180deg about Z axis))

COM1x -> 0

COM1y -> 0

```
COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

I2xy -> 0

I2yz -> 0

I2zx -> 0

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0

I3yz -> 0

I3zx -> 0

COM3x -> 0

COM3y -> 0

COM3z -> 0

FRP3x -> 0

FRP3y -> 0

FRP3z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03

                 -14
Mn1 -> 7.18688 10

                 -14
Mn2 -> 7.18688 10

                 -14
M11 -> 1.25444 10

                 -14
M12 -> 1.25444 10

                 -15
M21 -> 7.98114 10

                 -15
M22 -> 7.98114 10

                 -14
M31 -> 2.01062 10

                 -14
M32 -> 2.01062 10

temperature -> 290.

                        -23
boltzmann -> 1.38066 10

                      -7
alphasilica -> 3.9 10                                             AH measurement, summary email
by MB, 3/13/08
```

betasilica -> 0.000152	IFOModel v4.1
rhosilica -> 2200.	IFOModel v4.1
Csilica -> 772.	IFOModel v4.1
Ksilica -> 1.38	IFOModel v4.1
Ysilica -> $7.27\ 10^{10}$	IFOModel v4.1
phisilica -> $4.1\ 10^{-10}$	IFOModel v4.1
phissilica -> $3.\ 10^{-11}$	surface
rhosteel -> 7800.	IFOModel v4.1
Csteel -> 460.	IFOModel v4.1
Ksteel -> 49.	IFOModel v4.1
Ysteel -> $2.12\ 10^{11}$	measured, MB, 11/18/05
alphasteel -> 0.000012	IFOModel v4.1
betasteel -> -0.00025	IFOModel v4.1
phisteel -> 0.0001	IFOModel v4.1
rhomarag -> 7800.	IFOModel v4.1
Cmarag -> 460.	IFOModel v4.1
Kmarag -> 20.	IFOModel v4.1
Ymarag -> $1.87\ 10^{11}$	IFOModel v4.1
alphamarag -> 0.000011	IFOModel v4.1
betamarag -> -0.00025	Geppo's value - Bench v4.1 is wrong
phimarag -> 0.0001	IFOModel v4.1
tmU -> 0.0043	IFOModel v4.1
tmI -> 0.0046	IFOModel v4.1
tmL -> 0.0042	IFOModel v4.1
magicnumber -> 0.0737472	Zener, 1938, Phys. Rev. 53:90-99
deltabladeU -> 0.00182883	
deltabladeI -> 0.00182883	
deltabladeL -> 0.00182883	
deltawireU -> 0.00278559	
deltawireI -> 0.00278009	
deltawireL -> 0.00277408	
deltafibre -> $2.5317\ 10^{-9}$	
taubladeU -> 0.336093	
taubladeI -> 0.384626	
taubladeL -> 0.320643	
tauwireU -> 0.00653413	
tauwireI -> 0.00272987	

```
tauwireI -> 0.00272987

tauwireL -> 0.00217746

taufibre -> 0.058088

                                         0.003862 #1
damping[imag, bladeUtype] -> (0.0001 + --------------- & )
                                                     2
                                       1 + 4.45943 #1

                                        0.00441968 #1
damping[imag, bladeItype] -> (0.0001 + --------------- & )
                                                     2
                                       1 + 5.84032 #1

                                        0.00368446 #1
damping[imag, bladeLtype] -> (0.0001 + --------------- & )
                                                     2
                                       1 + 4.05884 #1

damping[imag, wireUtype] -> (0.0001 & )

damping[imag, wireItype] -> (0.0001 & )

damping[imag, wireLtype] -> (0.0001 & )

                                         0.000114363 #1
damping[imag, wireUatype] -> (0.0001 + ------------------ & )
                                                        2
                                       1 + 0.00168552 #1

                                        0.0000476847 #1
damping[imag, wireIatype] -> (0.0001 + ----------------- & )
                                                       2
                                       1 + 0.0002942 #1

                                        0.0000379532 #1
damping[imag, wireLatype] -> (0.0001 + ------------------ & )
                                                        2
                                       1 + 0.00018718 #1

                                     -8
damping[imag, fibretype] -> (3.116 10   & )

                                                       -10
                                      -8   9.24013 10    #1
damping[imag, fibreatype] -> (6.191 10   + ---------------- & )
                                                          2
                                           1 + 0.133209 #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx0 -> 0

kcony0 -> 0

kconz0 -> 0

kconyaw0 -> 0

kconpitch0 -> 0

kconroll0 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0
```

```
kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

ribbons -> False

dumbbell -> True

Inxz -> -0.00171841

Inzy -> -0.0000465463

Inyx -> -0.0375599

                   -6
I1xz -> -8.08401 10

I1zy -> 0.0000137417

I1yx -> -0.0132064

if -> 0.                                                          M050397-02/T010103-05

m2alt -> 39.9667                                                  calculated from shape, density

I2xalt -> 0.577518                                                calculated from shape, density

I2yalt -> 0.421981                                                calculated from shape, density
calculated from shape, density, cf. Brett's calc

I2zalt -> 0.421981                                                calculated from shape, density

tf -> 0.095                                                       M050397-02/T010103-05 (unused)

m3alt -> 39.5876                                                  calculated from shape, density

I3xalt -> 0.566907                                                calculated from shape,
density, cf. Brett's calc

I3zalt -> 0.410278                                                calculated from shape, density

tlnspec -> 0.416                                                  NR 4/3/06, T010103-05

tl1spec -> 0.277                                                  NR 4/3/06, T010103-05

tl2spec -> 0.341                                                  NR 4/3/06, T010103-05

tl3spec -> 0.602                                                  NR 4/3/06, T010103-05
derived from vertical heights, d's                  derived from vertical heights, d's   derived
from vertical heights, d's   derived from vertical heights, d's

r3m -> 0.0002                                                     T1000428-v2, midsection
T1000428-v2, midsection

r3n -> 0.0004                                                     800 µm diameter end sections
800 µm diameter end sections
```

```
t3m -> t3

W3m -> W3

                        1
t3n -> 3.11761 Sqrt[---------]
                    optstress

                        1
W3n -> 31.1761 Sqrt[---------]
                    optstress

r3m -> 0.0002                                                    T1000428-v2, midsection
T1000428-v2, midsection

r3n -> 0.0004                                                    800 µm diameter end sections
800 µm diameter end sections

                 -7
A3n -> 5.02655 10

                 -7
A3m -> 1.25664 10

nf -> 0.0257732                                                  l3/0.015

kw3alt -> 15829.1                                                original calculation, should
roughly match kw3

                 6
kw3n -> 2.3617 10                                                original calculation: net
longitudinal elasticity of one fibre neck

kw3m -> 16044.2                                                  original calculation: net
longitudinal elasticity of fibre midsection

ffn -> 0.807                                                     from Ian's data, 11/30/05,
linear fit version (not used)

ff1 -> 0.641                                                     from Ian's data, 11/30/05,
linear fit version (not used)

ff2 -> 0.608                                                     from Ian's data, 11/30/05,
linear fit version (not used)

kffn -> 0.69223                                                  not used

kff1 -> 0.67339                                                  not used

kff2 -> 1.06749                                                  not used

bend1 -> 0.00496695                                              calculated as usual

bend2 -> 0.00485987                                              calculated as usual

thetan -> -20.8756                                               calculated as usual

theta1 -> -27.0003                                               calculated as usual

theta2 -> 6.33399                                                calculated as usual

theta3 -> 0.                                                     calculated as usual

sigmasilica -> 0.17

                     10
Gsilica -> 3.10684 10                                            shear modulus

dssilica -> 0.015                                                IFOModel v4.1
```

Back to Top

Case 20120601TMproductionTMrehang of Mathematica model QuadLite2Lateral

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
{"mark.barton", "20120601TMproductionTMrehang"}
```

```
20120601TMproductionTMrehang, based on
^trunk/QUAD/Common/MatlabTools/QuadModel_Production/quadopt_fiber.m r2731 of 6/1/12, with
changes for wire rehang. Exported Matlab uploaded as
^trunk/QUAD/Common/MatlabTools/QuadModel_Production/quadopt_wirerehang.m
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeL1	modeT1	modeV1	modeP1	modeY1	modeR1
modeL2	modeT2	modeY2	modeP2	modeL3	modeT3
modeV2	modeY3	modeP3	modeR2	modeP4	modeY4
modeR3	modeL4	modeV3	modeT4	modeV4	modeR4

Mode Summary

Back to Top

```
N    f          name     type

1    0.434672   modeL1   pitch3   pitch2   pitch1

2    0.461331   modeT1   y3       roll3    roll2

3    0.548679   modeV1   z3       z2

4    0.562986   modeP1   pitch3   pitch2

5    0.601232   modeY1   yaw3     yaw2

6    0.838875   modeR1   roll1    roll2    roll3

7    0.983038   modeL2   x2       pitch0   x1       x3

8    1.03606    modeT2   y2       roll1    roll3    y1

9    1.36168    modeY2   yaw3     yaw1

10   1.44161    modeP2   pitch0   pitch1

11   2.00486    modeL3   x0       pitch0   x1

12   2.12165    modeT3   roll1    y0       y1

13   2.22184    modeV2   z0       z1

14   2.38749    modeY3   yaw0     yaw2

15   2.52198    modeP3   pitch1   pitch2   pitch3

16   2.63003    modeR2   roll1    roll0    pitch0

17   2.84646    modeP4   pitch1
```

18	3.03675	modeY4	yaw1	yaw2	
19	3.32094	modeR3	pitch0	pitch1	roll0
20	3.4283	modeL4	x1	x0	
21	3.56765	modeV3	z1	z0	
22	5.08241	modeT4	roll0	pitch0	
23	17.1158	modeV4	z3	z2	
24	24.011	modeR4	roll3	roll2	

Top	UIM	PUM/PRM	TM/CP/ERM
336.193	495.765	434.887	229.703

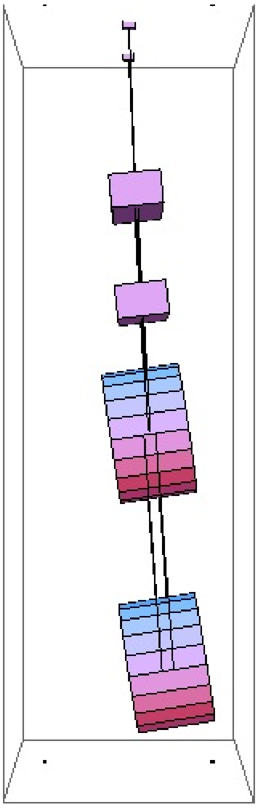
Mode Shapes

Mode #1 - modeL1

Back to Top

```
0.434672 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.0642433	0	0	0	-0.352924	0.00122569
Mass U	0.113647	0.000276062	0	0	-0.404523	0.00246869
Mass 2	0.185186	0.000493338	0	0	-0.505757	0.00212929
optic	0.342454	0.000916115	0	0	-0.536084	0.00209195

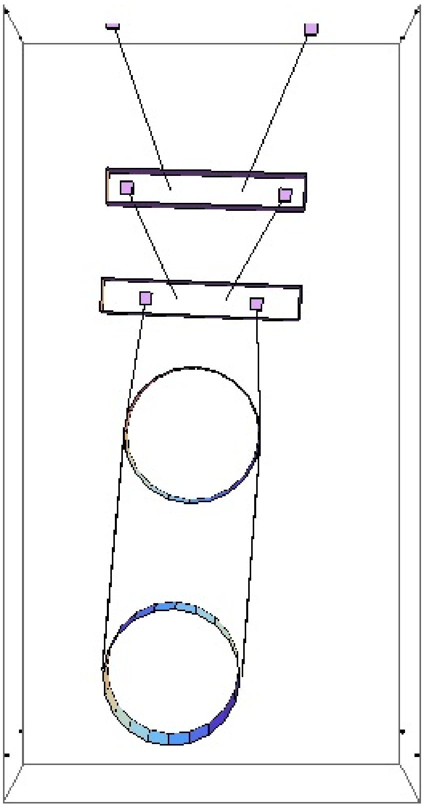


Mode #2 - modeT1

Back to Top

```
0.461331 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000473114	0.0983126	0	0	-0.0156734	0.212239
Mass U	-0.000796534	0.185651	0	0	-0.0128238	0.283487
Mass 2	-0.00117472	0.332391	0	-0.000112149	-0.0093873	0.349897
optic	-0.00237665	0.68544	0	-0.000135751	-0.00997622	0.356529

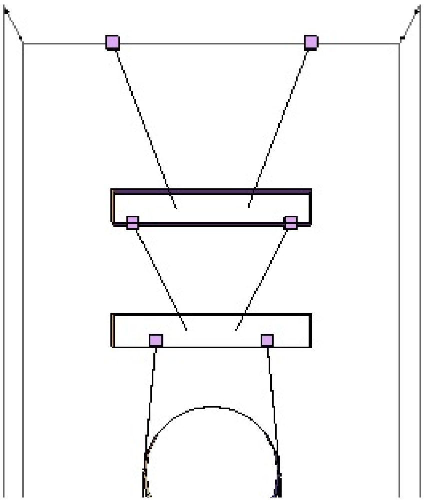


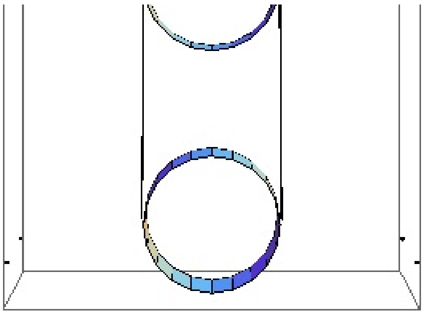
Mode #3 - modeV1

Back to Top

```
0.548679 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	-0.265139	0	0	0
Mass U	0	0	-0.473953	0	0	0
Mass 2	0	0	-0.593138	0	0	0
optic	0	0	-0.594354	0	0	0



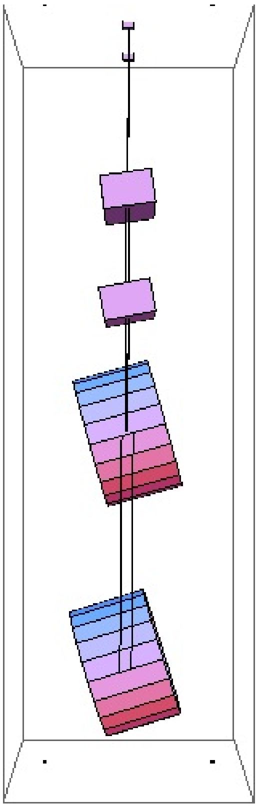


Mode #4 - modeP1

Back to Top

```
0.562986 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00461745	-0.000123956	0	0	-0.29791	0.00152298
Mass U	-0.00721975	0	0	0	-0.390559	0.00397715
Mass 2	-0.00522449	-0.000109764	0	0	-0.584804	0.0032626
optic	-0.0170184	-0.000433477	0	0	-0.645205	0.00316299

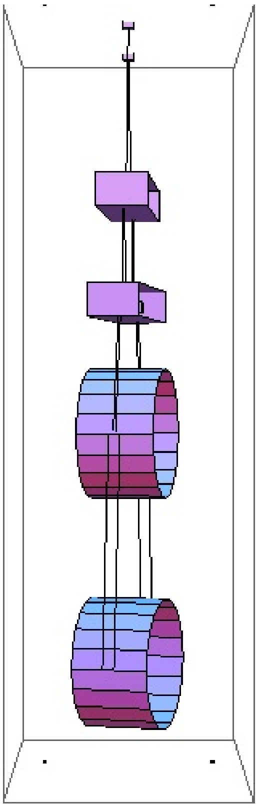


Mode #5 - modeY1

Back to Top

```
0.601232 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	-0.17159	0	0
Mass U	0	0	0	-0.391659	0	0
Mass 2	0	0	0	-0.520721	0	0
optic	0	0	0	-0.738924	0	0

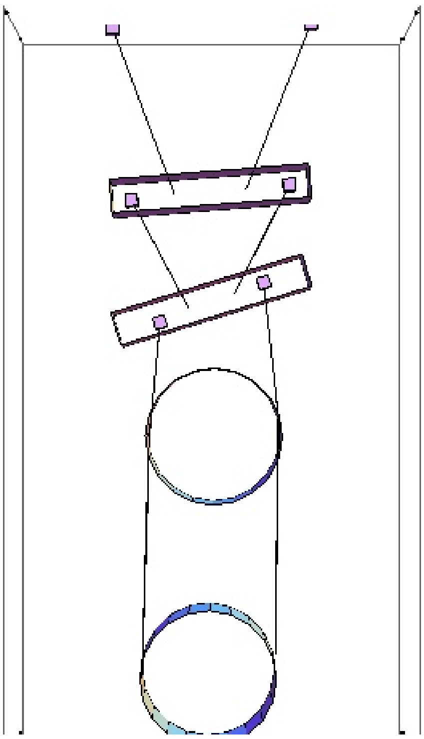


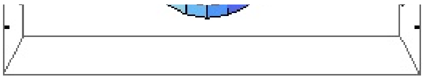
Mode #6 - modeR1

Back to Top

```
0.838875 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000116682	0.00714527	0	0	0.0399381	-0.150798
Mass U	-0.000264177	0.00163354	0	0	0.0257692	-0.575457
Mass 2	-0.000145973	-0.010125	0	0	-0.0167192	-0.574572
optic	0.000154311	0.0169859	0	-0.00023419	-0.0213725	-0.559049



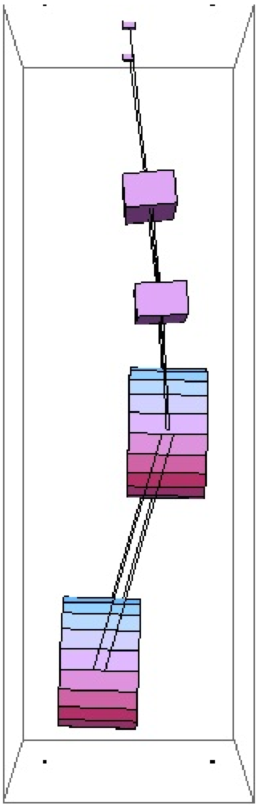


Mode #7 - modeL2

Back to Top

```
0.983038 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.269198	0.000130037	0	0	-0.479671	-0.0010352
Mass U	0.423505	0.00033879	0	0	-0.263792	-0.015601
Mass 2	0.509847	-0.000156363	0	0	0.109199	-0.0179657
optic	-0.382187	0.000163722	0	0	0.172327	-0.0175184

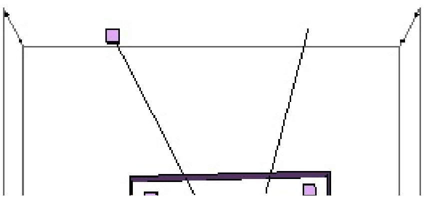


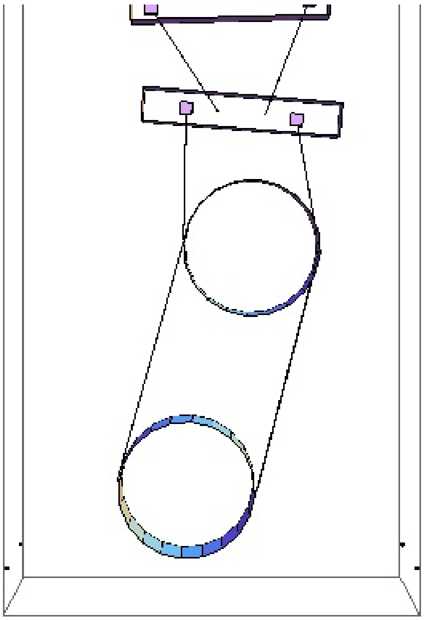
Mode #8 - modeT2

Back to Top

```
1.03606 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000151757	-0.231595	0	0.000131933	0.0255394	-0.136625
Mass U	0.000205531	-0.3771	0	0	-0.00175871	0.406665
Mass 2	0.000282397	-0.501522	0	0	0.000472397	0.352417
optic	-0.00017666	0.314216	0	-0.000241901	0.000719875	0.380797



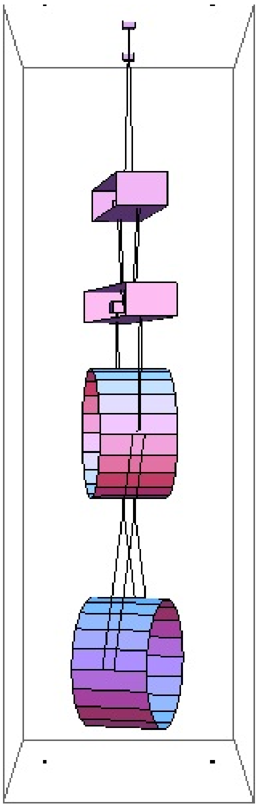


Mode #9 - modeY2

Back to Top

```
1.36168 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0.316017	-0.000335618	-0.000167041
Mass U	0	0	0	0.533703	-0.000166216	0
Mass 2	0	0	0	0.358976	0	0.000169956
optic	0	0	0	-0.697446	0	0.000165743

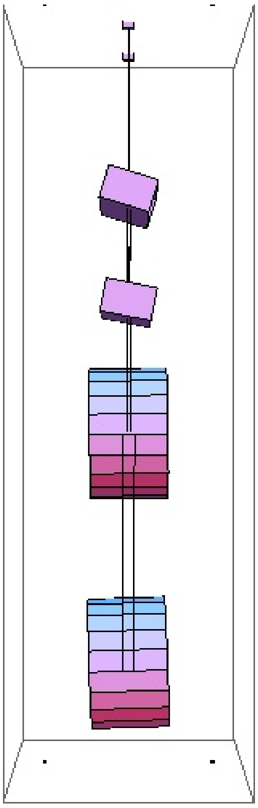


Mode #10 - modeP2

Back to Top

```
1.44161 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00174614	0.00069983	0	0	0.836533	-0.0113063
Mass U	0.000736192	-0.000621847	0	0	0.538097	0.00520078
Mass 2	0.00126232	0.000591362	0	0	-0.0322791	0.0176108
optic	-0.000345623	-0.000169699	0	0	-0.0941025	0.0172121

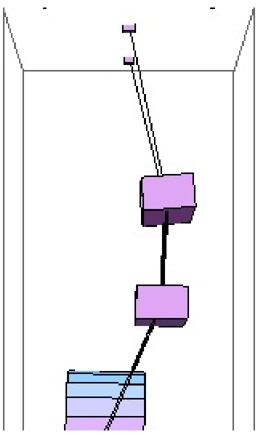


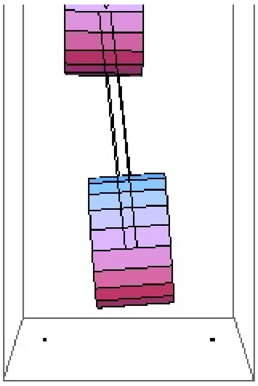
Mode #11 - modeL3

Back to Top

```
2.00486 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.530904	-0.00230943	0	0	-0.480014	0.0206289
Mass U	0.446104	0.00020156	0	0	0.166403	0.0039239
Mass 2	-0.32254	0	0	0	0.109679	-0.006715
optic	0.0370215	0	0	0	-0.378234	-0.00662102



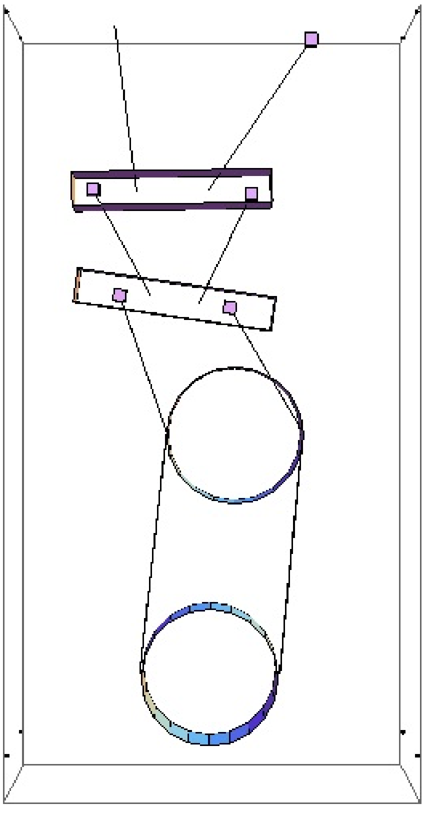


Mode #12 - modeT3

Back to Top

```
2.12165 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00140359	0.470303	0	0.000575722	0.150619	-0.0738388
Mass U	0.00093564	0.438228	0	0	-0.0997454	0.6877
Mass 2	-0.000704193	-0.269125	0	-0.000466888	-0.00620238	0.0269466
optic	0	0.027236	0	0.00017439	0.014372	0.0400893



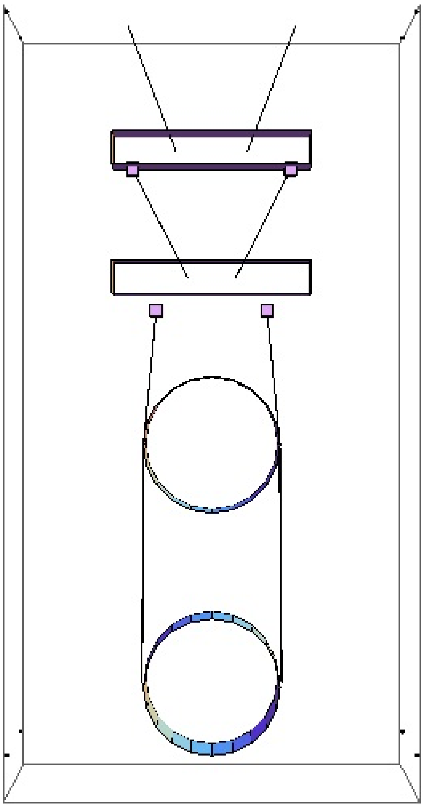
Mode #13 - modeV2

Back to Top

```
2.22184 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.834953	0	0	0
Mass U	0	0	0.469229	0	0	0

Mass 2	0	0	-0.19982	0	0	0
optic	0	0	-0.206759	0	0	0

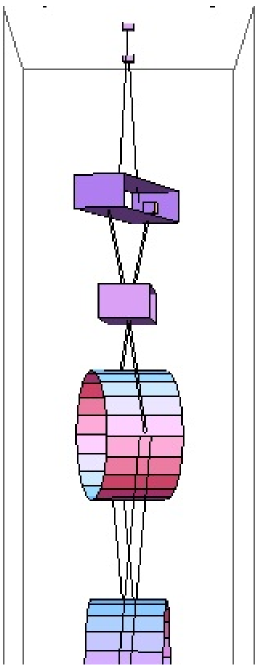


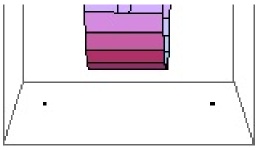
Mode #14 - modeY3

Back to Top

```
2.38749 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	-0.00031376	0	-0.745163	0.00109089	0.00416736
Mass U	0	0.00024736	0	-0.118619	0.00276715	0.00293885
Mass 2	0	0	0	0.632971	0.000607062	-0.00191034
optic	0	0	0	-0.173109	-0.000744421	-0.00188961



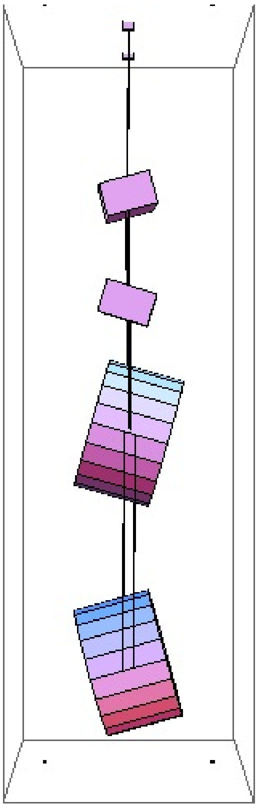


Mode #15 - modeP3

Back to Top

```
2.52198 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00641497	-0.00646401	0	0.00137481	-0.465402	0.0867923
Mass U	-0.00177261	0.00588962	0	0.000540847	0.538621	0.0607729
Mass 2	0.00264553	-0.00133298	0	-0.00139387	0.496303	-0.031479
optic	-0.000183364	0.000118648	0	0.000332198	-0.483371	-0.0311812

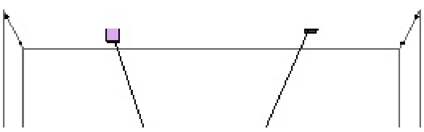


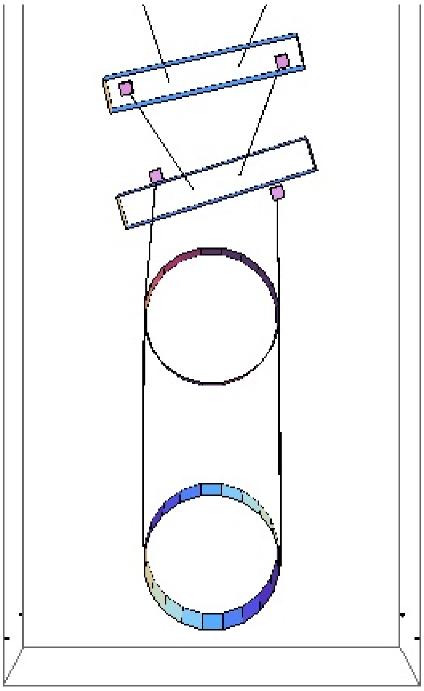
Mode #16 - modeR2

Back to Top

```
2.63003 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00174666	0.0420712	0	-0.00461512	-0.368506	-0.464096
Mass U	0.000378233	-0.0247759	0	-0.00324762	-0.191743	-0.624933
Mass 2	0.000203175	0.00103322	0	0.00585205	0.144429	0.30405
optic	0	-0.000311085	0	-0.00125834	-0.119873	0.30229



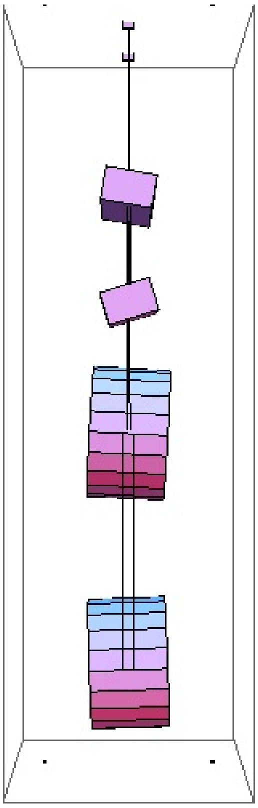


Mode #17 - modeP4

Back to Top

```
2.84646 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000282552	0.00322868	0	0	0.494969	-0.0451688
Mass U	0.000213982	-0.00568813	0	-0.000423863	-0.846136	0.0615914
Mass 2	-0.000101188	0.00196933	0	0.000467796	0.150543	-0.0272138
optic	0	0	0	0	-0.09504	-0.027259

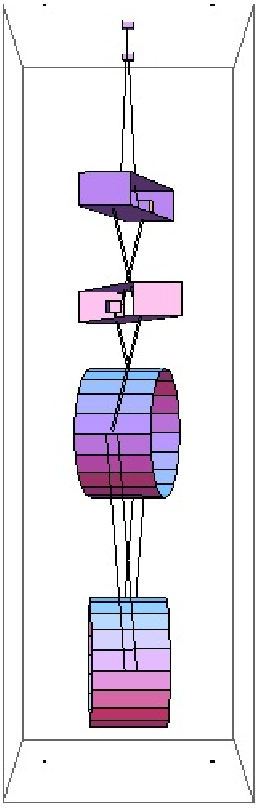


Mode #18 - modeY4

Back to Top

```
3.03675 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	-0.528735	0.00363168	0.000921891
Mass U	0	0.000180775	0	0.656414	-0.00672999	-0.00455013
Mass 2	0	0	0	-0.531835	0.000734751	0.0015519
optic	0	0	0	0.081402	-0.000378328	0.00155785

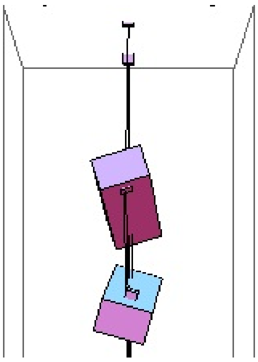


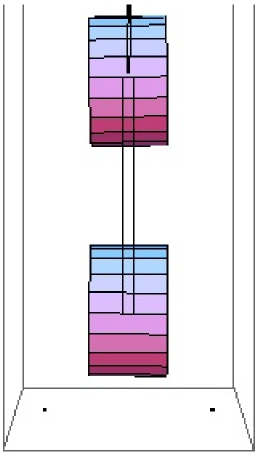
Mode #19 - modeR3

Back to Top

```
3.32094 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00381516	0.0646618	0	-0.00518536	-0.547486	-0.451124
Mass U	0.0033445	-0.0703594	0	0.00272544	0.508006	0.43887
Mass 2	-0.00044206	0.0142104	0	-0.00154878	-0.0362463	-0.131731
optic	0	-0.000455074	0	0.000193369	0.0143859	-0.133416



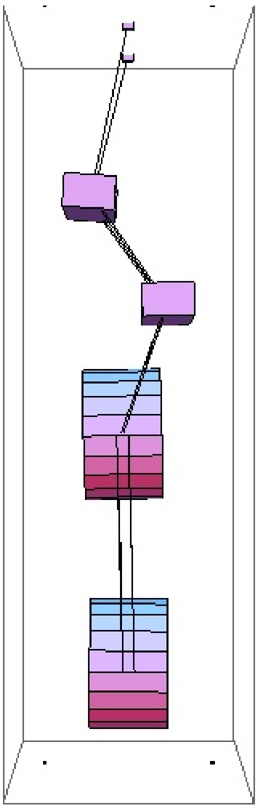


Mode #20 - modeL4

Back to Top

```
3.4283 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.613005	-0.0161684	0	0.00102226	0.426914	0.0974235
Mass U	0.619447	0.0167534	0	-0.000444192	-0.107337	-0.0714278
Mass 2	-0.0947071	-0.00298692	0	0.000225533	-0.138319	0.0210669
optic	0.00345601	0	0	0	0.0507127	0.0214207



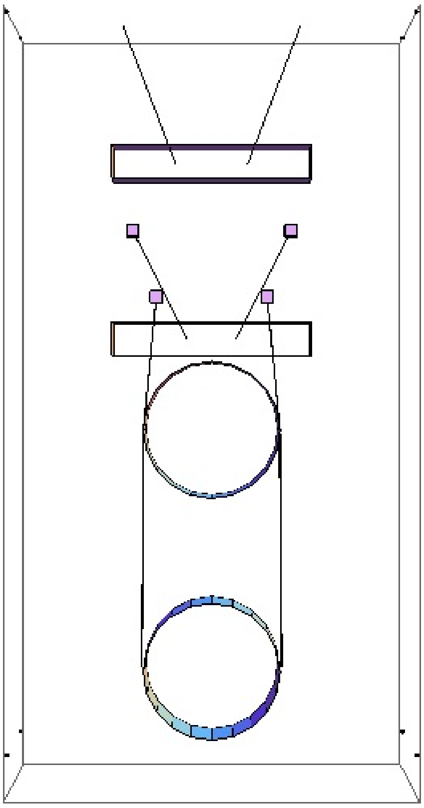
Mode #21 - modeV3

Back to Top

```
3.56765 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.548546	0	0	0

Mass U	0	0	-0.821719	0	0	0
Mass 2	0	0	0.104212	0	0	0
optic	0	0	0.114083	0	0	0

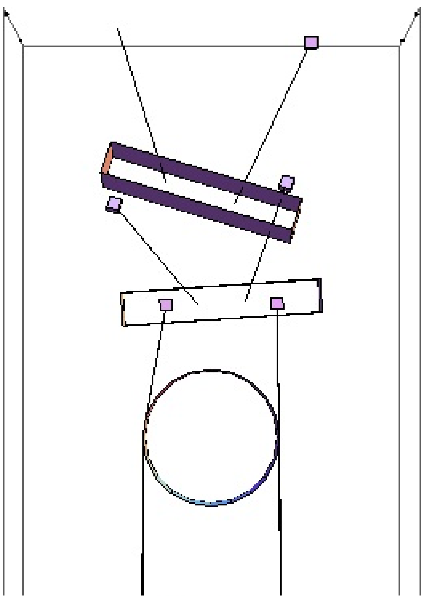


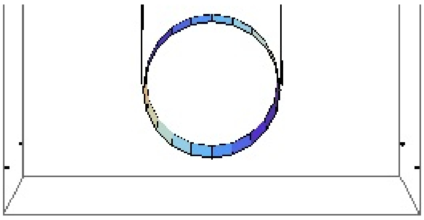
Mode #22 - modeT4

Back to Top

```
5.08241 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000178882	0.0747545	0	0.00404428	0.513565	0.822688
Mass U	0.000284513	-0.0677007	0	-0.000457041	-0.116615	-0.188182
Mass 2	0	0.00422328	0	0	0.00222514	0.010122
optic	0	0	0	0	-0.000306673	0.010501



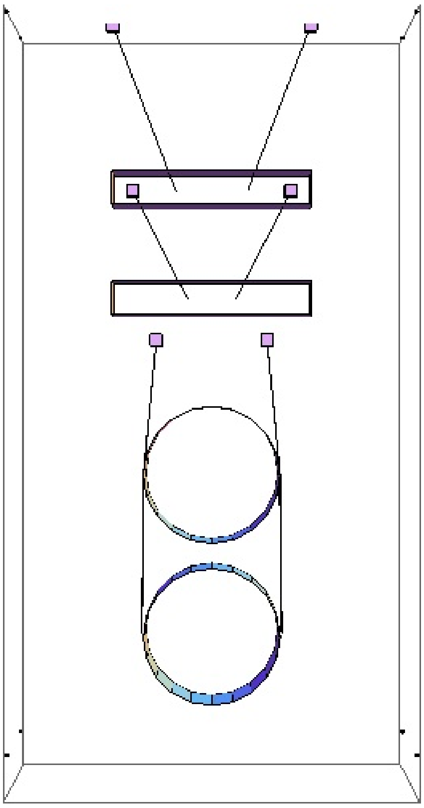


Mode #23 - modeV4

Back to Top

```
17.1158 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	-0.000183189	0	0	0
Mass U	0	0	0.0138464	0	0	0
Mass 2	0	0	-0.703992	0	0	0
optic	0	0	0.710073	0	0	0

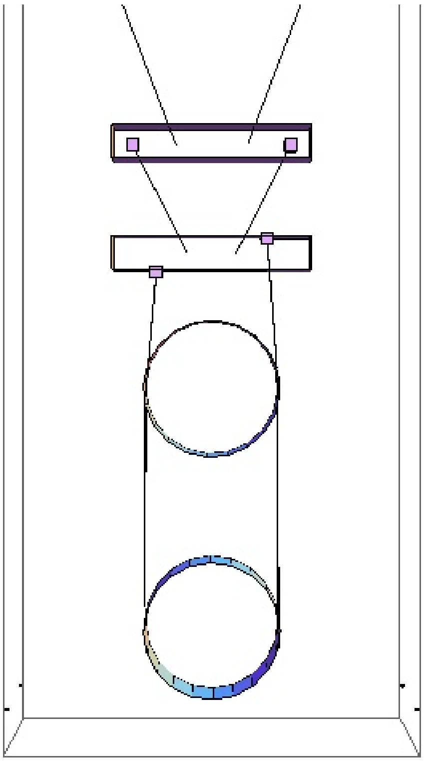


Mode #24 - modeR4

Back to Top

```
24.011 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0	0	0
Mass U	0	-0.000138507	0	0	0.00104449	0.00566447
Mass 2	0	0.000561678	0	0	0	-0.633903
optic	0	-0.000497433	0	0	0	0.773391



Parameters

Back to Top

```
g -> 9.81

nx -> 0.13                                                          T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

ny -> 0.5                                                           T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

nz -> 0.084                                                         T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

denn -> 683.94                                                      T1000286-v5 (info only)

mn -> 21.999                                                        measured value listed in
E1000186_v18_QUAD-2_Process_Traveler

Inx -> 0.460035                                                     T1000286-v9

Iny -> 0.0727234                                                    T1000286-v9

Inz -> 0.472177                                                     T1000286-v9

ux -> 0.13                                                          T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

uy -> 0.5                                                           T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

uz -> 0.084                                                         T040214-01, also T1000405-v1,
BNS 5/15/09 (display only)

den1 -> 740.934                                                     T1000286-v5 (info only)

m1 -> 21.526                                                        measured value listed in
E1000186_v18_QUAD-2_Process_Traveler (note, this is not the UIM with the pitch adjuster !)

I1x -> 0.50471                                                      T1000286-v9

I1y -> 0.0724024                                                    T1000286-v9

I1z -> 0.518216                                                     T1000286-v9

ix -> 0.2                                                           T1000405-v1, BNS 5/15/09

ir -> 0.17                                                          T1000405-v1, BNS 5/15/09
```

```
den2 -> 2201
http://www.sciner.com/Opticsland/FS.htm (T1000405-v1 says 2200)

m2 -> 40.391                                                     BNS -  measured 1/20/09,
+0.761 adjusted April 2009 for PUM magnet installation

I2x -> 0.680066                                                  Model Fit:
gradient_descent_fit_y_pitch_roll_Test_locked; BNS 5/9/09

I2y -> 0.439441                                                  Model Fit:
gradient_descent_fit_y_pitch_roll_Test_locked; BNS 5/9/09

I2z -> 0.41557                                                   %Model Fit:
gradient_descent_fit_yaw.m; BNS 7/29/2008

tx -> 0.2                                                        T1000405-v1

tr -> 0.17                                                       T1000405-v1

den3 -> 2201
http://www.sciner.com/Opticsland/FS.htm

m3 -> 39.631                                                     E1000186_v18_QUAD-
2_Process_Traveler - metal mass

I3x -> 0.56779                                                   Brett's calc ??

I3y -> 0.41955                                                   Brett's calc ??
calculated from shape, density

I3z -> 0.410601                                                  Brett's calc ??

ln -> 0.449192                                                   from QUAD01BUILD0101M0, out of
date                                                                                 derived
from tlnspec, d's, previously 0.449192

l1 -> 0.308585                                                   from QUAD01BUILD0101M0, out of
date                                                                                 derived
from tl1spec, d's, previously 0.308585

l2 -> 0.330787                                                   from QUAD01BUILD0101M0, out of
date                                                                                 derived
from vertical heights, d's, previously 0.5820   derived from tl2spec, d's, previously 0.330787

l3 -> 0.604322                                                   derived from tl3spec, d's,
previously 0.5820

nwn -> 2

nw1 -> 4

nw2 -> 4

nw3 -> 4

rn -> 0.00055                                                    T1000428-v2

r1 -> 0.0003555                                                  T1000428-v2

r2 -> 0.0003175                                                  T1000428-v2

r3 -> 0.0002285                                                  T1000428-v2

             11
Yn -> 2.12 10                                                    measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y1 -> 2.12 10                                                    measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y2 -> 2.12 10                                                    measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y3 -> 2.12 10                                                    IFOModel v4.1, unchanged,
different from new Ysilica

dm -> -0.00352908
```

```
dn -> 0.00423185                                                    Jeff's increase of 0.001

d0 -> -0.00174908

d1 -> 0.00399068                                                    Jeff's increase of 0.001

d2 -> 0.00708433                                                    M080134-00

d3 -> -0.00116094                                                   M080134-00

d4 -> -0.00116094                                                   M080134-00

sn -> 0                                                             T1000405-v1 (unused)

su -> 0.003                                                         T1000405-v1

si -> 0.003                                                         T1000405-v1

sl -> 0.015                                                         T1000405-v1

nn0 -> 0.25                                                         T1000405-v1

nn1 -> 0.09                                                         T1000405-v1

n0 -> 0.2                                                           T1000405-v1

n1 -> 0.06                                                          T1000405-v1

n2 -> 0.14                                                          T1000405-v1

n3 -> 0.17195                                                       D060358, D060337

n4 -> 0.17195                                                       D060358, D060337

n5 -> 0.17675                                                       D080658, D080697

tln -> 0.416201                                                     derived from l's, d's, should
match tlnspec

tl1 -> 0.277482                                                     derived from l's, d's, should
match tl1spec

tl2 -> 0.340315                                                     derived from l's, d's, should
match tl2spec

tl3 -> 0.601981                                                     derived from l's, d's, should
match tl3spec

ltotal -> 1.63598

bd -> 0.

unstretched -> False

vertblades -> True

matlabcompat -> False

uln -> 0.447746

ul1 -> 0.30756

ul2 -> 0.329816

ul3 -> 0.602633

sln -> 0.449192

sl1 -> 0.308585

sl2 -> 0.330787

sl3 -> 0.604322

sin -> -0.356195

si1 -> -0.453684

si2 -> 0.0965878

si3 -> 0.00794279
```

```
cn -> 0.934412

c1 -> 0.891163

c2 -> 0.995324

c3 -> 0.999968

pitchbul -> 0

pitchbur -> 0

pitchbil -> 0

pitchbir -> 0

pitchbll -> 0

pitchblr -> 0

rollbul -> 0

rollbur -> 0

rollbil -> 0

rollbir -> 0

rollbll -> 0

rollblr -> 0

                -7
An -> 9.50332 10

                -7
A1 -> 3.97035 10

                -7
A2 -> 3.16692 10

               -7
A3 -> 1.6403 10

kwn -> 448517.

kw1 -> 272766.

kw2 -> 202967.

kw3 -> 57542.6

flexn -> 0.00452908                                                    calculated as usual

flex1 -> 0.00274908                                                    calculated as usual

flex2 -> 0.00291567                                                    calculated as usual

flex3 -> 0.00216094                                                    calculated as usual

kbuz -> 1429.46                                                        Brett:
grad_descent_fit_z_bfgs_28July2010_part1

kbiz -> 1648.69                                                        Brett:
grad_descent_fit_z_bfgs_28July2010_part1

kblz -> 2382.97                                                        Brett:
grad_descent_fit_z_bfgs_28July2010_part1

kbux -> 100000.                                                        as for middle

kbix -> 100000.                                                        Justin 11/29/05

kblx -> 80000.                                                         Ian 12/09/05

bdu -> 0.423936

bdi -> 0.302114

bdl -> 0.164714
```

```
mn3 -> 123.547                                                      calculated as usual

m13 -> 101.548                                                      calculated as usual

m23 -> 80.022                                                       calculated as usual

Inxy -> -0.0375599                                                  T1000286-v9

Inyz -> -0.0000465463                                               T1000286-v9, Jeff K's sign
change

Inzx -> -0.00171841                                                 T1000286-v9, Jeff K's sign
change

COM0x -> 0

COM0y -> 0

COM0z -> 0

FRP0x -> 0

FRP0y -> 0

FRP0z -> 0

I1xy -> -0.0132064                                                  T1000286-v9

I1yz -> 0.0000137417                                                T1000286-v9, Jeff K's sign
change

                   -6
I1zx -> -8.08401 10                                                 T1000286-v9, Jeff K's sign
change

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

I2xy -> 0

I2yz -> 0

I2zx -> 0

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0

I3yz -> 0

I3zx -> 0

COM3x -> 0

COM3y -> 0

COM3z -> 0

FRP3x -> 0

FRP3y -> 0

FRP3z -> 0
```

btx -> 0.03	
bty -> 0.03	
btz -> 0.03	
Mn1 -> 7.18688 10^{-14}	
Mn2 -> 7.18688 10^{-14}	
M11 -> 1.25444 10^{-14}	
M12 -> 1.25444 10^{-14}	
M21 -> 7.98114 10^{-15}	
M22 -> 7.98114 10^{-15}	
M31 -> 2.14109 10^{-15}	
M32 -> 2.14109 10^{-15}	
temperature -> 290.	
boltzmann -> 1.38066 10^{-23}	
alphasilica -> 5.1 10^{-7}	IFOModel v4.1
betasilica -> 0.000152	IFOModel v4.1
rhosilica -> 2200.	IFOModel v4.1
Csilica -> 772.	IFOModel v4.1
Ksilica -> 1.38	IFOModel v4.1
Ysilica -> 7.27 10^{10}	IFOModel v4.1
phisilica -> 4.1 10^{-10}	IFOModel v4.1
phissilica -> 3. 10^{-11}	surface
rhosteel -> 7800.	IFOModel v4.1
Csteel -> 460.	IFOModel v4.1
Ksteel -> 49.	IFOModel v4.1
Ysteel -> 2.12 10^{11}	measured, MB, 11/18/05
alphasteel -> 0.000012	IFOModel v4.1
betasteel -> -0.00025	IFOModel v4.1
phisteel -> 0.0001	IFOModel v4.1
rhomarag -> 7800.	IFOModel v4.1
Cmarag -> 460.	IFOModel v4.1
Kmarag -> 20.	IFOModel v4.1
Ymarag -> 1.87 10^{11}	IFOModel v4.1
alphamarag -> 0.000011	IFOModel v4.1

```
betamarag -> -0.00025                                    Geppo's value - Bench v4.1 is
wrong

phimarag -> 0.0001                                       IFOModel v4.1

tmU -> 0.0043                                            IFOModel v4.1

tmI -> 0.0046                                            IFOModel v4.1

tmL -> 0.0042                                            IFOModel v4.1

magicnumber -> 0.0737472                                 Zener, 1938, Phys. Rev. 53:90-
99

deltabladeU -> 0.00182883

deltabladeI -> 0.00182883

deltabladeL -> 0.00182883

deltawireU -> 0.00278635

deltawireI -> 0.00278099

deltawireL -> 0.0027771

deltafibre -> 0.00276315

taubladeU -> 0.336093

taubladeI -> 0.384626

taubladeL -> 0.320643

tauwireU -> 0.00653413

tauwireI -> 0.00272987

tauwireL -> 0.00217746

taufibre -> 0.00112781

                                        0.003862 #1
damping[imag, bladeUtype] -> (0.0001 + --------------- & )
                                                     2
                                      1 + 4.45943 #1

                                       0.00441968 #1
damping[imag, bladeItype] -> (0.0001 + --------------- & )
                                                     2
                                      1 + 5.84032 #1

                                       0.00368446 #1
damping[imag, bladeLtype] -> (0.0001 + --------------- & )
                                                     2
                                      1 + 4.05884 #1

damping[imag, wireUtype] -> (0.0001 & )

damping[imag, wireItype] -> (0.0001 & )

damping[imag, wireLtype] -> (0.0001 & )

                                        0.000114394 #1
damping[imag, wireUatype] -> (0.0001 + ------------------ & )
                                                        2
                                      1 + 0.00168552 #1

                                       0.0000477003 #1
damping[imag, wireIatype] -> (0.0001 + ----------------- & )
                                                       2
                                      1 + 0.0002942 #1

                                       0.0000379945 #1
damping[imag, wireLatype] -> (0.0001 + ------------------ & )
                                                        2
                                      1 + 0.00018718 #1

damping[imag, fibretype] -> (0.0001 & )

                                        0.0000195802 #1
damping[imag, fibreatype] -> (0.0001 + ------------------ & )
```

```
damping[imag, fibreatype] -> (0.0001 + -------------------- & )
                                                           2
                                       1 + 0.0000502144 #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx0 -> 0

kcony0 -> 0

kconz0 -> 0

kconyaw0 -> 0

kconpitch0 -> 0

kconroll0 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

ribbons -> False

dumbbell -> True

Inxz -> -0.00171841

Inzy -> -0.0000465463

Inyx -> -0.0375599

                  -6
I1xz -> -8.08401 10

I1zy -> 0.0000137417

I1yx -> -0.0132064

if -> 0.                                                    M050397-02/T010103-05
```

```
tf -> 0.095                                                     M050397-02/T010103-05 (unused)

m3alt -> 39.5876                                                calculated from shape, density

I3xalt -> 0.566907                                              calculated from shape,
density, cf. Brett's calc

I3zalt -> 0.410278                                              calculated from shape, density

tlnspec -> 0.416                                                NR 4/3/06, T010103-05

tl1spec -> 0.277                                                NR 4/3/06, T010103-05

tl2spec -> 0.341                                                NR 4/3/06, T010103-05

tl3spec -> 0.602                                                NR 4/3/06, T010103-05

l3test -> 0.604322

ffn -> 0.807                                                    from Ian's data, 11/30/05,
linear fit version (not used)

ff1 -> 0.641                                                    from Ian's data, 11/30/05,
linear fit version (not used)

ff2 -> 0.608                                                    from Ian's data, 11/30/05,
linear fit version (not used)

kffn -> 0.692374                                                not used

kff1 -> 0.673672                                                not used

kff2 -> 1.059                                                   not used

bend1 -> 0.00498093                                             calculated as usual

bend2 -> 0.00490635                                             calculated as usual

thetan -> -20.8667                                              calculated as usual

theta1 -> -26.9803                                              calculated as usual

theta2 -> 5.54272                                               calculated as usual

theta3 -> 0.455093                                              calculated as usual

sigmasilica -> 0.17

                     10
Gsilica -> 3.10684 10                                           shear modulus

dssilica -> 0.015                                               IFOModel v4.1
```

Back to Top

Case 20120831TMproductionCP of Mathematica model QuadLite2Lateral

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
{"mark.barton", "20120831TMproductionCP"}
```

```
Corresponds to ^/trunk/QUAD/Common/MatlabTools/QuadModel_Production/quadopt_thincp.m r3304. Some minor MOI errors in 21020601TMproductionCP (cf. r2731) were corrected.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeL1	modeT1	modeV1	modeP1	modeY1	modeL2
modeT2	modeR1	modeY2	modeP2	modeL3	modeT3
modeV2	modeY3	modeR2	modeP3	modeY4	modeP4
modeR3	modeL4	modeV3	modeT4	modeV4	modeR4

Mode Summary

Back to Top

```
N    f          name     type

1    0.459444   modeL1   pitch3   pitch2   pitch1   pitch0

2    0.489339   modeT1   y3       roll3    roll2

3    0.550098   modeV1   z3       z2

4    0.611664   modeP1   pitch3   pitch2   pitch1

5    0.664019   modeY1   yaw3     yaw2

6    0.80953    modeL2   x3       pitch3   x2

7    0.815788   modeT2   roll1    roll3    roll2

8    0.87408    modeR1   roll1    roll3    roll2

9    1.34216    modeY2   yaw3

10   1.34265    modeP2   pitch0

11   1.90484    modeL3   x0       x1       pitch3

12   2.02972    modeT3   roll1    y1

13   2.22781    modeV2   z0

14   2.23941    modeY3   yaw0     yaw2

15   2.63028    modeR2   pitch1   roll1    roll0

16   2.72736    modeP3   pitch1   pitch0

17   2.95764    modeY4   yaw1     yaw0

18   3.28416    modeP4   pitch3
```

19	3.31786	modeR3	pitch3	pitch1	pitch0	roll1
20	3.40497	modeL4	pitch3	x0	x1	
21	3.58115	modeV3	z1	z0		
22	5.04888	modeT4	roll0	pitch0		
23	19.7505	modeV4	z3			
24	27.8594	modeR4	roll3			

Violin Modes	Top	UIM	PUM/PRM	TM/CP/ERM
f (Hz), n=1-4	335.2 671.439 1009.76 1351.25	494.64 990.467 1488.68 1990.49	433.145 867.311 1303.53 1742.84	163.897 327.918 492.188 656.831
Q, n=1-4	144772. 188715. 186069. 165419.	111109. 156990. 164331. 152783.	96976.4 143955. 157980. 152796.	69619. 86143.1 102464. 111658.

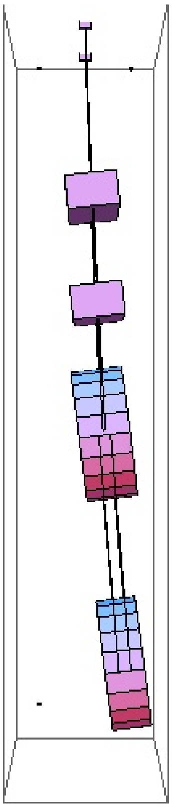
Mode Shapes

Mode #1 - modeL1

Back to Top

```
0.459444 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.0658833	0.00012168	0	0	-0.390695	0.00155456
Mass U	0.115897	0.000348393	0	0	-0.422576	0.00314138
Mass 2	0.187459	0.000621682	0	0	-0.478245	0.00266678
optic	0.382965	0.00127436	0	0	-0.490371	0.00266769

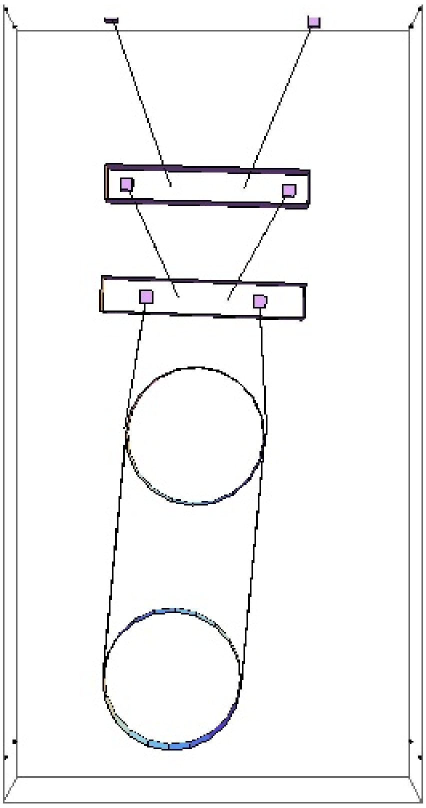


Mode #2 - modeT1

Back to Top

```
0.489339 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000562004	0.0917635	0	0	-0.0230799	0.199236
Mass U	-0.000957133	0.172672	0	0.00010304	-0.0188336	0.271499
Mass 2	-0.00138189	0.307511	0	0.000119167	-0.0127654	0.334351
optic	-0.00319996	0.727241	0	0.000141431	-0.0130902	0.334498

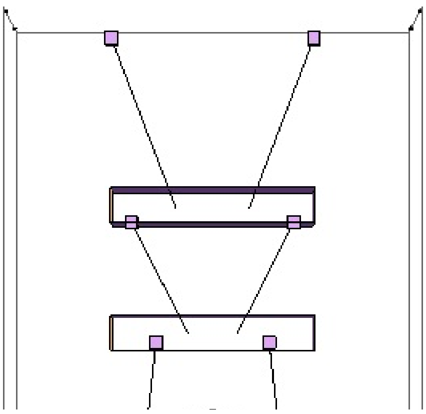


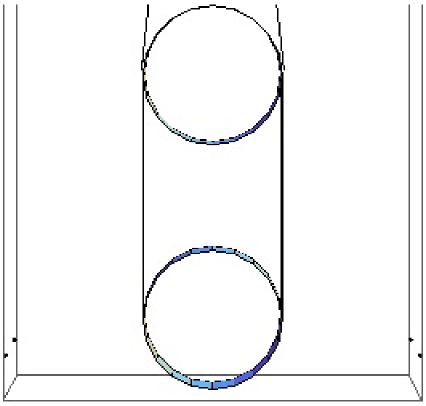
Mode #3 - modeV1

Back to Top

```
0.550098 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	-0.267937	0	0	0
Mass U	0	0	-0.475699	0	0	0
Mass 2	0	0	-0.592111	0	0	0
optic	0	0	-0.592727	0	0	0



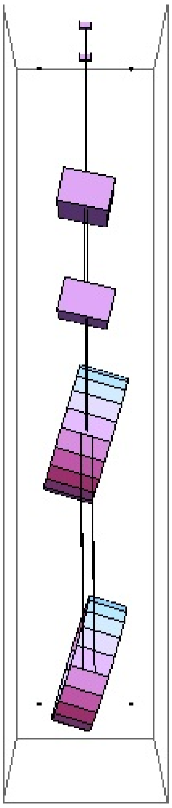


Mode #4 - modeP1

Back to Top

```
0.611664 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00375538	-0.000183004	0	0	-0.390747	0.00254736
Mass U	-0.00590433	0	0	0	-0.448553	0.00668239
Mass 2	-0.00318963	-0.000102674	0	-0.000109057	-0.555778	0.0054441
optic	-0.0206059	-0.000920923	0	-0.00014463	-0.580201	0.00544735



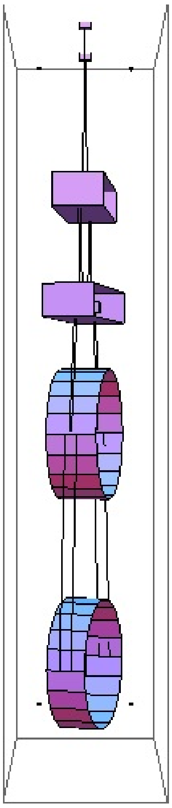
Mode #5 - modeY1

Back to Top

```
0.664019 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	-0.179145	0.000127067	0
Mass U	0	0	0	-0.403299	0.000121137	0

Mass 2	0	0	0	-0.519568	0.000169452	0
optic	0	0	0	-0.731646	0.000178431	0

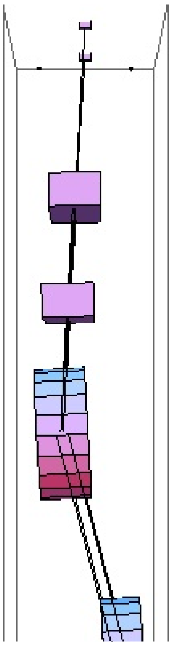


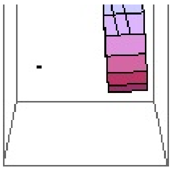
Mode #6 - modeL2

Back to Top

0.80953 Hz

	x	y	z	yaw	pitch	roll
Mass N	-0.17221	0.000925863	0	0	-0.0335264	0.0120793
Mass U	-0.283034	0.00235264	0	0	-0.148563	0.0388688
Mass 2	-0.381167	0.00450121	0	0	-0.35696	0.0377321
optic	0.659248	-0.00793129	0	0	-0.393861	0.0377723



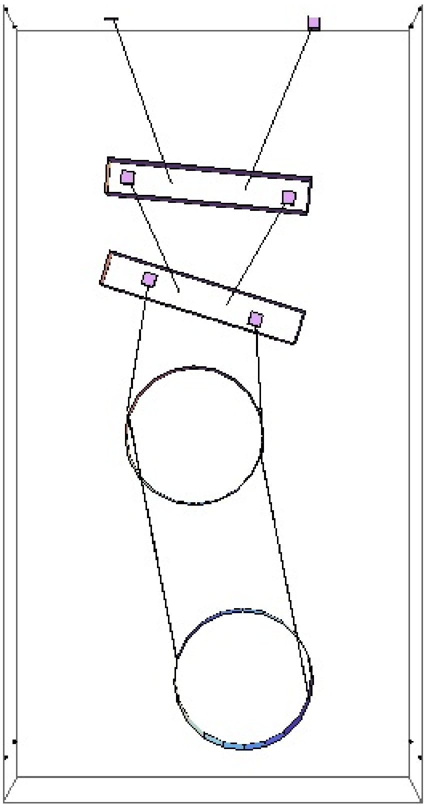


Mode #7 - modeT2

Back to Top

```
0.815788 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000865013	0.0204541	0	0	-0.0353392	0.181484
Mass U	0.0014402	0.0460723	0	-0.000195499	-0.0128784	0.562174
Mass 2	0.00148732	0.0839712	0	-0.00028973	0.0390694	0.554981
optic	-0.00233714	-0.141444	0	-0.000515086	0.0424093	0.555581

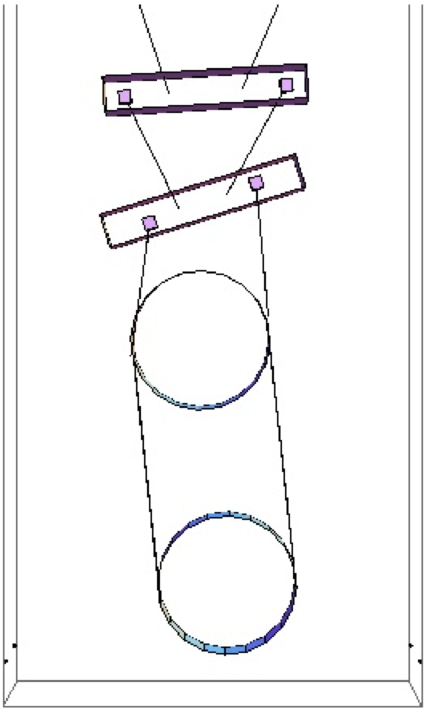


Mode #8 - modeR1

Back to Top

```
0.87408 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000161257	0.0345367	0	0	0.041596	-0.124486
Mass U	-0.000292327	0.0484113	0	0.00010311	0.0231547	-0.591645
Mass 2	0	0.0592148	0	0.000160278	-0.0308616	-0.555633
optic	0	-0.0692877	0	0.000322018	-0.0339252	-0.556343

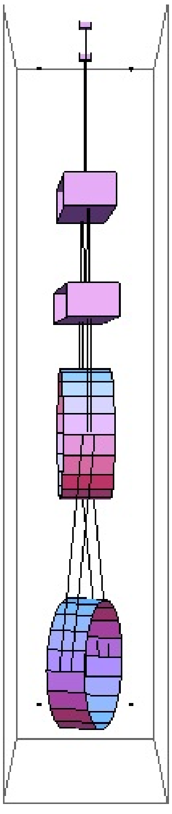


Mode #9 - modeY2

Back to Top

```
1.34216 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0.145522	-0.0220248	0.00029879
Mass U	0	0	0	0.248222	-0.0131408	-0.000272367
Mass 2	0	0	0	0.173469	0.0025484	-0.000547468
optic	0	0	0	-0.941511	0.00325904	-0.000549128

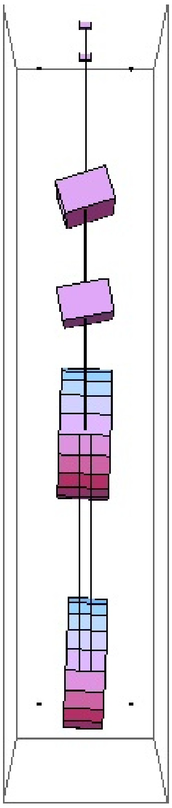


Mode #10 - modeP2

Back to Top

```
1.34265 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000327269	-0.000310958	0	-0.00148375	-0.846056	0.00862454
Mass U	0.00112238	0.000845414	0	-0.00260913	-0.507567	-0.00885703
Mass 2	-0.00101065	-0.000311998	0	-0.00182809	0.0986327	-0.0180518
optic	0.0003701	0.000104296	0	0.0098753	0.126163	-0.0181065

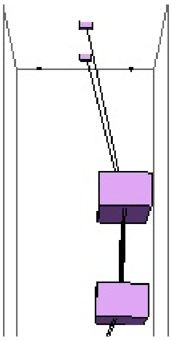


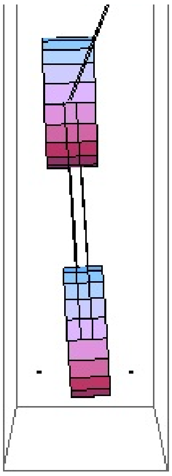
Mode #11 - modeL3

Back to Top

```
1.90484 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.613442	-0.000352214	0	0	0.0900156	-0.00391586
Mass U	0.568891	-0.000855347	0	0	0.197084	-0.0026419
Mass 2	-0.235909	0.00025356	0	0	-0.21433	0.00483675
optic	0.0304343	0	0	0	-0.38797	0.00486652



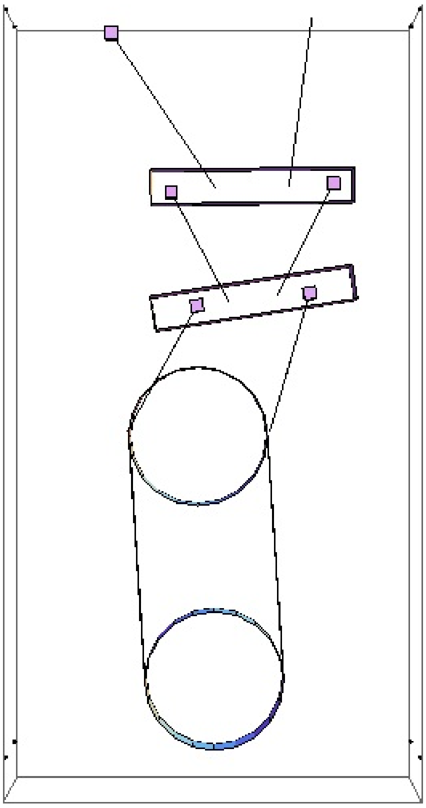


Mode #12 - modeT3

Back to Top

```
2.02972 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000619272	-0.450667	0	-0.000481932	-0.1939	-0.0579501
Mass U	-0.000469355	-0.460789	0	0	0.00650185	-0.718418
Mass 2	0.000162818	0.164509	0	0.000294695	-0.000249005	-0.00478286
optic	0	-0.018454	0	-0.000172501	-0.000499445	-0.00482532



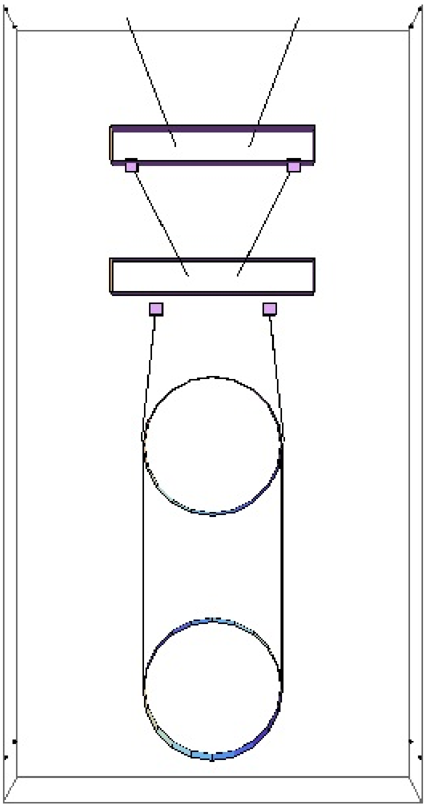
Mode #13 - modeV2

Back to Top

```
2.22781 Hz
```

	x	y	z	yaw	pitch	roll

Mass N	0	0	0.840122	0	0	0
Mass U	0	0	0.457372	0	0	0
Mass 2	0	0	-0.204382	0	0	0
optic	0	0	-0.207928	0	0	0

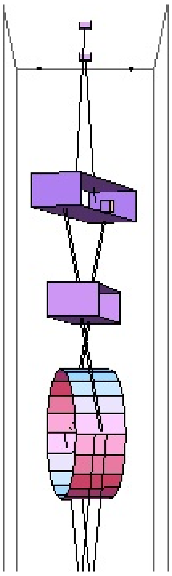


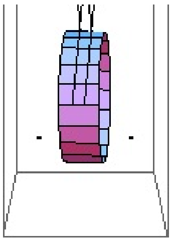
Mode #14 - modeY3

Back to Top

```
2.23941 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0.00021028	0	-0.726075	-0.000861733	-0.00251108
Mass U	0	-0.000119395	0	-0.302933	-0.00268214	-0.00115489
Mass 2	0	0	0	0.565962	0	0.00103822
optic	0	0	0	-0.246406	0.0002488	0.00104709



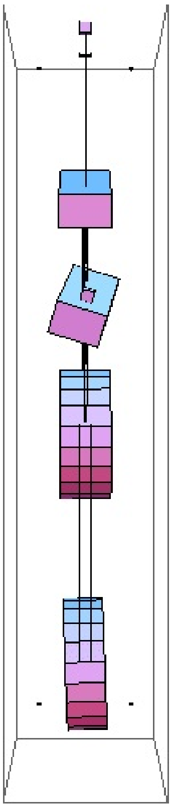


Mode #15 - modeR2

Back to Top

```
2.63028 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000132793	-0.0436101	0	-0.00110369	-0.00570904	0.454209
Mass U	-0.000587222	0.0243352	0	-0.0041887	0.606226	0.518516
Mass 2	0.000192727	-0.000163624	0	0.00344079	-0.010338	-0.272249
optic	0	0	0	-0.00096973	-0.0682386	-0.275471

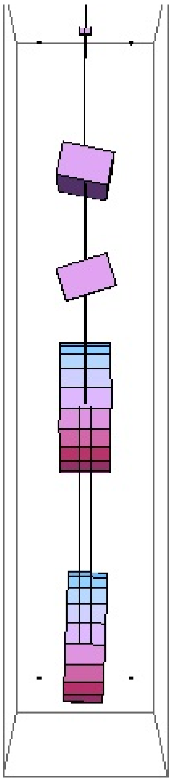


Mode #16 - modeP3

Back to Top

```
2.72736 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000289693	0.00184339	0	-0.000278979	-0.546153	-0.00431448
Mass U	-0.000718238	0.0020544	0	0	0.82347	-0.0949056
Mass 2	0.00023667	-0.00107566	0	0	-0.00864873	0.0482487
optic	0	0	0	0	-0.098935	0.0488634

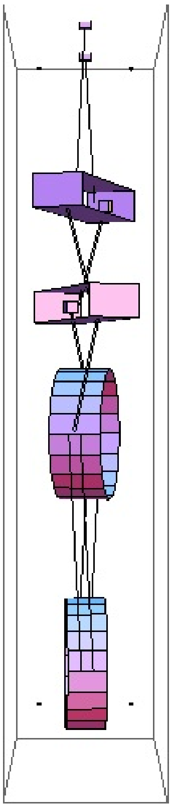


Mode #17 - modeY4

Back to Top

```
2.95764 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	-0.631848	-0.00232826	0
Mass U	0	0	0	0.682009	0.00362501	0.00549854
Mass 2	0	0	0	-0.360308	0	-0.00211692
optic	0	0	0	0.0758423	-0.000534352	-0.00214871

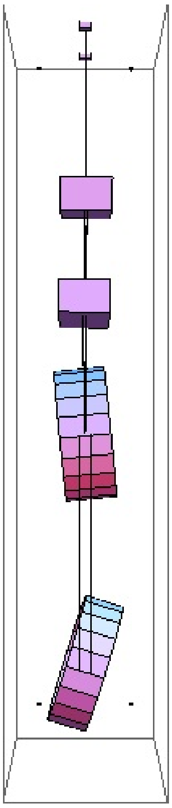


Mode #18 - modeP4

Back to Top

```
3.28416 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00725096	-0.0188208	0	-0.00147188	0.0808976	0.135953
Mass U	-0.00585105	0.0206221	0	0.000691722	0.051162	-0.135508
Mass 2	0.000269248	-0.00284385	0	-0.000249524	-0.301069	0.0451973
optic	0	0.000110301	0	0	0.926451	0.0460377

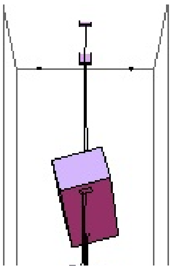


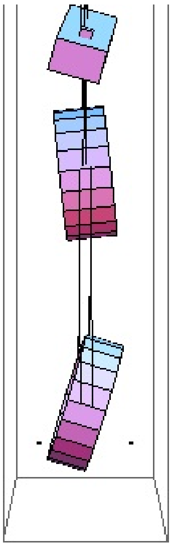
Mode #19 - modeR3

Back to Top

```
3.31786 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00537436	0.0543095	0	0.00387054	-0.432304	-0.373499
Mass U	-0.00494176	-0.0589686	0	-0.00188389	0.451912	0.390534
Mass 2	0.000355645	0.00793931	0	0.000724353	-0.175145	-0.12646
optic	0	-0.000301442	0	-0.00011614	0.497028	-0.12886



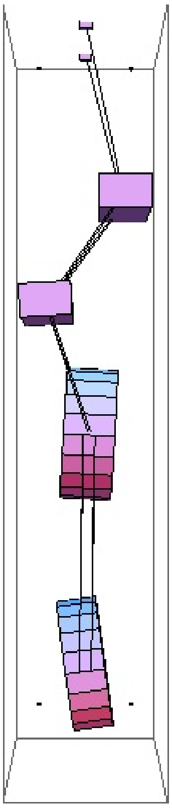


Mode #20 - modeL4

Back to Top

```
3.40497 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.46937	-0.00347374	0	-0.000218376	0.0167689	0.0203181
Mass U	-0.465584	0.00383038	0	0.000149605	-0.265421	-0.0362913
Mass 2	0.0455366	-0.000520055	0	0	0.272767	0.0104199
optic	-0.00170657	0	0	0	-0.643224	0.0106284

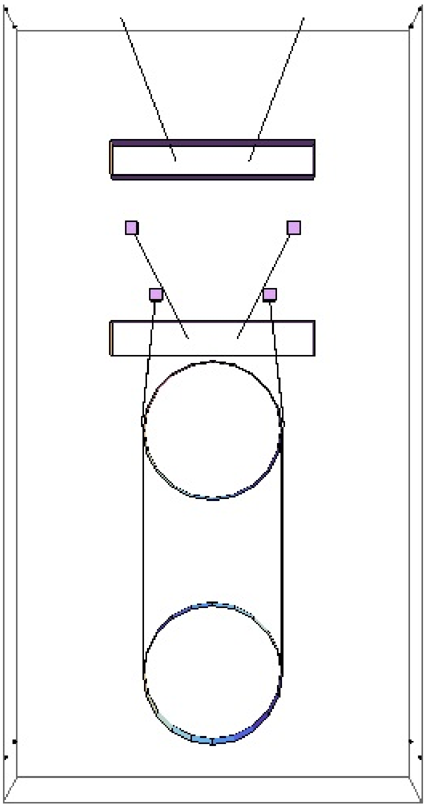


Mode #21 - modeV3

Back to Top

```
3.58115 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.539448	0	0	0
Mass U	0	0	-0.826475	0	0	0
Mass 2	0	0	0.11128	0	0	0
optic	0	0	0.116409	0	0	0

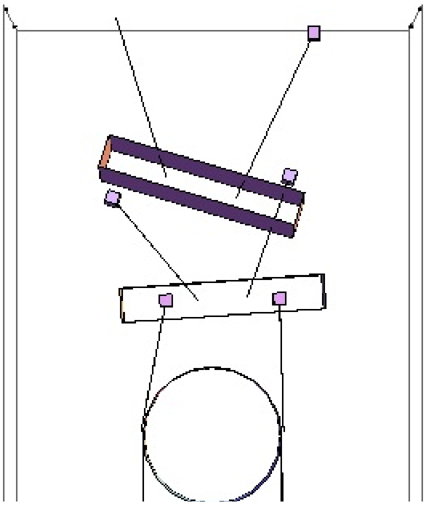


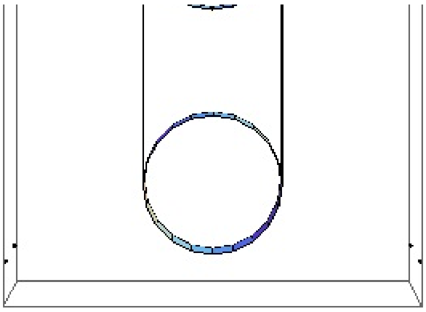
Mode #22 - modeT4

Back to Top

```
5.04888 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0.0764172	0	-0.00395125	0.498942	0.831374
Mass U	0.000109072	-0.0689644	0	0.000447547	-0.112557	-0.190685
Mass 2	0	0.00284022	0	0	0.00180687	0.0102902
optic	0	0	0	0	-0.000845634	0.0107543



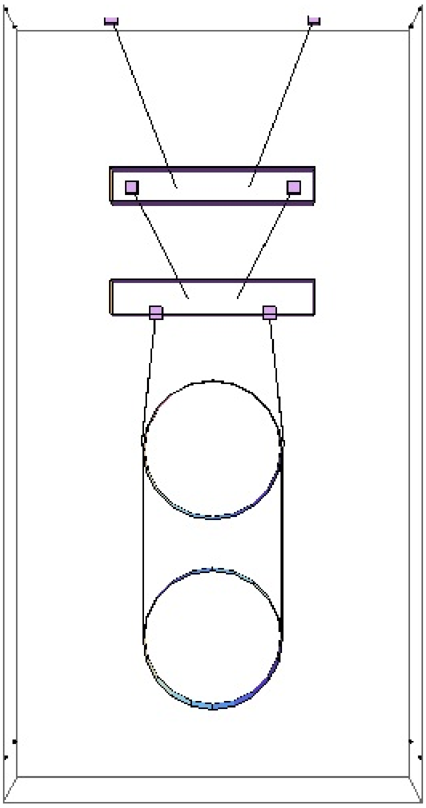


Mode #23 - modeV4

Back to Top

```
19.7505 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0	0	0
Mass U	0	0	0.00479676	0	0	0
Mass 2	0	0	-0.32203	0	0	0
optic	0	0	0.946717	0	0	0

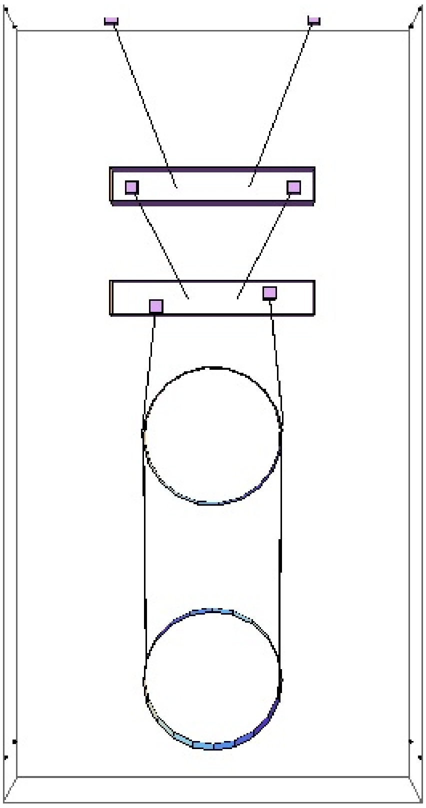


Mode #24 - modeR4

Back to Top

```
27.8594 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0	0	0
Mass U	0	0	0	0	0.000371137	0.00204675
Mass 2	0	0	0	0	0	-0.300833
optic	0	0	0	0	0	0.953675



Parameters

Back to Top

```
g -> 9.81

nx -> 0.13                                                       T040214-01 (display only)

ny -> 0.5                                                        T040214-01 (display only)

nz -> 0.084                                                      T040214-01 (display only)

denn -> 0                                                        (unused)

mn -> 22.002                                                     measured value listed in
E1000186_v18_QUAD-2_Process_Traveler

Inx -> 0.461176                                                  T1000286-v9 11.2 Reaction Top
Mass

Iny -> 0.0742518                                                 T1000286-v9 11.2 Reaction Top
Mass

Inz -> 0.472634                                                  T1000286-v9 11.2 Reaction Top
Mass

ux -> 0.13                                                       T040214-01 (display only)

uy -> 0.5                                                        T040214-01 (display only)

uz -> 0.084                                                      T040214-01 (display only)

den1 -> 0                                                        (unused)

m1 -> 21.532                                                     measured value listed in
E1000186_v18_QUAD-2_Process_Traveler

I1x -> 0.50522                                                   T1000286-v9, reaction UIM with
pitch adjuster, Joe's zz

I1y -> 0.073433                                                  T1000286-v9, reaction UIM with
pitch adjuster, Joe's yy

I1z -> 0.518171                                                  T1000286-v9, reaction UIM with
pitch adjuster, Joe's xx
```

```
ix -> 0.13                                                      T1000405-v1

ir -> 0.17                                                      T1000405-v1

den2 -> 0                                                       not used

m2 -> 59.229                                                    Calculated from ETM values
listed in E1000186_v18_QUAD-2_Process_Traveler by m2 = 26.130+53.138-20.039

I2x -> 0.923275                                                 T1000286-v9 11.8 PRM for CP

I2y -> 0.535571                                                 T1000286-v9 11.8 PRM for CP

I2z -> 0.52348                                                  T1000286-v9 11.8 PRM for CP

tx -> 0.10032                                                   C1000458

tr -> 0.170055                                                  C1000458

den3 -> 0                                                       unused

m3 -> 20.039                                                    E1000186_v18_QUAD-
2_Process_Traveler

I3x -> 0.289751                                                 0.5*m3*tr^2

I3y -> 0.161682                                                 (1/12)*m3*(3*tr^2 + tx^2)

I3z -> 0.161682                                                 (1/12)*m3*(3*tr^2 + tx^2)

ln -> 0.449192                                                  from QUAD01BUILD0101M0, out of
date

l1 -> 0.308585                                                  from QUAD01BUILD0101M0, out of
date

l2 -> 0.330787                                                  from QUAD01BUILD0101M0, out of
date

l3 -> 0.60406                                                   from QUAD01BUILD0101M0, out of
date

nwn -> 2

nw1 -> 4

nw2 -> 4

nw3 -> 4

rn -> 0.00055                                                   T1000428-v7

r1 -> 0.000355                                                  T1000428-v7

r2 -> 0.0003175                                                 T1000428-v7

r3 -> 0.0002285                                                 T1000428-v7

             11
Yn -> 2.12 10                                                   measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y1 -> 2.12 10                                                   measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y2 -> 2.12 10                                                   measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y3 -> 2.12 10                                                   measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

dm -> -0.0035428                                                effective value 0.001; BNS
6/18/08

dn -> 0.00319273                                                effective value 0.001 after
blade correction; BNS 6/18/08

d0 -> -0.00175151                                               effective value 0.001
```

```
d1 -> 0.00293649                                                  effective value 0.001 after
blade correction; BNS 6/18/08

d2 -> 0.00789637                                                  Originally 0.010 effective BNS
6/18/08; Adjustment for new wire clamp design BNS 2/16/09.  0.18 mm subtracted to match Joe's
updated value, BNS 7/21/09.

d3 -> -0.00203908                                                 0.0010 effective; BNS 6/18/08

d4 -> -0.00203908                                                 0.0010 effective; BNS 6/18/08

sn -> 0                                                           T1000405-v1 (unused)

su -> 0.003                                                       T1000405-v1

si -> 0.003                                                       T1000405-v1

sl -> 0.015                                                       T1000405-v1

nn0 -> 0.25                                                       T1000405-v1

nn1 -> 0.09                                                       T1000405-v1

n0 -> 0.2                                                         T1000405-v1

n1 -> 0.06                                                        T1000405-v1

n2 -> 0.14                                                        T1000405-v1

n3 -> 0.1762                                                      CT, email to NR, 9/22/04

n4 -> 0.1712                                                      CT, email to NR, 9/22/04

n5 -> 0.1712                                                      CT, email to NR, 9/22/04

tln -> 0.416187

tl1 -> 0.276441

tl2 -> 0.339633

tl3 -> 0.599982

ltotal -> 1.63224

bd -> 0.

unstretched -> False

vertblades -> True

matlabcompat -> False

uln -> 0.447755

ul1 -> 0.307565

ul2 -> 0.329823

ul3 -> 0.603206

sln -> 0.449192

sl1 -> 0.308585

sl2 -> 0.330787

sl3 -> 0.60406

sin -> -0.356195

si1 -> -0.453684

si2 -> 0.109436

si3 -> 0.

cn -> 0.934412

c1 -> 0.891163
```

```
c2 -> 0.993994

c3 -> 1.

pitchbul -> 0

pitchbur -> 0

pitchbil -> 0

pitchbir -> 0

pitchbll -> 0

pitchblr -> 0

rollbul -> 0

rollbur -> 0

rollbil -> 0

rollbir -> 0

rollbll -> 0

rollblr -> 0

                -7
An -> 9.50332 10

                -7
A1 -> 3.95919 10

                -7
A2 -> 3.16692 10

               -7
A3 -> 1.6403 10

kwn -> 448517.

kw1 -> 271999.

kw2 -> 202967.

kw3 -> 57567.6

flexn -> 0.0045428                                                   calculated as usual

flex1 -> 0.00275151                                                  calculated as usual

flex2 -> 0.00292363                                                  calculated as usual

flex3 -> 0.00303908                                                  calculated as usual

kbuz -> 1411.4640262910420948351545480 22                            Solved for given mass and
freqs; BNS 6/19/08

kbiz -> 1650.5240590453796728792143217 79                            Solved for given mass and
freqs; BNS 6/19/08

kblz -> 2423.5190152802382310519860608 10                            Solved for given mass and
freqs; BNS 6/19/08

kbux -> 100000.                                                      as for middle

kbix -> 100000.                                                      Justin 11/29/05

kblx -> 80000.                                                       Ian 12/09/05

bdu -> 0.426751

bdi -> 0.299556

bdl -> 0.160432

mn3 -> 122.802                                                       calculated as usual

m13 -> 100.8                                                         calculated as usual

m23 -> 79.268                                                        calculated as usual
```

```
Inxy -> -0.0375611                                         T1000286-v9 11.2 Reaction Top
Mass

Inyz -> -0.0000435986                                      T1000286-v9 11.2 Reaction Top
Mass

Inzx -> 0.00171393                                         T1000286-v9 11.2 Reaction Top
Mass

COM0x -> 0

COM0y -> 0

COM0z -> 0

FRP0x -> 0

FRP0y -> 0

FRP0z -> 0

I1xy -> -0.0132118                                         T1000286-v9, reaction UIM with
pitch adjuster, Joe's yz

I1yz -> 0.0000501027                                       T1000286-v9, reaction UIM with
pitch adjuster, -Joe's xy

I1zx -> -0.0000341209                                      T1000286-v9, reaction UIM with
pitch adjuster, -Joe's zx

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

I2xy -> -0.0000587085                                      T1000286-v9, PRM for CP

I2yz -> 0.0000237613                                       T1000286-v9, PRM for CP

I2zx -> 0.0000440743                                       T1000286-v9, PRM for CP

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0

I3yz -> 0

I3zx -> 0

COM3x -> 0

COM3y -> 0

COM3z -> 0

FRP3x -> 0

FRP3y -> 0

FRP3z -> 0

btx -> 0.03
```

bty -> 0.03	
btz -> 0.03	
Mn1 -> $7.18688\ 10^{-14}$	
Mn2 -> $7.18688\ 10^{-14}$	
M11 -> $1.24739\ 10^{-14}$	
M12 -> $1.24739\ 10^{-14}$	
M21 -> $7.98114\ 10^{-15}$	
M22 -> $7.98114\ 10^{-15}$	
M31 -> $2.14109\ 10^{-15}$	
M32 -> $2.14109\ 10^{-15}$	
temperature -> 290.	
boltzmann -> $1.38066\ 10^{-23}$	
alphasilica -> $5.1\ 10^{-7}$	IFOModel v4.1
betasilica -> 0.000152	IFOModel v4.1
rhosilica -> 2200.	IFOModel v4.1
Csilica -> 772.	IFOModel v4.1
Ksilica -> 1.38	IFOModel v4.1
Ysilica -> $7.27\ 10^{10}$	IFOModel v4.1
phisilica -> $4.1\ 10^{-10}$	IFOModel v4.1
phissilica -> $3.\ 10^{-11}$	surface
rhosteel -> 7800.	IFOModel v4.1
Csteel -> 460.	IFOModel v4.1
Ksteel -> 49.	IFOModel v4.1
Ysteel -> $2.12\ 10^{11}$	measured, MB, 11/18/05
alphasteel -> 0.000012	IFOModel v4.1
betasteel -> -0.00025	IFOModel v4.1
phisteel -> 0.0001	IFOModel v4.1
rhomarag -> 7800.	IFOModel v4.1
Cmarag -> 460.	IFOModel v4.1
Kmarag -> 20.	IFOModel v4.1
Ymarag -> $1.87\ 10^{11}$	IFOModel v4.1
alphamarag -> 0.000011	IFOModel v4.1
betamarag -> -0.00025	Geppo's value - Bench v4.1 is wrong

```
phimarag -> 0.0001                                                IFOModel v4.1

tmU -> 0.0043                                                     IFOModel v4.1

tmI -> 0.0046                                                     IFOModel v4.1

tmL -> 0.0042                                                     IFOModel v4.1

magicnumber -> 0.0737472                                          Zener, 1938, Phys. Rev. 53:90-
99

deltabladeU -> 0.00182883

deltabladeI -> 0.00182883

deltabladeL -> 0.00182883

deltawireU -> 0.00280734

deltawireI -> 0.00281891

deltawireL -> 0.002776

deltafibre -> 0.00261486

taubladeU -> 0.336093

taubladeI -> 0.384626

taubladeL -> 0.320643

tauwireU -> 0.00653413

tauwireI -> 0.00272219

tauwireL -> 0.00217746

taufibre -> 0.00112781

                                         0.003862 #1
damping[imag, bladeUtype] -> (0.0001 + --------------- & )
                                                    2
                                       1 + 4.45943 #1

                                        0.00441968 #1
damping[imag, bladeItype] -> (0.0001 + --------------- & )
                                                    2
                                       1 + 5.84032 #1

                                        0.00368446 #1
damping[imag, bladeLtype] -> (0.0001 + --------------- & )
                                                    2
                                       1 + 4.05884 #1

damping[imag, wireUtype] -> (0.0001 & )

damping[imag, wireItype] -> (0.0001 & )

damping[imag, wireLtype] -> (0.0001 & )

                                         0.000115256 #1
damping[imag, wireUatype] -> (0.0001 + ------------------ & )
                                                       2
                                       1 + 0.00168552 #1

                                         0.0000482147 #1
damping[imag, wireIatype] -> (0.0001 + ------------------- & )
                                                        2
                                       1 + 0.000292548 #1

                                        0.0000379795 #1
damping[imag, wireLatype] -> (0.0001 + ------------------ & )
                                                       2
                                       1 + 0.00018718 #1

damping[imag, fibretype] -> (0.0001 & )

                                         0.0000185295 #1
damping[imag, fibreatype] -> (0.0001 + -------------------- & )
                                                         2
                                       1 + 0.0000502144 #1
```

```
x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx0 -> 0

kcony0 -> 0

kconz0 -> 0

kconyaw0 -> 0

kconpitch0 -> 0

kconroll0 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

ribbons -> False

dumbbell -> True

Inxz -> 0.00171393

Inzy -> -0.0000435986

Inyx -> -0.0375611

I1xz -> -0.0000341209

I1zy -> 0.0000501027

I1yx -> -0.0132118

I2xz -> 0.0000440743

I2zy -> 0.0000237613

I2yx -> -0.0000587085
```

```
                -7
A3n -> 1.6403 10

bend1 -> 0.00494424                                  calculated as usual

bend2 -> 0.00486012                                  calculated as usual

thetan -> -20.8667                                   calculated as usual

theta1 -> -26.9803                                   calculated as usual

theta2 -> 6.2828                                     calculated as usual

theta3 -> 0.                                         calculated as usual

sigmasilica -> 0.17

                     10
Gsilica -> 3.10684 10                                shear modulus

dssilica -> 0.015                                    IFOModel v4.1
```

Back to Top

Case 20120831TMproductionERM of Mathematica model QuadLite2Lateral

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
{"mark.barton", "20120831TMproductionERM"}
```

```
Corresponds to ^/trunk/QUAD/Common/MatlabTools/QuadModel_Production/quadopt_erm.m r3304. Some
minor MOI errors in 21020601TMproductionERM (cf. r2731) were corrected.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeL1	modeT1	modeV1	modeP1	modeY1	modeR1
modeL2	modeT2	modeP2	modeY2	modeL3	modeT3
modeV2	modeY3	modeR2	modeP3	modeY4	modeP4
modeR3	modeL4	modeV3	modeT4	modeV4	modeR4

Mode Summary

Back to Top

```
N    f          name     type

1    0.450845   modeL1   pitch3   pitch2   pitch1   pitch0

2    0.479208   modeT1   y3       roll3    roll2

3    0.550444   modeV1   z3       z2

4    0.60238    modeP1   pitch3   pitch2   pitch1

5    0.648033   modeY1   yaw3     yaw2

6    0.834372   modeR1   roll1    roll3    roll2

7    0.859087   modeL2   x3       x2       x1

8    0.907233   modeT2   roll1    roll3    roll2

9    1.33693    modeP2   pitch0

10   1.34081    modeY2   yaw3

11   1.92516    modeL3   x0       x1       pitch3

12   2.0476     modeT3   roll1    y1

13   2.2276     modeV2   z0

14   2.28221    modeY3   yaw0     yaw2

15   2.62396    modeR2   pitch1   roll1    roll0

16   2.70885    modeP3   pitch1   pitch0

17   2.97614    modeY4   yaw1     yaw0

18   2.98639    modeP4   pitch3   pitch1
```

```
19    3.31274    modeR3   pitch0   roll1    roll0

20    3.40455    modeL4   x0       x1

21    3.58064    modeV3   z1       z0

22    5.04715    modeT4   roll0    pitch0

23    18.3013    modeV4   z3

24    25.6281    modeR4   roll3
```

Violin Modes	Top	UIM	PUM/PRM	TM/CP/ERM
f (Hz), n=1-4	335.017 671.073 1009.22 1350.52	494.31 989.808 1487.69 1989.18	432.778 866.577 1302.43 1741.38	186.428 372.964 559.718 746.799
Q, n=1-4	144648. 188553. 185901. 165262.	110989. 156818. 164144. 152600.	96841.1 143748. 157744. 152557.	80280.7 106206. 129541. 143394.

Mode Shapes

Mode #1 - modeL1

Back to Top

```
0.450845 Hz
```

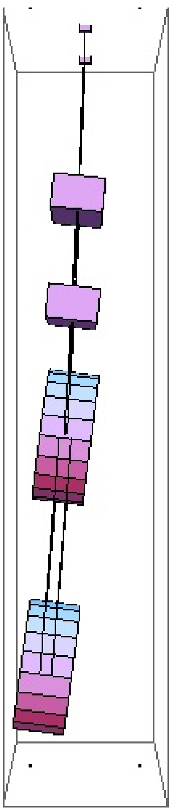
```
         x            y              z    yaw    pitch       roll

Mass N   -0.0664457   -0.000116542   0    0      0.390165    -0.0014854

Mass U   -0.117015    -0.000333793   0    0      0.422548    -0.00298759

Mass 2   -0.189692    -0.000596139   0    0      0.479361    -0.00253247

optic    -0.373672    -0.00117825    0    0      0.495672    -0.00253355
```

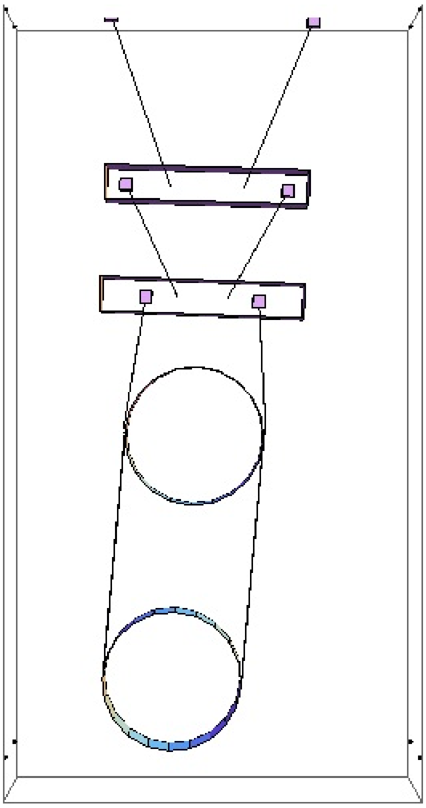


Mode #2 - modeT1

Back to Top

```
0.479208 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000538458	0.0943753	0	0	-0.0222017	0.20427
Mass U	-0.000918403	0.177725	0	0.000100482	-0.0179963	0.275869
Mass 2	-0.00133309	0.317309	0	0.000116425	-0.0120167	0.341301
optic	-0.00292961	0.711876	0	0.000138918	-0.0124309	0.341485

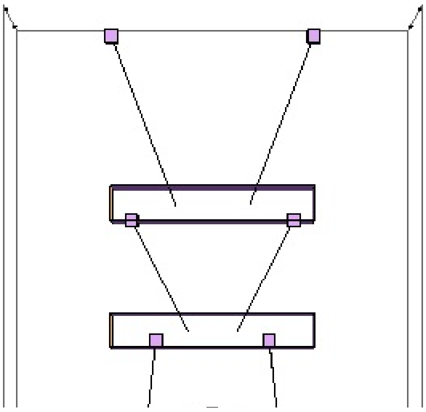


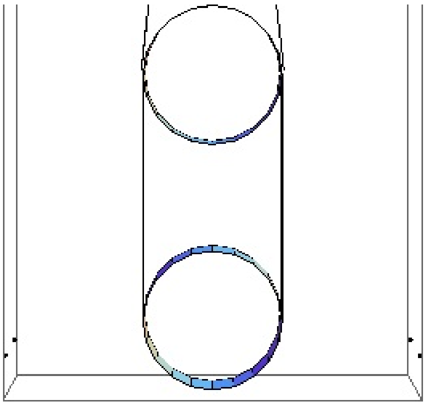
Mode #3 - modeV1

Back to Top

```
0.550444 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	-0.267935	0	0	0
Mass U	0	0	-0.475669	0	0	0
Mass 2	0	0	-0.592031	0	0	0
optic	0	0	-0.592831	0	0	0



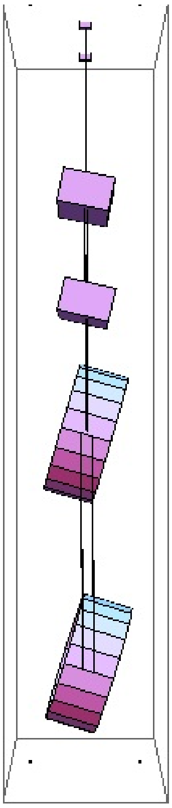


Mode #4 - modeP1

Back to Top

```
0.60238 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00378288	0.000173393	0	0	0.385983	-0.00240512
Mass U	0.00596688	0	0	0	0.444656	-0.00626962
Mass 2	0.00332783	0.000100995	0	0.000115135	0.554338	-0.00508784
optic	0.017387	0.000722443	0	0.00015472	0.587833	-0.00509167



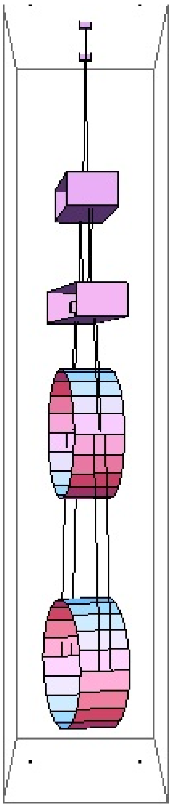
Mode #5 - modeY1

Back to Top

```
0.648033 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0.1767	-0.000132508	0
Mass U	0	0	0	0.399286	-0.000129922	0

Mass 2	0	0	0	0.517833	-0.000181272	0
optic	0	0	0	0.735661	-0.000194241	0

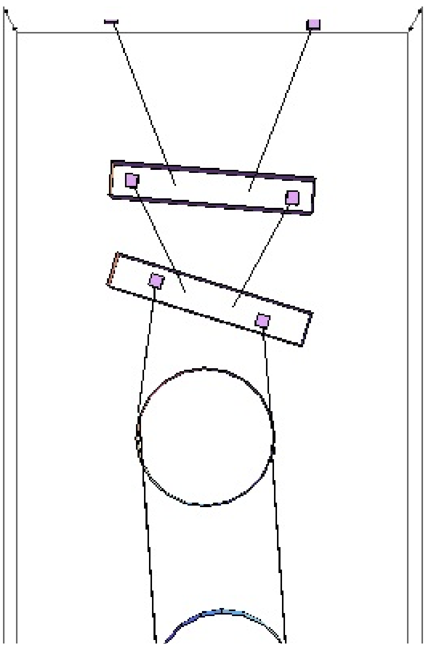


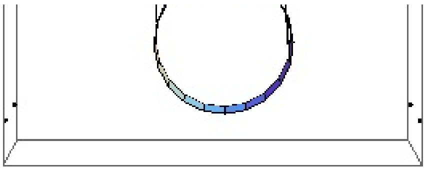
Mode #6 - modeR1

Back to Top

```
0.834372 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000144373	0.00171993	0	0	-0.0409348	0.163345
Mass U	0.00026406	0.013747	0	-0.000123373	-0.0199937	0.578416
Mass 2	0	0.0341723	0	-0.000197813	0.032416	0.560927
optic	0.00016593	-0.0521038	0	-0.000388528	0.0365946	0.56176



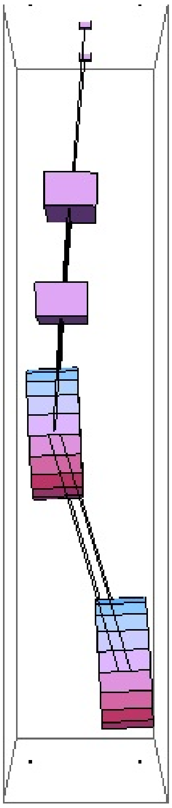


Mode #7 - modeL2

Back to Top

```
0.859087 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.222945	0	0	0	0.112251	0.00116836
Mass U	-0.362152	0.000149772	0	0	-0.034197	0.00594976
Mass 2	-0.474369	0.000403761	0	0	-0.297203	0.00592939
optic	0.608708	-0.000534061	0	0	-0.348402	0.00593874

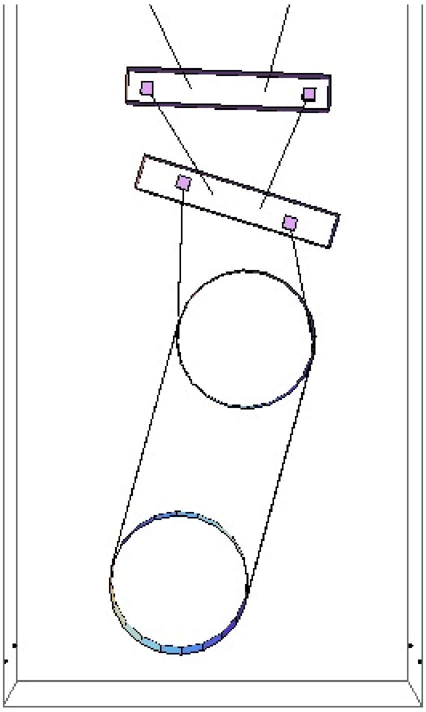


Mode #8 - modeT2

Back to Top

```
0.907233 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	-0.0754284	0	0	-0.0410657	0.0768963
Mass U	0.000107966	-0.11795	0	0	-0.0278383	0.587839
Mass 2	-0.000162053	-0.160243	0	0	0.02376	0.534893
optic	0.000216965	0.161902	0	-0.000179448	0.027495	0.535862

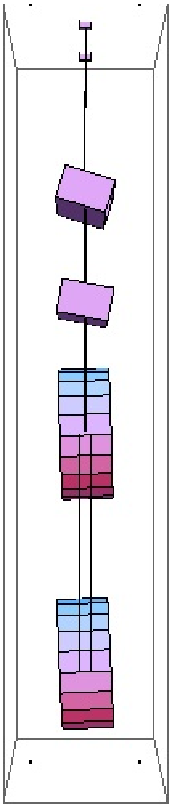


Mode #9 - modeP2

Back to Top

```
1.33693 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0.000345205	0	-0.000274822	0.844909	-0.00851527
Mass U	-0.000788147	-0.000781191	0	-0.000392055	0.511658	0.00874516
Mass 2	0.00114224	0.000365137	0	-0.000270362	-0.0879007	0.0179104
optic	-0.000410053	-0.000121282	0	0.00103883	-0.125738	0.0179803

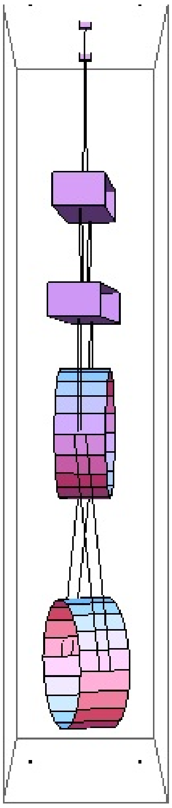


Mode #10 - modeY2

Back to Top

```
1.34081 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	-0.199076	-0.00351602	0
Mass U	0	0	0	-0.339731	-0.00222928	0
Mass 2	0	0	0	-0.237608	0.000389048	0
optic	0	0	0	0.887962	0.000557943	0

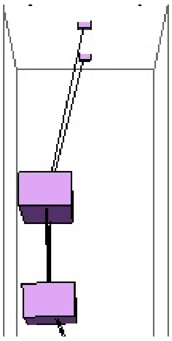


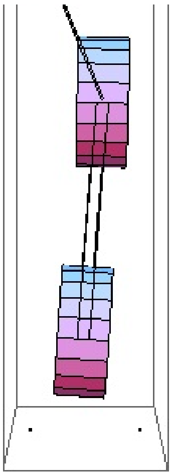
Mode #11 - modeL3

Back to Top

```
1.92516 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.614169	0.000504905	0	0	-0.0171074	0.00134469
Mass U	-0.557446	0.000689412	0	0	-0.198872	0.00230688
Mass 2	0.268566	-0.000243384	0	0	0.152759	-0.00362853
optic	-0.0337951	0	0	0	0.419009	-0.00365817



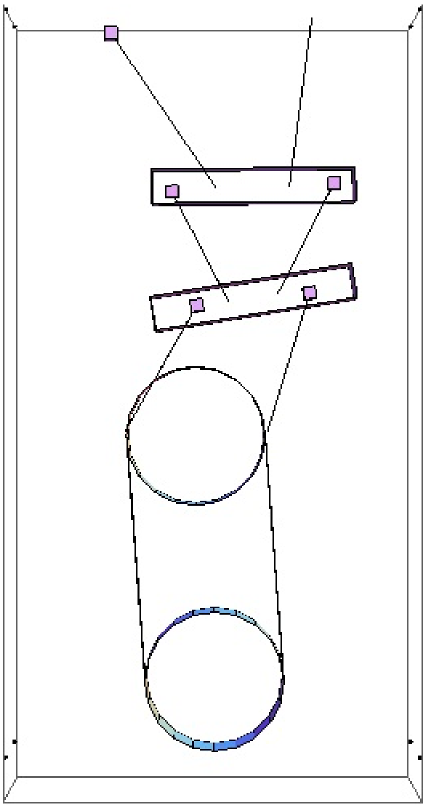


Mode #12 - modeT3

Back to Top

```
2.0476 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00063053	-0.444505	0	-0.000323924	-0.190448	-0.0427501
Mass U	-0.000483082	-0.446813	0	0	0.0222081	-0.727811
Mass 2	0.000198063	0.18452	0	0.000181215	-0.000732207	0.00176728
optic	0	-0.0202728	0	0	-0.00254286	0.00177061



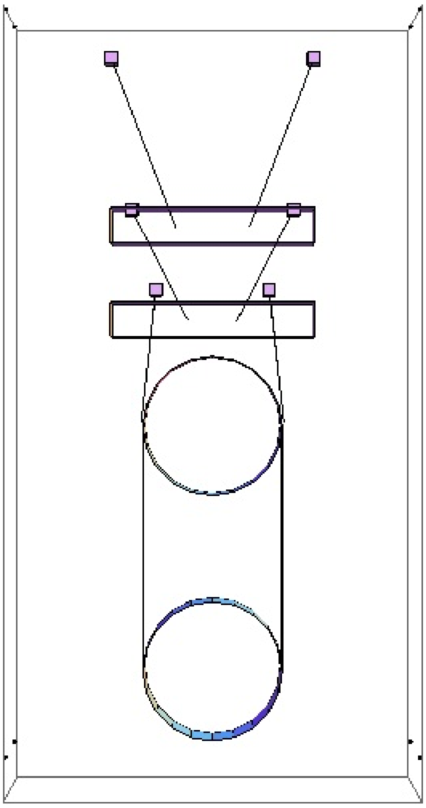
Mode #13 - modeV2

Back to Top

```
2.2276 Hz
```

	x	y	z	yaw	pitch	roll

Mass N	0	0	-0.839939	0	0	0
Mass U	0	0	-0.457475	0	0	0
Mass 2	0	0	0.2041	0	0	0
optic	0	0	0.208715	0	0	0

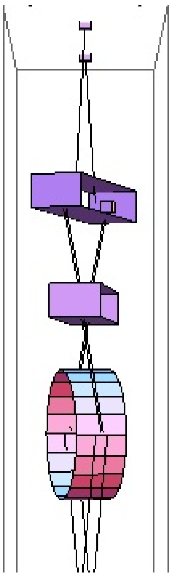


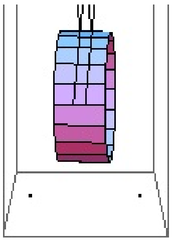
Mode #14 - modeY3

Back to Top

```
2.28221 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0.000239711	0	-0.734046	-0.000898499	-0.00289334
Mass U	0	-0.000144728	0	-0.247536	-0.00310605	-0.00149403
Mass 2	0	0	0	0.592253	0	0.00120532
optic	0	0	0	-0.221616	0.000346863	0.00121923



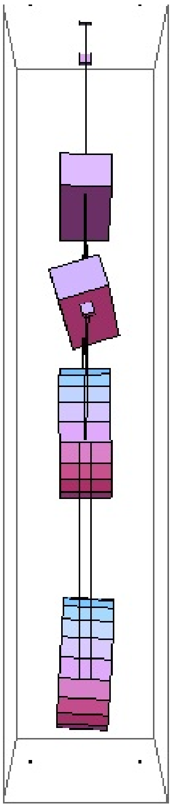


Mode #15 - modeR2

Back to Top

```
2.62396 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000199396	0.0417693	0	0.00196754	0.0507925	-0.436939
Mass U	0.00065775	-0.0233905	0	0.00373188	-0.639738	-0.497388
Mass 2	-0.000311808	0.000230514	0	-0.00380312	-0.0198462	0.256928
optic	0	0	0	0.000986632	0.114291	0.260866

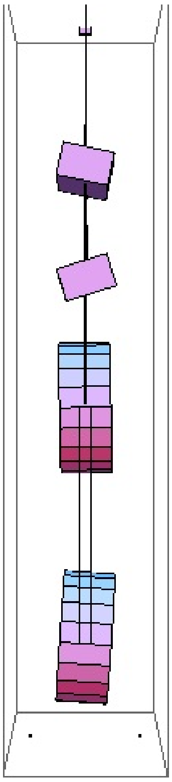


Mode #16 - modeP3

Back to Top

```
2.70885 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000792614	-0.0031225	0	0.000236552	0.555882	0.0165227
Mass U	0.000715542	-0.00136594	0	-0.000147895	-0.800426	0.106654
Mass 2	-0.00038013	0.00121727	0	0.000143614	-0.044112	-0.0536609
optic	0	0	0	0	0.175647	-0.0545386

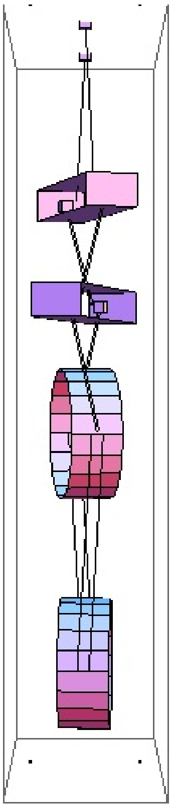


Mode #17 - modeY4

Back to Top

```
2.97614 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0.602862	-0.00062326	0.000803286
Mass U	0	0.000192896	0	-0.679844	0.00380208	-0.00570884
Mass 2	0	0	0	0.409909	-0.00607791	0.00215398
optic	0	0	0	-0.0781487	0.0118878	0.00219667

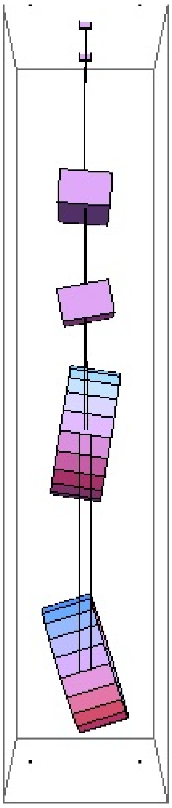


Mode #18 - modeP4

Back to Top

```
2.98639 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00381705	0.00361417	0	0.00471976	0.191106	-0.0387708
Mass U	0.00176082	-0.00466636	0	-0.00487164	-0.461766	0.032765
Mass 2	0.000323719	0.000947461	0	0.00286964	0.399833	-0.0140413
optic	0	0	0	-0.000542635	-0.766353	-0.0143216

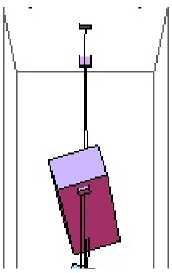


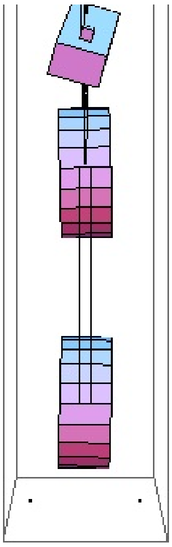
Mode #19 - modeR3

Back to Top

```
3.31274 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000825017	0.0677189	0	0.00495425	-0.509663	-0.471622
Mass U	-0.00106427	-0.0741016	0	-0.00246913	0.466549	0.486821
Mass 2	0.000131638	0.0113298	0	0.00106867	-0.0375539	-0.155638
optic	0	-0.000432935	0	-0.000158608	0.0429845	-0.15948



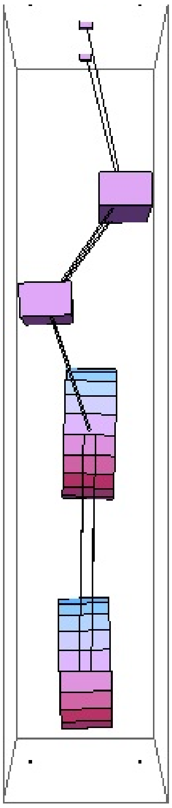


Mode #20 - modeL4

Back to Top

```
3.40455 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.65497	0.00160361	0	0.00012272	0.027761	-0.00898449
Mass U	0.651637	-0.00190408	0	-0.000109537	0.255725	0.027104
Mass 2	-0.0719544	0.000326404	0	0	-0.190138	-0.00733397
optic	0.0026723	0	0	0	0.194801	-0.00752544

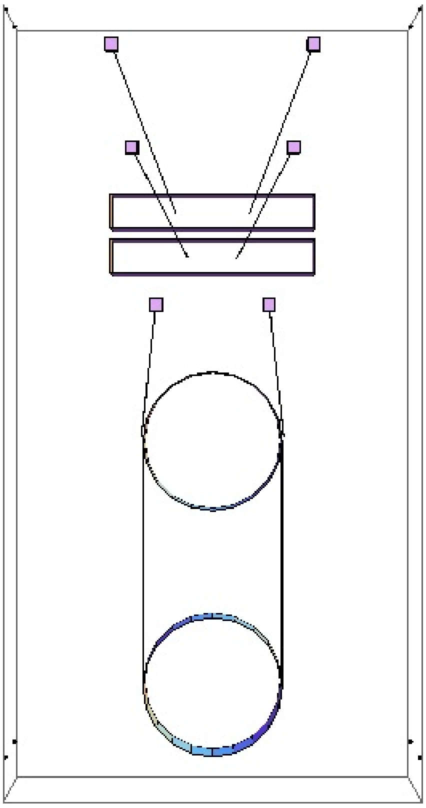


Mode #21 - modeV3

Back to Top

```
3.58064 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.539686	0	0	0
Mass U	0	0	-0.826316	0	0	0
Mass 2	0	0	0.11049	0	0	0
optic	0	0	0.117186	0	0	0

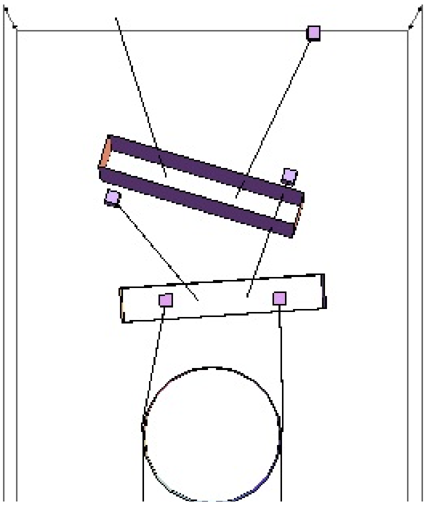


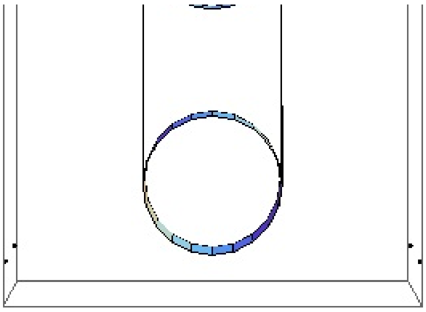
Mode #22 - modeT4

Back to Top

5.04715 Hz

	x	y	z	yaw	pitch	roll
Mass N	0	-0.0763941	0	0.0039506	-0.498982	-0.831359
Mass U	-0.000108884	0.0689805	0	-0.000447911	0.112627	0.190619
Mass 2	0	-0.00318782	0	0	-0.00193336	-0.0100341
optic	0	0	0	0	0.000576124	-0.0106286



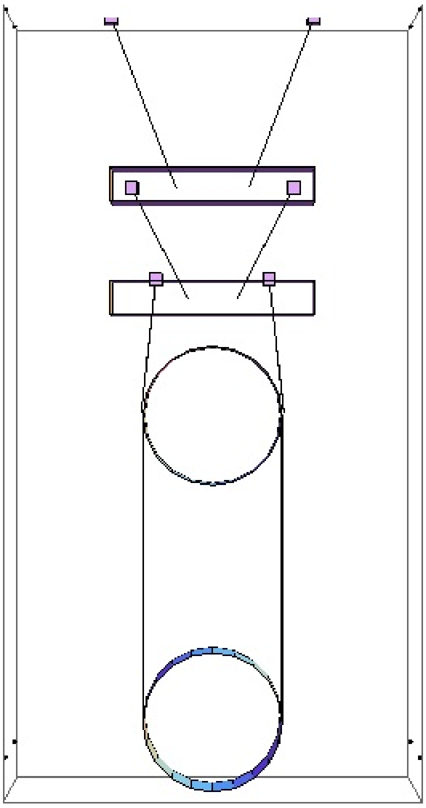


Mode #23 - modeV4

Back to Top

```
18.3013 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0	0	0
Mass U	0	0	-0.00769768	0	0	0
Mass 2	0	0	0.44188	0	0	0
optic	0	0	-0.897041	0	0	0

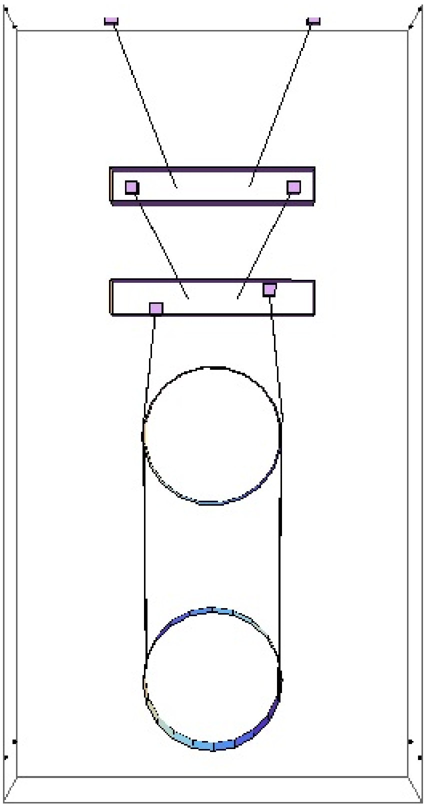


Mode #24 - modeR4

Back to Top

```
25.6281 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0	0	0
Mass U	0	0	0	0	-0.000593139	-0.00326645
Mass 2	0	0	0	0	0	0.40554
optic	0	0	0	0	0	-0.914071



Parameters

Back to Top

```
g -> 9.81

nx -> 0.13                                          T040214-01 (display only)

ny -> 0.5                                           T040214-01 (display only)

nz -> 0.084                                         T040214-01 (display only)

denn -> 0                                           (unused)

mn -> 22.002                                        measured value listed in
E1000186-v18

Inx -> 0.461176                                     T1000286-v9 11.2 Reaction Top
Mass

Iny -> 0.0742518                                    T1000286-v9 11.2 Reaction Top
Mass

Inz -> 0.472634                                     T1000286-v9 11.2 Reaction Top
Mass

ux -> 0.13                                          T040214-01 (display only)

uy -> 0.5                                           T040214-01 (display only)

uz -> 0.084                                         T040214-01 (display only)

den1 -> 0                                           (unused)

m1 -> 21.532                                        measured value listed in
E1000186-v18

I1x -> 0.50522                                      T1000286-v9, reaction UIM with
pitch adjuster, Joe's zz

I1y -> 0.073433                                     T1000286-v9, reaction UIM with
pitch adjuster, Joe's yy

I1z -> 0.518171                                     T1000286-v9, reaction UIM with
pitch adjuster, Joe's xx
```

```
ix -> 0.13                                          T1000405-v1

ir -> 0.17                                          T1000405-v1

den2 -> 0                                           not used

m2 -> 53.138                                        measured value listed in
E1000186_v18_QUAD-2_Process_Traveler

I2x -> 0.852587                                     T1000286-v9 11.4 PRM for ERM

I2y -> 0.498401                                     T1000286-v9 11.4 PRM for ERM

I2z -> 0.486309                                     T1000286-v9 11.4 PRM for ERM

tx -> 0.13028                                       ERM 01, C1001320-v1

tr -> 0.17004                                       ERM 01, C1001320-v1

den3 -> 3980                                        unused

m3 -> 25.993                                        ERM 01, from
<nebula.ligo.caltech.edu/optics/>

I3x -> 0.375776                                     0.5*m3*tr^2

I3y -> 0.224652                                     (1/12)*m3*(3*tr^2 + tx^2)

I3z -> 0.224652                                     (1/12)*m3*(3*tr^2 + tx^2))

ln -> 0.449192                                      from QUAD01BUILD0101M0

l1 -> 0.308585                                      from QUAD01BUILD0101M0

l2 -> 0.330787                                      from QUAD01BUILD0101M0

l3 -> 0.60406                                       from QUAD01BUILD0101M0

nwn -> 2

nw1 -> 4

nw2 -> 4

nw3 -> 4

rn -> 0.00055                                       T1000428-v7

r1 -> 0.000355                                      T1000428-v7

r2 -> 0.0003175                                     T1000428-v7

r3 -> 0.0002285                                     T1000428-v7

             11
Yn -> 2.12 10                                       measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y1 -> 2.12 10                                       measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y2 -> 2.12 10                                       measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

             11
Y3 -> 2.12 10                                       measured, MB, 11/18/05, via
IFOModel v4.1; cf. 2.2 in T010103-05

dm -> -0.00354533                                   effective value 0.001; BNS
6/18/08

dn -> 0.00318414                                    effective value 0.001 after
blade correction; BNS 6/18/08

d0 -> -0.00175338                                   effective value 0.001

d1 -> 0.00292556                                    effective value 0.001 after
blade correction; BNS 6/18/08
```

```
d2 -> 0.00789384                                                  Originally 0.010 effective BNS
6/18/08; Adjustment for new wire clamp design BNS 2/16/09.  0.18 mm subtracted to match Joe's
updated value, BNS 7/21/09.

d3 -> -0.00166841                                                 0.0010 effective; BNS 6/18/08

d4 -> -0.00166841                                                 0.0010 effective; BNS 6/18/08

sn -> 0                                                           T1000405-v1 (unused)

su -> 0.003                                                       T1000405-v1

si -> 0.003                                                       T1000405-v1

sl -> 0.015                                                       T1000405-v1

nn0 -> 0.25                                                       T1000405-v1

nn1 -> 0.09                                                       T1000405-v1

n0 -> 0.2                                                         T1000405-v1

n1 -> 0.06                                                        T1000405-v1

n2 -> 0.14                                                        T1000405-v1

n3 -> 0.1762                                                      CT, email to NR, 9/22/04

n4 -> 0.1712                                                      CT, email to NR, 9/22/04

n5 -> 0.1712                                                      CT, email to NR, 9/22/04

tln -> 0.416185

tl1 -> 0.27643

tl2 -> 0.33962

tl3 -> 0.600723

ltotal -> 1.63296

bd -> 0.

unstretched -> False

vertblades -> True

matlabcompat -> False

uln -> 0.447756

ul1 -> 0.307567

ul2 -> 0.329825

ul3 -> 0.602953

sln -> 0.449192

sl1 -> 0.308585

sl2 -> 0.330787

sl3 -> 0.60406

sin -> -0.356195

si1 -> -0.453684

si2 -> 0.109436

si3 -> 0.

cn -> 0.934412

c1 -> 0.891163

c2 -> 0.993994

c3 -> 1.
```

```
pitchbul -> 0

pitchbur -> 0

pitchbil -> 0

pitchbir -> 0

pitchbll -> 0

pitchblr -> 0

rollbul -> 0

rollbur -> 0

rollbil -> 0

rollbir -> 0

rollbll -> 0

rollblr -> 0

                -7
An -> 9.50332 10

                -7
A1 -> 3.95919 10

                -7
A2 -> 3.16692 10

               -7
A3 -> 1.6403 10

kwn -> 448517.

kw1 -> 271999.

kw2 -> 202967.

kw3 -> 57567.6

flexn -> 0.00454533                                 calculated as usual

flex1 -> 0.00275338                                 calculated as usual

flex2 -> 0.00292616                                 calculated as usual

flex3 -> 0.00266841                                 calculated as usual

kbuz -> 1411.464026291042094835154548022            Solved for given mass and
freqs; BNS 6/19/08

kbiz -> 1650.524059045379672879214321779            Solved for given mass and
freqs; BNS 6/19/08

kblz -> 2423.519015280238231051986060810            Solved for given mass and
freqs; BNS 6/19/08

kbux -> 100000.                                     as for middle

kbix -> 100000.                                     Justin 11/29/05

kblx -> 80000.                                      Ian 12/09/05

bdu -> 0.426275

bdi -> 0.299149

bdl -> 0.160155

mn3 -> 122.665                                      calculated as usual

m13 -> 100.663                                      calculated as usual

m23 -> 79.131                                       calculated as usual

Inxy -> -0.0375611                                  T1000286-v9 11.2 Reaction Top
Mass
```

```
Inyz -> -0.0000435986                                   T1000286-v9 11.2 Reaction Top
Mass

Inzx -> 0.00171393                                      T1000286-v9 11.2 Reaction Top
Mass

COM0x -> 0

COM0y -> 0

COM0z -> 0

FRP0x -> 0

FRP0y -> 0

FRP0z -> 0

I1xy -> -0.0132118                                      T1000286-v9, reaction UIM with
pitch adjuster, Joe's yz

I1yz -> 0.0000501027                                    T1000286-v9, reaction UIM with
pitch adjuster, -Joe's xy

I1zx -> -0.0000341209                                   T1000286-v9, reaction UIM with
pitch adjuster, -Joe's zx

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

I2xy -> -0.0000586847                                   T1000286-v9 11.4 PRM for ERM

I2yz -> 0.0000237548                                    T1000286-v9 11.4 PRM for ERM

I2zx -> 0.000044073                                     T1000286-v9 11.4 PRM for ERM

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0

I3yz -> 0

I3zx -> 0

COM3x -> 0

COM3y -> 0

COM3z -> 0

FRP3x -> 0

FRP3y -> 0

FRP3z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03
```

Mn1 -> $7.18688\ 10^{-14}$	
Mn2 -> $7.18688\ 10^{-14}$	
M11 -> $1.24739\ 10^{-14}$	
M12 -> $1.24739\ 10^{-14}$	
M21 -> $7.98114\ 10^{-15}$	
M22 -> $7.98114\ 10^{-15}$	
M31 -> $2.14109\ 10^{-15}$	
M32 -> $2.14109\ 10^{-15}$	
temperature -> 290.	
boltzmann -> $1.38066\ 10^{-23}$	
alphasilica -> $5.1\ 10^{-7}$	IFOModel v4.1
betasilica -> 0.000152	IFOModel v4.1
rhosilica -> 2200.	IFOModel v4.1
Csilica -> 772.	IFOModel v4.1
Ksilica -> 1.38	IFOModel v4.1
Ysilica -> $7.27\ 10^{10}$	IFOModel v4.1
phisilica -> $4.1\ 10^{-10}$	IFOModel v4.1
phissilica -> $3.\ 10^{-11}$	surface
rhosteel -> 7800.	IFOModel v4.1
Csteel -> 460.	IFOModel v4.1
Ksteel -> 49.	IFOModel v4.1
Ysteel -> $2.12\ 10^{11}$	measured, MB, 11/18/05
alphasteel -> 0.000012	IFOModel v4.1
betasteel -> -0.00025	IFOModel v4.1
phisteel -> 0.0001	IFOModel v4.1
rhomarag -> 7800.	IFOModel v4.1
Cmarag -> 460.	IFOModel v4.1
Kmarag -> 20.	IFOModel v4.1
Ymarag -> $1.87\ 10^{11}$	IFOModel v4.1
alphamarag -> 0.000011	IFOModel v4.1
betamarag -> -0.00025	Geppo's value - Bench v4.1 is wrong
phimarag -> 0.0001	IFOModel v4.1
tmU -> 0.0043	IFOModel v4.1

```
tmI -> 0.0046                                                  IFOModel v4.1

tmL -> 0.0042                                                  IFOModel v4.1

magicnumber -> 0.0737472                                       Zener, 1938, Phys. Rev. 53:90-
99

deltabladeU -> 0.00182883

deltabladeI -> 0.00182883

deltabladeL -> 0.00182883

deltawireU -> 0.00280695

deltawireI -> 0.00281841

deltawireL -> 0.00277545

deltafibre -> 0.00265949

taubladeU -> 0.336093

taubladeI -> 0.384626

taubladeL -> 0.320643

tauwireU -> 0.00653413

tauwireI -> 0.00272219

tauwireL -> 0.00217746

taufibre -> 0.00112781

                                        0.003862 #1
damping[imag, bladeUtype] -> (0.0001 + --------------- & )
                                                    2
                                      1 + 4.45943 #1

                                       0.00441968 #1
damping[imag, bladeItype] -> (0.0001 + --------------- & )
                                                    2
                                      1 + 5.84032 #1

                                       0.00368446 #1
damping[imag, bladeLtype] -> (0.0001 + --------------- & )
                                                    2
                                      1 + 4.05884 #1

damping[imag, wireUtype] -> (0.0001 & )

damping[imag, wireItype] -> (0.0001 & )

damping[imag, wireLtype] -> (0.0001 & )

                                        0.000115239 #1
damping[imag, wireUatype] -> (0.0001 + ------------------ & )
                                                       2
                                      1 + 0.00168552 #1

                                        0.0000482062 #1
damping[imag, wireIatype] -> (0.0001 + ------------------- & )
                                                        2
                                      1 + 0.000292548 #1

                                        0.000037972 #1
damping[imag, wireLatype] -> (0.0001 + ------------------ & )
                                                       2
                                      1 + 0.00018718 #1

damping[imag, fibretype] -> (0.0001 & )

                                        0.0000188457 #1
damping[imag, fibreatype] -> (0.0001 + -------------------- & )
                                                         2
                                      1 + 0.0000502144 #1

x00 -> 0

y00 -> 0
```

```
z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx0 -> 0

kcony0 -> 0

kconz0 -> 0

kconyaw0 -> 0

kconpitch0 -> 0

kconroll0 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

ribbons -> False

dumbbell -> True

Inxz -> 0.00171393

Inzy -> -0.0000435986

Inyx -> -0.0375611

I1xz -> -0.0000341209

I1zy -> 0.0000501027

I1yx -> -0.0132118

I2xz -> 0.000044073

I2zy -> 0.0000237548

I2yx -> -0.0000586847

                -7
A3n -> 1.6403 10

bend1 -> 0.00493752                                                   calculated as usual
```

```
bend1 -> 0.00495752                                                   calculated as usual

bend2 -> 0.00485172                                                   calculated as usual

thetan -> -20.8667                                                    calculated as usual

theta1 -> -26.9803                                                    calculated as usual

theta2 -> 6.2828                                                      calculated as usual

theta3 -> 0.                                                          calculated as usual

sigmasilica -> 0.17

                     10
Gsilica -> 3.10684 10                                                 shear modulus

dssilica -> 0.015                                                     IFOModel v4.1
```

Back to Top

Case 20121115TMproductionWire of Mathematica model QuadLite2Lateral

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
"20121115TMproductionWire"
```

```
Old-style metal build of main chain, without prisms. Corresponds to
^/trunk/QUAD/Common/MatlabTools/QuadModel_Production/quadopt_wire.m r3736 of 11/15/12.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeL1	modeT1	modeV1	modeP1	modeY1	modeR1
modeL2	modeT2	modeP2	modeY2	modeL3	modeT3
modeV2	modeY3	modeP3	modeR2	modeP4	modeY4
modeR3	modeL4	modeV3	modeT4	modeV4	modeR4

Mode Summary

Back to Top

```
N    f          name     type

1    0.434017   modeL1   pitch3   pitch2   pitch1   pitch0

2    0.460117   modeT1   y3       roll3    roll2

3    0.546505   modeV1   z3       z2

4    0.588103   modeP1   pitch3   pitch2   pitch1

5    0.604666   modeY1   yaw3     yaw2

6    0.847285   modeR1   roll1    roll3    roll2

7    0.981483   modeL2   pitch0   x2       x1       x3

8    1.03398    modeT2   y2       y1       roll1    y3

9    1.33001    modeP2   pitch0

10   1.35465    modeY2   yaw3     yaw1

11   1.98651    modeL3   x0       x1       pitch3

12   2.1007     modeT3   roll1    y0

13   2.21871    modeV2   z0       z1

14   2.38956    modeY3   yaw0     yaw2

15   2.52201    modeP3   pitch1   pitch0   pitch2

16   2.6257     modeR2   roll1    roll0    pitch0

17   2.8107     modeP4   pitch1

18   3.0358     modeY4   yaw1     yaw0
```

19	3.31136	modeR3	pitch0	pitch1	roll0
20	3.39588	modeL4	x0	x1	
21	3.53664	modeV3	z1	z0	
22	5.04259	modeT4	roll0	pitch0	
23	17.1245	modeV4	z3	z2	
24	23.2379	modeR4	roll3	roll2	

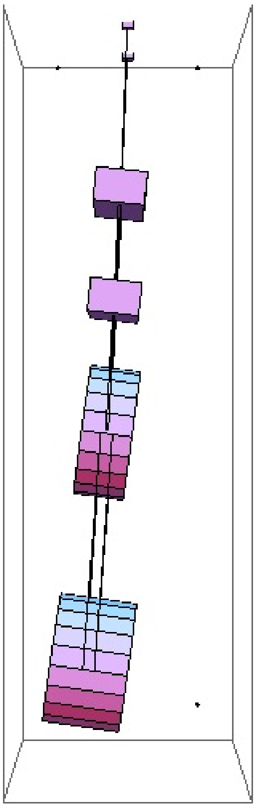
Mode Shapes

Mode #1 - modeL1

Back to Top

0.434017 Hz

	x	y	z	yaw	pitch	roll
Mass N	-0.0658938	-0.000103917	0	0	0.384612	-0.00131095
Mass U	-0.117217	-0.00029917	0	0	0.419429	-0.00260164
Mass 2	-0.192486	-0.000537492	0	0	0.483538	-0.00219019
optic	-0.358685	-0.00101092	0	0	0.508465	-0.00219138



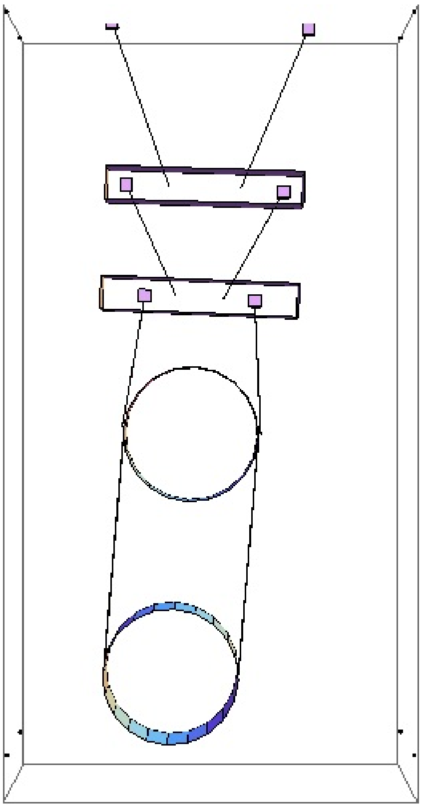
Mode #2 - modeT1

Back to Top

0.460117 Hz

	x	y	z	yaw	pitch	roll
Mass N	-0.000474589	0.0956553	0	0	-0.0191302	0.211032
Mass U	-0.000819611	0.182576	0	0	-0.014936	0.287249

Mass 2	-0.00123988	0.331321	0	0.00010476	-0.00872164	0.356724
optic	-0.00242017	0.682069	0	0.000126973	-0.00906991	0.357029

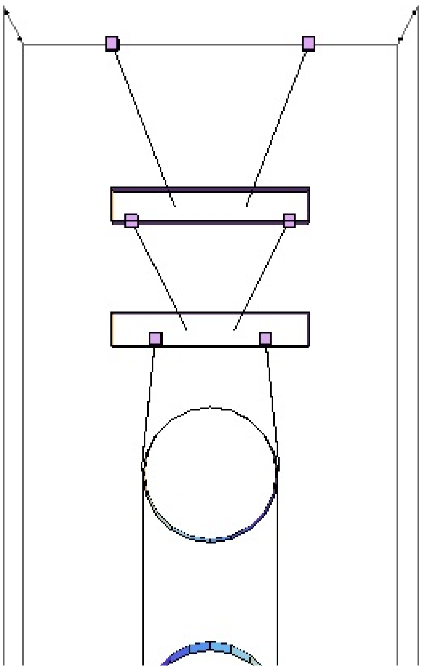


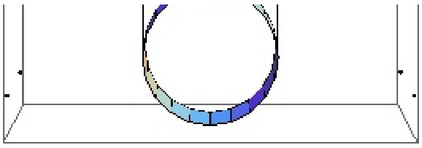
Mode #3 - modeV1

Back to Top

```
0.546505 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	-0.267799	0	0	0
Mass U	0	0	-0.475816	0	0	0
Mass 2	0	0	-0.591802	0	0	0
optic	0	0	-0.593004	0	0	0



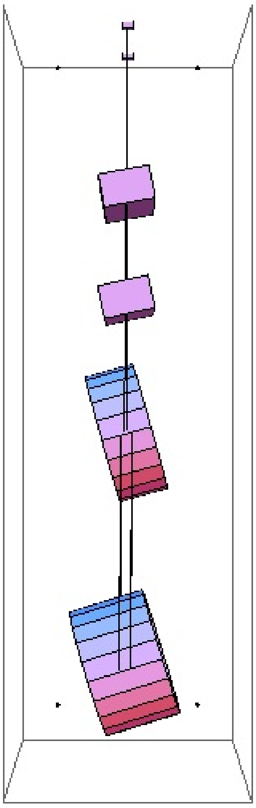


Mode #4 - modeP1

Back to Top

```
0.588103 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00450564	-0.000149061	0	0	-0.370438	0.00207024
Mass U	-0.00731126	0	0	0	-0.432599	0.00530386
Mass 2	-0.00597588	-0.000104549	0	0	-0.554458	0.00423371
optic	-0.0139454	-0.000470452	0	0	-0.606496	0.00423795

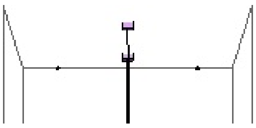


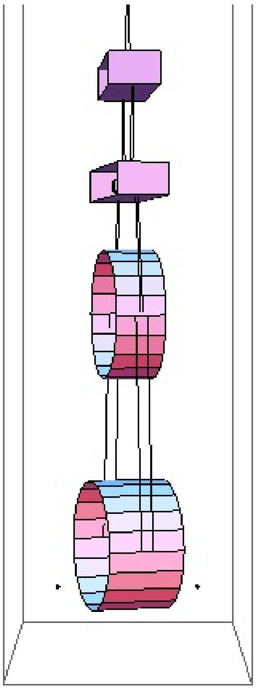
Mode #5 - modeY1

Back to Top

```
0.604666 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0.16865	0	0
Mass U	0	0	0	0.389482	0	0
Mass 2	0	0	0	0.518186	0	0
optic	0	0	0	0.742525	0	0



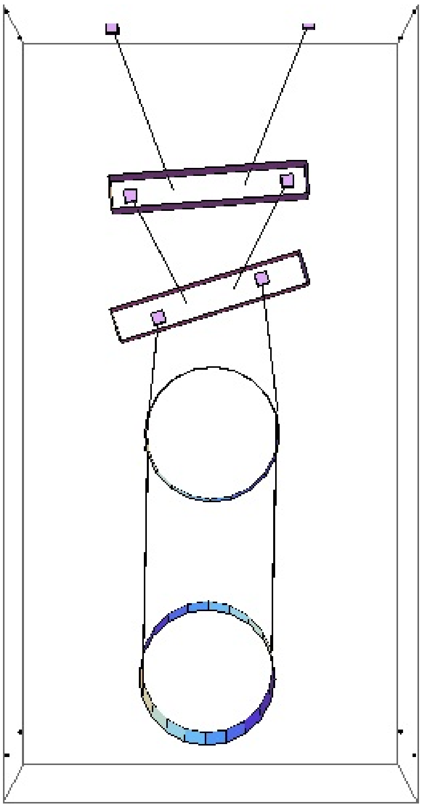


Mode #6 - modeR1

Back to Top

```
0.847285 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000240362	0.00770538	0	0	0.051285	-0.152379
Mass U	-0.000436085	0.00272255	0	0	0.03131	-0.583325
Mass 2	-0.000297319	-0.00845504	0	0	-0.0227273	-0.561114
optic	0.000184284	0.0163913	0	0.000228804	-0.0286096	-0.562415

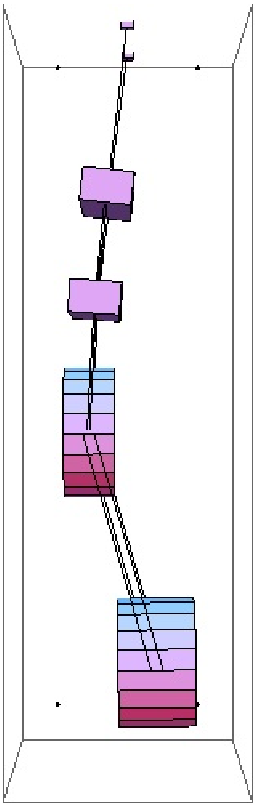


Mode #7 - modeL2

Back to Top

```
0.981483 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.247193	-0.000227219	0	0	0.550655	0.00194143
Mass U	-0.392548	-0.000467533	0	0	0.350966	0.0208453
Mass 2	-0.47396	0.000148461	0	0	-0.0137963	0.0228696
optic	0.357559	-0.000221395	0	0	-0.0651089	0.0229426

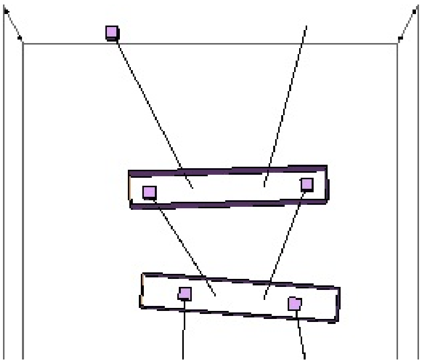


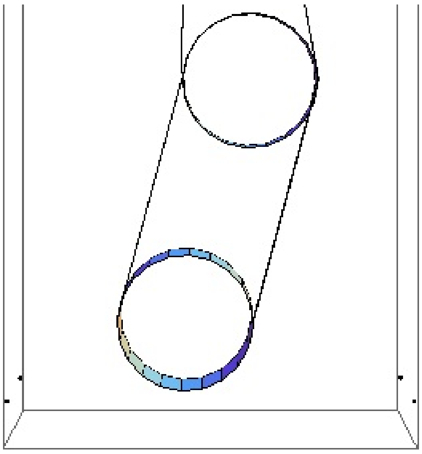
Mode #8 - modeT2

Back to Top

```
1.03398 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00020605	-0.241105	0	-0.000148323	0.0443797	-0.155748
Mass U	0.000305622	-0.398149	0	0	0.00785713	0.397187
Mass 2	0.000431106	-0.532573	0	0	-0.00213153	0.317825
optic	-0.000283249	0.33576	0	0.000285814	-0.00307915	0.319223



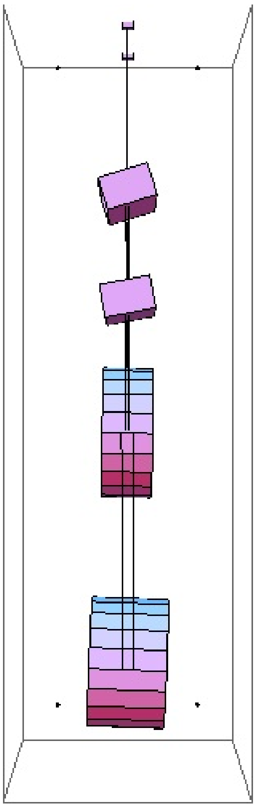


Mode #9 - modeP2

Back to Top

```
1.33001 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.000831797	-0.00049602	0	-0.000158305	-0.839981	0.00812793
Mass U	-0.000476079	0.000498225	0	-0.000239336	-0.528962	-0.00845436
Mass 2	-0.00169326	-0.000634388	0	-0.000154551	0.0492757	-0.0176342
optic	0.000669037	0.000228268	0	0.000334702	0.106933	-0.0177404



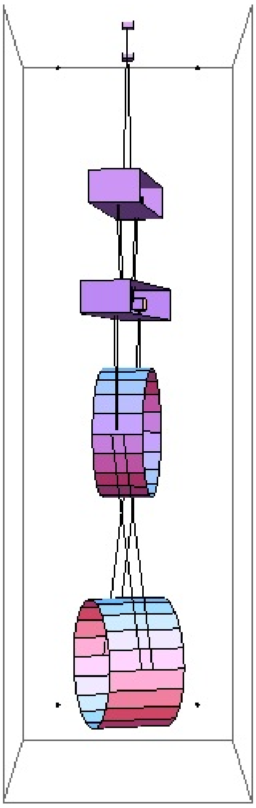
Mode #10 - modeY2

Back to Top

```
1.35465 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	-0.306539	0.00218989	-0.000185112

Mass U	0	0	0	-0.528124	0.00140087	0
Mass 2	0	0	0	-0.363357	-0.000120042	0.00021079
optic	0	0	0	0.703622	-0.000273026	0.000212107

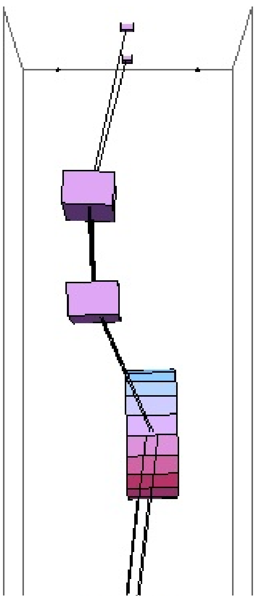


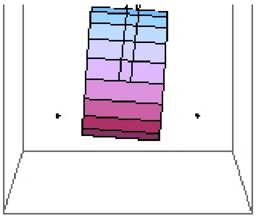
Mode #11 - modeL3

Back to Top

1.98651 Hz

	x	y	z	yaw	pitch	roll
Mass N	-0.575796	0.0012786	0	0	0.22976	-0.00872467
Mass U	-0.502561	0.000242381	0	0	-0.231926	0.000296353
Mass 2	0.361559	-0.000198535	0	0	-0.110967	0.000354316
optic	-0.0423083	0	0	0	0.405469	0.000359274



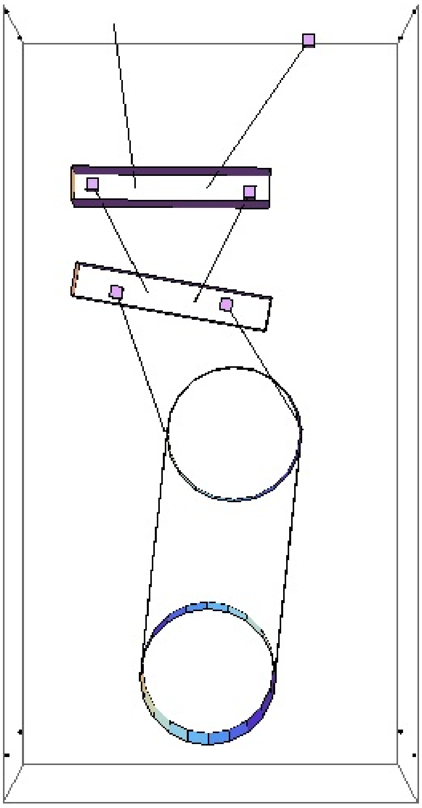


Mode #12 - modeT3

Back to Top

```
2.1007 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000798575	0.413805	0	0.000332342	0.202135	0.0393166
Mass U	0.000607738	0.408828	0	0	-0.0366548	0.743512
Mass 2	-0.000389714	-0.248528	0	-0.000283633	-0.00183527	-0.0358825
optic	0	0.0257776	0	0.000107168	0.00483507	-0.0363535

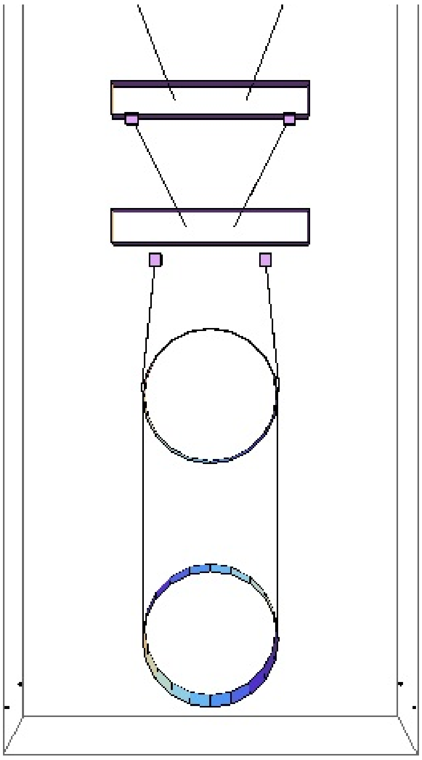


Mode #13 - modeV2

Back to Top

```
2.21871 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.833968	0	0	0
Mass U	0	0	0.466789	0	0	0
Mass 2	0	0	-0.204527	0	0	0
optic	0	0	-0.211598	0	0	0

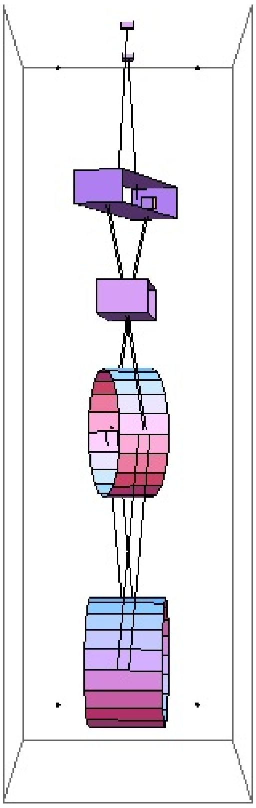


Mode #14 - modeY3

Back to Top

```
2.38956 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0.000382368	0	-0.743869	-0.000794733	-0.00425877
Mass U	0	-0.000196589	0	-0.131	-0.00298574	-0.00290397
Mass 2	0	0	0	0.632842	-0.000605515	0.00192383
optic	0	0	0	-0.170191	0.00076039	0.00196264

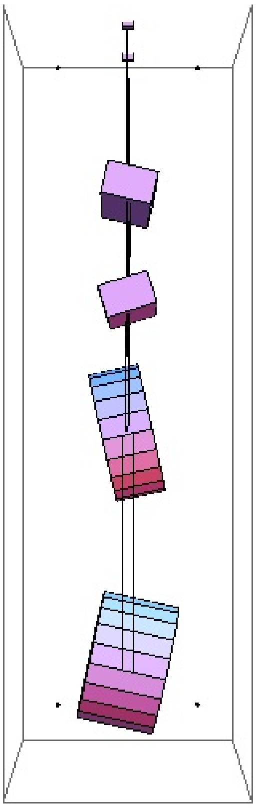


Mode #15 - modeP3

Back to Top

```
2.52201 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.0047795	0.00713176	0	0.00138178	0.473796	-0.0852955
Mass U	0.00193722	-0.00514784	0	0.000554119	-0.609374	-0.0552161
Mass 2	-0.00229058	0.000967314	0	-0.00137181	-0.443626	0.0284188
optic	0.000159239	0	0	0.000322321	0.441912	0.0290594

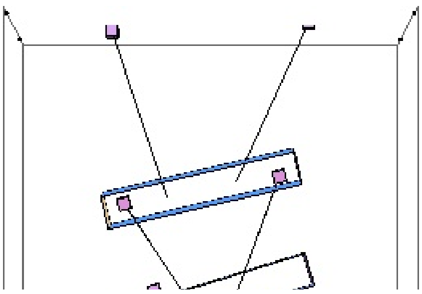


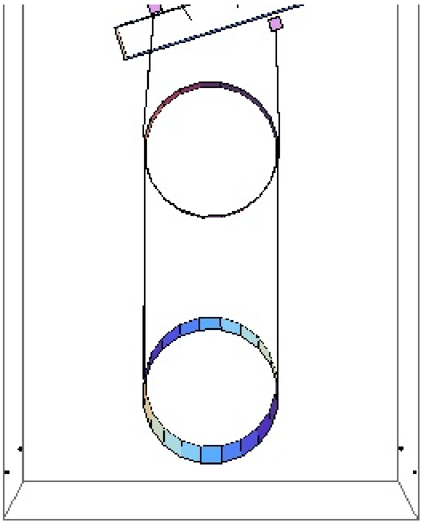
Mode #16 - modeR2

Back to Top

```
2.6257 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	-0.00165237	0.0478619	0	0.00476174	-0.368859	-0.471472
Mass U	0	-0.0210597	0	0.00340974	-0.154877	-0.621325
Mass 2	0.000405174	-0.00159193	0	-0.00605031	0.15032	0.304329
optic	0	0	0	0.0012881	-0.128025	0.311787



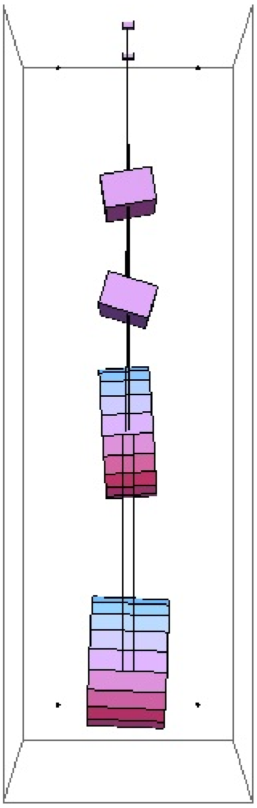


Mode #17 - modeP4

Back to Top

```
2.8107 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.00117068	-0.0025988	0	0	-0.482116	0.0382951
Mass U	-0.000843079	0.00480848	0	-0.00025398	0.847288	-0.0571076
Mass 2	0	-0.00184656	0	0.00029492	-0.173368	0.0260702
optic	0	0	0	0	0.115968	0.0268063

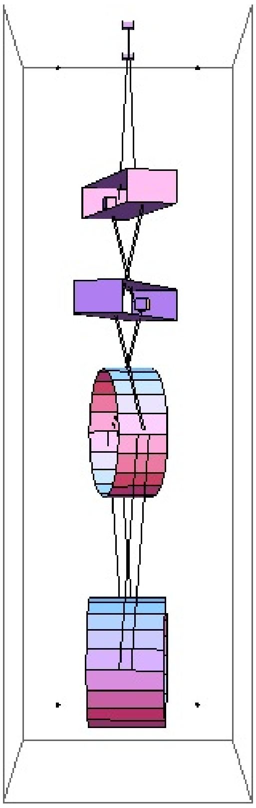


Mode #18 - modeY4

Back to Top

```
3.0358 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	-0.000101767	0	0.535903	0.00281476	0.0010808
Mass U	0	0.000199106	0	-0.655611	-0.00524544	-0.00473838
Mass 2	0	0	0	0.525925	0.000567001	0.00161257
optic	0	0	0	-0.0794965	-0.000296117	0.00166597

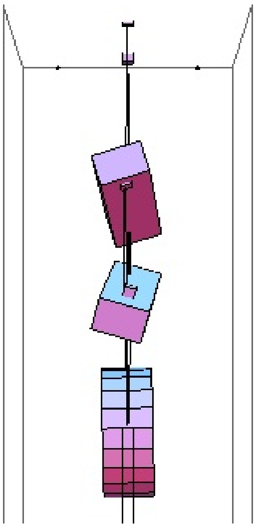


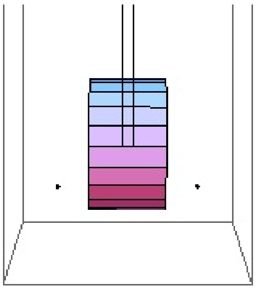
Mode #19 - modeR3

Back to Top

```
3.31136 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.000816824	0.0696695	0	0.00549609	-0.5285	-0.464101
Mass U	-0.00105204	-0.0708106	0	-0.00294004	0.491757	0.460513
Mass 2	0.000197265	0.0147276	0	0.00167557	-0.033888	-0.138532
optic	0	-0.000543304	0	-0.000207862	0.013704	-0.144032



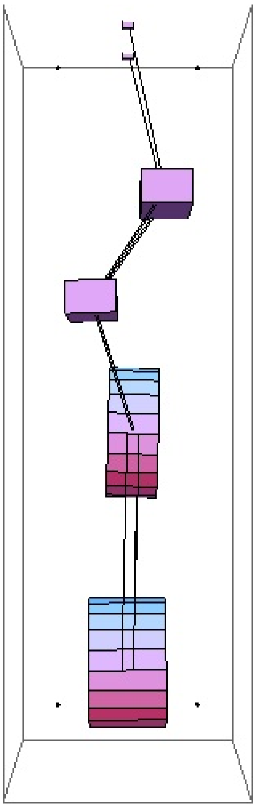


Mode #20 - modeL4

Back to Top

```
3.39588 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0.684175	-0.00216248	0	-0.000112032	-0.0202773	0.0118415
Mass U	-0.642979	0.00233217	0	0	-0.272423	-0.0298156
Mass 2	0.0977826	-0.00052593	0	0	0.169792	0.00768476
optic	-0.00363207	0	0	0	-0.0655108	0.00800627

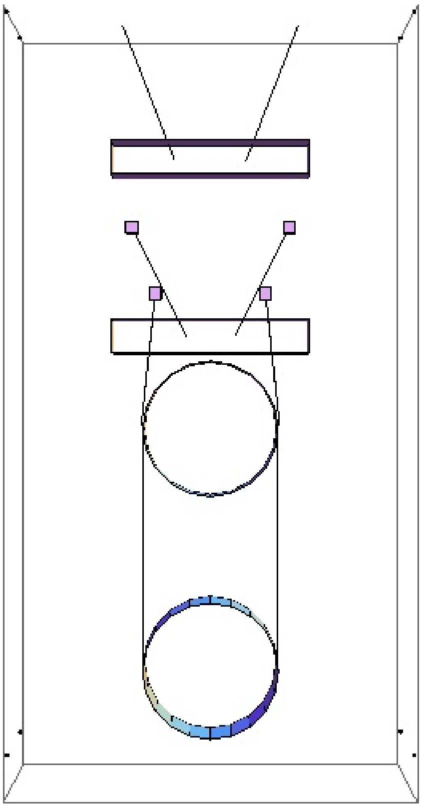


Mode #21 - modeV3

Back to Top

```
3.53664 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.564061	0	0	0
Mass U	0	0	-0.810389	0	0	0
Mass 2	0	0	0.106966	0	0	0
optic	0	0	0.116892	0	0	0

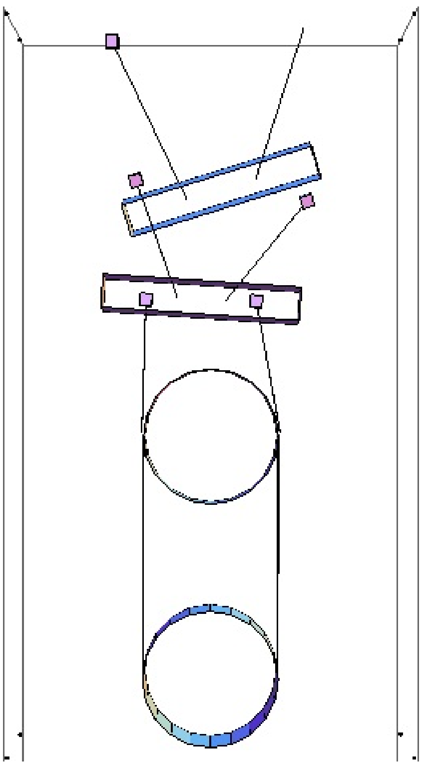


Mode #22 - modeT4

Back to Top

```
5.04259 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	-0.0750957	0	0.00409212	-0.508625	-0.825889
Mass U	-0.000105376	0.0642703	0	-0.000470534	0.117841	0.187889
Mass 2	0	-0.00393407	0	0	-0.00225671	-0.0098105
optic	0	0	0	0	0.000318508	-0.0107643

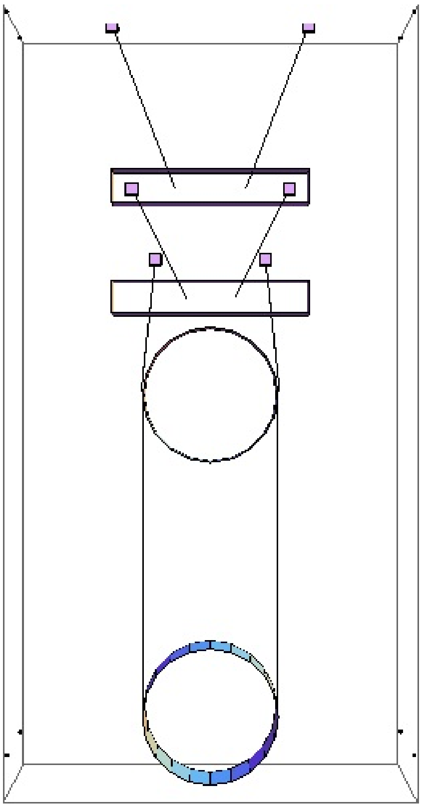


Mode #23 - modeV4

Back to Top

```
17.1245 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0.000179692	0	0	0
Mass U	0	0	-0.0135359	0	0	0
Mass 2	0	0	0.703774	0	0	0
optic	0	0	-0.710295	0	0	0

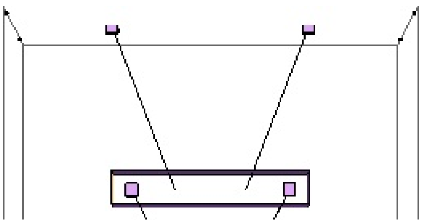


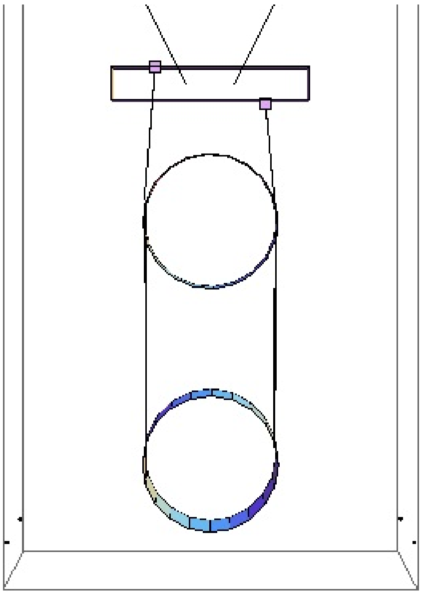
Mode #24 - modeR4

Back to Top

```
23.2379 Hz
```

	x	y	z	yaw	pitch	roll
Mass N	0	0	0	0	0	0
Mass U	0	0.000163143	0	0	-0.00121891	-0.00648287
Mass 2	0	0	0	0	0	0.663912
optic	0	0	0	0	0	-0.747781





Parameters

Back to Top

```
g -> 9.81
nx -> 0.13                                                          T040214-01 (display only)
ny -> 0.5                                                           T040214-01 (display only)
nz -> 0.084                                                         T040214-01 (display only)
denn -> 0                                                           (unused)
mn -> 21.924                                                        T1000286-v5
Inx -> 0.460035                                                     T1000286-v5
Iny -> 0.0727234                                                    T1000286-v5
Inz -> 0.472177                                                     T1000286-v5
ux -> 0.13                                                          T040214-01 (display only)
uy -> 0.5                                                           T040214-01 (display only)
uz -> 0.084                                                         T040214-01 (display only)
den1 -> 0                                                           (unused)
m1 -> 22.3383                                                       T1000286-v5
I1x -> 0.508756                                                     T1000286-v5
I1y -> 0.0710543                                                    T1000286-v5
I1z -> 0.518081                                                     T1000286-v5
ix -> 0.13                                                          T1000405-v1
ir -> 0.17                                                          T1000405-v1
den2 -> 0                                                           not used
m2 -> 40.391                                                        BNS -  measured 1/20/09,
+0.761 adjusted April 2009 for PUM magnet installation
I2x -> 0.680066                                                     Model Fit:
gradient_descent_fit_y_pitch_roll_Test_locked; BNS 5/15/09
I2y -> 0.439441                                                     Model Fit:
gradient_descent_fit_y_pitch_roll_Test_locked; BNS 5/15/09
I2z -> 0.41557                                                      Model Fit:
gradient_descent_fit_yaw.m; BNS 7/29/2008
```

```
tx -> 0.2                                                                     C1000458

tr -> 0.17                                                                    C1000458

den3 -> 0                                                                     unused

m3 -> 39.6                                                                    E1000186_v18_QUAD-
2_Process_Traveler

I3x -> 0.597753                                                               Model Fit:
gradient_descent_fit_y_pitch_roll; BNS 5/9/09

I3y -> 0.418423                                                               Model Fit:
gradient_descent_fit_y_pitch_roll; BNS 5/9/09

I3z -> 0.40046                                                                Model Fit:
gradient_descent_fit_yaw.m; BNS 7/29/2008

ln -> 0.445                                                                   RAL spreadsheet. Updated by
BNS 8 May 2009

l1 -> 0.311                                                                   RAL spreadsheet. Updated by
BNS 8 May 2009

l2 -> 0.339                                                                   RAL spreadsheet. Updated by
BNS 8 May 2009

l3 -> 0.604028                                                                calculated for updated wire
hang pend.n5, pend.l3 = sqrt(0.604 + (pend.n5-pend.n4)^2), BNS 5/9/09

nwn -> 2

nw1 -> 4

nw2 -> 4

nw3 -> 4

rn -> 0.00055                                                                 T1000428-v7

r1 -> 0.000355                                                                T1000428-v7

r2 -> 0.0003175                                                               T1000428-v7

r3 -> 0.0002285                                                               T1000428-v7

             11
Yn -> 2.12 10                                                                 "As Designed Parameter Set
2"; BNS 6/18/08

             11
Y1 -> 2.12 10                                                                 "As Designed Parameter Set
2"; BNS 6/18/08

             11
Y2 -> 2.12 10                                                                 "As Designed Parameter Set
2"; BNS 6/18/08

             11
Y3 -> 2.12 10                                                                 "As Designed Parameter Set
2"; BNS 6/18/08

dm -> -0.00350687                                                             effective value 0.001; BNS
6/18/08

dn -> 0.00328017                                                              effective value 0.001 after
blade correction; BNS 6/18/08

d0 -> -0.00173908                                                             effective value 0.001

d1 -> 0.00299279                                                              effective value 0.001 after
blade correction; BNS 6/18/08

d2 -> 0.00708835                                                              effective value 0.001; BNS
6/18/08

d3 -> 0.001                                                                   0.001 physical; T1000518

d4 -> 0.001                                                                   0.001 physical; T1000518

sn -> 0                                                                       T040214-01 (unused)
```

```
su -> 0.003                                                         T040214-01

si -> 0.003                                                         T040214-01

sl -> 0.015                                                         T040214-01

nn0 -> 0.25                                                         T040214-01, also "As
Designed Parameter Set 2"

nn1 -> 0.09                                                         T040214-01, also "As
Designed Parameter Set 2"

n0 -> 0.2                                                           T040214-01, also "As
Designed Parameter Set 2"

n1 -> 0.06                                                          T040214-01, also "As
Designed Parameter Set 2"

n2 -> 0.14                                                          T040214-01, also "As
Designed Parameter Set 2"

n3 -> 0.1762                                                        CT, email to NR, 9/22/04,
also "As Designed Parameter Set 2"

n4 -> 0.1712                                                        CT, email to NR, 9/22/04,
also "As Designed Parameter Set 2"

n5 -> 0.1712                                                        CT, email to NR, 9/22/04,
also "As Designed Parameter Set 2"

tln -> 0.411734

tl1 -> 0.279248

tl2 -> 0.347143

tl3 -> 0.606028

ltotal -> 1.64415

bd -> 0.

unstretched -> False

vertblades -> True

matlabcompat -> False

uln -> 0.443557

ul1 -> 0.309959

ul2 -> 0.338004

ul3 -> 0.602341

sln -> 0.445

sl1 -> 0.311

sl2 -> 0.339

sl3 -> 0.604028

sin -> -0.359551

si1 -> -0.450161

si2 -> 0.106785

si3 -> 0.

cn -> 0.933126

c1 -> 0.892948

c2 -> 0.994282

c3 -> 1.

pitchbul -> 0
```

```
pitchbur -> 0

pitchbil -> 0

pitchbir -> 0

pitchbll -> 0

pitchblr -> 0

rollbul -> 0

rollbur -> 0

rollbil -> 0

rollbir -> 0

rollbll -> 0

rollblr -> 0

                -7
An -> 9.50332 10

                -7
A1 -> 3.95919 10

                -7
A2 -> 3.16692 10

               -7
A3 -> 1.6403 10

kwn -> 452742.

kw1 -> 269887.

kw2 -> 198049.

kw3 -> 57570.7

flexn -> 0.00450687                                      calculated as usual

flex1 -> 0.00273908                                      calculated as usual

flex2 -> 0.00291165                                      calculated as usual

flex3 -> 0.00216189                                      calculated as usual

kbuz -> 1411.464026291042094835154548022                 Solved for given mass and
freqs; BNS 6/19/08

kbiz -> 1650.524059045379672879214321779                 Solved for given mass and
freqs; BNS 6/19/08

kblz -> 2423.519015280238231051986060810                 Solved for given mass and
freqs; BNS 6/19/08

kbux -> 100000.                                          as for middle

kbix -> 100000.                                          Justin 11/29/05

kblx -> 80000.                                           Ian 12/09/05

bdu -> 0.431794

bdi -> 0.3041

bdl -> 0.161895

mn3 -> 124.253                                           calculated as usual

m13 -> 102.329                                           calculated as usual

m23 -> 79.991                                            calculated as usual

Inxy -> -0.0375599                                       T1000286-v5

Inyz -> 0.0000465463                                     T1000286-v5
```

```
Inzx -> 0.00171841                                                     T1000286-v5

COM0x -> 0

COM0y -> 0

COM0z -> 0

FRP0x -> 0

FRP0y -> 0

FRP0z -> 0

I1xy -> -0.0132117                                                     T1000286-v5

I1yz -> 0                                                              T1000286-v5

I1zx -> 0                                                              T1000286-v5

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

I2xy -> 0

I2yz -> 0

I2zx -> 0

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0

I3yz -> 0

I3zx -> 0

COM3x -> 0

COM3y -> 0

COM3z -> 0

FRP3x -> 0

FRP3y -> 0

FRP3z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03

                 -14
Mn1 -> 7.18688 10

                 -14
Mn2 -> 7.18688 10

                 -14
M11 -> 1.24739 10
```

M11 -> 1.24739 10	
M12 -> 1.24739 10^{-14}	
M21 -> 7.98114 10^{-15}	
M22 -> 7.98114 10^{-15}	
M31 -> 2.14109 10^{-15}	
M32 -> 2.14109 10^{-15}	
temperature -> 290.	
boltzmann -> 1.38066 10^{-23}	
alphasilica -> 5.1 10^{-7}	IFOModel v4.1
betasilica -> 0.000152	IFOModel v4.1
rhosilica -> 2200.	IFOModel v4.1
Csilica -> 772.	IFOModel v4.1
Ksilica -> 1.38	IFOModel v4.1
Ysilica -> 7.27 10^{10}	IFOModel v4.1
phisilica -> 4.1 10^{-10}	IFOModel v4.1
phissilica -> 3. 10^{-11}	surface
rhosteel -> 7800.	IFOModel v4.1
Csteel -> 460.	IFOModel v4.1
Ksteel -> 49.	IFOModel v4.1
Ysteel -> 2.12 10^{11}	measured, MB, 11/18/05
alphasteel -> 0.000012	IFOModel v4.1
betasteel -> -0.00025	IFOModel v4.1
phisteel -> 0.0001	IFOModel v4.1
rhomarag -> 7800.	IFOModel v4.1
Cmarag -> 460.	IFOModel v4.1
Kmarag -> 20.	IFOModel v4.1
Ymarag -> 1.87 10^{11}	IFOModel v4.1
alphamarag -> 0.000011	IFOModel v4.1
betamarag -> -0.00025	Geppo's value - Bench v4.1 is wrong
phimarag -> 0.0001	IFOModel v4.1
tmU -> 0.0043	IFOModel v4.1
tmI -> 0.0046	IFOModel v4.1
tmL -> 0.0042	IFOModel v4.1
magicnumber -> 0.0737472	Zener, 1938, Phys. Rev. 53:90-99

```
deltabladeU -> 0.00182883

deltabladeI -> 0.00182883

deltabladeL -> 0.00182883

deltawireU -> 0.00278823

deltawireI -> 0.0027844

deltawireL -> 0.00277698

                        -6
deltafibre -> 6.57741 10

taubladeU -> 0.336093

taubladeI -> 0.384626

taubladeL -> 0.320643

tauwireU -> 0.00653413

tauwireI -> 0.00272219

tauwireL -> 0.00217746

taufibre -> 0.0189557

                                          0.003862 #1
damping[imag, bladeUtype] -> (0.0001 + --------------- & )
                                                    2
                                        1 + 4.45943 #1

                                         0.00441968 #1
damping[imag, bladeItype] -> (0.0001 + --------------- & )
                                                    2
                                        1 + 5.84032 #1

                                         0.00368446 #1
damping[imag, bladeLtype] -> (0.0001 + --------------- & )
                                                    2
                                        1 + 4.05884 #1

damping[imag, wireUtype] -> (0.0001 & )

damping[imag, wireItype] -> (0.0001 & )

damping[imag, wireLtype] -> (0.0001 & )

                                          0.000114471 #1
damping[imag, wireUatype] -> (0.0001 + ------------------ & )
                                                       2
                                        1 + 0.00168552 #1

                                          0.0000476244 #1
damping[imag, wireIatype] -> (0.0001 + ------------------- & )
                                                        2
                                        1 + 0.000292548 #1

                                         0.0000379928 #1
damping[imag, wireLatype] -> (0.0001 + ------------------ & )
                                                       2
                                        1 + 0.00018718 #1

                                        -8
damping[imag, fibretype] -> (5.42393 10   & )

                                                         -7
                                         -7    7.83382 10   #1
damping[imag, fibreatype] -> (1.08069 10   + ----------------- & )
                                                             2
                                              1 + 0.0141853 #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0
```

```
pitch00 -> 0

roll00 -> 0

kconx0 -> 0

kcony0 -> 0

kconz0 -> 0

kconyaw0 -> 0

kconpitch0 -> 0

kconroll0 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

ribbons -> False

dumbbell -> True

Inxz -> 0.00171841

Inzy -> 0.0000465463

Inyx -> -0.0375599

I1xz -> 0

I1zy -> 0

I1yx -> -0.0132117

I2xz -> 0

I2zy -> 0

I2yx -> 0

                 -7
A3n -> 1.6403 10

bend1 -> 0.00501925                                                       calculated as usual

bend2 -> 0.00490445                                                       calculated as usual

thetan -> -21.0726                                                        calculated as usual
```

[illegible]	calculated as usual
theta1 -> -26.754	calculated as usual
theta2 -> 6.13	calculated as usual
theta3 -> 0.	calculated as usual
sigmasilica -> 0.17	
Gsilica -> $3.10684\ 10^{10}$	shear modulus
dssilica -> 0.015	IFOModel v4.1

Back to Top

aLIGO SUS Operation Manual - Info on BSFM Suspensions

Back to Operation Manual main page

References

- T040027: Conceptual Design of Beamsplitter Suspension for Advanced LIGO
- D1000392: aLIGO BS/FM MAIN ASSEMBLY
- T1100602: BS/FM Triple Suspension Control Ranges
- T1100479: BSFM Electronics Design
- E1100108: Beamsplitter-Folding Mirror Controls Arrangement Poster
- D1100022: aLIGO SUS ITM and BS Wiring Diagrams (for H1/L1)
- D1001725: aLIGO SUS ITM, BS, and FM System Wiring Diagrams (for H2)

Models

The BSFM suspension has been extensively modelled. Key results are at Suspensions/OpsManual/BSFM/Models .

Screens

Annotated screenshots of the BSFM are at /Screens.

aLIGO: Suspensions/OpsManual/BSFM (last edited 2014-02-06 12:15:06 by MarkBarton)

aLIGO SUS Operations Manual - Overview of BSFM MEDM screens

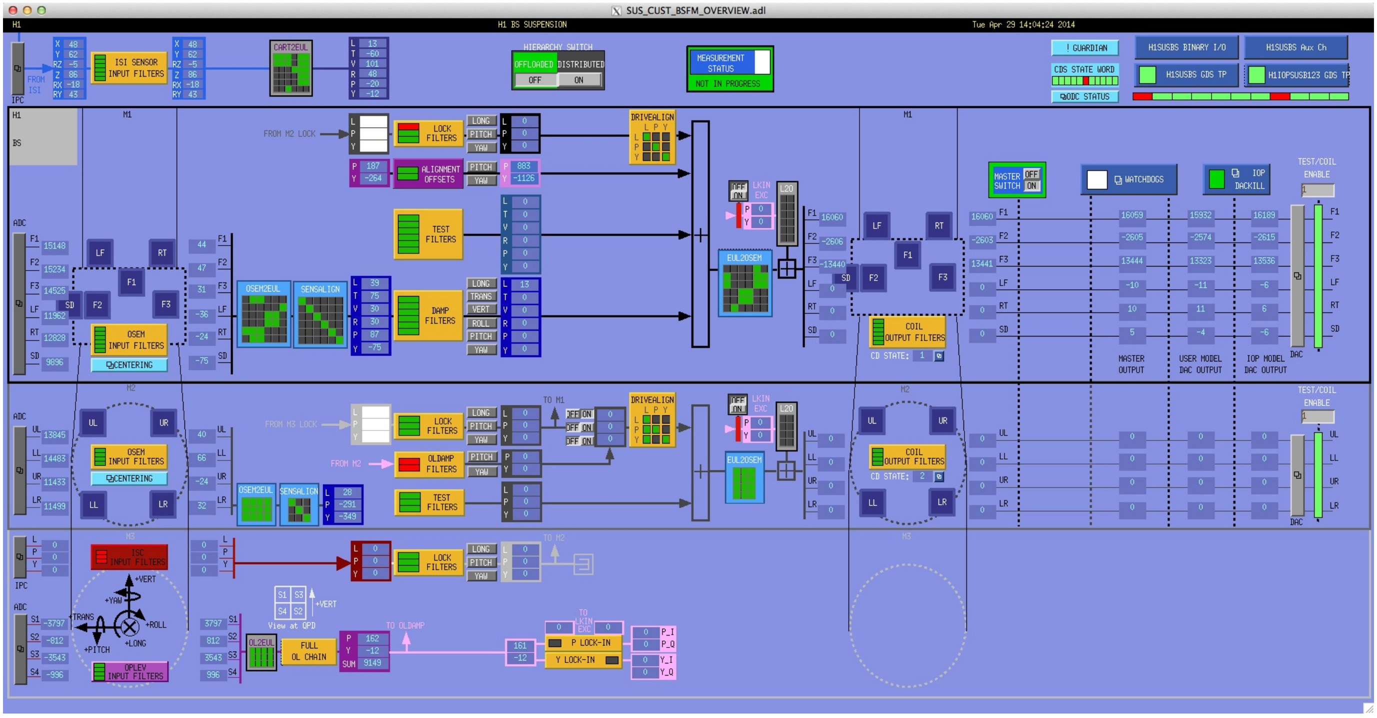
Back to Operation Manual main page
Back to BSFM main page

This page makes extensive use of text fragments in-lined from Suspensions/OpsManual/Boilerplate. Use the scripts there to update.

Except where noted, the BSFM screens described below live at `/opt/rtcds/userapps/release/sus/common/medm/bsfm/`. They are all generic screens which rely for their execution on site-, interferometer- and suspension-specific arguments passed in when they are called. They can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_BSFM_OVERVIEW.adl

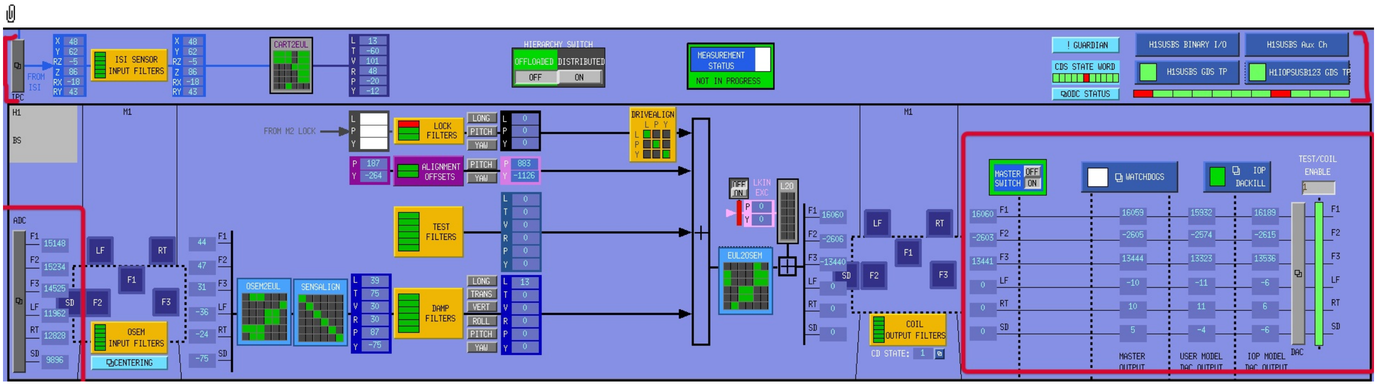


This is the overview screen. The functionality is divided up according to the three sensor-actuator groups, plus an odds-and-ends area at the top:

- Other - ISI feedforward stuff, links to CDS utility screens.
- M1 - 6 BOSEMs on the structure engaging the upper top mass
- M2 - 4 BOSEMs on the structure engaging the intermediate mass
- M3 - no OSEMs on the optic, but various optical lever stuff

M1 is used for local damping (relative to the structure) and the control loops are already functional. DC pitch and yaw offsets are also injected at M1. M2 is intended for implementing global control (relative to other optics) and has placeholder inputs for actuation requests from ISC. The M3 level just processes optical lever signals.

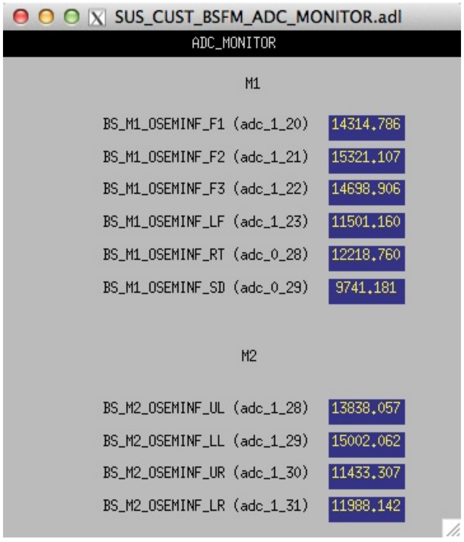
Other Screens



- MASTER SWITCH - a manual on/off switch for all actuation (common to all sensor-actuator groups).

- ADC - the ADC overview
- DAC - the DAC channel overview
- H1SUSBS Aux Ch (or the like) - the coil driver monitor channels
- WD ("WATCHDOGS") - a block implementing the watchdogs on the various sensor actuator groups.
- IOP DACKILL - a screen for watchdogs on all IOP processes serving SUS models.
- H1SUSBS GDS TP (or the like) - controls for user model process
- H1IOPSUSB123 GDS TP (or the like) - controls for IOP model process
- H1SUSBS BINARY I/O (or the like) - the Binary Input/Output controls
- IPC - the IPC channel monitor
- ODC - ODC status
- !GUARDIAN - Guardian
- SUS_CUST_BSFM_M1_CART2EUL - monitor from signals from the ISI

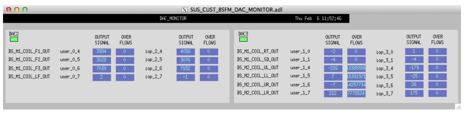
Screen SUS_CUST_BSFM_ADC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

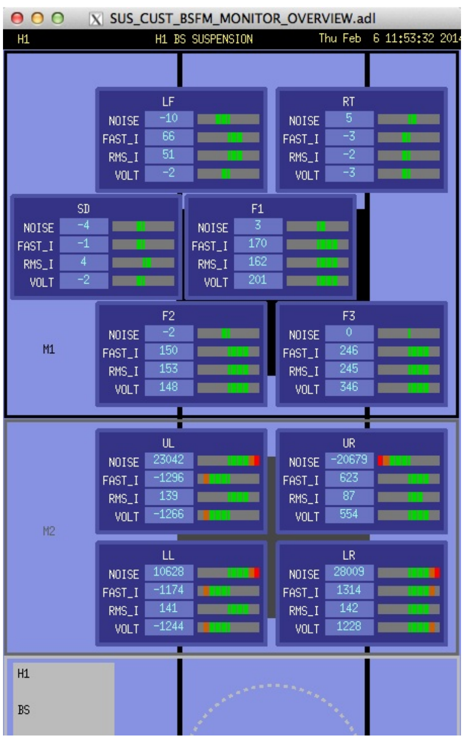
Screen SUS_CUST_BSFM_DAC_MONITOR.adl

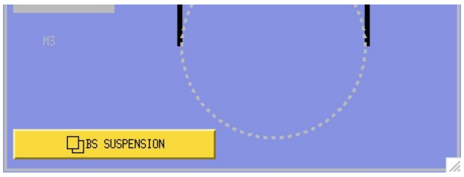


Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen SUS_CUST_BSFM_MONITOR_OVERVIEW.adl



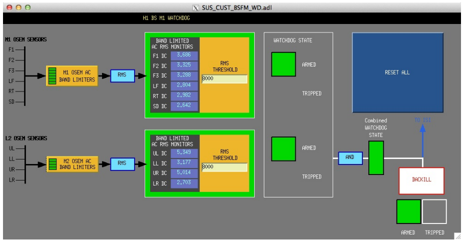


Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

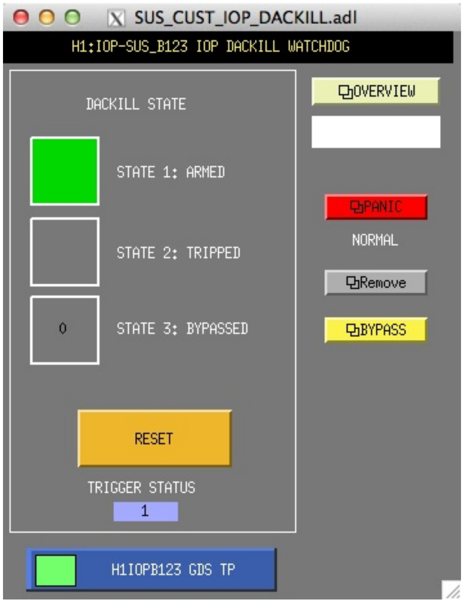
Screen SUS_CUST_BSFM_WD.adl



Suspensions/OpsManual/Boilerplate/WD:

This screen allows for setting the watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

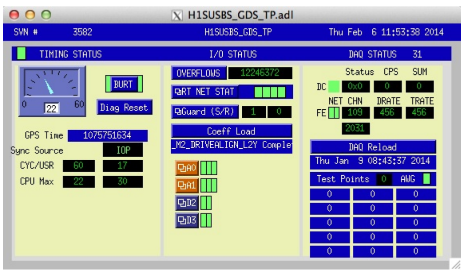
Screen SUS_CUST_IOP_DACKILL.adl



Suspensions/OpsManual/Boilerplate/IOP_DACKILL:

The watchdog controls for the IOP process. If this WD is tripped it disables IO to/from all user models on the front end. RESET resets from the tripped state if there are no outstanding triggers. BYPASS is a manual override that allows for operation of the suspension if there is a trigger condition that is known by the operator to be harmless.

Screen H1SUSBS_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

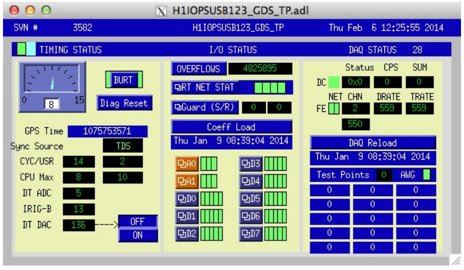
This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1IOPSUSB123_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen SUS_CUST_BSFM_BIO.adl

Suspensions/OpsManual/Boilerplate/BSFM_BIO:

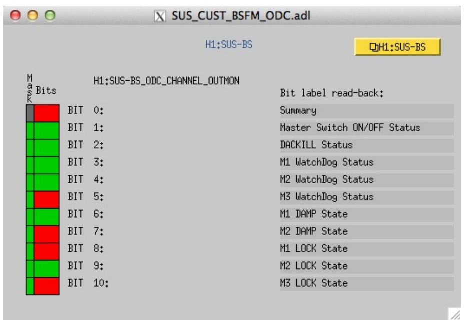
The Binary Input-Output control screen. State Requests for the various sensor-actuator groups can be input. Positive state values lock out manual control of the filters on the COILOUTF screen and set the digital dewhitening filters to match the analog whitening filters selected via the binary outputs. Negative state values allow the dewhitening filters to be manually overridden.

Screen GUARD.adl

Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

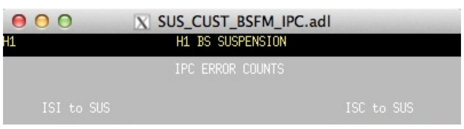
Screen SUS_CUST_BSFM_ODC.adl



Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

Screen SUS_CUST_BSFM_IPC.adl



Suspensions/OpsManual/Boilerplate/IPC:

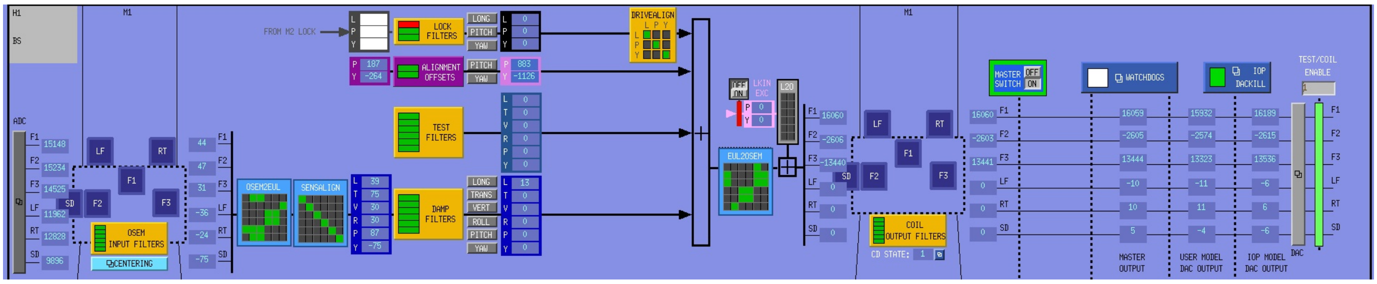
Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

Screen SUS_CUST_BSFM_M1_CART2EUL.adl

Suspensions/OpsManual/Boilerplate/M1_CART2EUL:

A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

Sensor Actuator Group M1 - Main Chain Top Mass



This section is laid out to reflect the flow of the main chain local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to ±15000 counts.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- DAMP FILTERS - a filter bank applying the damping filters.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ????.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

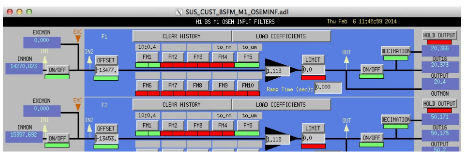
There are the following auxiliary inputs:

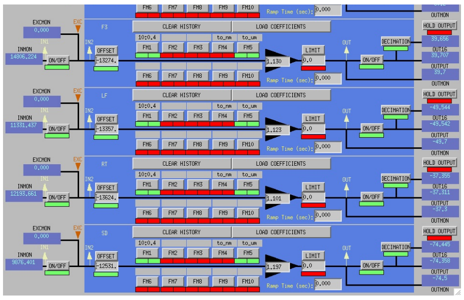
- TEST - a filter bank (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.
- LOCK - filters for global control signals.

There are the following additional controls:

- MEASUREMENT STATUS - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- HIERARCHY SWITCH - enables various ISC inputs and outputs.

Screen SUS_CUST_BSFM_M1_OSEMINF.adl





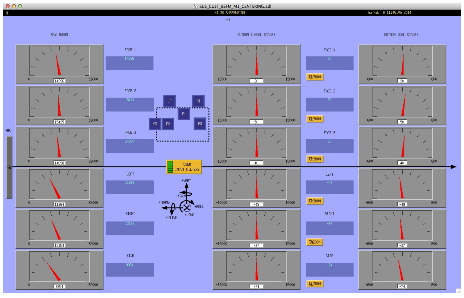
Suspensions/OpsManual/Boilerplate/M1_OSEMINF:

This block has 6 filter groups corresponding to the 6 M1 BOSEMs, T1/T2/T3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_BSFM_M1_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M1_CENTERING:

This screen gives various views of the M1 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_BSFM_M1_OSEM2EUL.adl

Suspensions/OpsManual/Boilerplate/M1_OSEM2EUL:

This screen allows entry of the matrix which converts from the M1 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

Screen SUS_CUST_BSFM_M1_SENSALIGN.adl

Suspensions/OpsManual/Boilerplate/M1_SENSALIGN:

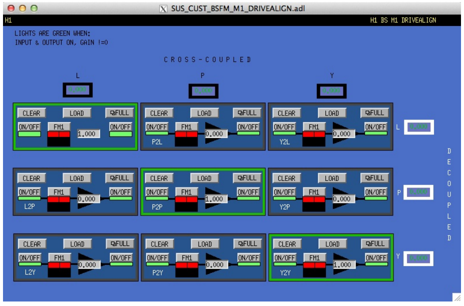
This screen is reserved for tweaking the M1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_BSFM_M1_DAMP.adl

Suspensions/OpsManual/Boilerplate/M1_DAMP:

These filters implement the local damping. See Suspensions/OpsManual/Damping.

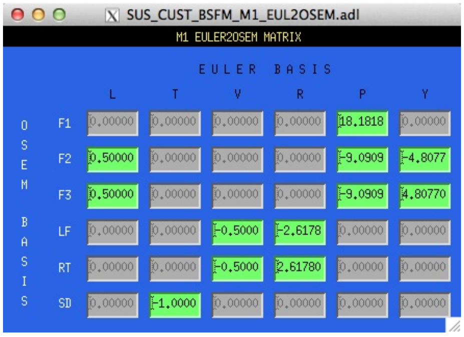
Screen SUS_CUST_BSFM_M1_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_DRIVEALIGN:

This screen is reserved for tweaking the M1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

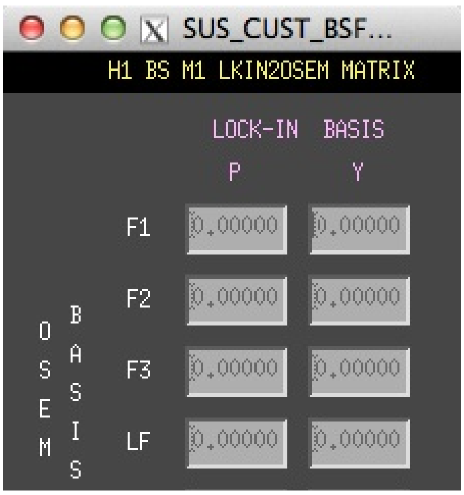
Screen SUS_CUST_BSFM_M1_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_BSFM_M1_LKIN2OSEM.adl



Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_BSFM_M1_COILOUTF.adl

Suspensions/OpsManual/Boilerplate/M1_COILOUTF:

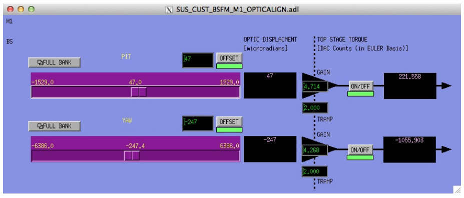
This screen applies compensation for the hardware filters in the M1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

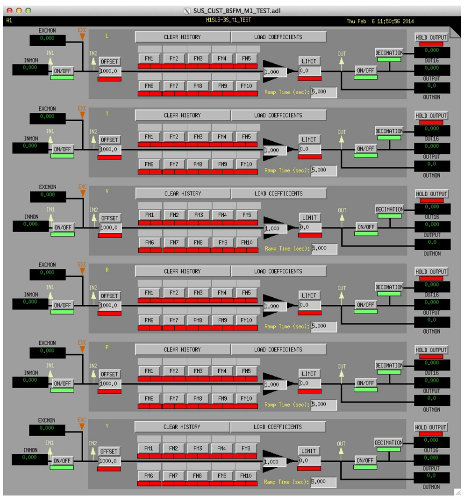
Screen SUS_CUST_BSFM_M1_OPTICALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets. See Suspensions/OpsManual/Alignments for more info.

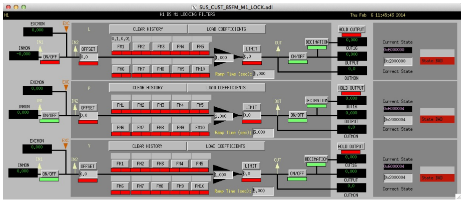
Screen SUS_CUST_BSFM_M1_TEST.adl



Suspensions/OpsManual/Boilerplate/M1_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

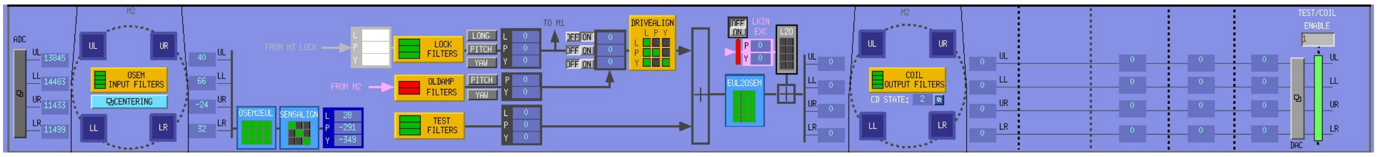
Screen SUS_CUST_BSFM_M1_LOCK.adl



Suspensions/OpsManual/Boilerplate/M1_LOCK:

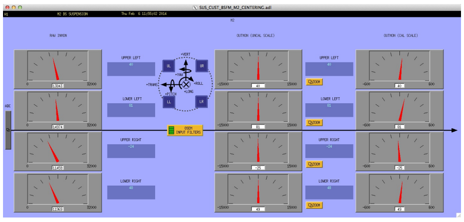
Filters for the locking signals.

Sensor Actuator Group M2 - Between Upper Intermediate Masses



- CENTERING - the raw inputs as panel meters for ease of adjustment
- OSEM INPUT FILTERS - OSEM sensor dewhitening and calibration filters.
- OSEM2EUL - transformation matrix for OSEM sensor to Euler basis
- SENSALIGN - for tweaking sensor diagonalization
- LOCK FILTERS - lock input filters
- OLDAMP FILTERS - filters for optical lever damping.
- TEST FILTERS - for test inputs
- DRIVEALIGN - for tweaking drive diagonalization
- EUL2OSEM - transformation from Euler to OSEM basis
- COIL OUTPUT FILTERS - coil dewhitening and magnet sign correction filters
- L2O - ????.

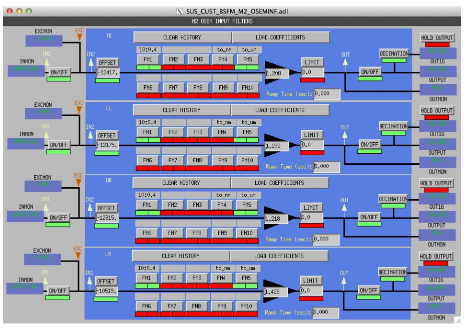
Screen SUS_CUST_BSFM_M2_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M2_CENTERING:

This screen gives various views of the OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_BSFM_M2_OSEMINF.adl



Suspensions/OpsManual/Boilerplate/M2_OSEMINF:

This block has 4 filter banks corresponding to the 4 M2 OSEMs, UL/LL/UR/LR. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_BSFM_M2_OSEM2EUL.adl

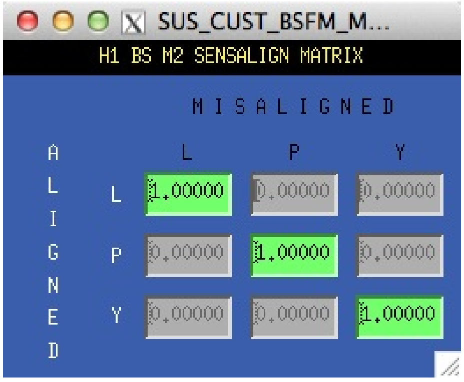
Suspensions/OpsManual/Boilerplate/M2_OSEM2EUL:

This screen allows entry of the matrix which converts from the OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOF L are dimensionless.

The entries for the angular DOFs P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

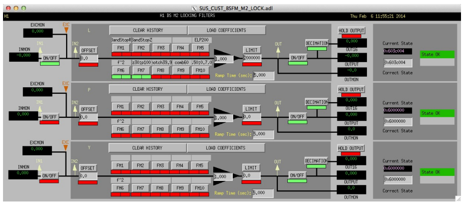
Screen SUS_CUST_BSFM_M2_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M2_SENSALIGN:

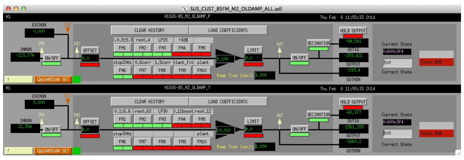
This screen is reserved for tweaking the M2 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_BSFM_M2_LOCK.adl


Suspensions/OpsManual/Boilerplate/M2_LOCK:

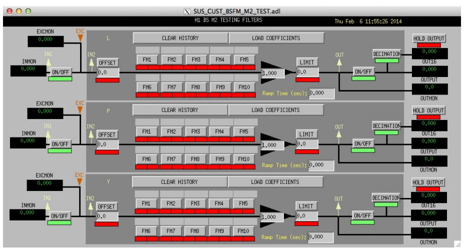
Filters for the locking signals.

Screen SUS_CUST_BSFM_M2_OLDAMP_ALL.adl


Suspensions/OpsManual/Boilerplate/M2_OLDAMP:

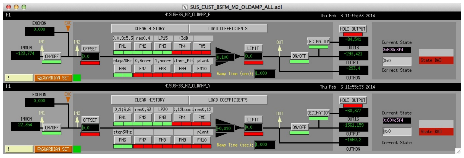
Filters for the optical lever locking signals.

Screen SUS_CUST_BSFM_M2_TEST.adl


Suspensions/OpsManual/Boilerplate/M2_TEST:

This is a deliberately empty set of 3 filter banks corresponding to the M2 L, P and Y DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

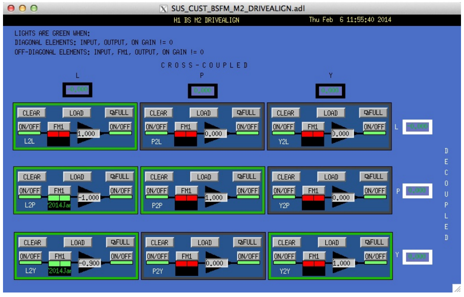
Screen SUS_CUST_BSFM_M2_OLDAMP_ALL.adl



Suspensions/OpsManual/Boilerplate/M2_OLDAMP_ALL:

Filters for optical lever damping.

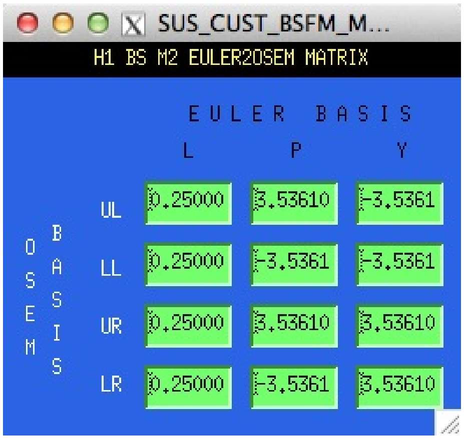
Screen SUS_CUST_BSFM_M2_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M2_DRIVEALIGN:

This screen is reserved for tweaking the actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

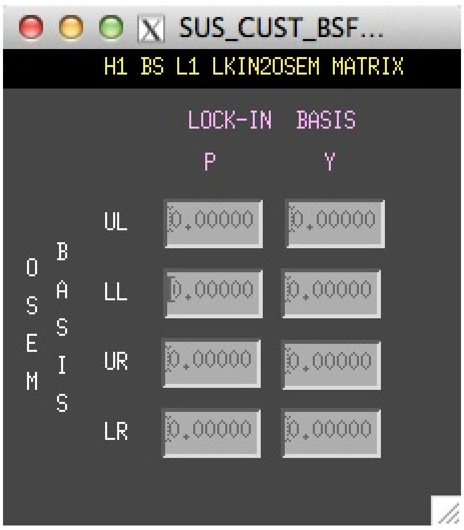
Screen SUS_CUST_BSFM_M2_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M2_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

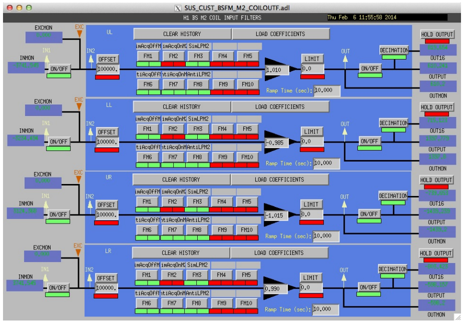
Screen SUS_CUST_BSFM_M2_LKIN2OSEM.adl



Suspensions/OpsManual/Boilerplate/M2_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_BSFM_M2_COILOUTF.adl



Suspensions/OpsManual/Boilerplate/M2_COILOUTF:

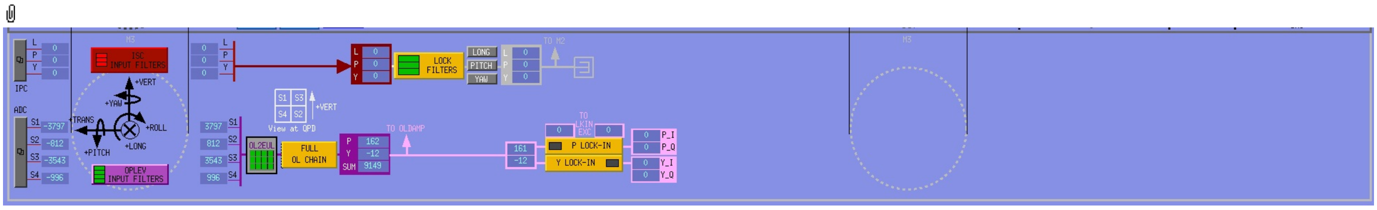
This screen applies compensation for the hardware filters in the actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

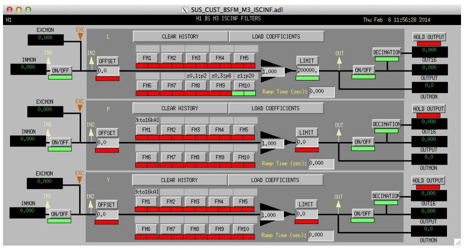
The sign convention at the output is positive for a positive current (rather than force).

Sensor Actuator Group M3 - Optic



- LOCK FILTERS - filters for lock signals
- OPLEV INPUT FILTERS - filters for individual OL QPD segment signals
- OL2EUL - transformation matrix for OL QPD signals to P/Y/SUM
- FULL OL CHAIN - overview of OL signal processing
- P/Y LOCK-IN - ????

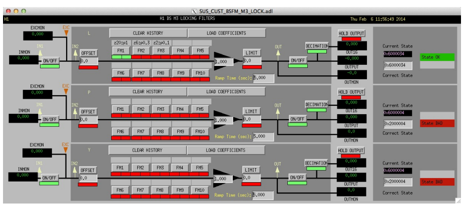
Screen SUS_CUST_BSFM_M3_ISCINF.adl



Suspensions/OpsManual/Boilerplate/M3_ISCINF:

Filters for the control signals from ISC.

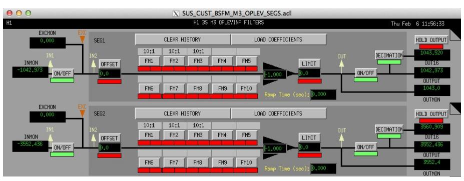
Screen SUS_CUST_BSFM_M3_LOCK.adl

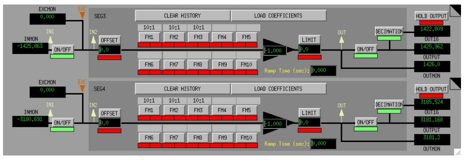


Suspensions/OpsManual/Boilerplate/M3_LOCK:

Filters for the locking signals.

Screen SUS_CUST_BSFM_M3_OPLEV_SEGS.adl

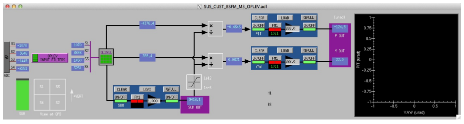




Suspensions/OpsManual/Boilerplate/M3_OPLEV_SEGS:

Filters for the optical lever QPD segment signals.

Screen SUS_CUST_BSFM_M3_OPLEV.adl



Suspensions/OpsManual/Boilerplate/M3_OPLEV:

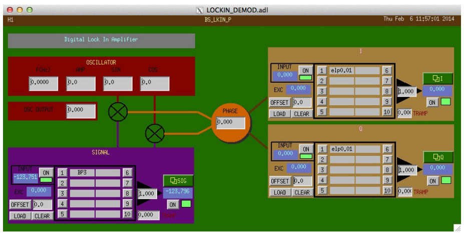
Overview of the Optical Lever signal processing.

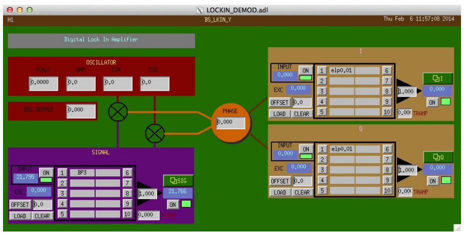
Screen SUS_CUST_BSFM_M3_OPLEV_MTRX.adl

Suspensions/OpsManual/Boilerplate/M3_OPLEV_MTRX:

Transformation matrix from the optical lever QPD segment basis into the P/Y/SUM basis.

Screen LOCKIN_DEMOD.adl (two versions: P and Y)





Suspensions/OpsManual/Boilerplate/LOCKIN_DEMOD:

Controls for the lockin demodulation. Comes in P and Y versions for pitch and yaw.

aLIGO SUS Operations Manual - Models for BSFM Suspensions

Back to Operation Manual main page

The BSFM suspensions have been extensively modelled. There are Mathematica (T020205) and Matlab (T080188) programs that take a parameter set describing a single chain and calculate the dynamics. The parameter set needs to be formatted differently for Mathematica vs Matlab, but the parameter names are the same as far as was practical.

The Mathematica triple model, which covers BSFM as well as HLTS and HSTS, and parameter sets for it lives in the SUS SVN at `^/trunk/Common/MathematicaModels/TripleLite2/`. Parameter sets for Matlab live at `^/trunk/Common/MatlabTools/TripleModel_Production`. Mark Barton maintains the Mathematica, Jeff Kissel maintains the Matlab. Unfortunately two different naming conventions are in use and it's difficult to tell what corresponds to what - if in doubt, ask. References given below are to Mathematica versions as subdirectories of `^/trunk/Common/MathematicaModels/TripleLite2/`.

Key cases of the triple model for BSFM are given below, with results such as mode frequencies and mode shapes. T1200404 has transfer functions for many of the same models.

Generic BSFM

Current best generic BSFM model is .../mark.barton/20120120bsNW.

aLIGO: Suspensions/OpsManual/BSFM/Models (last edited 2014-05-27 09:11:44 by MarkBarton)

Case 20120120bsNW of Mathematica model TripleLite2

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
{"mark.barton", "20120120bsNW"}
```

```
Same as Jeff K's Matlab model bsfmopt_metal.m Rev 2005 of 1/19/12 (no wedge)
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeL1	modeT1	modeP1	modeY1	modeR1	modeL2
modeP2	modeV1	modeP3	modeY2	modeT2	modeL3
modeT3	modeY3	modeR2	modeV2	modeV3	modeR3

Mode Summary

Back to Top

```
N    f          name     type

1    0.419315   modeL1   pitch3   pitch2

2    0.423602   modeT1   y3       y2       roll3

3    0.468279   modeP1   pitch3

4    0.493035   modeY1   yaw3     yaw2

5    1.0506     modeR1   y2       y3       y1      roll1

6    1.05136    modeL2   pitch1   pitch2

7    1.07596    modeP2   pitch1   pitch2

8    1.08389    modeV1   z3       z2

9    1.3915     modeP3   pitch1

10   1.3963     modeY2   yaw1

11   1.55266    modeT2   roll3    roll2

12   1.69857    modeL3   x1       pitch2

13   2.18939    modeT3   roll1    roll3

14   2.25325    modeY3   yaw2     yaw3

15   3.20625    modeR2   roll1    roll3

16   3.76046    modeV2   z1

17   17.5222    modeV3   z2       z3

18   25.9715    modeR3   roll3    roll2
```

Violin Modes	UM	IM	Optic
f (Hz), n=1-	237.8 475.75 714.002 952.705	225.496 451.061 676.762	303.012 606.065 909.203

4	[illegible]	902.67	1212.47
Q, n=1-4	114198. 168946. 194804. 200082.	103165. 133634. 161126. 177500.	158925. 150489. 168099. 185542.

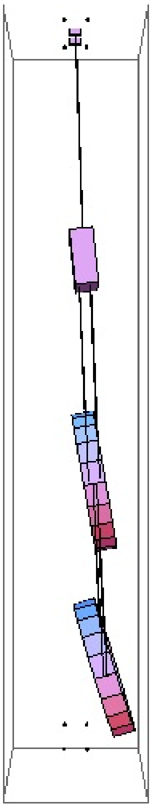
Mode Shapes

Mode #1 - modeL1

Back to Top

```
0.419315 Hz
```

```
        x          y   z   yaw   pitch      roll
Mass U  -0.057342  0   0   0     0.299037   0
Mass I  -0.1312    0   0   0     0.463081   0
optic   -0.203265  0   0   0     0.796439   0
```

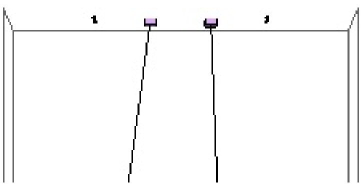


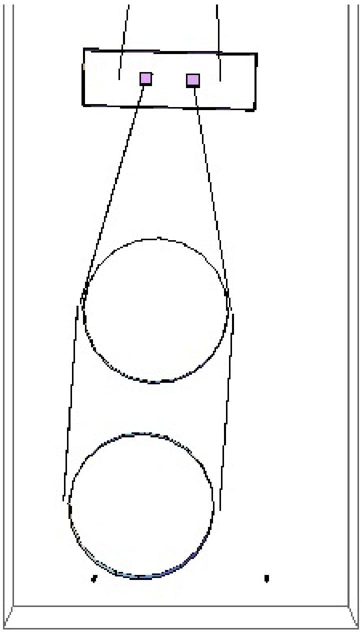
Mode #2 - modeT1

Back to Top

```
0.423602 Hz
```

```
        x   y          z   yaw   pitch   roll
Mass U  0   0.196907   0   0     0       0.176372
Mass I  0   0.438473   0   0     0       0.366259
optic   0   0.68515    0   0     0       0.36645
```



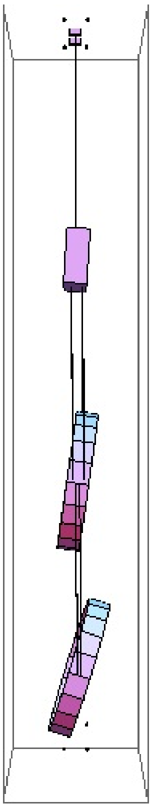


Mode #3 - modeP1

Back to Top

```
0.468279 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.0093894	0	0	0	0.207023	0
Mass I	0.0159212	0	0	0	0.421663	0
optic	0.0276447	0	0	0	0.882177	0

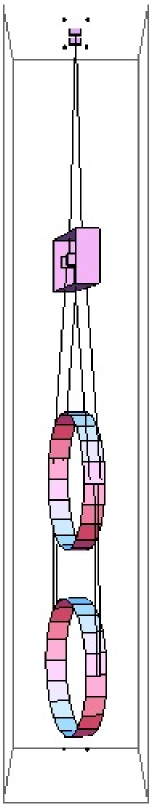


Mode #4 - modeY1

Back to Top

```
0.493035 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	0.332994	0	0
Mass I	0	0	0	0.627688	0	0
optic	0	0	0	0.703649	0	0

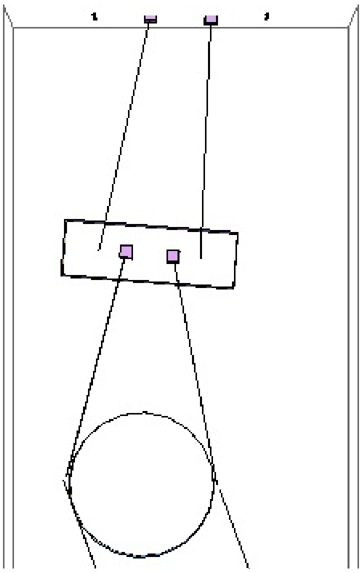


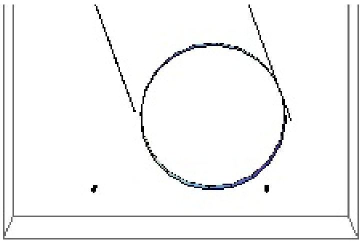
Mode #5 - modeR1

Back to Top

```
1.0506 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.423331	0	0	0	0.410844
Mass I	0	0.535269	0	0	0	0.291088
optic	0	-0.442127	0	0	0	0.292023



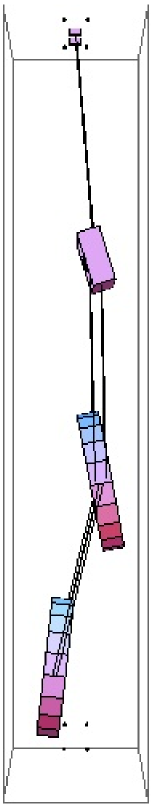


Mode #6 - modeL2

Back to Top

```
1.05136 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.136171	0	0	0	0.775799	0
Mass I	-0.171869	0	0	0	0.489702	0
optic	0.141912	0	0	0	-0.300178	0

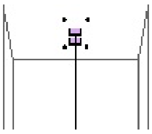


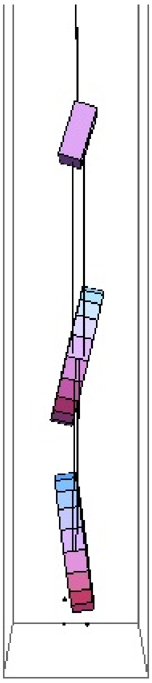
Mode #7 - modeP2

Back to Top

```
1.07596 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.0142387	0	0	0	-0.743736	0
Mass I	-0.0103628	0	0	0	-0.580644	0
optic	0.00741414	0	0	0	0.330672	0



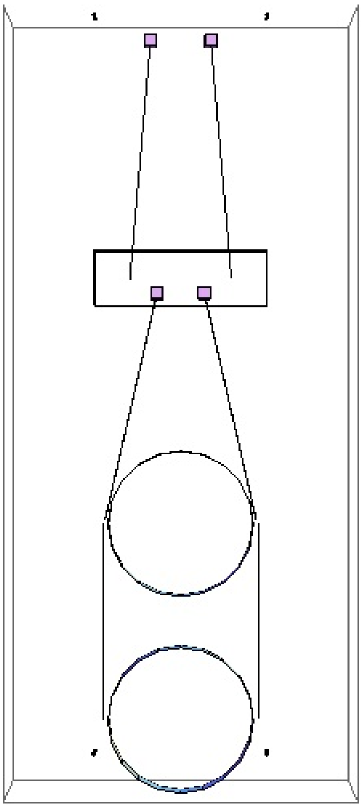


Mode #8 - modeV1

Back to Top

```
1.08389 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0.403867	0	0	0
Mass I	0	0	0.644297	0	0	0
optic	0	0	0.649441	0	0	0

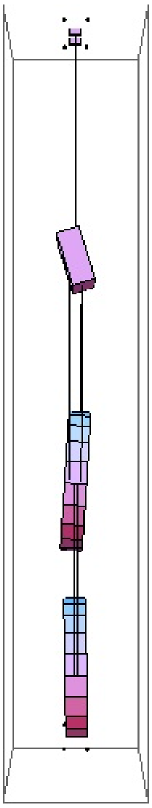


Mode #9 - modeP3

Back to Top

```
1.3915 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.000492053	0	0	0	0.976302	0
Mass I	0.00131969	0	0	0	-0.20855	0
optic	-0.000530777	0	0	0	0.0577796	0

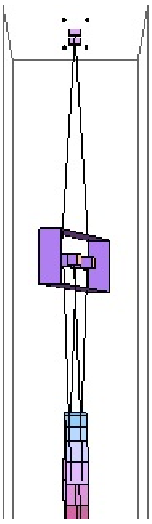


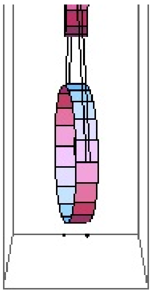
Mode #10 - modeY2

Back to Top

```
1.3963 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	0.874878	0	0
Mass I	0	0	0	-0.0644056	0	0
optic	0	0	0	-0.480043	0	0



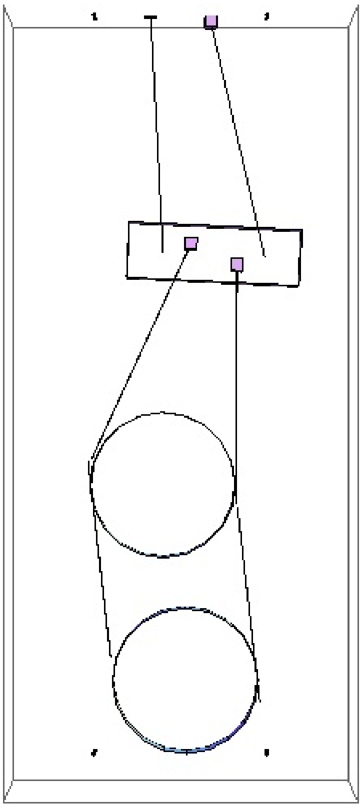


Mode #11 - modeT2

Back to Top

```
1.55266 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.195084	0	0	0	-0.111339
Mass I	0	-0.104872	0	0	0	-0.682353
optic	0	0.0275608	0	0	0	-0.687155

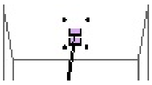


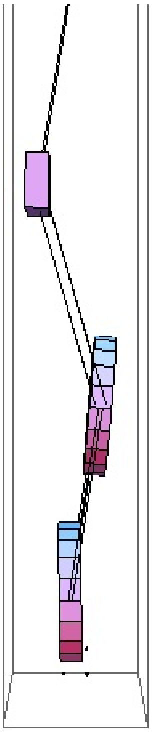
Mode #12 - modeL3

Back to Top

```
1.69857 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.6685	0	0	0	-0.147779	0
Mass I	-0.432943	0	0	0	-0.571119	0
optic	0.0904494	0	0	0	0.0973239	0



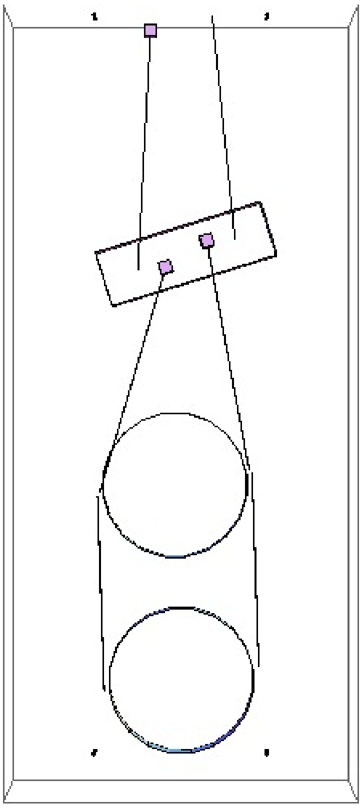


Mode #13 - modeT3

Back to Top

```
2.18939 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.0386426	0	0	0	-0.817925
Mass I	0	0.0402597	0	0	0	-0.402042
optic	0	-0.00463276	0	0	0	-0.407707

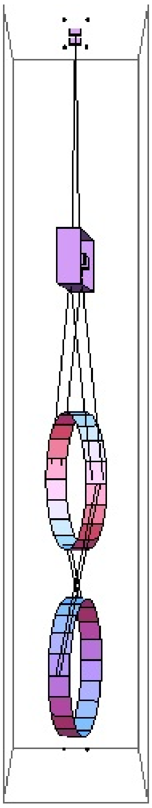


Mode #14 - modeY3

Back to Top

```
2.25325 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	-0.208787	0	0
Mass I	0	0	0	0.764786	0	0
optic	0	0	0	-0.609516	0	0

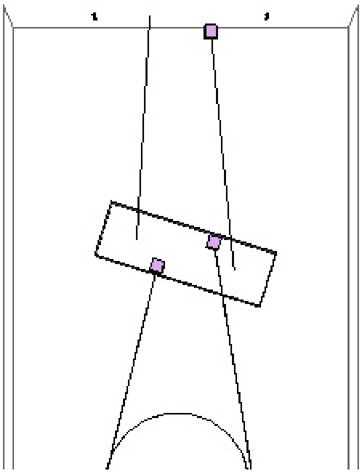


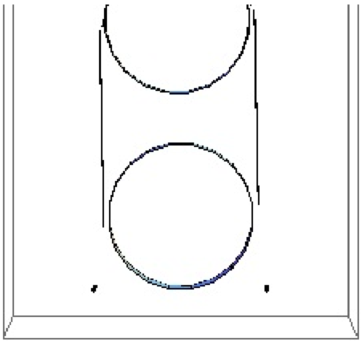
Mode #15 - modeR2

Back to Top

```
3.20625 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.0349124	0	0	0	0.824356
Mass I	0	0.0249148	0	0	0	-0.39304
optic	0	-0.00125139	0	0	0	-0.405111



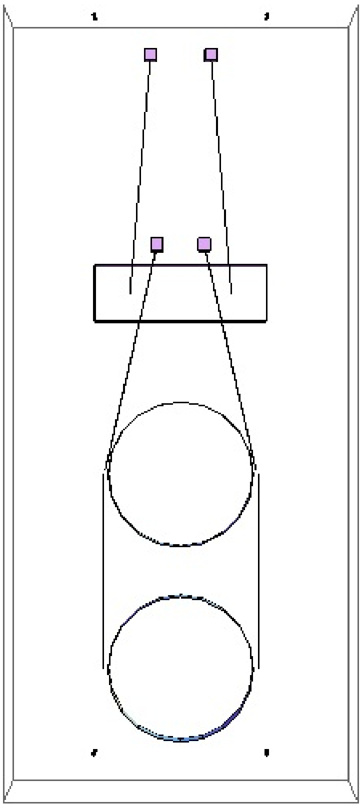


Mode #16 - modeV2

Back to Top

```
3.76046 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.928065	0	0	0
Mass I	0	0	0.249845	0	0	0
optic	0	0	0.276174	0	0	0

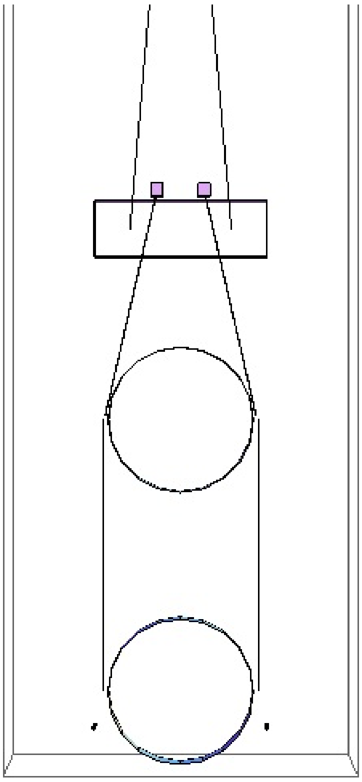


Mode #17 - modeV3

Back to Top

```
17.5222 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0.0172332	0	0	0
Mass I	0	0	-0.730446	0	0	0
optic	0	0	0.682753	0	0	0

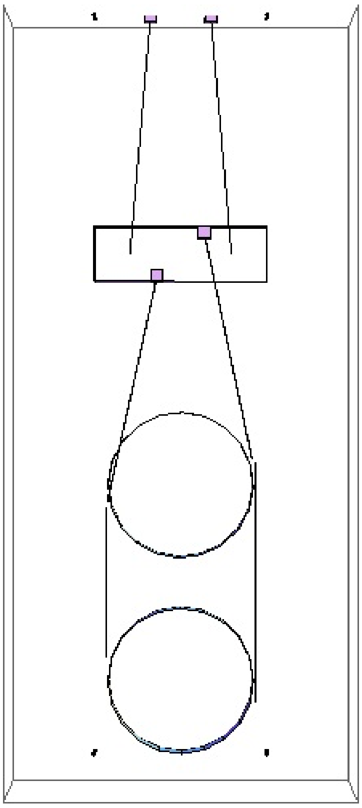


Mode #18 - modeR3

Back to Top

```
25.9715 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.000330179	0	0	0	0.00453115
Mass I	0	0.000305891	0	0	0	-0.690706
optic	0	0	0	0	0	0.723122



Parameters

Back to Top

```
g -> 9.81

ux -> 0.06                                                         display only

uy -> 0.44                                                         display only

uz -> 0.142                                                        display only

den1 -> 4000

m1 -> 12.63                                                        Measured from BSFM01,
https://ics-redux.ligo-la.caltech.edu/JIRA/browse/ASSY-D070435-001

I1x -> 0.1659                                                      T080222-v3

I1y -> 0.02473                                                     T080222-v3

I1z -> 0.1643                                                      T080222-v3

ix -> 0.05709                                                      (54 + 2*1.55 [mm]) from
D080369-v2, display only

ir -> 0.185                                                        D080369-v2, info only

den2 -> 2700
<http://en.wikipedia.org/wiki/6061_aluminium_alloy>, info only

m2 -> 13.575                                                       D080369-v2

I2x -> 0.2592                                                      D080369-v2

I2y -> 0.12976                                                     D080369-v2

I2z -> 0.13587                                                     D080369-v2

tx -> 0.05709                                                      (54 + 2*1.55 [mm]) from
D080369-v2

tr -> 0.185                                                        D080369-v2

den3 -> 2700
<http://en.wikipedia.org/wiki/6061_aluminium_alloy>, info only

m3 -> 14.21                                                        T080222-v3

I3x -> 0.243169                                                    cylinder: m3, tx, tr

I3y -> 0.125444                                                    cylinder: m3, tx, tr

I3z -> 0.125444                                                    cylinder: m3, tx, tr

l1 -> 0.612                                                        As-built, see, e.g.,
<https://alog.ligo-la.caltech.edu/aLOG/index.php?callRep=1769>

l2 -> 0.596                                                        As-built, see, e.g.,
<https://alog.ligo-la.caltech.edu/aLOG/index.php?callRep=1769>

l3 -> 0.5                                                          T080222-v3

nw1 -> 2

nw2 -> 4

nw3 -> 4

r1 -> 0.0003125                                                    T080222-v3

r2 -> 0.0002                                                       T080222-v3

r3 -> 0.000125                                                     T080222-v3

              11
Y1 -> 2.119 10

              11
Y2 -> 2.119 10
```

```
              11
Y3 -> 2.119 10

ufc1 -> 2.32                                                      Fit to H2SUSFMY and L1SUSBS
data

ufc2 -> 2.57                                                      Fit to H2SUSFMY and L1SUSBS
data

ufc3 -> 0

d0 -> -0.00181363                                                 T080222-v3, flexure corrected,
+0.000 tweak from fitting

d1 -> 0.0000976211                                                T080222-v3, flexure corrected,
+0.001 tweak from fitting

d2 -> 0.00809762                                                  T080222-v3, flexure corrected,
+0.000 tweak from fitting

d3 -> -0.0000797641                                               T080222-v3, flexure corrected,
+0.000 tweak from fitting

d4 -> -0.00107976                                                 T080222-v3, flexure corrected,
-0.001 tweak from fitting

su -> 0                                                           T080222-v3, not used

si -> 0.015                                                       T080222-v3

sl -> 0.005                                                       T080222-v3

n0 -> 0.077                                                       T080222-v3

n1 -> 0.13                                                        T080222-v3

n2 -> 0.06                                                        T080222-v3

n3 -> 0.1931                                                      D080560-v1 + T080222-v3

n4 -> 0.1995                                                      D080560-v1 + T080222-v3

n5 -> 0.1995                                                      D080560-v1 + T080222-v3

tl1 -> 0.607887

tl2 -> 0.589143

tl3 -> 0.49884

ltotal -> 1.69587

leverarmrt -> 0.03

leverarmrz -> 0.08

leverarmrl -> 0.08

gain -> 0.4

gainrtzrtl -> 0.4

gaint -> 0.8

gainlrz -> 0.4

b1 -> 0.03

b2 -> 0.03

b3 -> 0.03

b4 -> 0.03

b5 -> 0.03

b6 -> 0.03

unstretched -> False

vertblades -> True
```

```
ul1 -> 0.610127

ul2 -> 0.594435

ul3 -> 0.498325

sl1 -> 0.612

sl2 -> 0.596

sl3 -> 0.5

si1 -> 0.0866013

si2 -> 0.223322

si3 -> 0.

c1 -> 0.996243

c2 -> 0.974745

c3 -> 1.

pitchbul -> 0

pitchbur -> 0

pitchbll -> 0

pitchblr -> 0

pitchbllf -> 0

pitchblrf -> 0

pitchbllb -> 0

pitchblrb -> 0

rollbul -> 0

rollbur -> 0

rollbll -> 0

rollblr -> 0

rollbllf -> 0

rollblrf -> 0

rollbllb -> 0

rollblrb -> 0

                -7
A1 -> 3.06796 10

                -7
A2 -> 1.25664 10

                -8
A3 -> 4.90874 10

kw1 -> 106226.

kw2 -> 44678.1

kw3 -> 20803.2

flex1 -> 0.00281363                                              calculated

flex2 -> 0.00190238                                              calculated

flex3 -> 0.00107976                                              calculated

kbuz -> 1341.87

kblz -> 884.924
```

```
bdu -> 0.147731

bdl -> 0.0770041

m13 -> 40.415

m23 -> 27.785

I1xy -> 0

I1yz -> 0

I1zx -> 0

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

Ibtxyl -> 0

Ibtyzl -> 0

Ibtzxl -> 0

I2xy -> 0.

I2yz -> 0

I2zx -> 0

COM2x -> 0

COM2y -> 0.

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0.                                                          cylinder: m3, tx, tr

I3yz -> 0

I3zx -> 0.                                                          cylinder: m3, tx, tr

COM3x -> 0.

COM3y -> 0.

COM3z -> 0.

FRP3x -> 0.

FRP3y -> 0

FRP3z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03

phib -> 0.001

bd -> 0.001

                 -15
M11 -> 7.49014 10

                 -15
```

```
M12 -> 7.49014 10

                 -15
M21 -> 1.25664 10

                 -15
M22 -> 1.25664 10

                 -16
M31 -> 1.91748 10

                 -16
M32 -> 1.91748 10

temperature -> 290.

                       -23
boltzmann -> 1.38066 10

                     -7
alphasilica -> 3.9 10                                            measurements by PW, AH - cf.
5.1 10^-7 from gwinc

betasilica -> 0.000152                                           gwinc/IFOModel v1.0

rhosilica -> 2200.                                               gwinc/IFOModel v1.0

Csilica -> 772.                                                  gwinc/IFOModel v1.0

Ksilica -> 1.38                                                  gwinc/IFOModel v1.0

                 10
Ysilica -> 7.2 10                                                spec sheet for silica

                   -10
phisilica -> 4.1 10                                              gwinc/IFOModel v1.0

                   -11
phissilica -> 3. 10                                              Phil Willems

rhosteel -> 7800.                                                gwinc/IFOModel v1.0

Csteel -> 486.                                                   gwinc/IFOModel v1.0

Ksteel -> 49.                                                    gwinc/IFOModel v1.0

                  11
Ysteel -> 2.119 10                                               measured, MB, 11/18/05

alphasteel -> 0.000012                                           gwinc/IFOModel v1.0

betasteel -> -0.00025                                            gwinc/IFOModel v1.0

phisteel -> 0.0002                                               Geppo's value

rhomarag -> 7800.                                                gwinc/IFOModel v1.0

Cmarag -> 460.                                                   gwinc/IFOModel v1.0

Kmarag -> 20.                                                    gwinc/IFOModel v1.0

                 11
Ymarag -> 1.87 10                                                gwinc/IFOModel v1.0

alphamarag -> 0.000011                                           gwinc/IFOModel v1.0

betamarag -> -0.00025                                            Geppo's value - gwinc/IFOModel
v1.0 is wrong

phimarag -> 0.0001                                               gwinc/IFOModel v1.0

tmU -> 0.0025                                                    upper blade thickness, NAR
8/4/06

tmL -> 0.0017                                                    lower blade thickness, NAR
8/4/06

magicnumber -> 0.0737472                                         Zener, 1938, Phys. Rev. 53:90-
99

deltabladeU -> 0.00182883                                        cf Bench delta_v1
```

```
deltabladeL -> 0.00182883                                           cf Bench delta_v3

deltawireU -> 0.00264152                                            cf Bench delta_h1

deltawireL -> 0.00259665                                            cf Bench delta_h3

deltafibre -> 0.00267157

taubladeU -> 0.113606

taubladeL -> 0.0525316

tauwireU -> 0.00222864                                              cf Bench tau_steel1

tauwireL -> 0.000912852                                             cf Bench tau_steel3

tausilica -> 0.00782827

                                        0.00130543 #1
damping[imag, bladeUtype] -> (0.0001 + ---------------- & )
                                                     2
                                      1 + 0.509525 #1

                                        0.000603633 #1
damping[imag, bladeLtype] -> (0.0001 + ---------------- & )
                                                     2
                                      1 + 0.108943 #1

damping[imag, wireUtype] -> (0.0002 & )

damping[imag, wireLtype] -> (0.0002 & )

                                         0.0000369891 #1
damping[imag, wireUatype] -> (0.0002 + ------------------- & )
                                                        2
                                      1 + 0.000196083 #1

                                         0.0000148934 #1
damping[imag, wireLatype] -> (0.0002 + -------------------- & )
                                                         2
                                      1 + 0.0000328973 #1

damping[imag, fibretype] -> (0.0002 & )

                                                   -6
                                         5.98559 10   #1
damping[imag, fibreatype] -> (0.0002 + -------------------- & )
                                                     -6   2
                                      1 + 5.01974 10   #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0
```

```
kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

lockedblades -> False

kw1usual -> 106226.

kw2usual -> 44678.1

kw3usual -> 20803.2

kbuzusual -> 1341.87

kblzusual -> 884.924

                  7
kbuy -> 1.34187 10

                  6
kbly -> 8.84924 10

kbux -> 134187.

kblx -> 88492.4

ifs -> 0.354                                                    D080369-v2, info only

wedge -> 0                                                      no wedge

wabh3 -> 0                                                      no horizontal wedge on back

wabv3 -> 0                                                      no vertical wedge on back

wafh3 -> 0                                                      no horizontal wedge on front

wafv3 -> 0                                                      no vertical wedge on front

dl -> 0.                                                        correction to wire length to
offset wedge mass imbalance

bssteel -> 2000000000                                           breaking stress of steel

                  8
wssilica -> 7.7 10                                              working stress of silica

r1opt -> 0.000308233

r2opt -> 0.000182698

r3opt -> 0.000120028

dssilica -> 0.015                                               gwinc/IFOModel v1.0

taufibre -> 0.000356583
```

Back to Top

aLIGO SUS Operation Manual - Info on HLTS Suspensions

Back to Operation Manual main page

References

- HLTS Wiki Page (old)
- T010103: aLIGO Suspension System Conceptual Design
- D070447: HLTS Overall Assembly
- T1000061: aLIGO HAM Triple Suspension Controls Design Description
- E1100109: HAM Suspensions Controls Arrangement Poster
- T1300083: HAM Large Triple Suspension (HLTS) Control Ranges
- D0902810: aLIGO HAM2 Suspension Controls Wiring Diagrams
- D1002740: aLIGO SUS HAM 5-6 Wiring Diagrams

Models

The HLTS suspension has been extensively modelled. Key results are at Suspensions/OpsManual/HLTS/Models .

Screens

HLTS MEDM screens are documented at /Screens.

aLIGO: Suspensions/OpsManual/HLTS (last edited 2013-04-05 11:05:37 by MarkBarton)

aLIGO SUS Operations Manual - Overview of HLTS MEDM screens

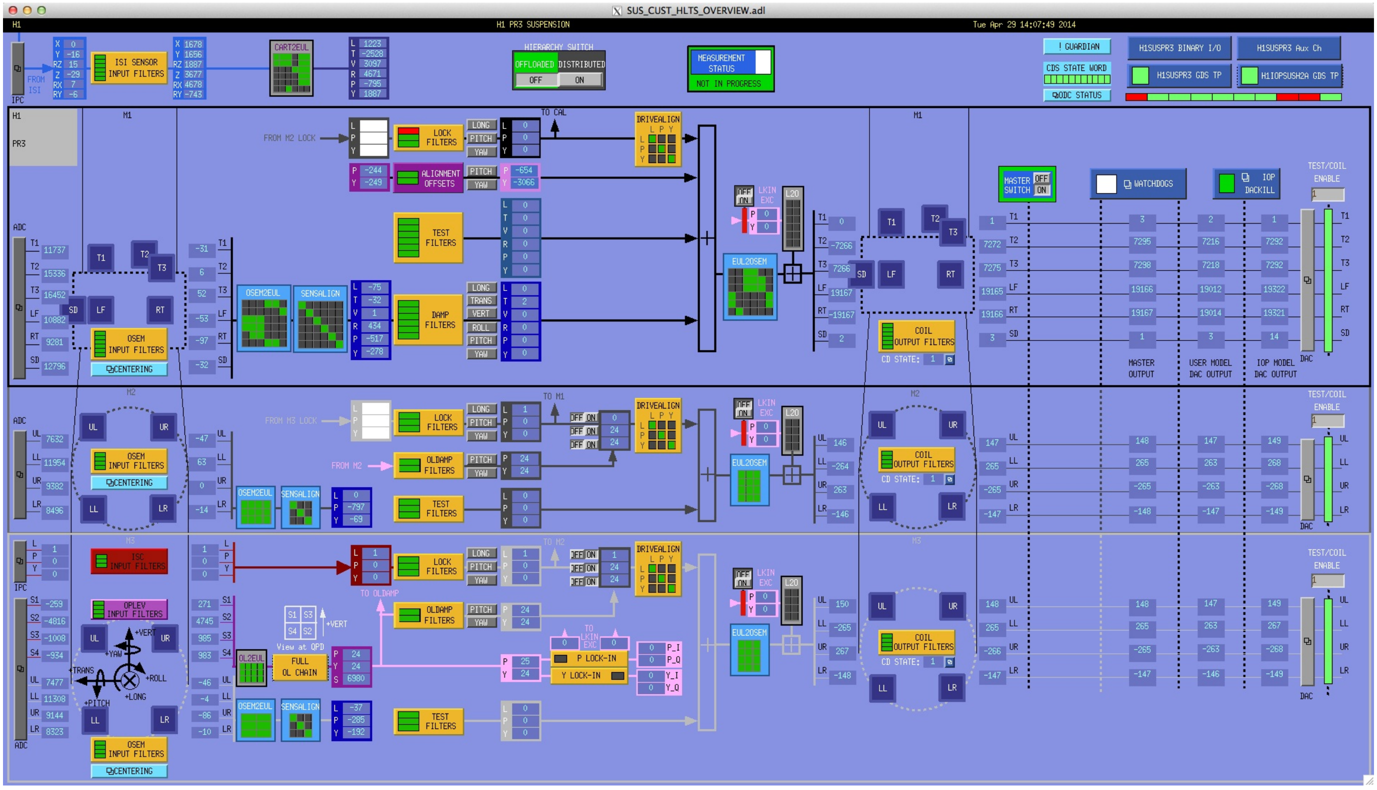
Back to Operation Manual main page
Back to HLTS main page

This page makes extensive use of text fragments in-lined from Suspensions/OpsManual/Boilerplate. Use the extract.py script there to update.

Except where noted, the HLTS screens described below are common to HLTS and HSTS and live at `/opt/rtcds/userapps/release/sus/common/medm/hxts/`. (Note `hxts` not `hlts`.) They are all generic screens which rely for their execution on site-, interferometer- and suspension-specific arguments passed in when they are called. They can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_HLTS_OVERVIEW.adl

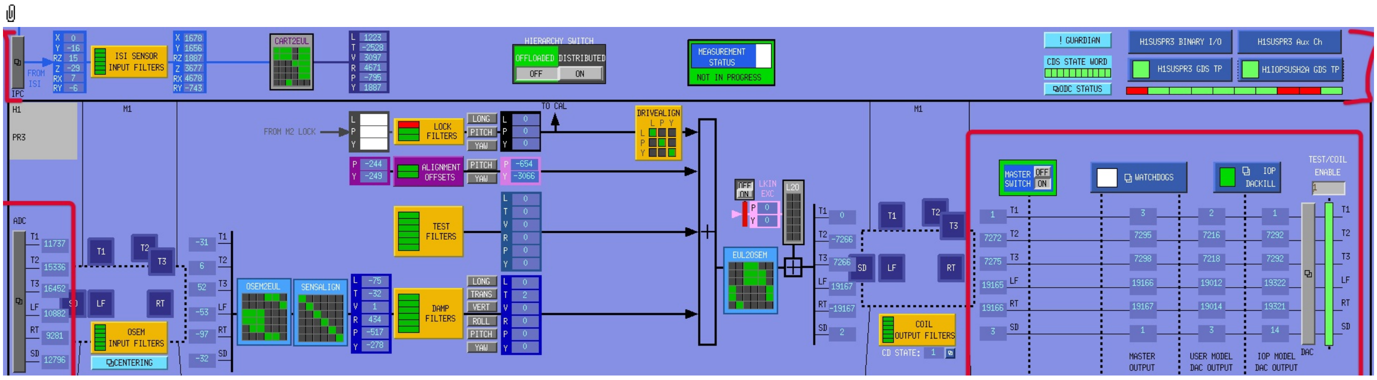


This is the overview screen and one of only a few specific to HLTS (with HLTS rather than HXTS in the name). The functionality is divided up according to the three sensor-actuator groups, plus an odds-and-ends area at the top:

- Other - subscreens not associated with a sensor actuator group
- M1 - 6 BOSEMs on the structure engaging the upper top mass
- M2 - 4 BOSEMs on the structure engaging the intermediate mass
- M3 - 4 AOSEMs on the structure engaging the optic, also an optical lever

M1 is used for local damping (relative to the structure). DC pitch and yaw offsets are also injected at M1. M2 and M3 are for implementing global control (relative to other optics) and have inputs for actuation requests from ISC.

Other Screens



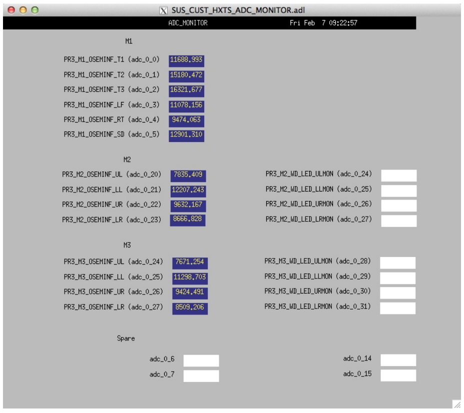
There are the following additional controls/displays that apply to all sensor-actuator groups and have been crammed into the M1 area:

- MASTER SWITCH - a manual on/off switch for all actuation (common to all sensor-actuator groups).
- MEASUREMENT STATUS - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- HIERARCHY SWITCH - enables various ISC inputs and outputs.
- ADC - a monitor screen for the semi-raw ADC values as reported by the IOP process after downsampling.
- DAC - a popup allowing access to all the DAC outputs as reported by the IOP process.
- H1SUSPR3 MONITORS (or the like; depends on the suspension name) - readbacks from coil driver channels for
- WATCHDOG - a block implementing the watchdogs on the various sensor-actuator groups.
- IOP DACKILL - a watchdog that shuts off the IOP process (potentially other suspensions on the same front-end).
- H1IOPSUSH34 GDS TP (or the like; depends on the front-end computer name) - status of the IOP process for the front-end.
- H1SUSPR3 GDS TP (or the like; depends on the suspension name) - status of the suspension process.
- H1SUSPR3 BIO (or the like; depends on the suspension name) - binary input/output controls.

all OSEMs as reported by the corresponding auxiliary front-end processor.

- ODC - ODC status
- GUARDIAN Guardian controls
- ISIINF - ISI input filters.
- CART2EUL - transformation from the ISI basis to SUS.

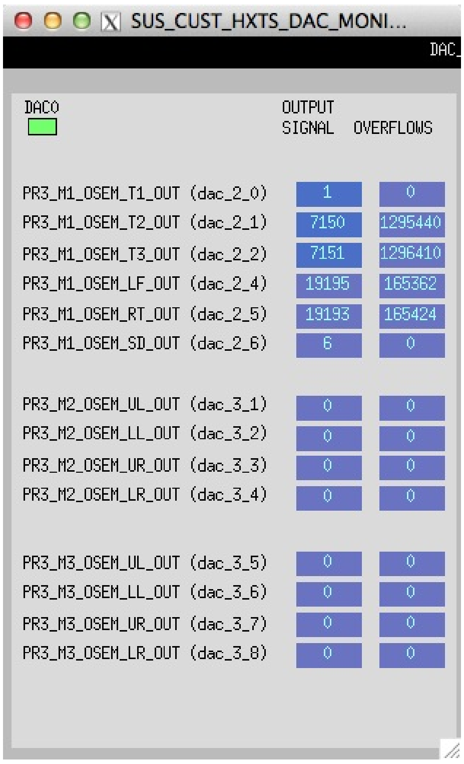
Screen SUS_CUST_HXTS_ADC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen SUS_CUST_HXTS_DAC_MONITOR.adl

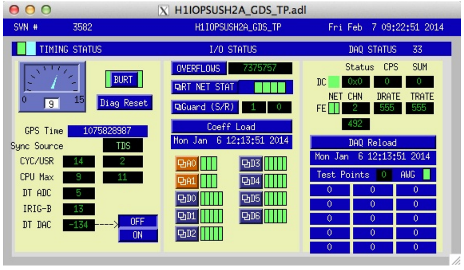


Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen H1IOPSUSH34_GDS_TP.adl

Screen H1IOPSUSH2A_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

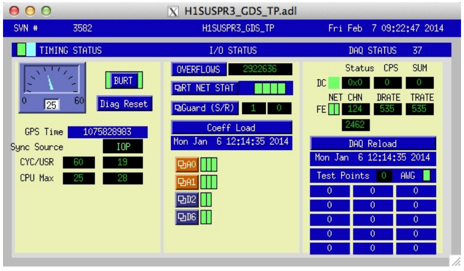
This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1SUSPR3_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

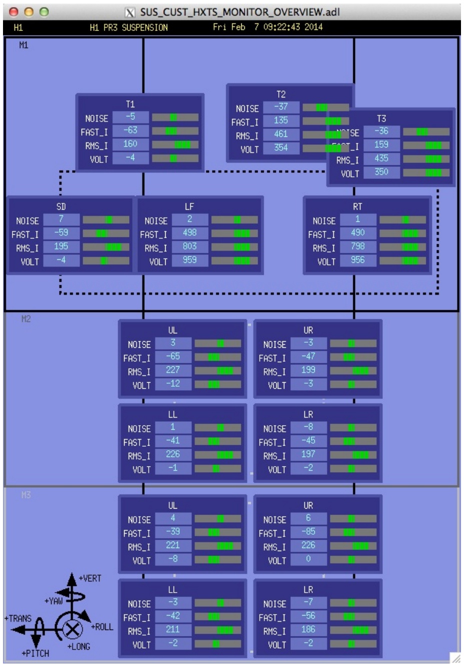
This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen SUS_CUST_HXTS_MONITOR_OVERVIEW.adl

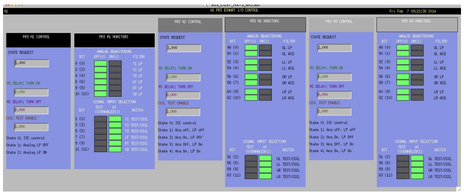


Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

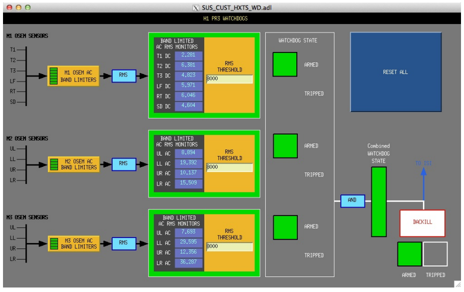
Screen SUS_CUST_HXTS_BIO.adl



Suspensions/OpsManual/Boilerplate/HLTS_BIO:

The Binary Input-Output control screen. State Requests for the various sensor-actuator groups can be input. Positive state values lock out manual control of the filters on the COILOUTF screen and set the digital dewhitening filters to match the analog whitening filters selected via the binary outputs. Negative state values allow the dewhitening filters to be manually overridden.

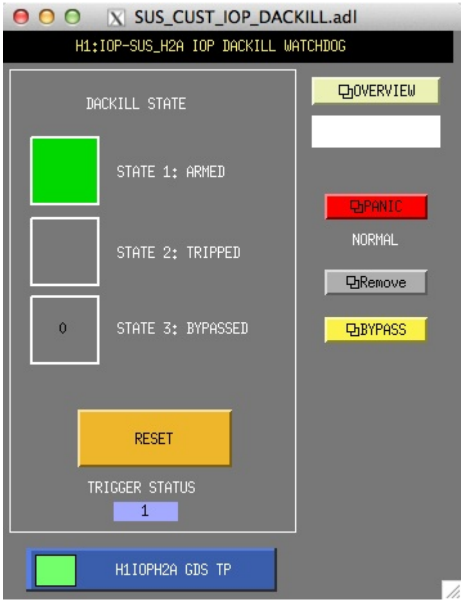
Screen SUS_CUST_HXTS_WD.adl



Suspensions/OpsManual/Boilerplate/WD:

This screen allows for setting the watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

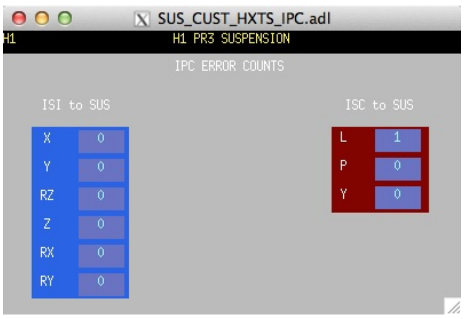
Screen SUS_CUST_IOP_DACKILL.adl



Suspensions/OpsManual/Boilerplate/IOP_DACKILL:

The watchdog controls for the IOP process. If this WD is tripped it disables IO to/from all user models on the front end. RESET resets from the tripped state if there are no outstanding triggers. BYPASS is a manual override that allows for operation of the suspension if there is a trigger condition that is known by the operator to be harmless.

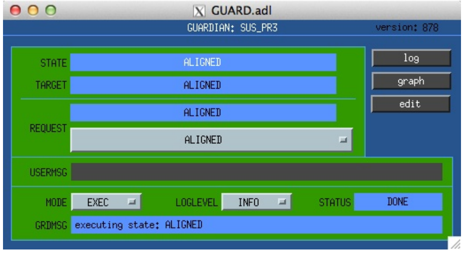
Screen SUS_CUST_HXTS_IPC.adl



Suspensions/OpsManual/Boilerplate/IPC:

Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

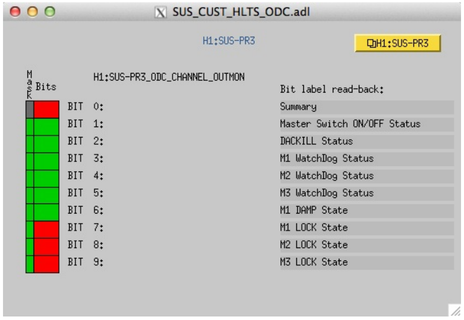
Screen GUARD.adl



Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

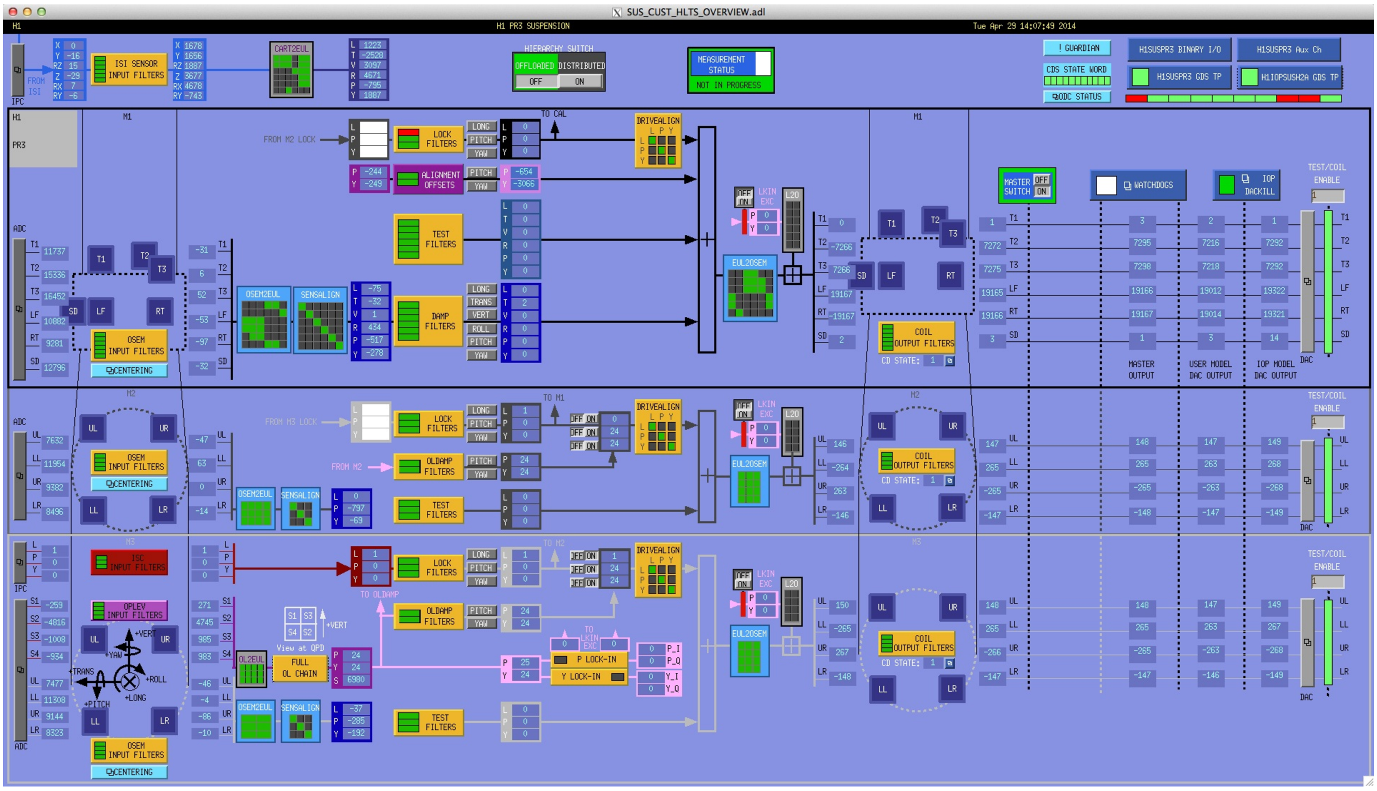
Screen SUS_CUST_HLTS_ODC.adl



Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

Sensor Actuator Group M1 - Upper Mass

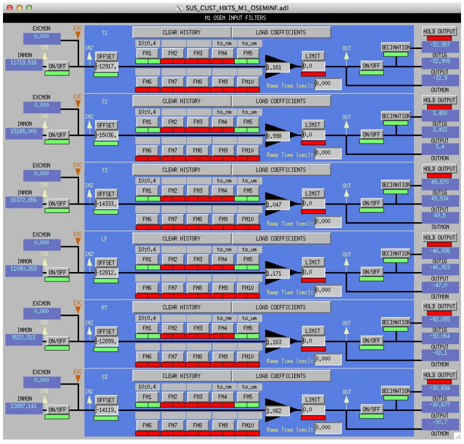


The M1 section of the main HLTS screen is excerpted above. Most of the items on it correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- DAMP FILTERS - a filter bank applying the damping filters.
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.
- LOCK FILTERS - filters for global control signals.

- DRIVEALIGN - a block reserved for applying frequency-dependent corrections to the actuation (defaults to the identity matrix).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

Screen SUS_CUST_HXTS_M1_OSEMINF.adl



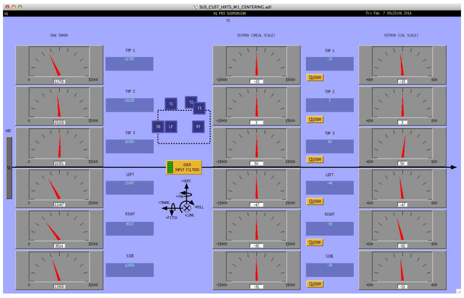
Suspensions/OpsManual/Boilerplate/M1_OSEMINF:

This block has 6 filter groups corresponding to the 6 M1 BOSEMs, T1/T2/T3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_HXTS_M1_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M1_CENTERING:

This screen gives various views of the M1 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_HXTS_M1_OSEM2EUL.adl

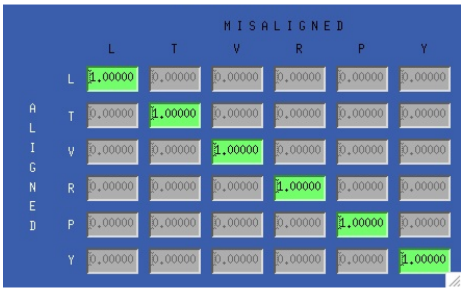
Suspensions/OpsManual/Boilerplate/M1_OSEM2EUL:

This screen allows entry of the matrix which converts from the M1 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

Screen SUS_CUST_HXTS_M1_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_SENSALIGN:

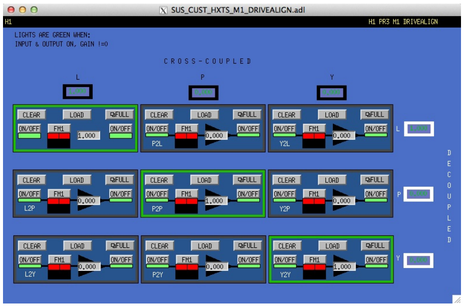
This screen is reserved for tweaking the M1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_HXTS_M1_DAMP.adl


Suspensions/OpsManual/Boilerplate/M1_DAMP:

These filters implement the local damping. See Suspensions/OpsManual/Damping.

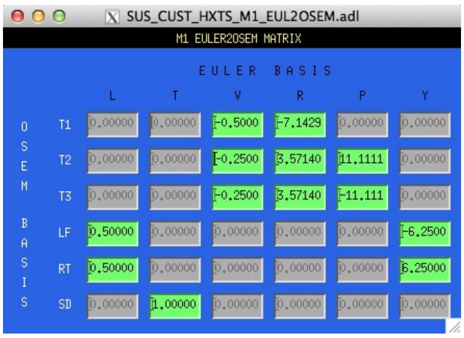
Screen SUS_CUST_HXTS_M1_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_DRIVEALIGN:

This screen is reserved for tweaking the M1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

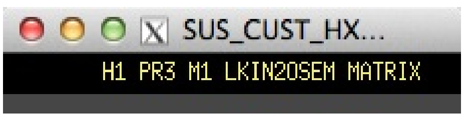
Screen SUS_CUST_HXTS_M1_EUL2OSEM.adl

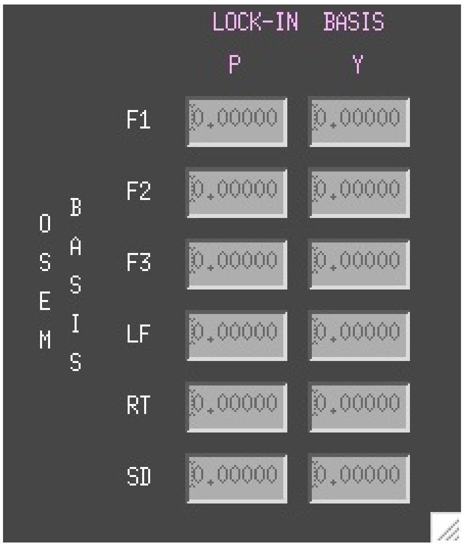


Suspensions/OpsManual/Boilerplate/M1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_HXTS_M1_LKIN2OSEM.adl

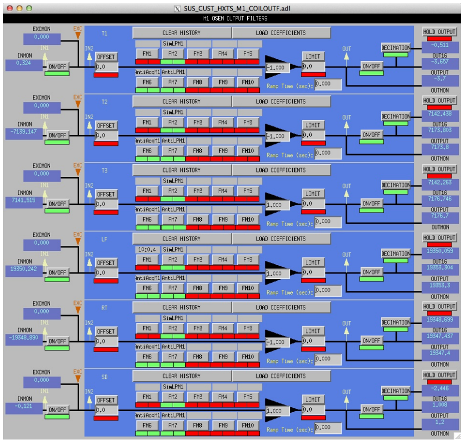




Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_HXTS_M1_COILOUTF.adl



Suspensions/OpsManual/Boilerplate/M1_COILOUTF:

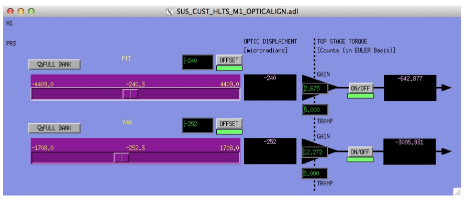
This screen applies compensation for the hardware filters in the M1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

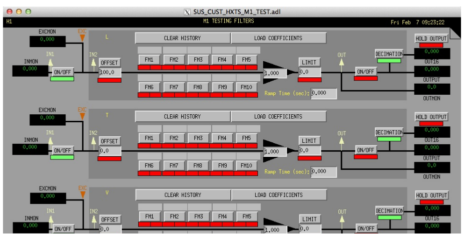
Screen SUS_CUST_HLTS_M1_OPTICALIGN.adl

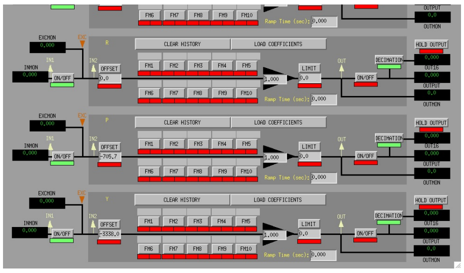


Suspensions/OpsManual/Boilerplate/M1_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets. See Suspensions/OpsManual/Alignments for more info.

Screen SUS_CUST_HXTS_M1_TEST.adl

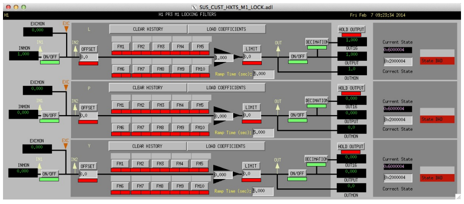




Suspensions/OpsManual/Boilerplate/M1_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

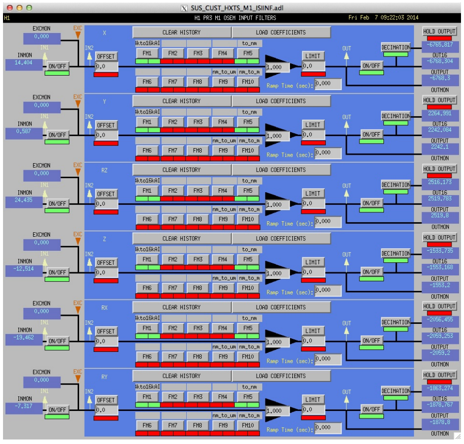
Screen SUS_CUST_HXTS_M1_LOCK.adl



Suspensions/OpsManual/Boilerplate/M1_LOCK:

Filters for the locking signals.

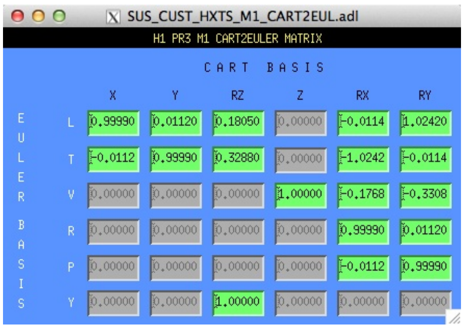
Screen SUS_CUST_HXTS_M1_ISIINF.adl



Suspensions/OpsManual/Boilerplate/M1_ISIINF:

Filters for the diagnostic signals from the ISI.

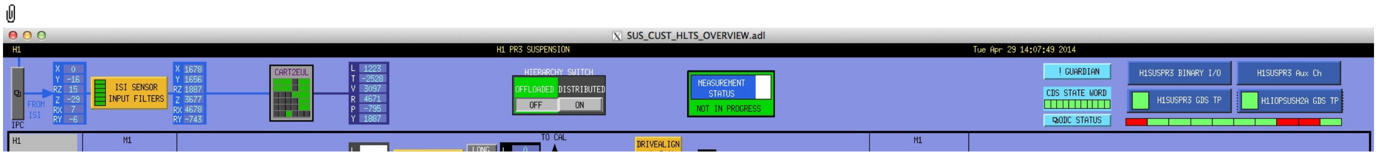
Screen SUS_CUST_HXTS_M1_CART2EUL.adl

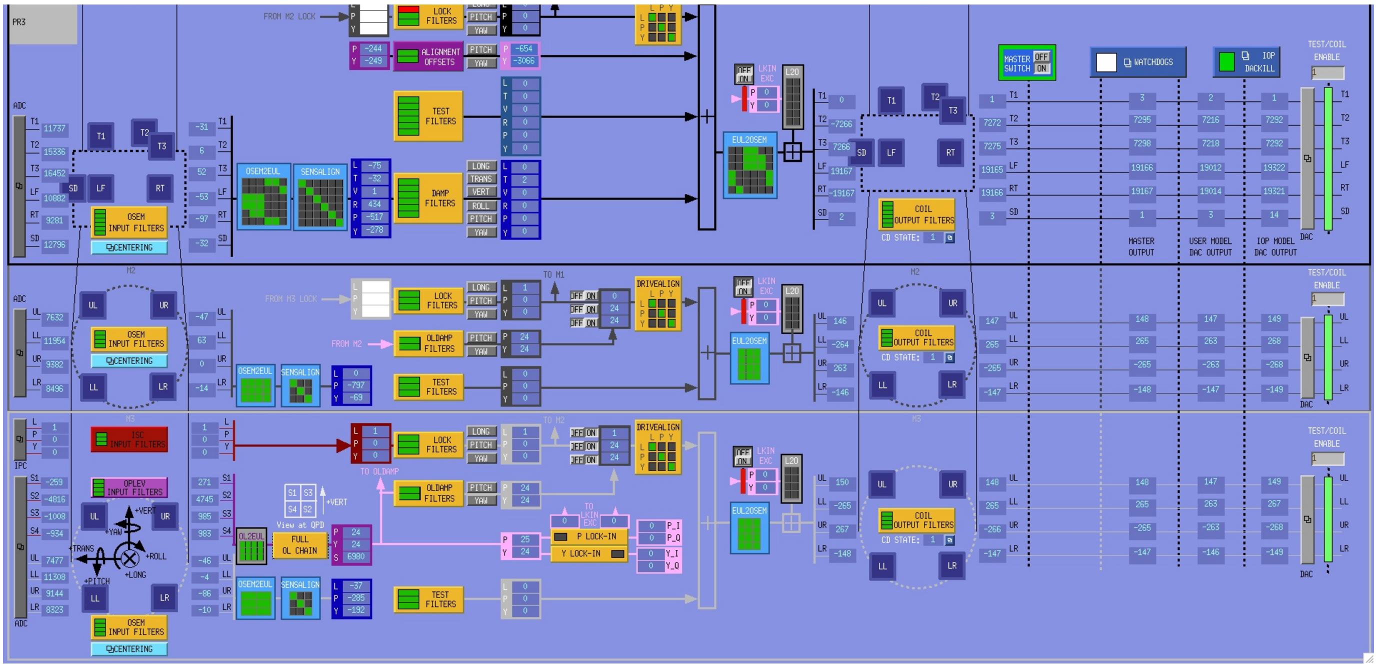


Suspensions/OpsManual/Boilerplate/M1_CART2EUL:

A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

Sensor Actuator Group M2 - Intermediate Mass

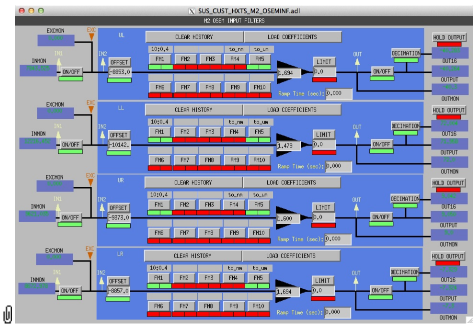




The M2 level has the following blocks in the main path:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/pitch/yaw (L/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- LOCK FILTERS - filters for global control signals.
- OLDAMP FILTERS - filters for optical lever damping.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

Screen SUS_CUST_HXTS_M2_OSEMINF.adl



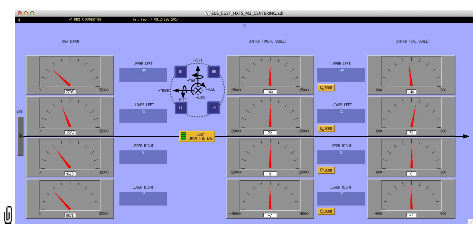
Suspensions/OpsManual/Boilerplate/M2_OSEMINF:

This block has 4 filter banks corresponding to the 4 M2 OSEMs, UL/LL/UR/LR. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_HXTS_M2_CENTERING.adl

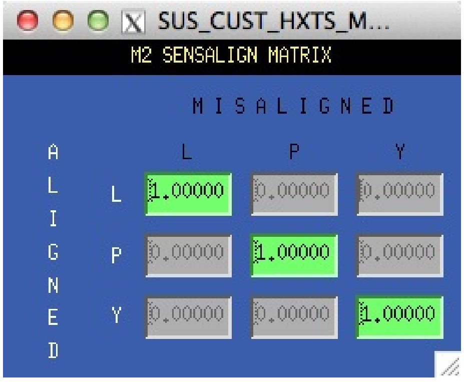


Suspensions/OpsManual/Boilerplate/M2_CENTERING:

This screen gives various views of the OSEM signals as meter panels for ease of viewing when mechanically adjusting

OSEMs.

Screen SUS_CUST_HXTS_M2_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M2_SENSALIGN:

This screen is reserved for tweaking the M2 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_HXTS_M2_OSEM2EUL.adl

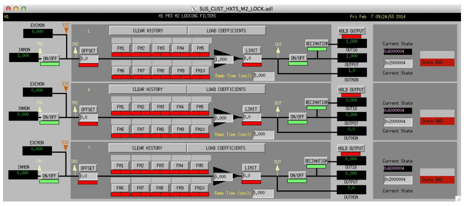
Suspensions/OpsManual/Boilerplate/M2_OSEM2EUL:

This screen allows entry of the matrix which converts from the OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOF L are dimensionless.

The entries for the angular DOFs P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

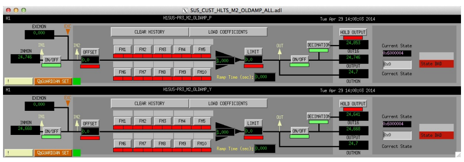
Screen SUS_CUST_HXTS_M2_LOCK.adl



Suspensions/OpsManual/Boilerplate/M2_LOCK:

Filters for the locking signals.

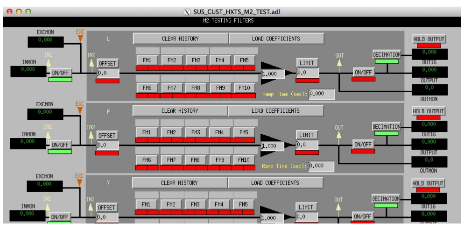
Screen SUS_CUST_HLTS_M2_OLDAMP_ALL.adl

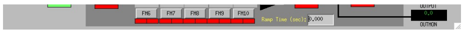


Suspensions/OpsManual/Boilerplate/M2_OLDAMP:

Filters for the optical lever locking signals.

Screen SUS_CUST_HXTS_M2_TEST.adl

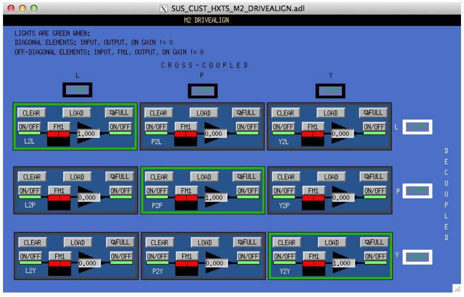




Suspensions/OpsManual/Boilerplate/M2_TEST:

This is a deliberately empty set of 3 filter banks corresponding to the M2 L, P and Y DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

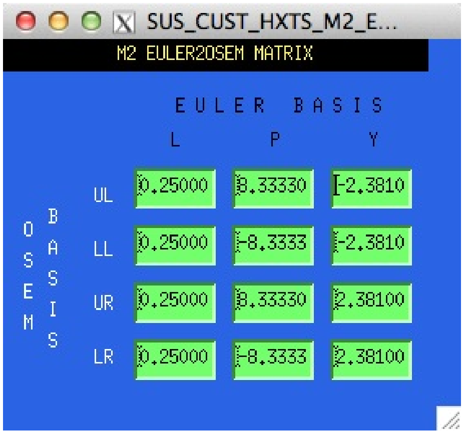
Screen SUS_CUST_HXTS_M2_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M2_DRIVEALIGN:

This screen is reserved for tweaking the actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

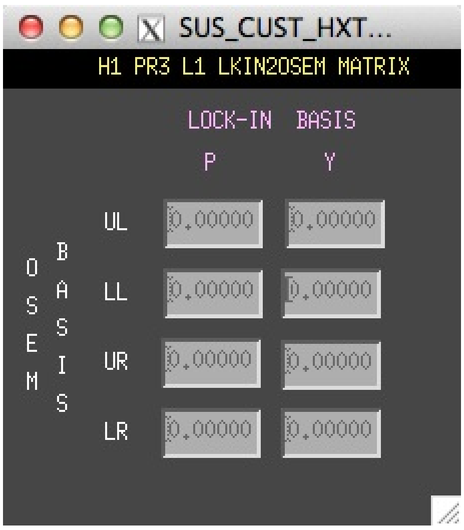
Screen SUS_CUST_HXTS_M2_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M2_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

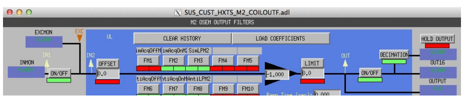
Screen SUS_CUST_HXTS_M2_LKIN2OSEM.adl

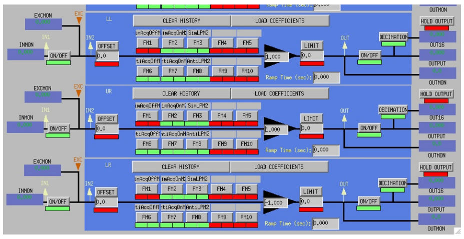


Suspensions/OpsManual/Boilerplate/M2_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_HXTS_M2_COILOUTF.adl





Suspensions/OpsManual/Boilerplate/M2_COILOUTF:

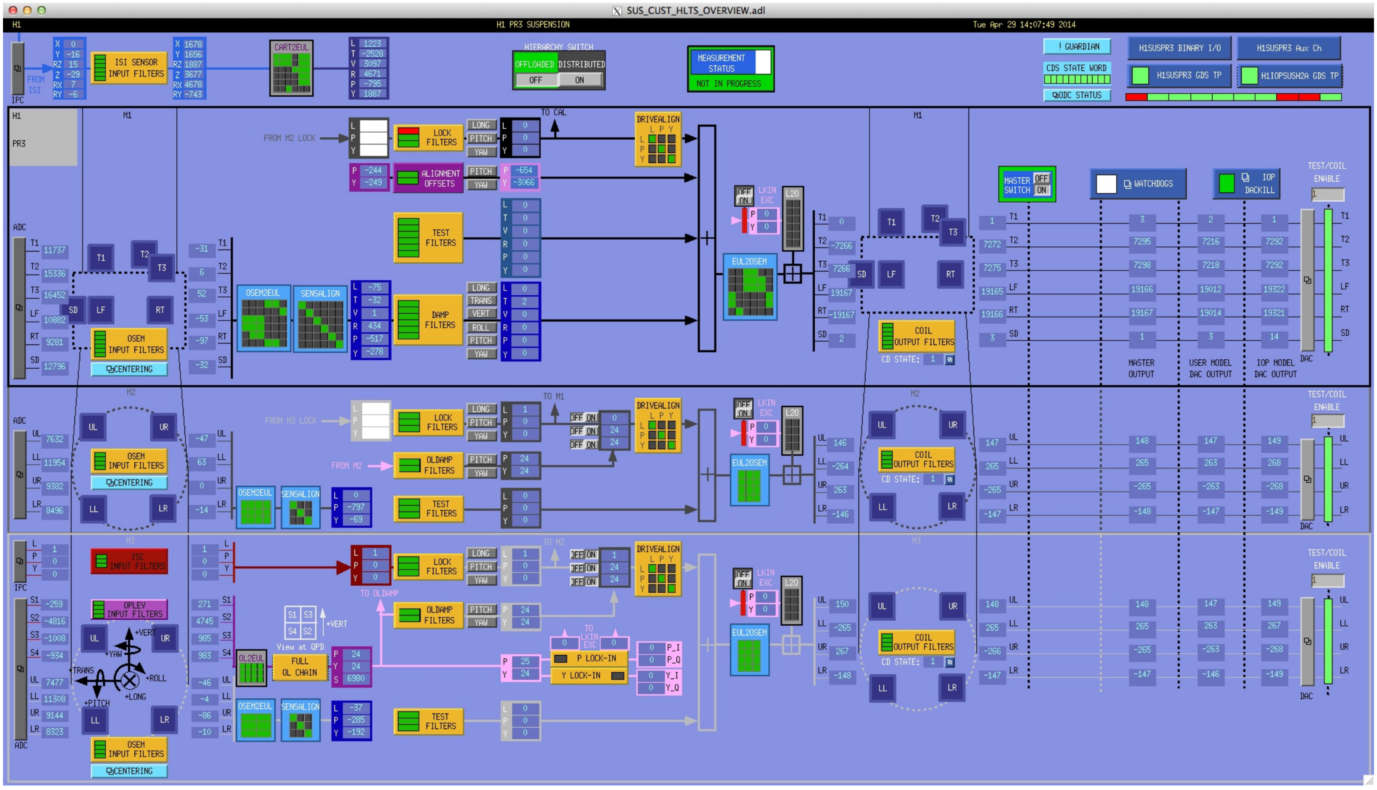
This screen applies compensation for the hardware filters in the actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

Sensor Actuator Group M3 - Optic



The M3 level has the following blocks in the main path:

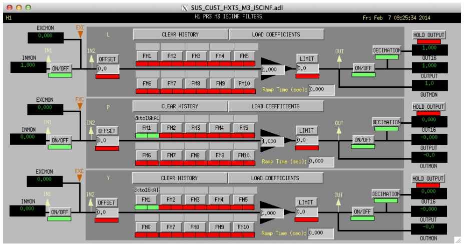
- ISC INPUT FILTERS - ISC input filters.
- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to μm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/pitch/yaw (L/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- LOCK FILTERS - filters for global control signals.
- OLDAMP FILTERS - filters for optical lever damping.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

HLTS has provision for optical levers:

- OPLEV INPUT FILTERS - input filters for optical levers.
- OL2EUL - a block that diagonalizes the OL signal to signals in a pitch/yaw/sum (P/Y/S) basis.

- FULL OL CHAIN - overview of optical levers.

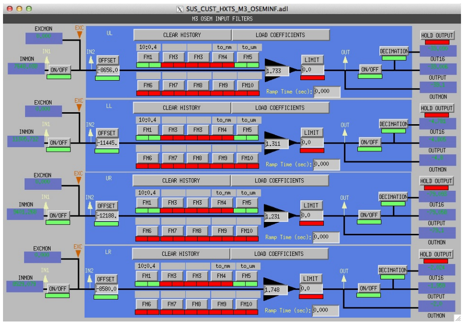
Screen SUS_CUST_HXTS_M3_ISCINF.adl



Suspensions/OpsManual/Boilerplate/M3_ISCINF:

Filters for the control signals from ISC.

Screen SUS_CUST_HXTS_M3_OSEMINF.adl



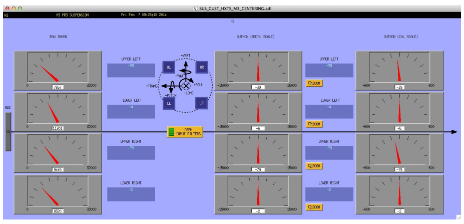
Suspensions/OpsManual/Boilerplate/M3_OSEMINF:

This block has 4 filter banks corresponding to the 4 M3 AOSEMs, UL/LL/UR/LR. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_HXTS_M3_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M3_CENTERING:

This screen gives various views of the OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_HXTS_M3_OSEM2EUL.adl

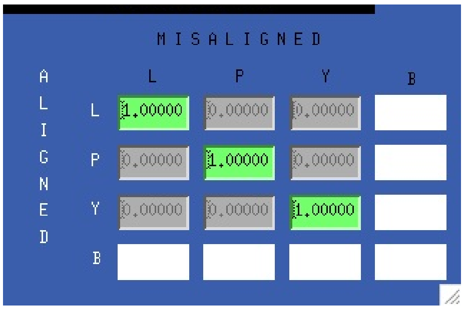
Suspensions/OpsManual/Boilerplate/M3_OSEM2EUL:

This screen allows entry of the matrix which converts from the OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOF L are dimensionless.

The entries for the angular DOFs P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

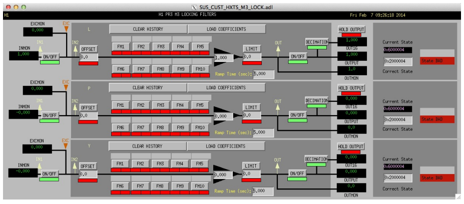
Screen SUS_CUST_HXTS_M3_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M3_SENSALIGN:

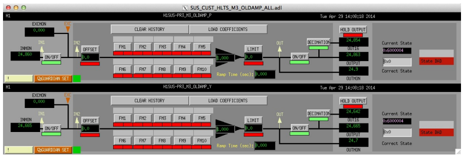
This screen is reserved for tweaking the M3 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_HXTS_M3_LOCK.adl


Suspensions/OpsManual/Boilerplate/M3_LOCK:

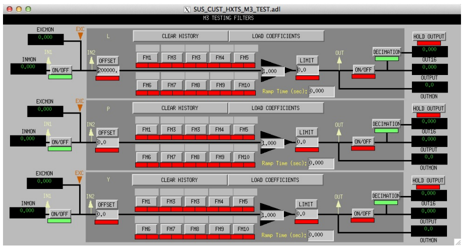
Filters for the locking signals.

Screen SUS_CUST_HLTS_M3_OLDAMP_ALL.adl


Suspensions/OpsManual/Boilerplate/M3_OLDAMP:

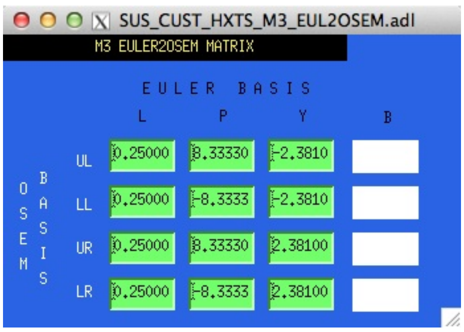
Filters for the optical lever locking signals.

Screen SUS_CUST_HXTS_M3_TEST.adl


Suspensions/OpsManual/Boilerplate/M3_TEST:

This is a deliberately empty set of 3 filter banks corresponding to the L, P and Y DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

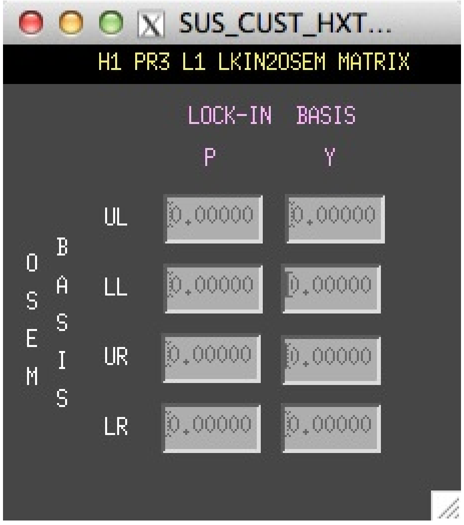
Screen SUS_CUST_HXTS_M3_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M3_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

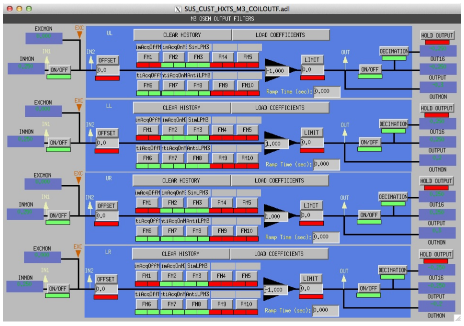
Screen SUS_CUST_HXTS_M3_LKIN2OSEM.adl



Suspensions/OpsManual/Boilerplate/M3_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_HXTS_M3_COILOUTF.adl



Suspensions/OpsManual/Boilerplate/M3_COILOUTF:

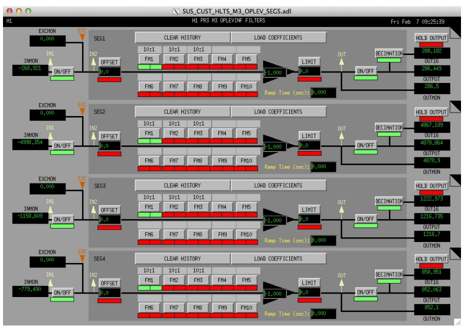
This screen applies compensation for the hardware filters in the actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

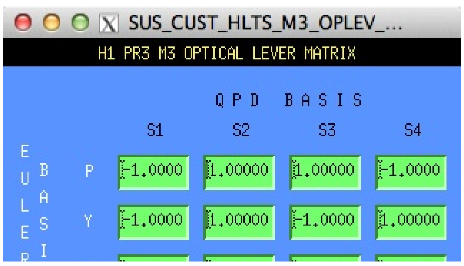
Screen SUS_CUST_HLTS_M3_OPLEV_SEGS.adl



Suspensions/OpsManual/Boilerplate/M3_OPLEV_SEGS:

Filters for the optical lever QPD segment signals.

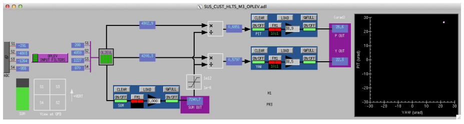
Screen SUS_CUST_HLTS_M3_OPLEV_MTRX.adl



Suspensions/OpsManual/Boilerplate/M3_OPLEV_MTRX:

Transformation matrix from the optical lever QPD segment basis into the P/Y/SUM basis.

Screen SUS_CUST_HLTS_M3_OPLEV.adl


Suspensions/OpsManual/Boilerplate/M3_OPLEV:

Overview of the Optical Lever signal processing.

aLIGO SUS Operations Manual - Models for HLTS Suspensions

Back to Operation Manual main page

The HLTS suspensions have been extensively modelled. There are Mathematica (T020205) and Matlab (T080188) programs that take a parameter set describing a single chain and calculate the dynamics. The parameter set needs to be formatted differently for Mathematica vs Matlab, but the parameter names are the same as far as was practical.

The Mathematica triple model, which covers HLTS as well as HSTS and BSFM, and parameter sets for it lives in the SUS SVN at `^/trunk/Common/MathematicaModels/TripleLite2/`. Parameter sets for Matlab live at `^/trunk/Common/MatlabTools/TripleModel_Production`. Mark Barton maintains the Mathematica, Jeff Kissel maintains the Matlab. Unfortunately two different naming conventions are in use and it's difficult to tell what corresponds to what - if in doubt, ask. References given below are to Mathematica versions as subdirectories of `^/trunk/Common/MathematicaModels/TripleLite2/`.

Key cases of the triple model for HLTS are given below, with results such as mode frequencies and mode shapes. T1200404 has transfer functions for many of the same models.

Generic HLTS

Current best generic HLTS model is .../mark.barton/20120120hlts.

Case 20120120hlts of Mathematica model TripleLite2

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
{"mark.barton", "20120120hlts"}
```

```
Equivalent to Jeff K's hltsopt_metal.m revision 2034 of 1/24/12 for PR metal build.
```

Links to Result Sections

Pendulum Mode Summary	Violin Mode Summary	Pendulum Mode Shapes	Parameters

modeP1	modeT1	modeL1	modeY1	modeV1	modeR1
modeL2	modeR2	modeP2	modeY2	modeT2	modeL3
modeY3	modeV2	modeP3	modeT3	modeV3	modeR3

Pendulum Mode Summary

Back to Top

```
N    f          name     type

1    0.659538   modeP1   pitch3   pitch2

2    0.692337   modeT1   roll3    roll2    y3

3    0.744758   modeL1   pitch3   pitch2

4    0.989521   modeY1   yaw3     yaw2

5    1.06994    modeV1   z3       z2

6    1.50732    modeR1   roll3    roll2

7    1.5814     modeL2   pitch1

8    1.97903    modeR2   roll3    roll2

9    2.12717    modeP2   pitch1

10   2.23634    modeY2   yaw3     yaw1

11   2.50427    modeT2   roll1

12   2.85421    modeL3   pitch2   x1

13   3.34348    modeY3   yaw2     yaw1

14   3.50334    modeV2   z1

15   3.51862    modeP3   pitch2

16   3.68116    modeT3   roll1    roll3

17   28.1003    modeV3   z2       z3

18   44.7247    modeR3   roll2
```

Violin Mode Summary

Back to Top

Violin Modes	UM	IM	Optic
f (Hz), n=1-4	724.247 1452.39 2188.39 2936.35	744.19 1489.44 2236.81 2987.37	513.273 1026.98 1541.56 2057.44
Q, n=1-4	72423.3 85167.2 77775.7 65222.4	67514.1 96724. 107132. 105288.	63747.6 81281.3 94421.3 99492.5

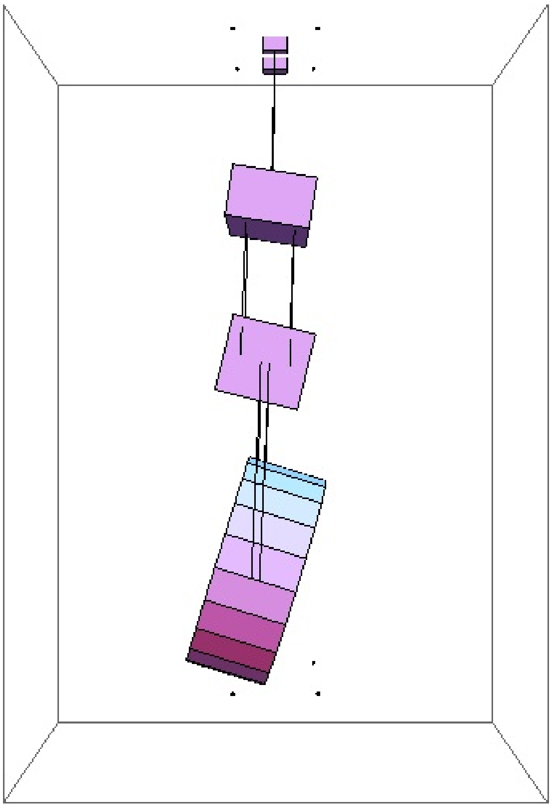
Mode Shapes

Mode #1 - modeP1

Back to Top

```
0.659538 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.0122691	0	0	0	0.379662	0
Mass I	-0.0287355	0	0	0	0.57534	0
optic	-0.0569338	0	0	0	0.721542	0

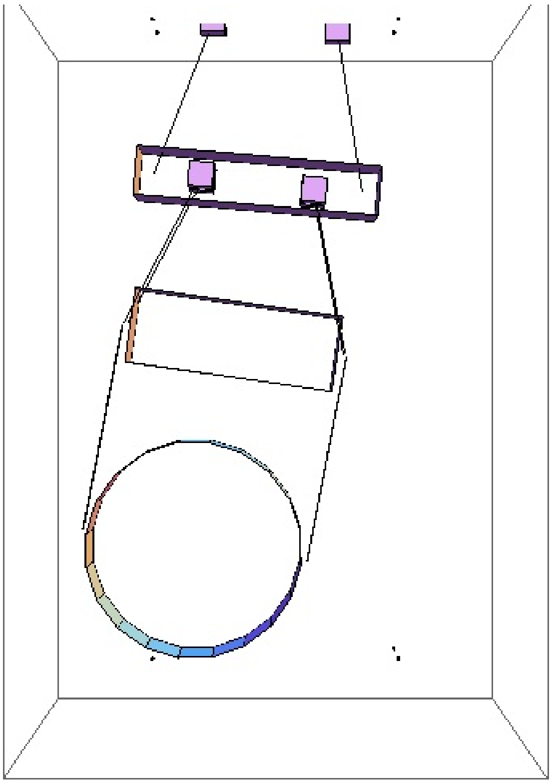


Mode #2 - modeT1

Back to Top

```
0.692337 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.0883701	0	0	0	0.352577
Mass I	0	0.206134	0	0	0	0.574222
optic	0	0.406826	0	0	0	0.574587

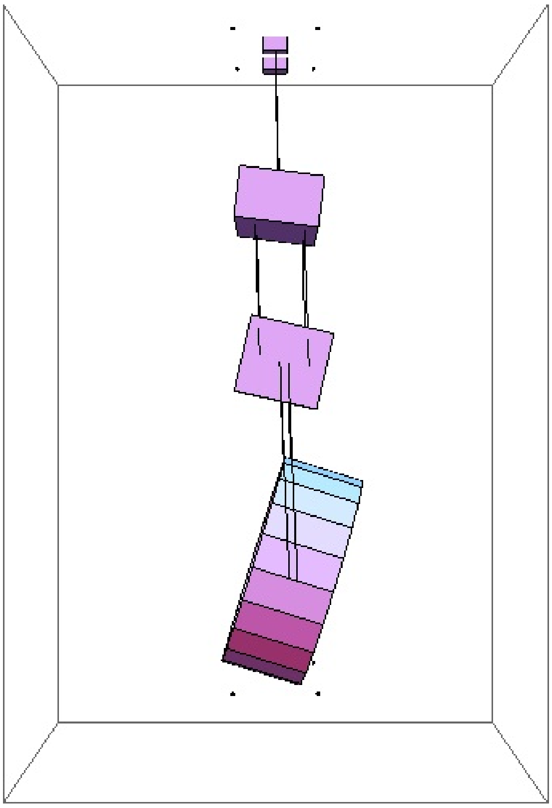


Mode #3 - modeL1

Back to Top

```
0.744758 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.0116095	0	0	0	0.316161	0
Mass I	0.0260904	0	0	0	0.565622	0
optic	0.0525815	0	0	0	0.759299	0

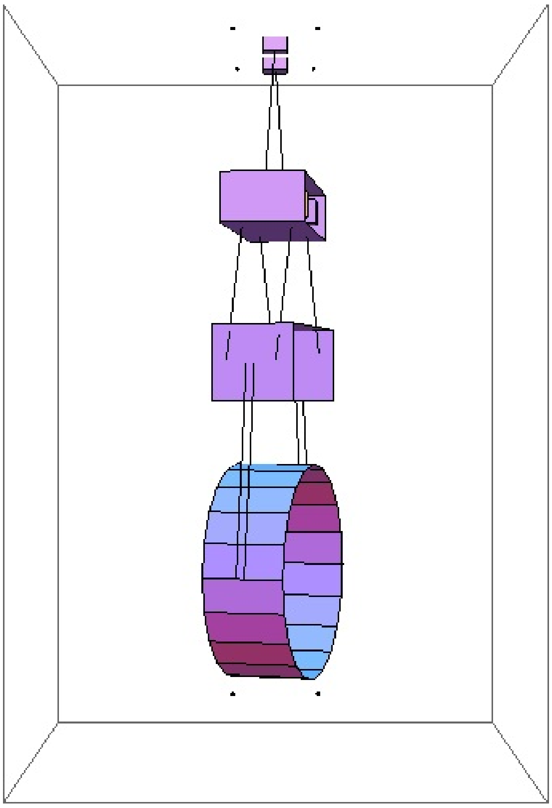


Mode #4 - modeY1

Back to Top

```
0.989521 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	0.275828	0	0
Mass I	0	0	0	0.554968	0	0
optic	0	0	0	0.784812	0	0

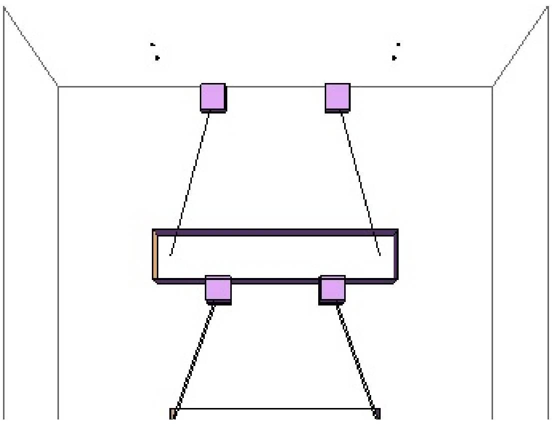


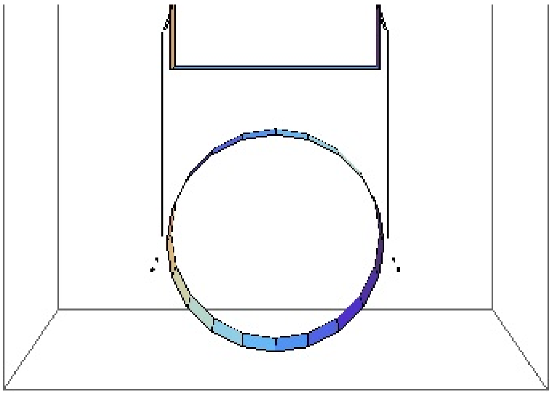
Mode #5 - modeV1

Back to Top

```
1.06994 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.382597	0	0	0
Mass I	0	0	-0.652357	0	0	0
optic	0	0	-0.654255	0	0	0



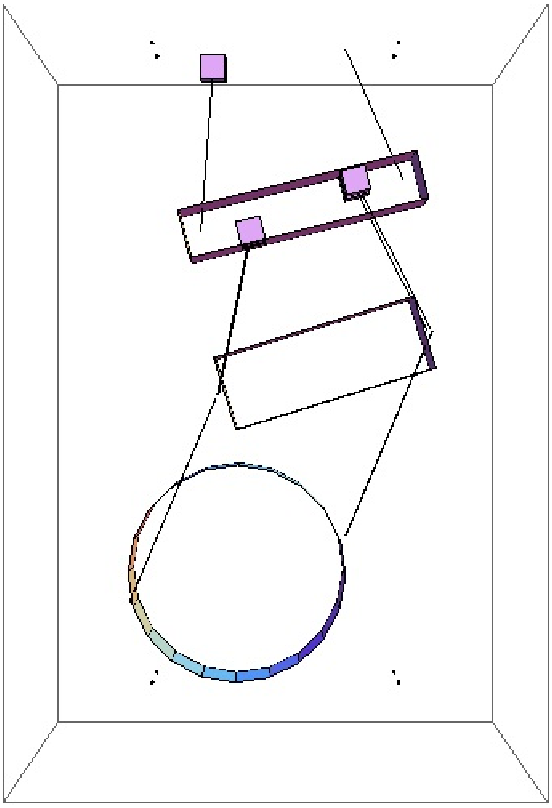


Mode #6 - modeR1

Back to Top

```
1.50732 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.067815	0	0	0	-0.49431
Mass I	0	-0.121181	0	0	0	-0.602246
optic	0	0.0949076	0	0	0	-0.603876

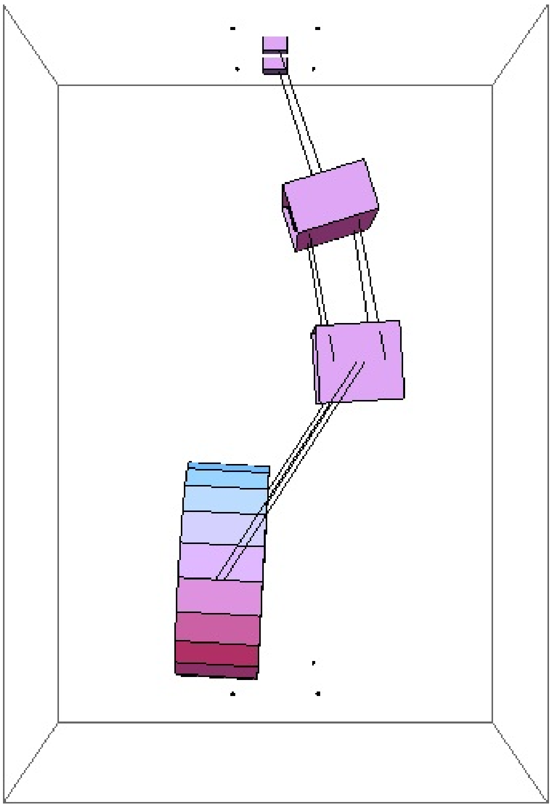


Mode #7 - modeL2

Back to Top

```
1.5814 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.191935	0	0	0	-0.882633	0
Mass I	0.294221	0	0	0	-0.181135	0
optic	-0.191187	0	0	0	0.167902	0

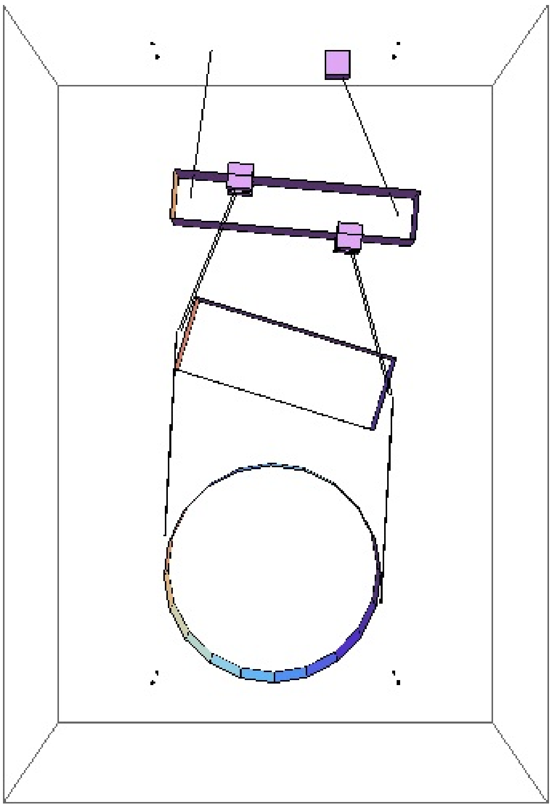


Mode #8 - modeR2

Back to Top

```
1.97903 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.0549799	0	0	0	-0.20392
Mass I	0	0.0277194	0	0	0	-0.689153
optic	0	-0.00821975	0	0	0	-0.692551

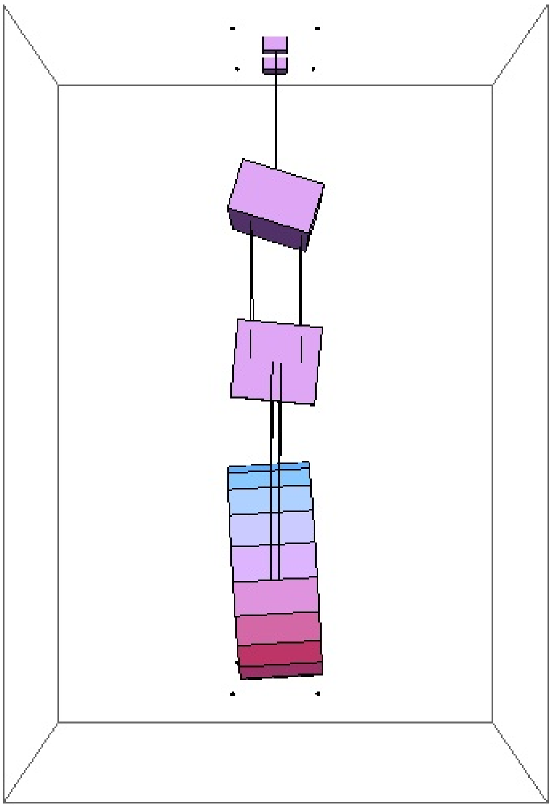


Mode #9 - modeP2

Back to Top

```
2.12717 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.00192117	0	0	0	-0.933946	0
Mass I	-0.00366319	0	0	0	-0.301692	0
optic	0.000947106	0	0	0	0.191596	0

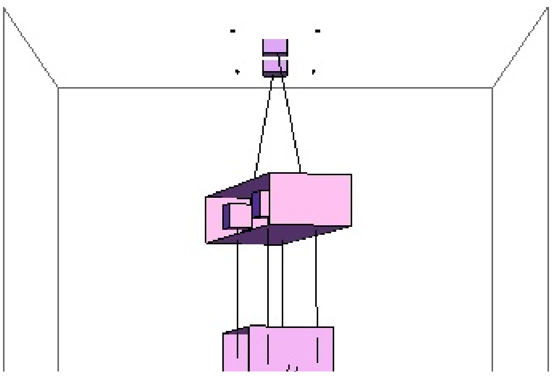


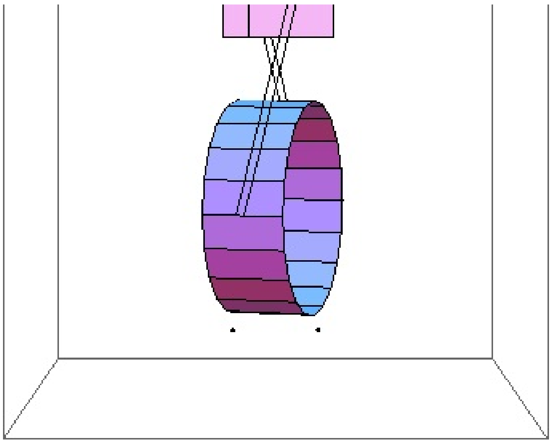
Mode #10 - modeY2

Back to Top

```
2.23634 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	-0.647503	0	0
Mass I	0	0	0	-0.338551	0	0
optic	0	0	0	0.682733	0	0



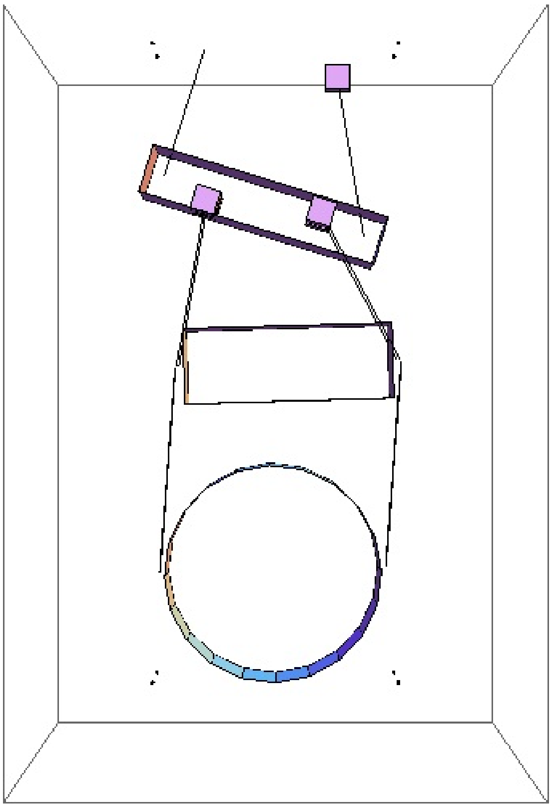


Mode #11 - modeT2

Back to Top

```
2.50427 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.047002	0	0	0	-0.988271
Mass I	0	0.0524729	0	0	0	0.0951857
optic	0	-0.00986026	0	0	0	0.0959227



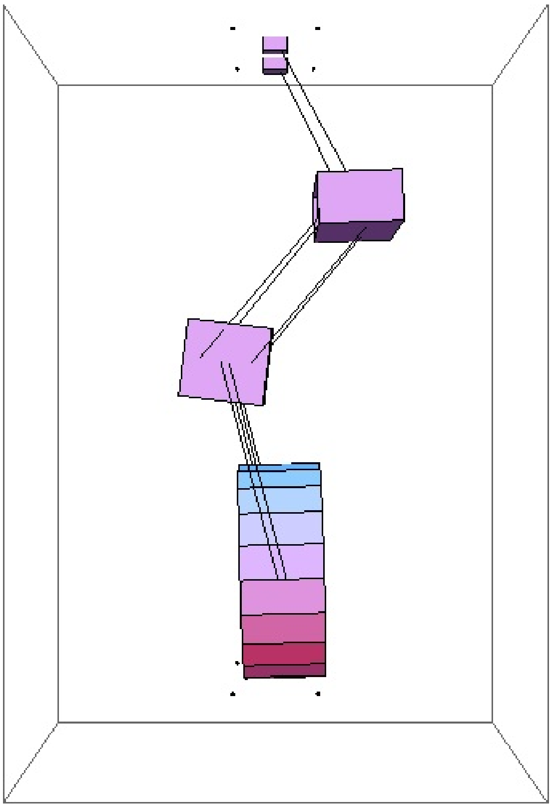
Mode #12 - modeL3

Back to Top

```
2.85421 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.574201	0	0	0	-0.212197	0
Mass I	-0.339828	0	0	0	0.69352	0

optic	0.0469087	0	0	0	-0.163131	0

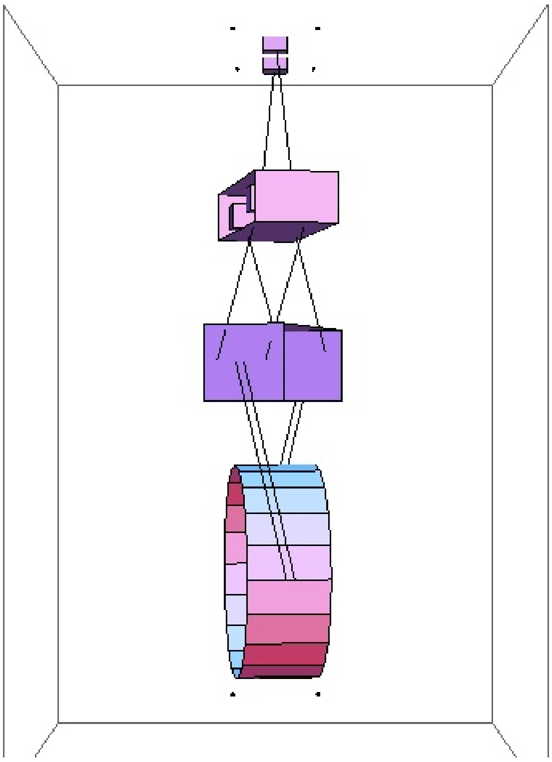


Mode #13 - modeY3

Back to Top

```
3.34348 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	-0.462262	0	0
Mass I	0	0	0	0.8156	0	0
optic	0	0	0	-0.34801	0	0

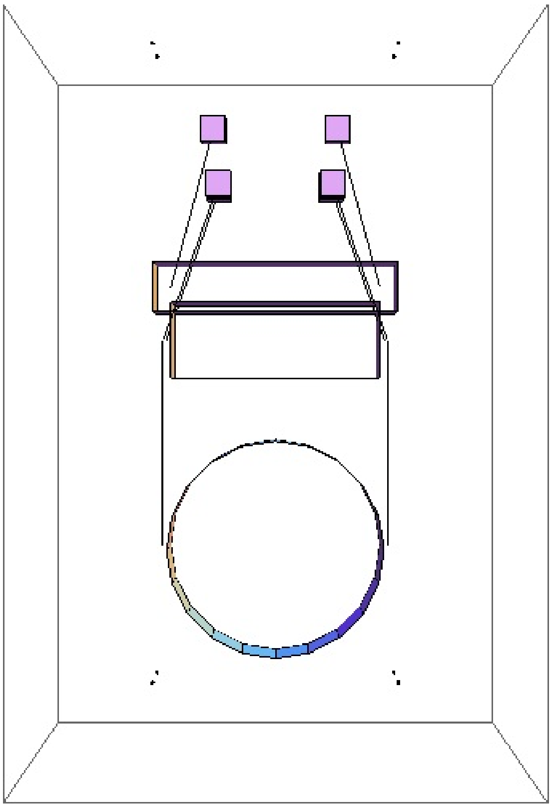


Mode #14 - modeV2

Back to Top

```
3.50334 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.924973	0	0	0
Mass I	0	0	0.264448	0	0	0
optic	0	0	0.272933	0	0	0

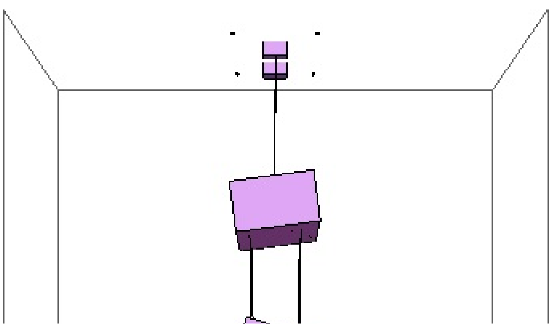


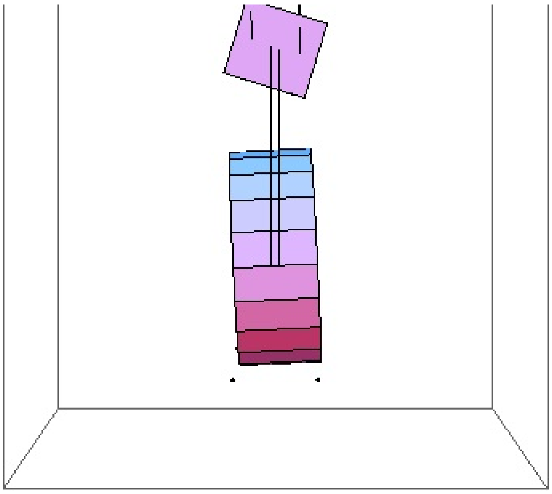
Mode #15 - modeP3

Back to Top

```
3.51862 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00188339	0	0	0	0.398527	0
Mass I	-0.000714155	0	0	0	-0.905318	0
optic	0	0	0	0	0.146873	0



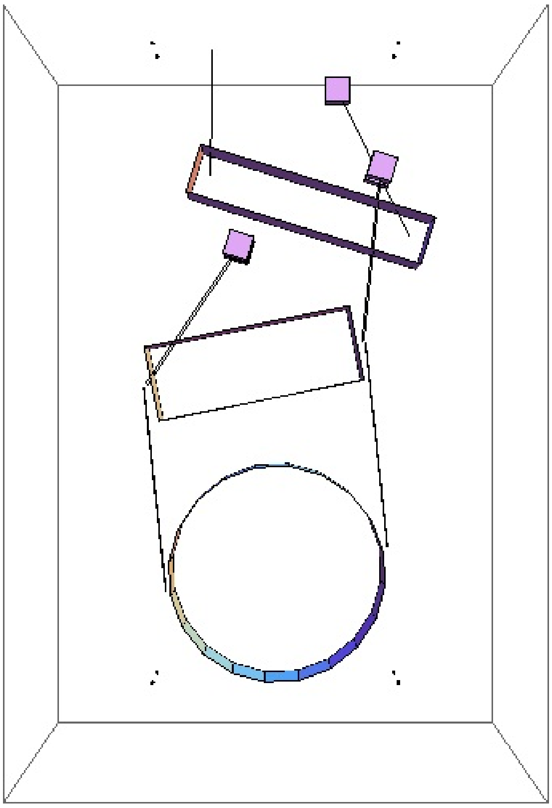


Mode #16 - modeT3

Back to Top

```
3.68116 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.104123	0	0	0	-0.721683
Mass I	0	-0.0623914	0	0	0	0.477695
optic	0	0.00471243	0	0	0	0.48603



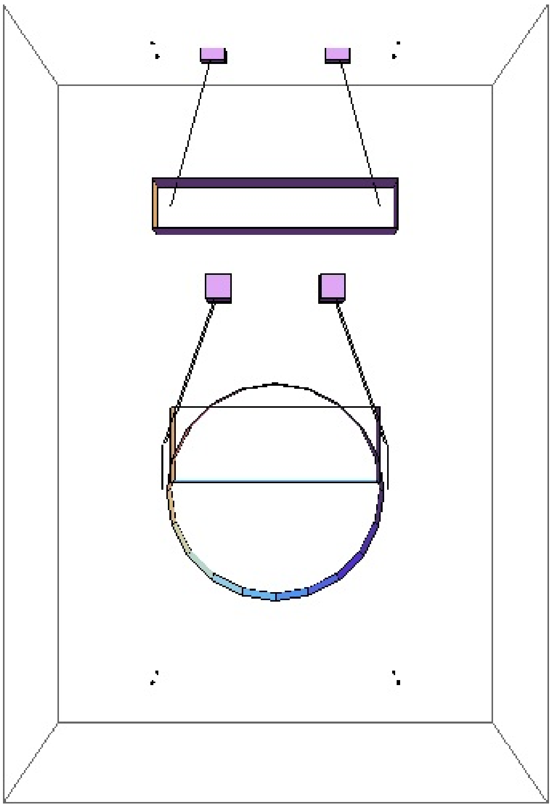
Mode #17 - modeV3

Back to Top

```
28.1003 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.0050742	0	0	0

Mass I	0	0	0.707132	0	0	0
optic	0	0	-0.707063	0	0	0

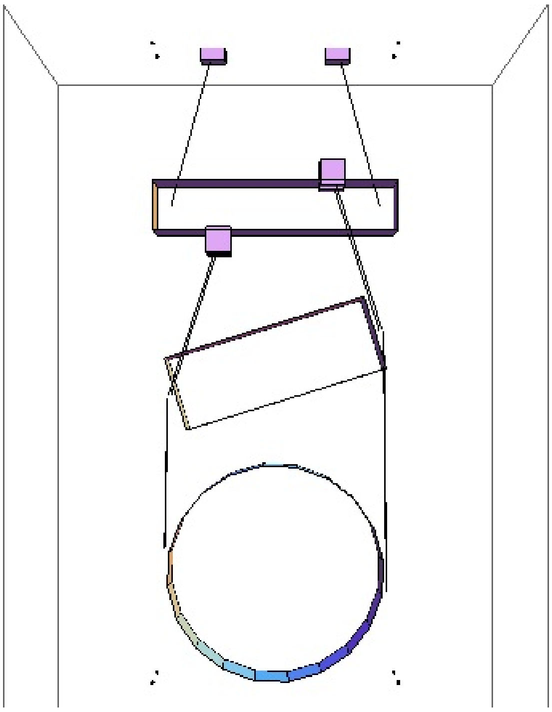


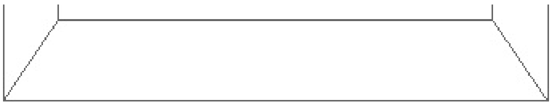
Mode #18 - modeR3

Back to Top

44.7247 Hz

	x	y	z	yaw	pitch	roll
Mass U	0	-0.000117055	0	0	0	0.00118095
Mass I	0	0.000115054	0	0	0	-0.83814
optic	0	0	0	0	0	0.545454





Parameters

Back to Top

```
g -> 9.81

ux -> 0.1                                                            display only

uy -> 0.3                                                            display only

uz -> 0.06                                                           display only

den1 -> 6715.                                                        average computed from m1, ux,
uy, uz

m1 -> 12.087                                                         SolidWorks model of D070335
with wire, blade clamps

I1x -> 0.127                                                         SolidWorks model of D070335
with wire, blade clamps

I1y -> 0.02                                                          SolidWorks model of D070335
with wire, blade clamps

I1z -> 0.129                                                         SolidWorks model of D070335
with wire, blade clamps

ix -> 0.1                                                            D070336-v3, display only

ir -> 0.09

den2 -> 7999.49                                                      300-series SSTL, SolidWorks -
LIGO material properties

m2 -> 12.227                                                         SolidWorks model of D070334
with wire clamps

I2x -> 0.0786                                                        SolidWorks model of D070334
with wire clamps

I2y -> 0.0201                                                        SolidWorks model of D070334
with wire clamps

I2z -> 0.079                                                         SolidWorks model of D070334
with wire clamps

tx -> 0.1

tr -> 0.1325

den3 -> 2698.79

m3 -> 12.142                                                         SolidWorks model of D070337
with metal prisms, magnets

I3x -> 0.12                                                          SolidWorks model of D070337

I3y -> 0.0699                                                        SolidWorks model of D070337

I3z -> 0.0707                                                        SolidWorks model of D070337

l1 -> 0.2025                                                         D070340

l2 -> 0.2036                                                         D070393

l3 -> 0.255                                                          D070436

nw1 -> 2

nw2 -> 4

nw3 -> 4

r1 -> 0.0003048                                                      0.024" diameter
```

```
r2 -> 0.00017018                                                    0.0079" diameter

r3 -> 0.00013462                                                    0.047" in diameter

              11
Y1 -> 2.119 10

              11
Y2 -> 2.119 10

              11
Y3 -> 2.119 10

ufc1 -> 2.26                                                        Fit to PR3 Metal Data

ufc2 -> 2.36                                                        Fit to PR3 Metal Data

ufc3 -> 0

d0 -> 0.001                                                         D070447-v2 (Rounded to nearest
mm)

d1 -> 0.001                                                         D070447-v2 (Rounded to nearest
mm)

d2 -> 0.001                                                         D070447-v2 (Rounded to nearest
mm)

d3 -> 0.001                                                         D070447-v2 (Rounded to nearest
mm)

d4 -> 0.001                                                         D070447-v2 (Rounded to nearest
mm)

su -> 0                                                             D070447-v2

si -> 0.03                                                          D070447-v2

sl -> 0.005                                                         D070447-v2

n0 -> 0.077                                                         D070447-v2

n1 -> 0.13                                                          D070447-v2

n2 -> 0.0699                                                        D070447-v2

n3 -> 0.1375                                                        calculation from D070447-v2

n4 -> 0.1405                                                        D070447-v2

n5 -> 0.1405                                                        calculation from D070447-v2

tl1 -> 0.196441

tl2 -> 0.19405

tl3 -> 0.257

ltotal -> 0.647491

leverarmrt -> 0.03

leverarmrz -> 0.08

leverarmrl -> 0.08

gain -> 0.4

gainrtzrtl -> 0.4

gaint -> 0.8

gainlrz -> 0.4

b1 -> 0.03

b2 -> 0.03

b3 -> 0.03

b4 -> 0.03
```

```
b5 -> 0.03

b6 -> 0.03

unstretched -> False

vertblades -> True

ul1 -> 0.201893

ul2 -> 0.202931

ul3 -> 0.254371

sl1 -> 0.2025

sl2 -> 0.2036

sl3 -> 0.255

si1 -> 0.261728

si2 -> 0.332024

si3 -> 0.

c1 -> 0.965142

c2 -> 0.943271

c3 -> 1.

pitchbul -> 0

pitchbur -> 0

pitchbll -> 0

pitchblr -> 0

pitchbllf -> 0

pitchblrf -> 0

pitchbllb -> 0

pitchblrb -> 0

rollbul -> 0

rollbur -> 0

rollbll -> 0

rollblr -> 0

rollbllf -> 0

rollblrf -> 0

rollbllb -> 0

rollblrb -> 0

                -7
A1 -> 2.91864 10

                -8
A2 -> 9.09844 10

                -8
A3 -> 5.69337 10

kw1 -> 305412.

kw2 -> 94693.5

kw3 -> 47310.7

flex1 -> 0.00268734
```

```
flex2 -> 0.0014001

flex3 -> 0.00135481

kbuz -> 1218.61

kblz -> 672.115

bdu -> 0.146738

bdl -> 0.0889207

m13 -> 36.456

m23 -> 24.369

I1xy -> 0

I1yz -> 0

I1zx -> 0

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

Ibtxyl -> 0

Ibtyzl -> 0

Ibtzxl -> 0

I2xy -> 0

I2yz -> 0

I2zx -> 0

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0

I3yz -> 0

I3zx -> 0

COM3x -> 0

COM3y -> 0

COM3z -> 0

FRP3x -> 0

FRP3y -> 0

FRP3z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03
```

```
phib -> 0.001

bd -> 0.001

                 -15
M11 -> 6.77875 10

                 -15
M12 -> 6.77875 10

                 -16
M21 -> 6.58755 10

                 -16
M22 -> 6.58755 10

                 -16
M31 -> 2.57946 10

                 -16
M32 -> 2.57946 10

temperature -> 290.

                       -23
boltzmann -> 1.38066 10

                     -7
alphasilica -> 3.9 10                                        measurements by PW, AH - cf.
5.1 10^-7 from gwinc

betasilica -> 0.000152                                       gwinc/IFOModel v1.0

rhosilica -> 2200.                                           gwinc/IFOModel v1.0

Csilica -> 772.                                              gwinc/IFOModel v1.0

Ksilica -> 1.38                                              gwinc/IFOModel v1.0

                 10
Ysilica -> 7.2 10                                            spec sheet for silica

                   -10
phisilica -> 4.1 10                                          gwinc/IFOModel v1.0

                   -11
phissilica -> 3. 10                                          Phil Willems

rhosteel -> 7800.                                            gwinc/IFOModel v1.0

Csteel -> 486.                                               gwinc/IFOModel v1.0

Ksteel -> 49.                                                gwinc/IFOModel v1.0

                  11
Ysteel -> 2.119 10                                           measured by MB, 11/18/05

alphasteel -> 0.000012                                       gwinc/IFOModel v1.0

betasteel -> -0.00025                                        gwinc/IFOModel v1.0

phisteel -> 0.0002                                           gwinc/IFOModel v1.0 = Geppo's
value

rhomarag -> 7800.                                            gwinc/IFOModel v1.0

Cmarag -> 460.                                               gwinc/IFOModel v1.0

Kmarag -> 20.                                                gwinc/IFOModel v1.0

                 11
Ymarag -> 1.87 10                                            gwinc/IFOModel v1.0

alphamarag -> 0.000011                                       gwinc/IFOModel v1.0

betamarag -> -0.00025                                        Geppo's value - gwinc/IFOModel
v1.0 is wrong

phimarag -> 0.0001                                           gwinc/IFOModel v1.0

tmU -> 0.0025                                                upper blade thickness, NAR
8/4/06
```

```
8/4/06

tmL -> 0.0017                                                      lower blade thickness, NAR
8/4/06

magicnumber -> 0.0737472                                           Zener, 1938, Phys. Rev. 53:90-
99

deltabladeU -> 0.00182883                                          cf Bench delta_v1

deltabladeL -> 0.00182883                                          cf Bench delta_v3

deltawireU -> 0.00263479                                           cf Bench delta_h1

deltawireL -> 0.0026649                                            cf Bench delta_h3

deltafibre -> 0.00258057

taubladeU -> 0.113606

taubladeL -> 0.0525316

tauwireU -> 0.00212017                                             cf Bench tau_steel1

tauwireL -> 0.000660933                                            cf Bench tau_steel3

tausilica -> 0.00907956

                                       0.00130543 #1
damping[imag, bladeUtype] -> (0.0001 + ---------------- & )
                                                      2
                                      1 + 0.509525 #1

                                       0.000603633 #1
damping[imag, bladeLtype] -> (0.0001 + ---------------- & )
                                                      2
                                      1 + 0.108943 #1

damping[imag, wireUtype] -> (0.0002 & )

damping[imag, wireLtype] -> (0.0002 & )

                                       0.0000350991 #1
damping[imag, wireUatype] -> (0.0002 + ------------------ & )
                                                        2
                                      1 + 0.00017746 #1

                                        0.0000110667 #1
damping[imag, wireLatype] -> (0.0002 + -------------------- & )
                                                          2
                                      1 + 0.0000172455 #1

damping[imag, fibretype] -> (0.0002 & )

                                                    -6
                                        6.70587 10    #1
damping[imag, fibreatype] -> (0.0002 + -------------------- & )
                                                      -6   2
                                      1 + 6.75273 10    #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0
```

```
kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

lockedblades -> False                                        False for maximum realism.
True for compatibility with very old Matlab.

kw1usual -> 305412.

kw2usual -> 94693.5

kw3usual -> 47310.7

kbuzusual -> 1218.61

kblzusual -> 672.115

                  7
kbuy -> 1.21861 10

                  6
kbly -> 6.72115 10

kbux -> 121861.

kblx -> 67211.5

boxymassi -> True

iy -> 0.255                                                  D070336-v3, display only

iz -> 0.092                                                  D070336-v3, display only

nb -> 6061 Al, SolidWorks - LIGO material properties

nb -> D070338-v4

nb -> D070338-v4

dl -> 0.                                                     correction to wire length to
offset wedge mass imbalance

dssilica -> 0.015                                            gwinc/IFOModel v1.0

taufibre -> 0.00041358
```

Back to Top

aLIGO SUS Operation Manual - Info on HSTS Suspensions

Back to Operation Manual main page

References

- HSTS wiki page (old)
- T010103: aLIGO Suspension System Conceptual Design
- D020700: HSTS Overall Assembly
- T1000061: aLIGO HAM Triple Suspension Controls Design Description
- E1100109: HAM Suspensions Controls Arrangement Poster
- T1300079: HAM Small Triple Suspension (HSTS) Actuation Ranges
- D0902810: aLIGO HAM2 Suspension Controls Wiring Diagrams
- D1000599: aLIGO HAM 3-4 Suspension Controls Wiring Diagram
- D1002740: aLIGO SUS HAM 5-6 Wiring Diagrams

Models

The HSTS suspension has been extensively modelled. Key results are at Suspensions/OpsManual/HSTS/Models .

Screens

HSTS MEDM screens are at /Screens.

aLIGO: Suspensions/OpsManual/HSTS (last edited 2013-09-13 14:14:08 by MarkBarton)

aLIGO SUS Operations Manual - Overview of HSTS MEDM screens

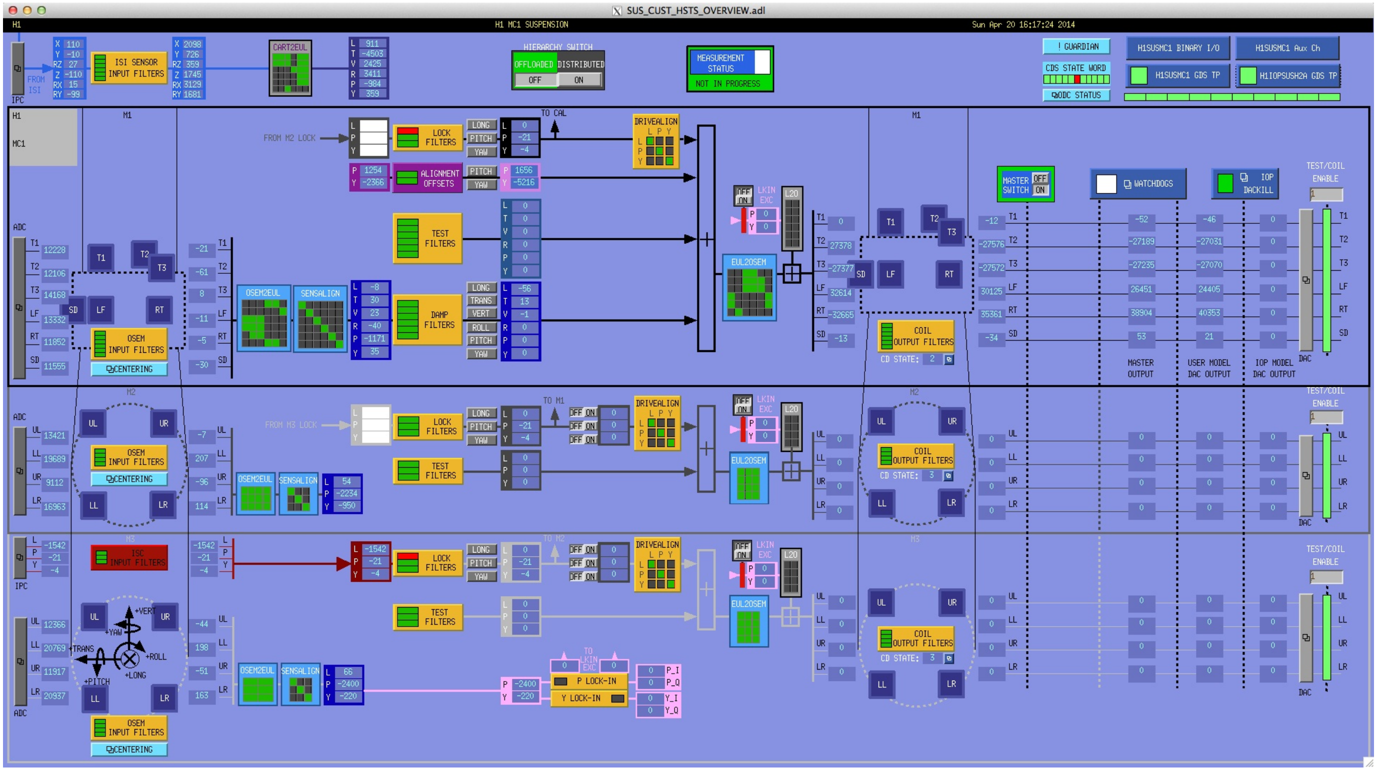
Back to Operation Manual main page
Back to HSTS main page

This page makes extensive use of text fragments in-lined from Suspensions/OpsManual/Boilerplate. Use the extract.py script there to update.

Except where noted, the HSTS screens described below are common to HSTS and HLTS and live at `/opt/rtcds/userapps/release/sus/common/medm/hxts/`. (Note `hxts` not `hsts`.) They are all generic screens which rely for their execution on site-, interferometer- and suspension-specific arguments passed in when they are called. They can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_HSTS_OVERVIEW.adl

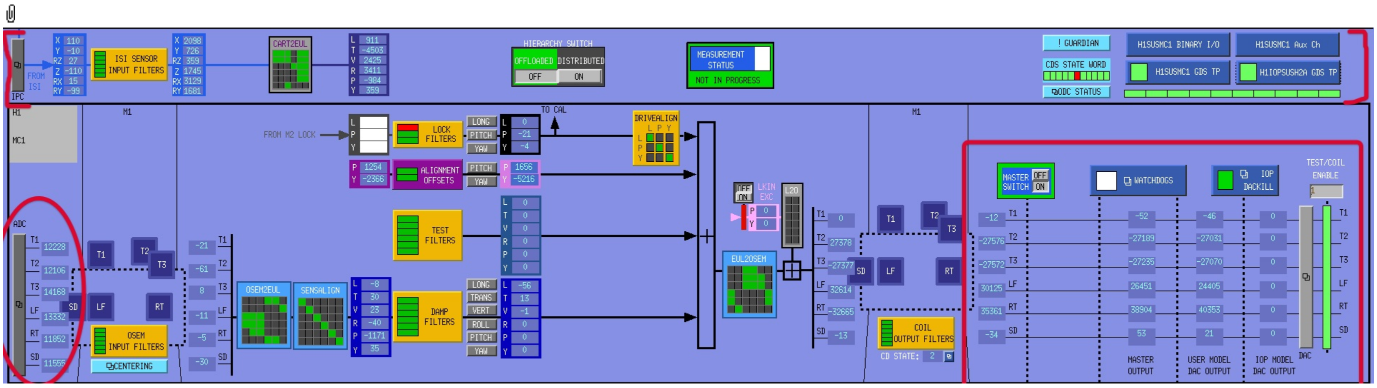


This is the overview screen and one of only a few specific to HSTS (with HSTS rather than HXTS in the name). The functionality is divided up according to the three sensor-actuator groups, plus an odds-and-ends area at the top:

- Other - subscreens not specific to a sensor/actuator group
- M1 - 6 BOSEMs on the structure engaging the main chain top mass
- M2 - 4 BOSEMs on the structure engaging the intermediate mass
- M3 - 4 AOSEMs on the structure engaging the optic

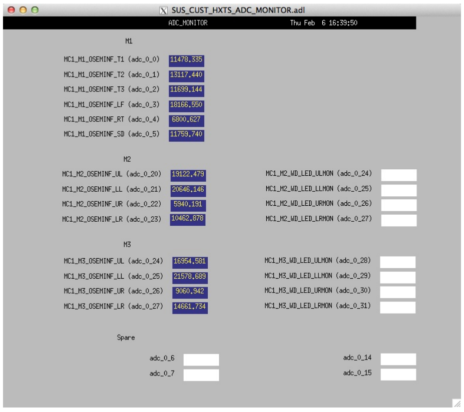
M1 is used for local damping (relative to the structure). DC pitch and yaw offsets are also injected at M1. M2 and M3 are for implementing global control (relative to other optics) and have inputs for actuation requests from ISC.

Other Screens



- MASTER SWITCH - switches all actuation.
- MEASUREMENT STATUS - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- HIERARCHY SWITCH - enables various ISC inputs and outputs.
- ADC - a monitor screen (common to all sensor-actuator groups) for the semi-raw ADC values as reported by the IOP process after downsampling.
- DAC - a popup allowing access to all the DAC outputs as reported by the IOP process.
- H1SUSMC1 MONITORS (or the like; depends on the suspension name) - readbacks from coil driver channels for all OSEMs as reported by the corresponding auxiliary front-end processor.
- WATCHDOGS - a block implementing the watchdogs on the various sensor-actuator groups.
- IOP DACKILL - a watchdog that shuts off the IOP process (potentially other suspensions on the same front-end).
- H1IOPSUSH2A GDS TP (or the like; depends on the front-end computer name) - status of the IOP process for the front-end.
- H1SUSMC1 GDS TP (or the like; depends on the suspension name) - status of the suspension process.
- H1SUSMC1 BIO (or the like; depends on the suspension name) - binary input/output controls.
- ODC STATUS - ODC status
- !GUARDIAN Guardian controls
- ISIINF - ISI input filters.
- CART2EUL - transformation from the ISI basis to SUS.

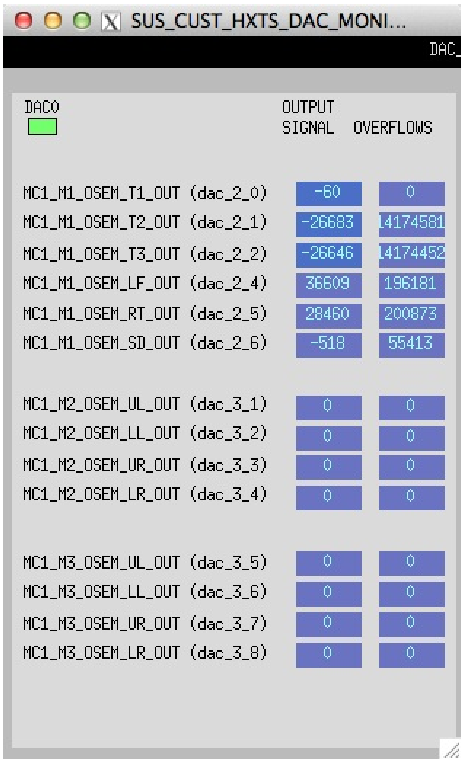
Screen SUS_CUST_HXTS_ADC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

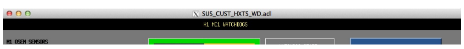
Screen SUS_CUST_HXTS_DAC_MONITOR.adl

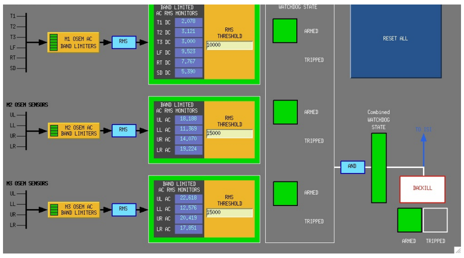


Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

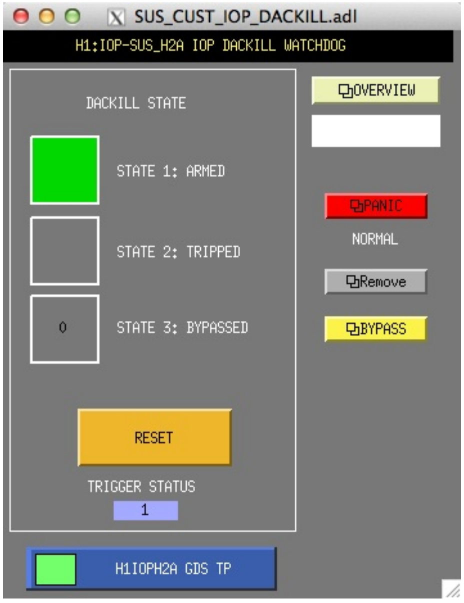
Screen SUS_CUST_HXTS_WD.adl




Suspensions/OpsManual/Boilerplate/WD:

This screen allows for setting the watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

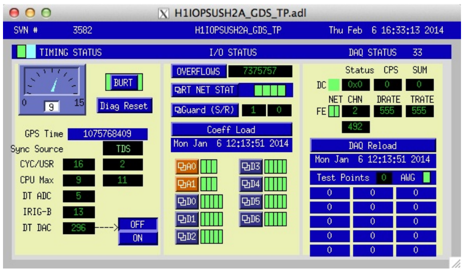
Screen SUS_CUST_IOP_DACKILL.adl



Suspensions/OpsManual/Boilerplate/IOP_DACKILL:

The watchdog controls for the IOP process. If this WD is tripped it disables IO to/from all user models on the front end. RESET resets from the tripped state if there are no outstanding triggers. BYPASS is a manual override that allows for operation of the suspension if there is a trigger condition that is known by the operator to be harmless.

Screen H1IOPSUSH2A_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

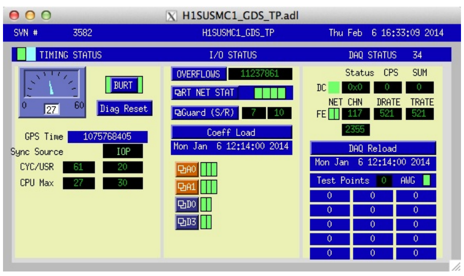
This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1SUSMC1_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

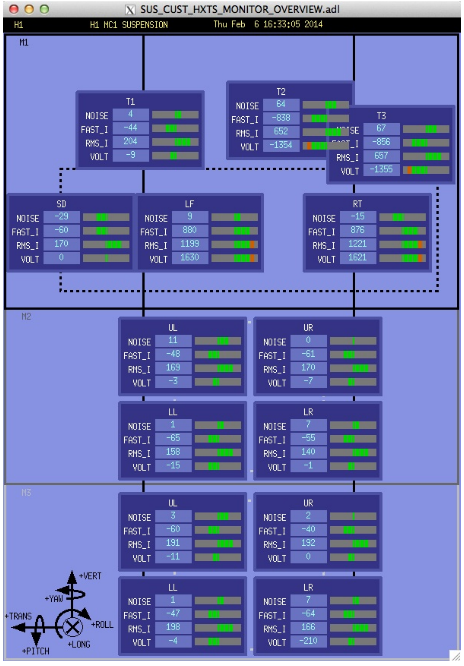
This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen SUS_CUST_HXTS_MONITOR_OVERVIEW.adl

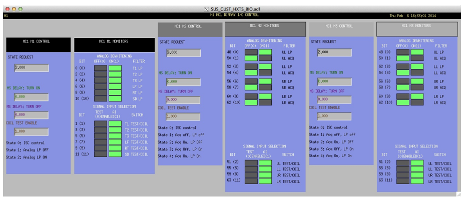


Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

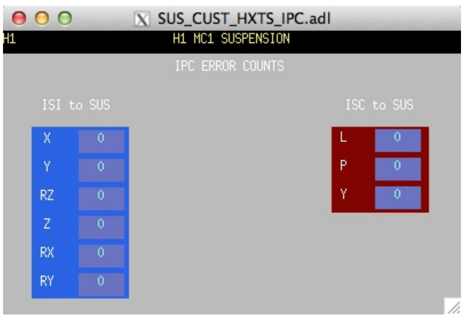
Screen SUS_CUST_HXTS_BIO.adl



Suspensions/OpsManual/Boilerplate/HSTS_BIO:

The Binary Input-Output control screen. State Requests for the various sensor-actuator groups can be input. Positive state values lock out manual control of the filters on the COILOUTF screen and set the digital dewhitening filters to match the analog whitening filters selected via the binary outputs. Negative state values allow the dewhitening filters to be manually overridden.

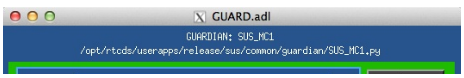
Screen SUS_CUST_HXTS_IPC.adl



Suspensions/OpsManual/Boilerplate/IPC:

Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

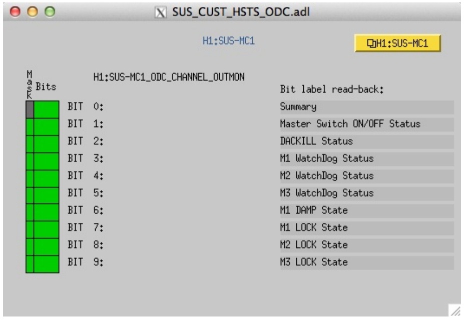
Screen GUARD.adl



Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

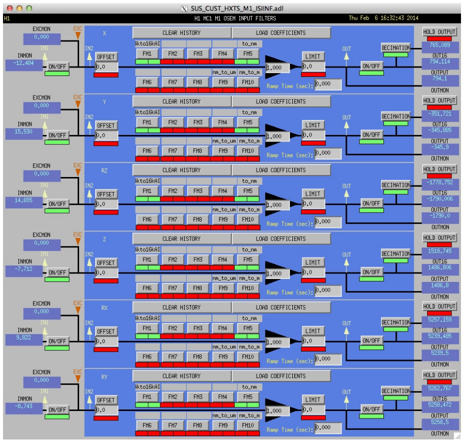
Screen SUS_CUST_HSTS_ODC.adl



Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

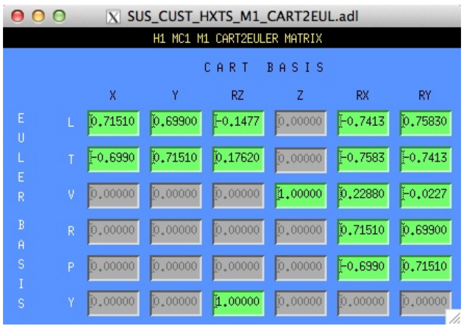
Screen SUS_CUST_HXTS_M1_ISIINF.adl



Suspensions/OpsManual/Boilerplate/M1_ISIINF:

Filters for the diagnostic signals from the ISI.

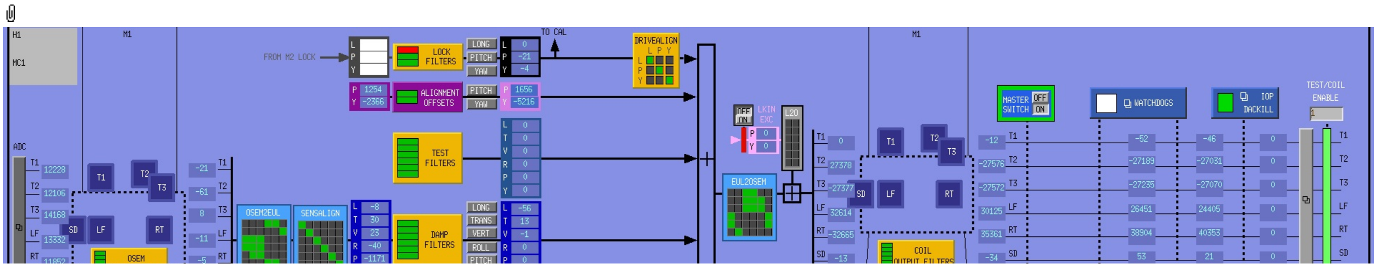
Screen SUS_CUST_HXTS_M1_CART2EUL.adl



Suspensions/OpsManual/Boilerplate/M1_CART2EUL:

A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

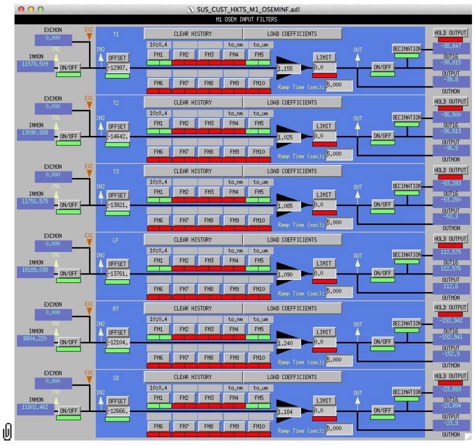
Sensor Actuator Group M1 - Upper Mass



The M1 section of the main HLTS screen is excerpted above. Most of the items on it correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- DAMP FILTERS - a filter bank applying the damping filters.
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.
- LOCK FILTERS - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

Screen SUS_CUST_HXTS_M1_OSEMINF.adl



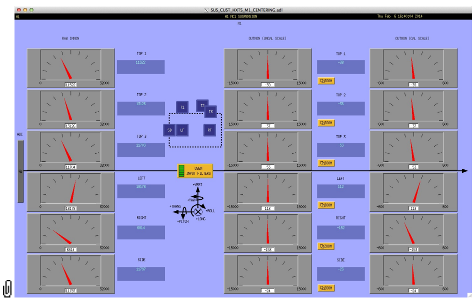
Suspensions/OpsManual/Boilerplate/M1_OSEMINF:

This block has 6 filter groups corresponding to the 6 M1 BOSEMs, T1/T2/T3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

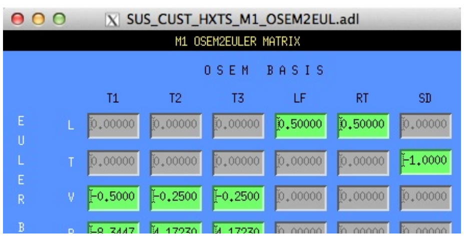
Screen SUS_CUST_HXTS_M1_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M1_CENTERING:

This screen gives various views of the M1 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_HXTS_M1_OSEM2EUL.adl



Suspensions/OpsManual/Boilerplate/M1_OSEM2EUL:

This screen allows entry of the matrix which converts from the M1 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

Screen SUS_CUST_HXTS_M1_SENSALIGN.adl

Suspensions/OpsManual/Boilerplate/M1_SENSALIGN:

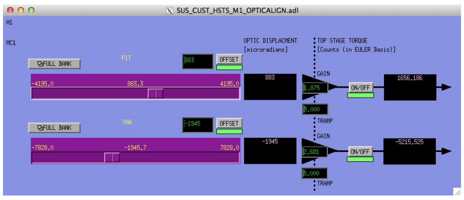
This screen is reserved for tweaking the M1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_HXTS_M1_DAMP.adl

Suspensions/OpsManual/Boilerplate/M1_DAMP:

These filters implement the local damping. See Suspensions/OpsManual/Damping.

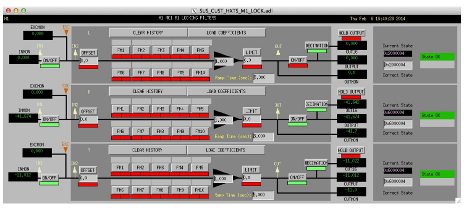
Screen SUS_CUST_HSTS_M1_OPTICALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets. See Suspensions/OpsManual/Alignments for more info.

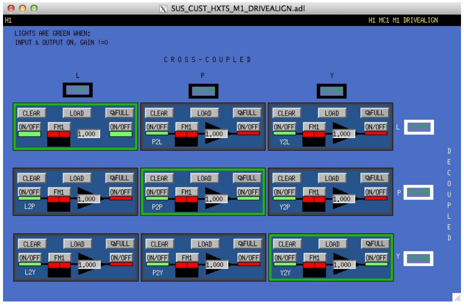
Screen SUS_CUST_HXTS_M1_LOCK.adl



Suspensions/OpsManual/Boilerplate/M1_LOCK:

Filters for the locking signals.

Screen SUS_CUST_HXTS_M1_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_DRIVEALIGN:

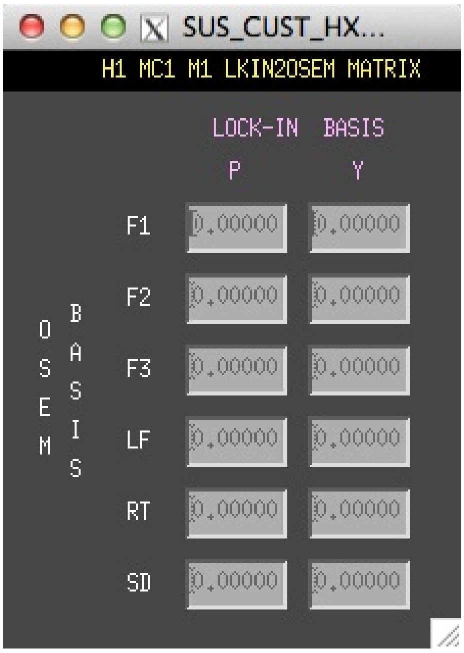
This screen is reserved for tweaking the M1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_HXTS_M1_EUL2OSEM.adl

Suspensions/OpsManual/Boilerplate/M1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

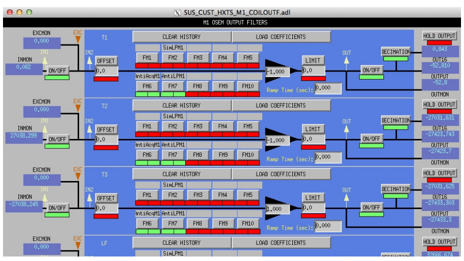
Screen SUS_CUST_HXTS_M1_LKIN2OSEM.adl

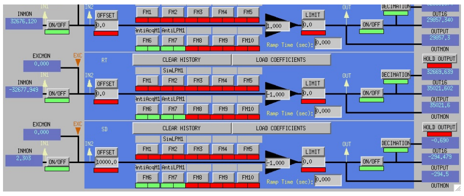


Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_HXTS_M1_COILOUTF.adl





Suspensions/OpsManual/Boilerplate/M1_COILOUTF:

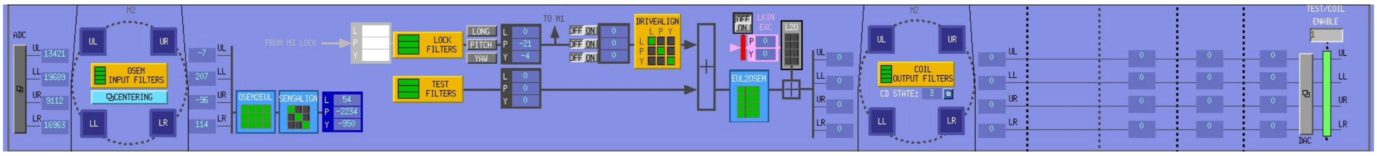
This screen applies compensation for the hardware filters in the M1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

Sensor Actuator Group M2 - Intermediate Mass

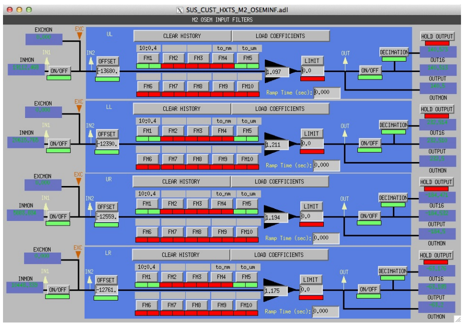


The M2 level has the following blocks:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/pitch/yaw (L/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- LOCK FILTERS - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

There are the following auxiliary inputs:

Screen SUS_CUST_HXTS_M2_OSEMINF.adl



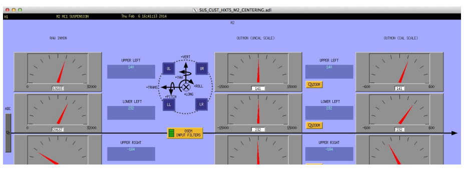
Suspensions/OpsManual/Boilerplate/M2_OSEMINF:

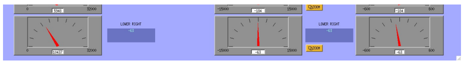
This block has 4 filter banks corresponding to the 4 M2 OSEMs, UL/LL/UR/LR. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_HXTS_M2_CENTERING.adl

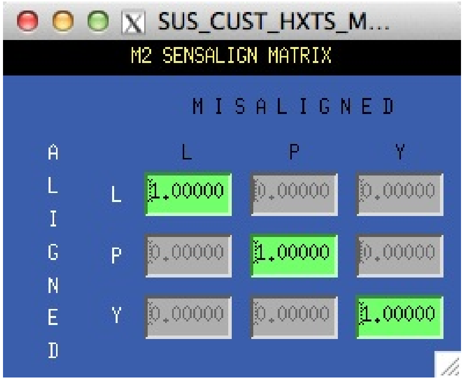




Suspensions/OpsManual/Boilerplate/M2_CENTERING:

This screen gives various views of the OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_HXTS_M2_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M2_SENSALIGN:

This screen is reserved for tweaking the M2 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_HXTS_M2_OSEM2EUL.adl

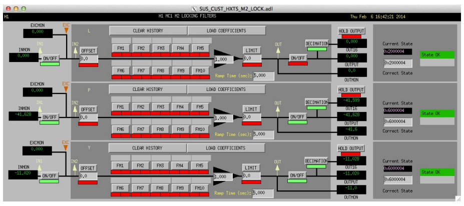
Suspensions/OpsManual/Boilerplate/M2_OSEM2EUL:

This screen allows entry of the matrix which converts from the OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOF L are dimensionless.

The entries for the angular DOFs P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

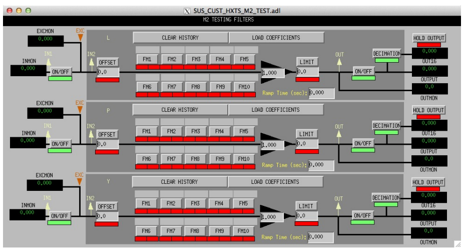
Screen SUS_CUST_HXTS_M2_LOCK.adl



Suspensions/OpsManual/Boilerplate/M2_LOCK:

Filters for the locking signals.

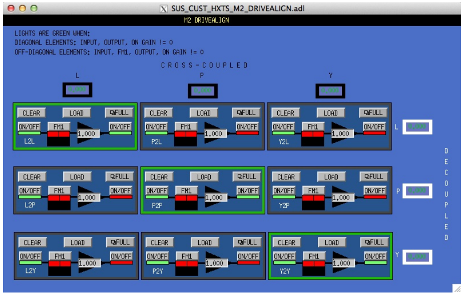
Screen SUS_CUST_HXTS_M2_TEST.adl



Suspensions/OpsManual/Boilerplate/M2_TEST:

This is a deliberately empty set of 3 filter banks corresponding to the M2 L, P and Y DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

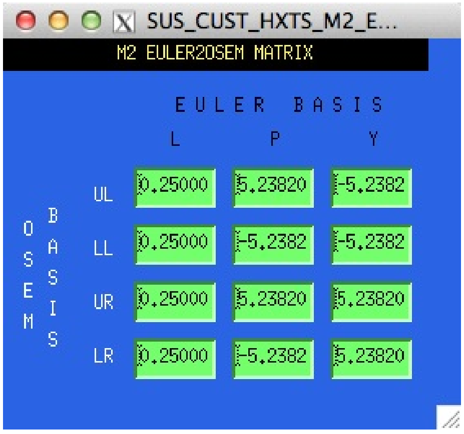
Screen SUS_CUST_HXTS_M2_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M2_DRIVEALIGN:

This screen is reserved for tweaking the actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

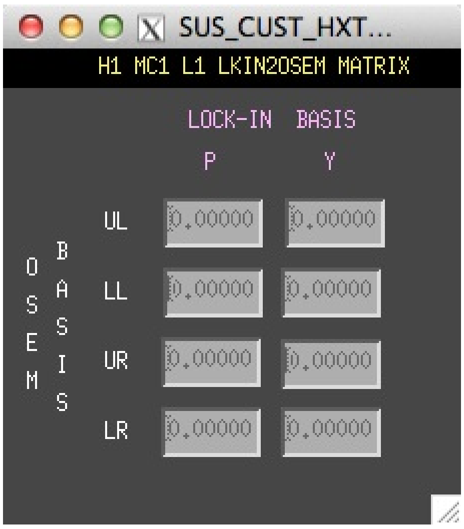
Screen SUS_CUST_HXTS_M2_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M2_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

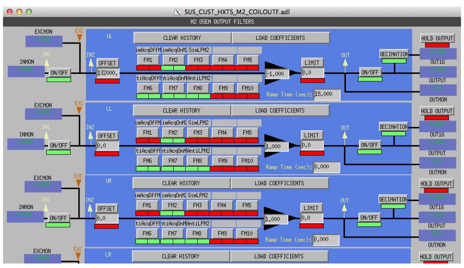
Screen SUS_CUST_HXTS_M2_LKIN2OSEM.adl

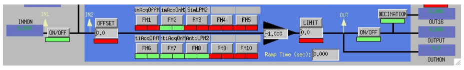


Suspensions/OpsManual/Boilerplate/M2_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_HXTS_M2_COILOUTF.adl





Suspensions/OpsManual/Boilerplate/M2_COILOUTF:

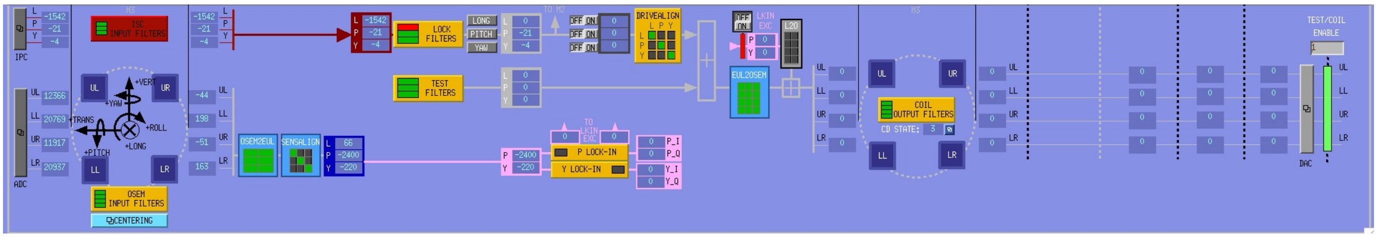
This screen applies compensation for the hardware filters in the actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

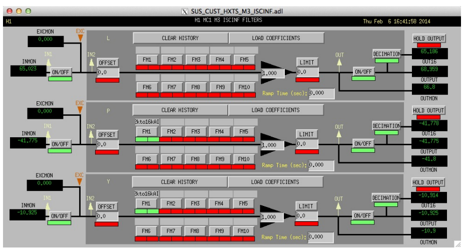
Sensor Actuator Group M3 - Optic



The M3 level has the following blocks in the main path:

- ISC INPUT FILTERS - ISC input filters.
- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to ±15000 counts.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/pitch/yaw (L/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- LOCK FILTERS - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

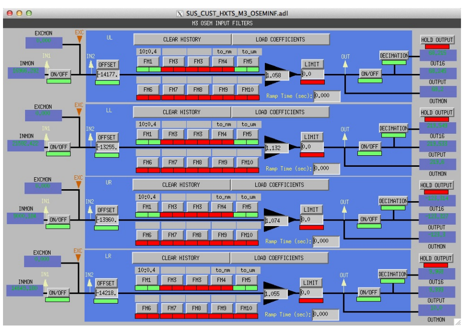
Screen SUS_CUST_HXTS_M3_ISCINF.adl



Suspensions/OpsManual/Boilerplate/M3_ISCINF:

Filters for the control signals from ISC.

Screen SUS_CUST_HXTS_M3_OSEMINF.adl



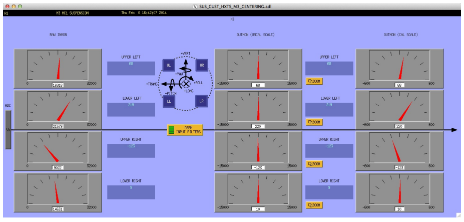
Suspensions/OpsManual/Boilerplate/M3_OSEMINF:

This block has 4 filter banks corresponding to the 4 M3 AOSEMs, UL/LL/UR/LR. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_HXTS_M3_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M3_CENTERING:

This screen gives various views of the OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_HXTS_M3_OSEM2EUL.adl

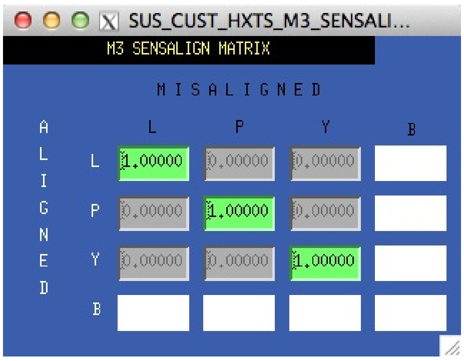
Suspensions/OpsManual/Boilerplate/M3_OSEM2EUL:

This screen allows entry of the matrix which converts from the OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOF L are dimensionless.

The entries for the angular DOFs P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

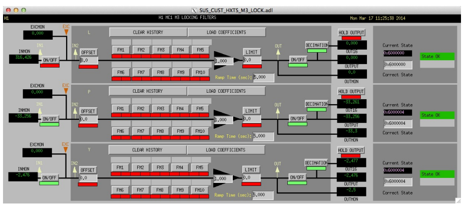
Screen SUS_CUST_HXTS_M3_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M3_SENSALIGN:

This screen is reserved for tweaking the M3 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

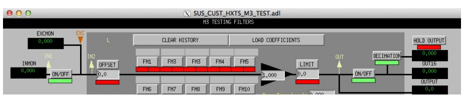
Screen SUS_CUST_HXTS_M3_LOCK.adl

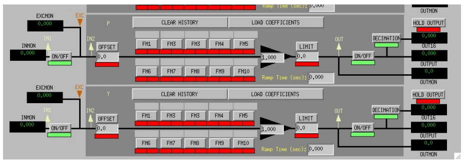


Suspensions/OpsManual/Boilerplate/M3_LOCK:

Filters for the locking signals.

Screen SUS_CUST_HXTS_M3_TEST.adl

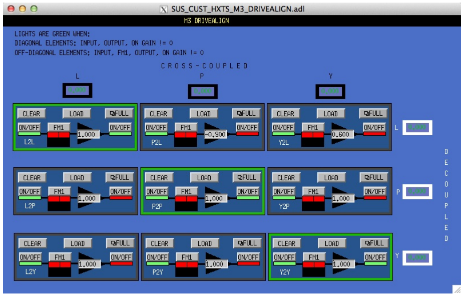




Suspensions/OpsManual/Boilerplate/M3_TEST:

This is a deliberately empty set of 3 filter banks corresponding to the L, P and Y DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

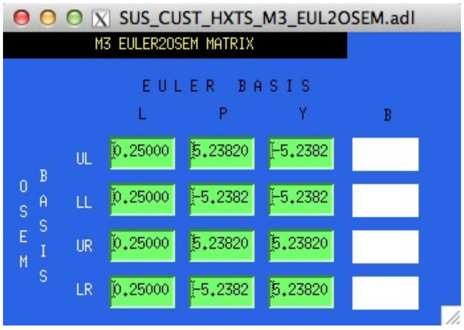
Screen SUS_CUST_HXTS_M3_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M3_DRIVEALIGN:

This screen is reserved for tweaking the M3 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

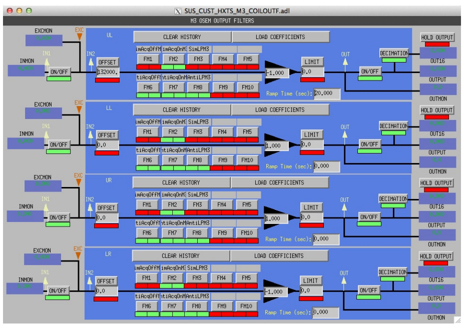
Screen SUS_CUST_HXTS_M3_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M3_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_HXTS_M3_COILOUTF.adl



Suspensions/OpsManual/Boilerplate/M3_COILOUTF:

This screen applies compensation for the hardware filters in the actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

aLIGO SUS Operations Manual - Models for HSTS Suspensions

Back to Operation Manual main page

The HSTS suspensions have been extensively modelled. There are Mathematica (T020205) and Matlab (T080188) programs that take a parameter set describing a single chain and calculate the dynamics. The parameter set needs to be formatted differently for Mathematica vs Matlab, but the parameter names are the same as far as was practical.

The Mathematica triple model, which covers HSTS as well as HLTS and BSFM, and parameter sets for it lives in the SUS SVN at `^/trunk/Common/MathematicaModels/TripleLite2/`. Parameter sets for Matlab live at `^/trunk/Common/MatlabTools/TripleModel_Production`. Mark Barton maintains the Mathematica, Jeff Kissel maintains the Matlab. Unfortunately two different naming conventions are in use and it's difficult to tell what corresponds to what - if in doubt, ask. References given below are to Mathematica versions as subdirectories of `^/trunk/Common/MathematicaModels/TripleLite2/`.

Key cases of the triple model for HSTS are given below, with results such as mode frequencies and mode shapes. T1200404 has transfer functions for many of the same models.

Generic HSTS

Current best generic HSTS model is .../mark.barton/20120120hsts.

aLIGO: Suspensions/OpsManual/HSTS/Models (last edited 2014-05-27 09:12:14 by MarkBarton)

Case 20120120hsts of Mathematica model TripleLite2

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
{"mark.barton", "20120120hsts"}
```

```
Equivalent to Jeff K's hstsopt_metal.m revision 2007 of 1/19/12.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeL1	modeT1	modeV1	modeP1	modeY1	modeR1
modeL2	modeT2	modeY2	modeR2	modeV2	modeL3
modeT3	modeP2	modeY3	modeP3	modeV3	modeR3

Mode Summary

Back to Top

```
N    f          name     type

1    0.672497   modeL1   pitch3   pitch2

2    0.675834   modeT1   roll3    roll2

3    0.848391   modeV1   z3       z2

4    1.0051     modeP1   pitch3   pitch2

5    1.09179    modeY1   yaw3     yaw2

6    1.51197    modeR1   roll3    roll2

7    1.51559    modeL2   pitch1   x2

8    1.52673    modeT2   roll3    roll2

9    2.0381     modeY2   yaw3     yaw1

10   2.18447    modeR2   roll1

11   2.76171    modeV2   z1

12   2.80671    modeL3   pitch3   x1       x2

13   2.98169    modeT3   roll3    roll2

14   3.20926    modeP2   pitch1   pitch3

15   3.42401    modeY3   yaw2

16   3.78136    modeP3   pitch1

17   27.3201    modeV3   z3       z2

18   40.369     modeR3   roll3    roll2
```

Violin Modes	UM	IM	Optic
f (Hz), n=1-	409.941 820.413 1231.95	744.632 1490.12 2237.33	650.585 1301.29 1952.23

4	1645.09	2987.11	2603.54
Q, n=1-4	58337.5 80281.6 91684. 93427.4	51727.9 59199.6 66273.1 68426.4	201248. 135720. 119571. 114931.

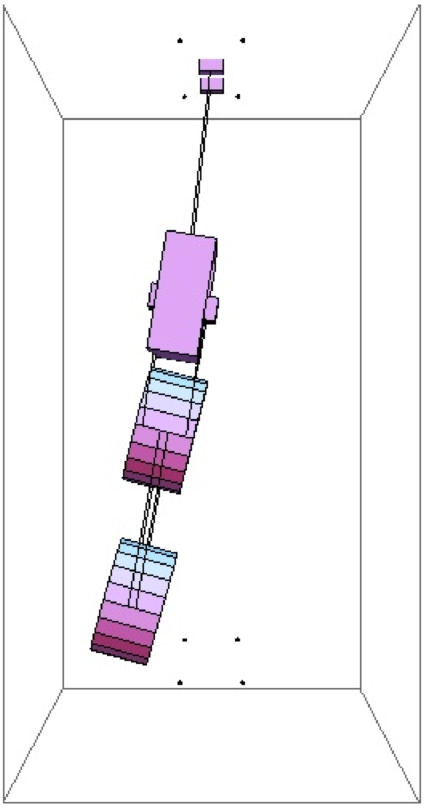
Mode Shapes

Mode #1 - modeL1

Back to Top

```
0.672497 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.082953	0	0	0	0.390667	0
Mass I	-0.14013	0	0	0	0.593636	0
optic	-0.23611	0	0	0	0.642425	0

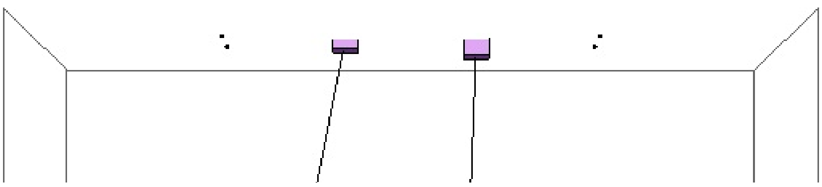


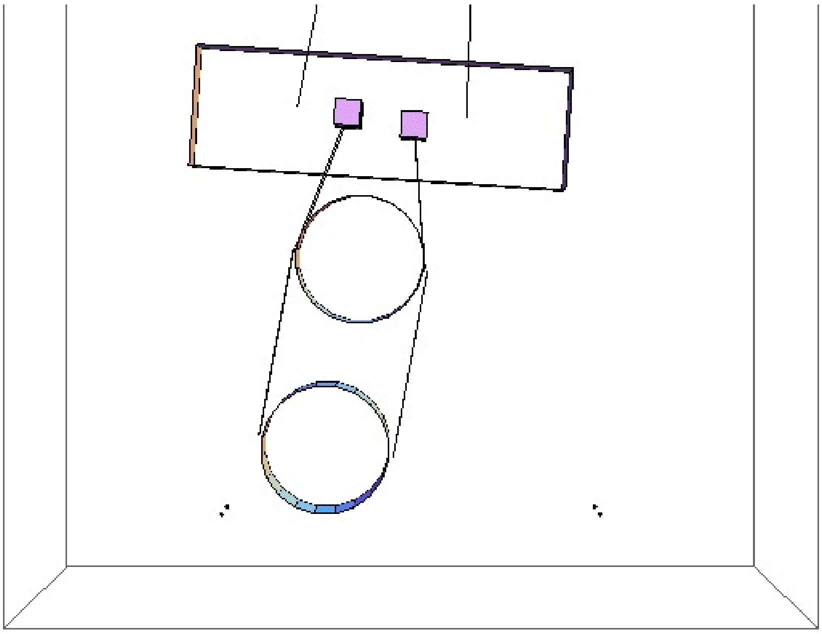
Mode #2 - modeT1

Back to Top

```
0.675834 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.127606	0	0	0	-0.239276
Mass I	0	-0.217712	0	0	0	-0.609766
optic	0	-0.367451	0	0	0	-0.610107



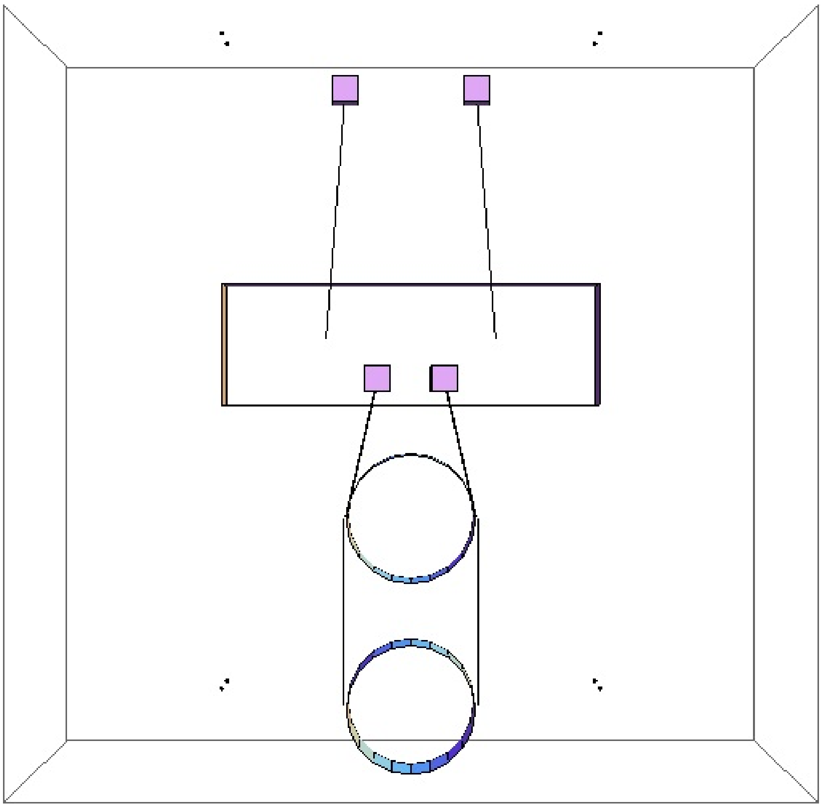


Mode #3 - modeV1

Back to Top

```
0.848391 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.400198	0	0	0
Mass I	0	0	-0.647395	0	0	0
optic	0	0	-0.64863	0	0	0

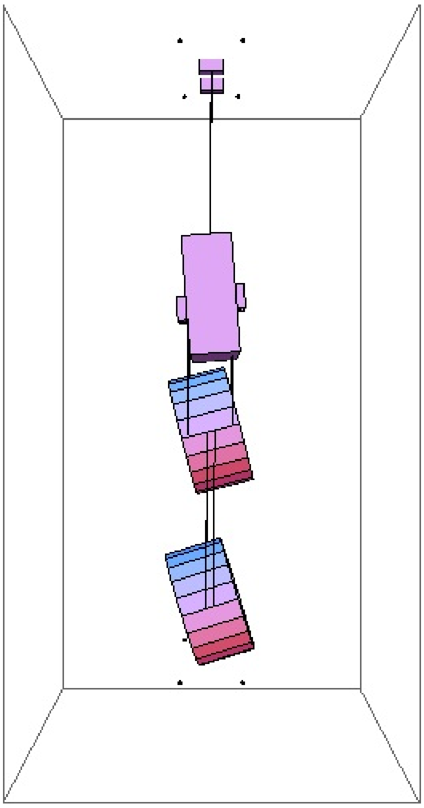


Mode #4 - modeP1

Back to Top

```
1.0051 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.00215427	0	0	0	-0.188765	0
Mass I	-0.0026432	0	0	0	-0.632475	0
optic	-0.00457764	0	0	0	-0.751207	0

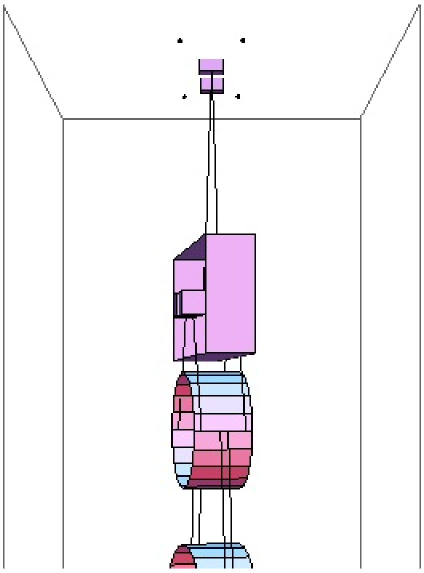


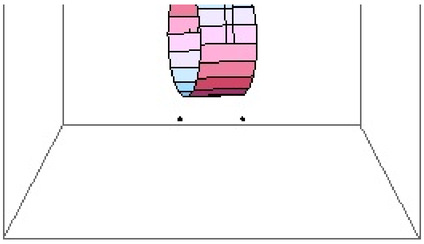
Mode #5 - modeY1

Back to Top

```
1.09179 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	-0.278214	0	0
Mass I	0	0	0	-0.542756	0	0
optic	0	0	0	-0.792472	0	0



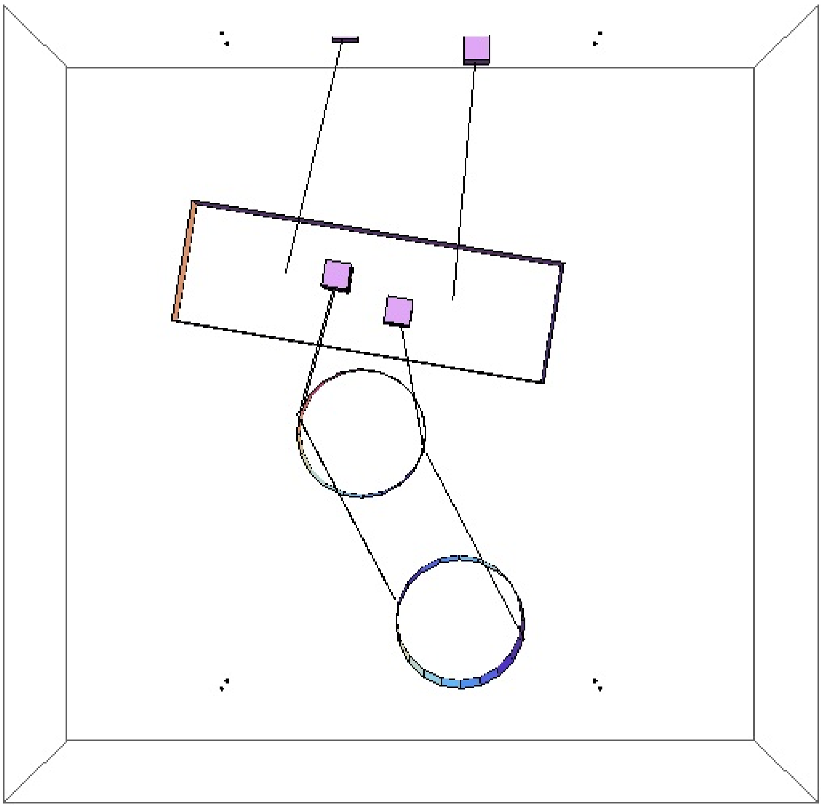


Mode #6 - modeR1

Back to Top

```
1.51197 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.107399	0	0	0	0.359908
Mass I	0	0.123736	0	0	0	0.642788
optic	0	-0.124003	0	0	0	0.644257

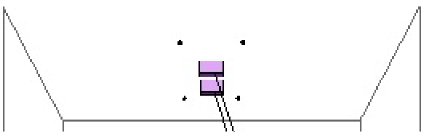


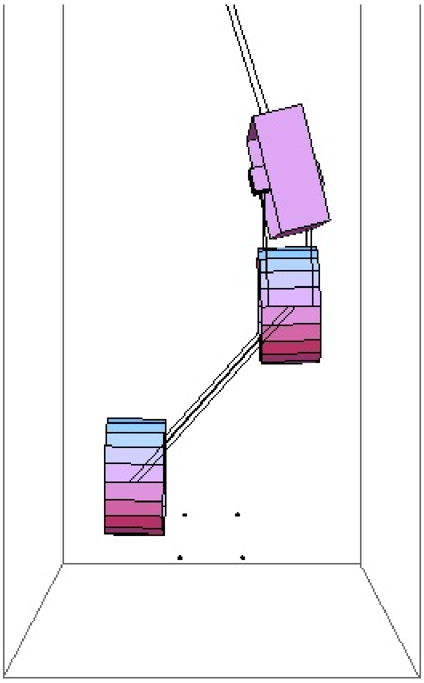
Mode #7 - modeL2

Back to Top

```
1.51559 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.32395	0	0	0	-0.810535	0
Mass I	0.340372	0	0	0	-0.091548	0
optic	-0.332592	0	0	0	0.0569024	0



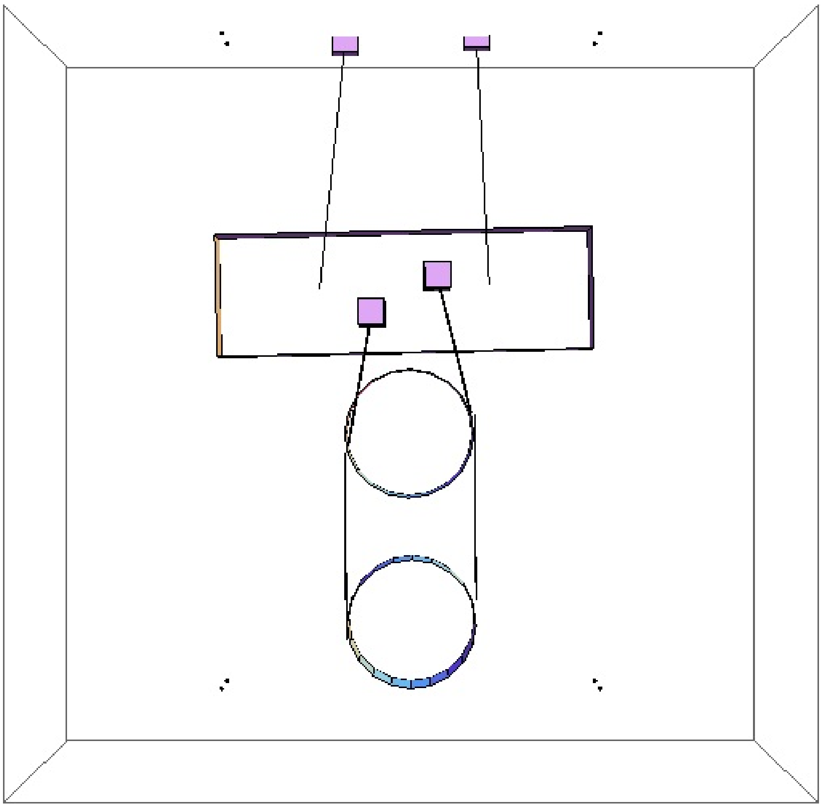


Mode #8 - modeT2

Back to Top

```
1.52673 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.0176642	0	0	0	-0.052556
Mass I	0	0.00430684	0	0	0	-0.705085
optic	0	-0.00201584	0	0	0	-0.706936

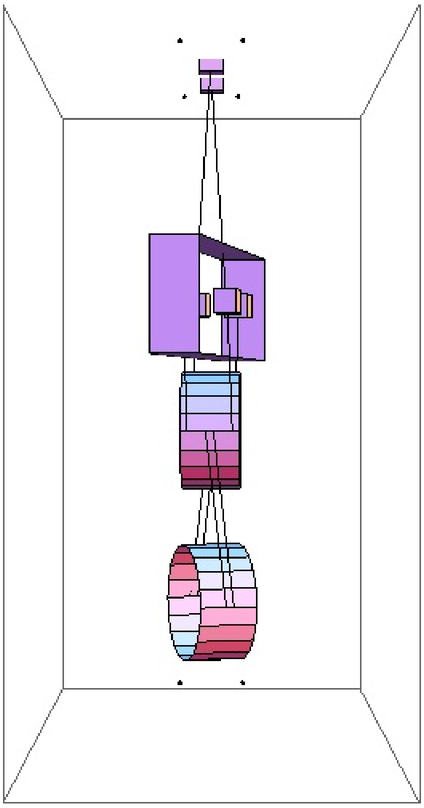


Mode #9 - modeY2

Back to Top

```
2.0381 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	0.559492	0	0
Mass I	0	0	0	0.0809058	0	0
optic	0	0	0	-0.824878	0	0

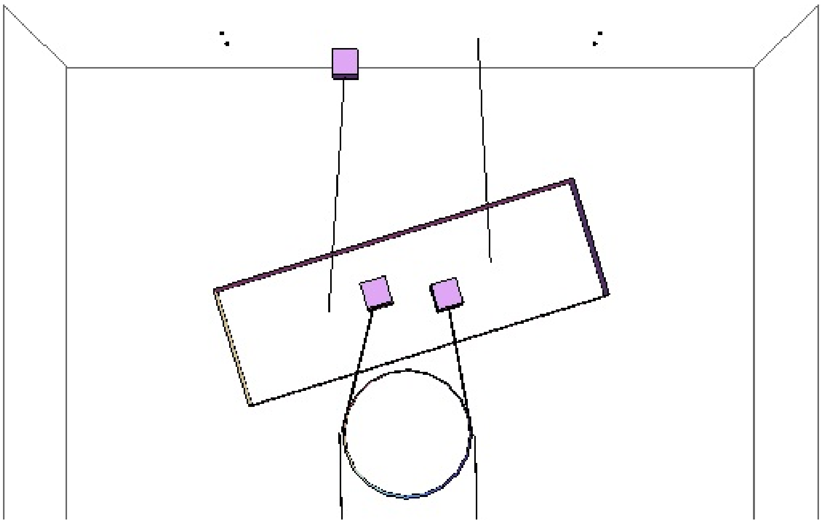


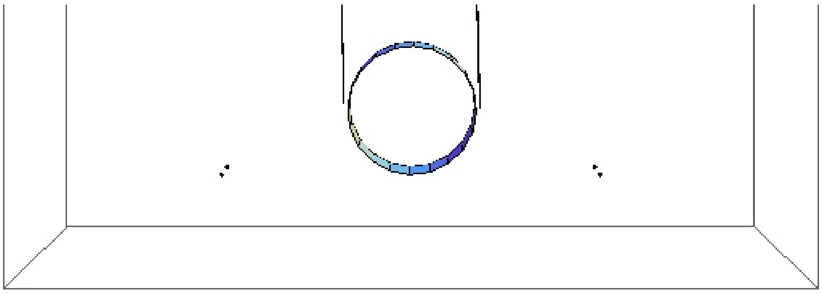
Mode #10 - modeR2

Back to Top

```
2.18447 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.000981251	0	0	0	0.989093
Mass I	0	-0.0139883	0	0	0	-0.103348
optic	0	0.00446618	0	0	0	-0.103908



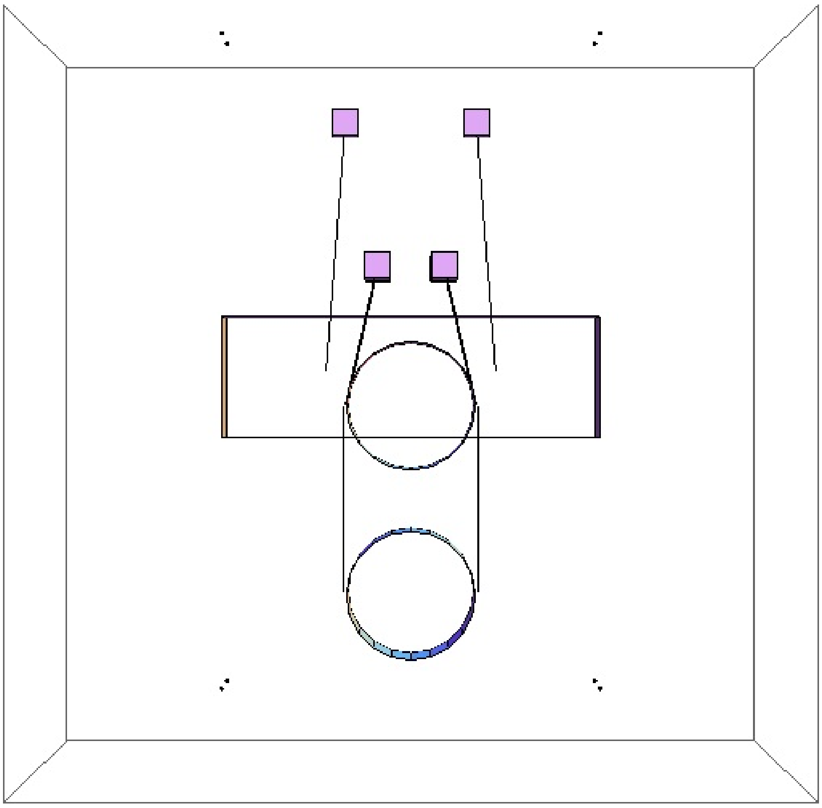


Mode #11 - modeV2

Back to Top

```
2.76171 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0.907309	0	0	0
Mass I	0	0	-0.29427	0	0	0
optic	0	0	-0.300326	0	0	0

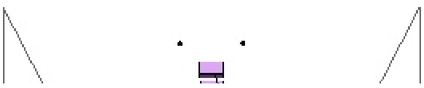


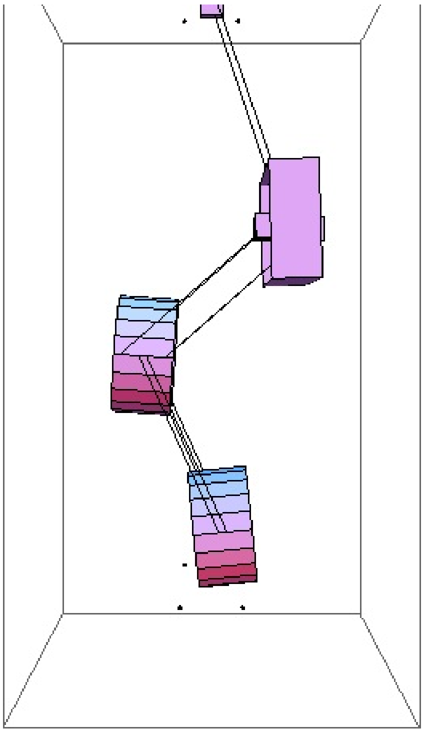
Mode #12 - modeL3

Back to Top

```
2.80671 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.54538	0	0	0	0.16219	0
Mass I	0.467556	0	0	0	-0.385581	0
optic	-0.0786733	0	0	0	0.550258	0



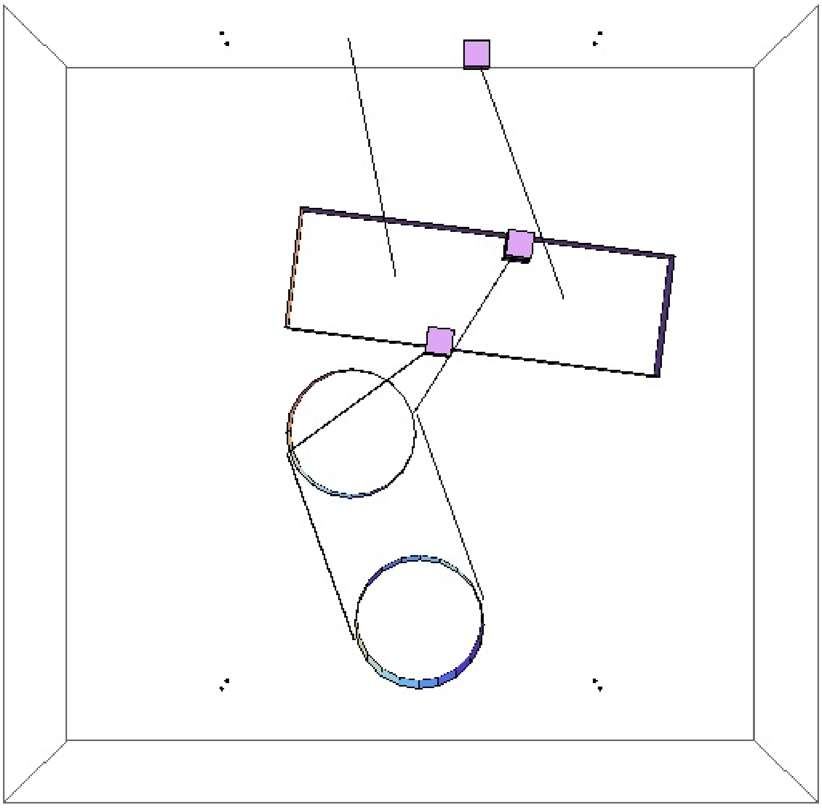


Mode #13 - modeT3

Back to Top

```
2.98169 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.177203	0	0	0	-0.286543
Mass I	0	-0.152925	0	0	0	0.65323
optic	0	0.0220861	0	0	0	0.660234

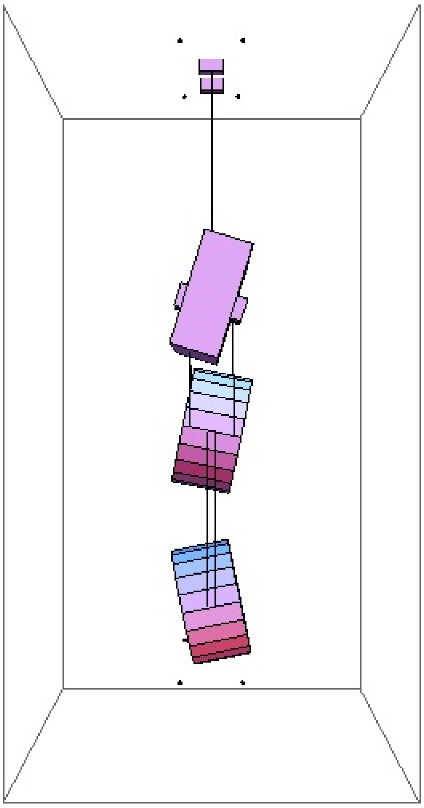


Mode #14 - modeP2

Back to Top

```
3.20926 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	0	-0.722402	0
Mass I	-0.00158387	0	0	0	-0.487615	0
optic	0.000196697	0	0	0	0.490269	0

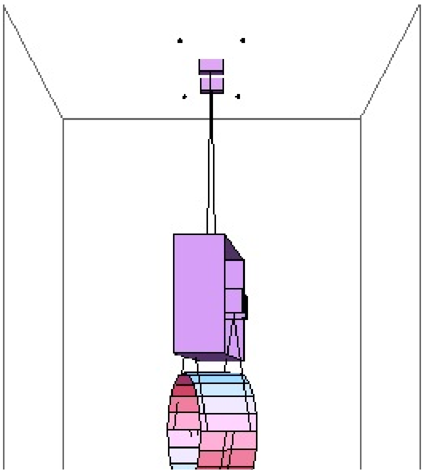


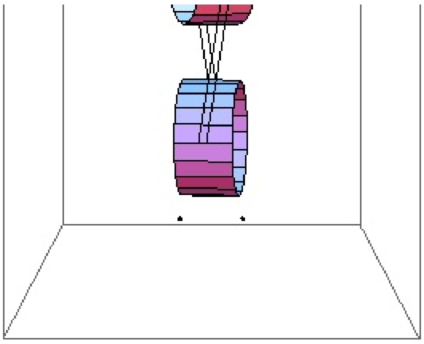
Mode #15 - modeY3

Back to Top

```
3.42401 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	-0.19789	0	0
Mass I	0	0	0	0.884947	0	0
optic	0	0	0	-0.421554	0	0



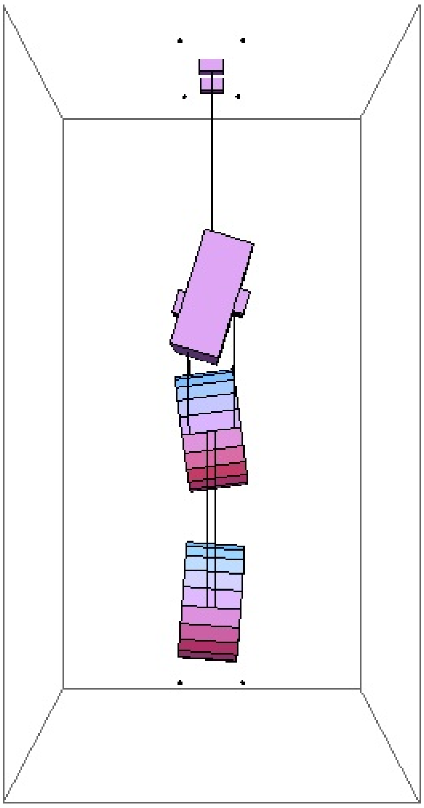


Mode #16 - modeP3

Back to Top

```
3.78136 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00126625	0	0	0	0.891275	0
Mass I	0.000223569	0	0	0	-0.395678	0
optic	0	0	0	0	0.22151	0

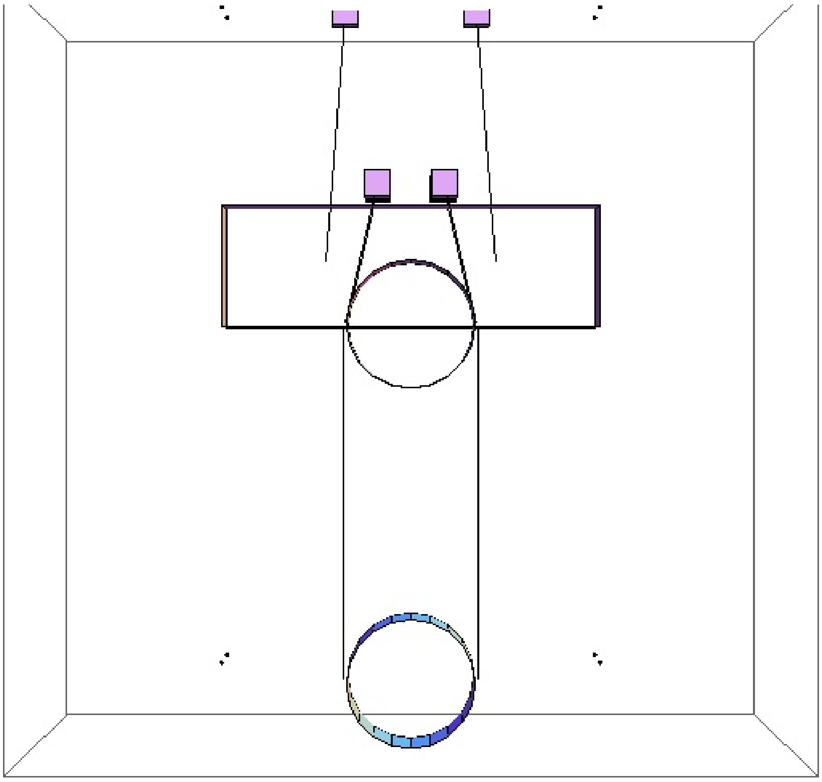


Mode #17 - modeV3

Back to Top

```
27.3201 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.00335283	0	0	0
Mass I	0	0	0.697492	0	0	0
optic	0	0	-0.716585	0	0	0

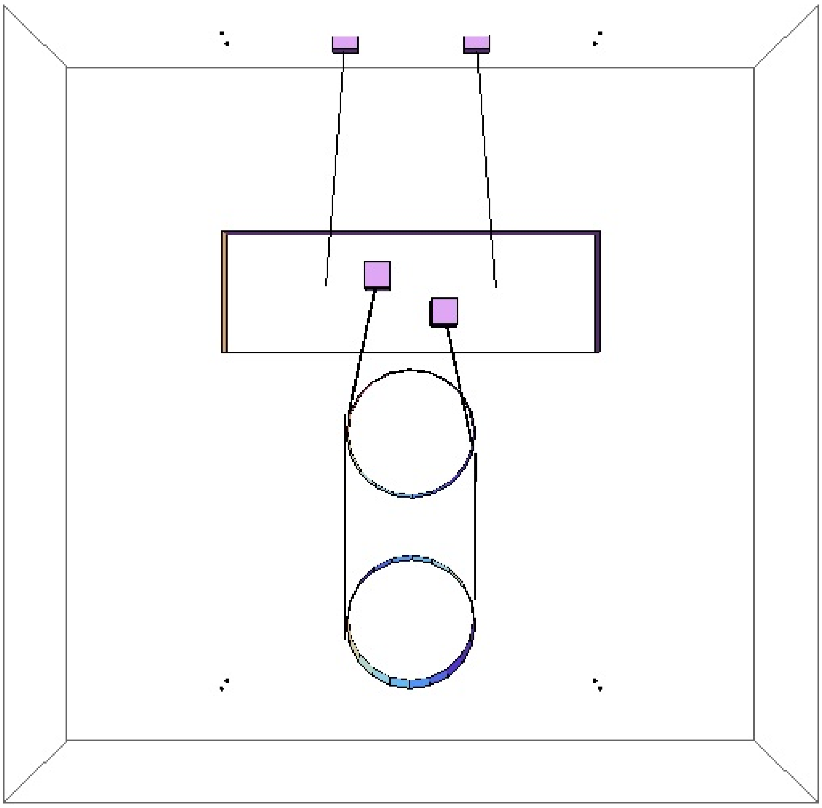


Mode #18 - modeR3

Back to Top

```
40.369 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	0	0	0.000138119
Mass I	0	0	0	0	0	-0.684383
optic	0	0	0	0	0	0.729123



Parameters

Back to Top

```
g -> 9.81

ux -> 0.06                                                          display only

uy -> 0.44                                                          display only

uz -> 0.142                                                         display only

den1 -> 830.933                                                     average computed from m1, ux,
uy, uz

m1 -> 3.115                                                         SolidWorks model of D020534
with wire, blade clamps added

I1x -> 0.0213                                                       SolidWorks model of D020534
with wire, blade clamps added

I1y -> 0.0025                                                       SolidWorks model of D020534
with wire, blade clamps added

I1z -> 0.0216                                                       SolidWorks model of D020534
with wire, blade clamps added

ix -> 0.075                                                         D0901792-v2, display only

ir -> 0.075                                                         D0901792-v2, display only,
radius/diameter error in hstsopt_metal.m corrected

den2 -> 2202

m2 -> 2.983                                                         SolidWorks model of D0901873
with wire clamps

I2x -> 0.00889                                                      SolidWorks model of D0901873
with wire clamps

I2y -> 0.00531                                                      SolidWorks model of D0901873
with wire clamps

I2z -> 0.00635                                                      SolidWorks model of D0901873
with wire clamps

tx -> 0.07435                                                       txMC

tr -> 0.075

den3 -> 2202

m3 -> 2.889                                                         m3MC

I3x -> 0.00832                                                      I3xMC

I3y -> 0.0054                                                       I3yMC

I3z -> 0.00557                                                      I3zMC

l1 -> 0.295                                                         D020700

l2 -> 0.167                                                         D020700

l3 -> 0.22                                                          D020700

nw1 -> 2

nw2 -> 4

nw3 -> 4

r1 -> 0.000178                                                      Actual wire is 0.014"
diameter

r2 -> 0.0001                                                        Actual wire is 0.0079"
diameter

r3 -> 0.0000597                                                     Actual wire is 0.047" in
diameter
```

```
               11
Y1 -> 2.119 10

               11
Y2 -> 2.119 10

               11
Y3 -> 2.119 10

ufc1 -> 1.71                                                       Fit to Dirty Build HSTS data

ufc2 -> 1.93                                                       Fit to Dirty Build HSTS data

ufc3 -> 0

d0 -> 0.006                                                        D020700-v1 (Rounded to
nearest mm)

d1 -> 0.002                                                        D020700-v1 (Rounded to
nearest mm)

d2 -> 0.001                                                        D020700-v1 (Rounded to
nearest mm)

d3 -> 0.001                                                        D020700-v1 (Rounded to
nearest mm)

d4 -> 0.001                                                        D020700-v1 (Rounded to
nearest mm)

su -> 0                                                            D020700-v1

si -> 0.0285                                                       D020700-v1

sl -> 0.005                                                        D020700-v1

n0 -> 0.0772                                                       D020700-v1

n1 -> 0.1                                                          D020700-v1

n2 -> 0.039                                                        D020700-v1

n3 -> 0.0765                                                       calculation from D020700-v1

n4 -> 0.08                                                         D020700-v1

n5 -> 0.08                                                         calculation from D020700-v1

tl1 -> 0.300118

tl2 -> 0.165735

tl3 -> 0.222

ltotal -> 0.687853

leverarmrt -> 0.03

leverarmrz -> 0.08

leverarmrl -> 0.08

gain -> 0.4

gainrtzrtl -> 0.4

gaint -> 0.8

gainlrz -> 0.4

b1 -> 0.03

b2 -> 0.03

b3 -> 0.03

b4 -> 0.03

b5 -> 0.03

b6 -> 0.03
```

```
unstretched -> False

vertblades -> True

ul1 -> 0.294382

ul2 -> 0.166629

ul3 -> 0.219343

sl1 -> 0.295

sl2 -> 0.167

sl3 -> 0.22

si1 -> 0.0772881

si2 -> 0.224551

si3 -> 0.

c1 -> 0.997009

c2 -> 0.974462

c3 -> 1.

pitchbul -> 0

pitchbur -> 0

pitchbll -> 0

pitchblr -> 0

pitchbllf -> 0

pitchblrf -> 0

pitchbllb -> 0

pitchblrb -> 0

rollbul -> 0

rollbur -> 0

rollbll -> 0

rollblr -> 0

rollbllf -> 0

rollblrf -> 0

rollbllb -> 0

rollblrb -> 0

                -8
A1 -> 9.95382 10

                -8
A2 -> 3.14159 10

                -8
A3 -> 1.11969 10

kw1 -> 71498.8

kw2 -> 39862.5

kw3 -> 10784.7

flex1 -> 0.00193808

flex2 -> 0.0010341

flex3 -> 0.000546237
```

```
kbuz -> 179.796

kblz -> 109.665

bdu -> 0.245174

bdl -> 0.131319

m13 -> 8.987

m23 -> 5.872

I1xy -> 0

I1yz -> 0

I1zx -> 0

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

Ibtxyl -> 0

Ibtyzl -> 0

Ibtzxl -> 0

I2xy -> 0                                                    SolidWorks model of D0901873
with wire clamps

I2yz -> 0                                                    SolidWorks model of D0901873
with wire clamps

I2zx -> 0                                                    SolidWorks model of D0901873
with wire clamps

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

I3xy -> 0.                                                   I3xyMC

I3yz -> 0.                                                   I3yzMC

I3zx -> 0.                                                   I3zxMC

COM3x -> 0

COM3y -> 0

COM3z -> 0

FRP3x -> 0

FRP3y -> 0

FRP3z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03

phib -> 0.001
```

phis -> 0.001

bd -> 0.001

M11 -> $7.88442\ 10^{-16}$

M12 -> $7.88442\ 10^{-16}$

M21 -> $7.85398\ 10^{-17}$

M22 -> $7.85398\ 10^{-17}$

M31 -> $9.97671\ 10^{-18}$

M32 -> $9.97671\ 10^{-18}$

temperature -> 290.

boltzmann -> $1.38066\ 10^{-23}$

alphasilica -> $3.9\ 10^{-7}$ measurements by PW, AH - cf. 5.1 10^-7 from gwinc

betasilica -> 0.000152 gwinc/IFOModel v1.0

rhosilica -> 2200. gwinc/IFOModel v1.0

Csilica -> 772. gwinc/IFOModel v1.0

Ksilica -> 1.38 gwinc/IFOModel v1.0

Ysilica -> $7.2\ 10^{10}$ spec sheet for silica

phisilica -> $4.1\ 10^{-10}$ gwinc/IFOModel v1.0

phissilica -> $3.\ 10^{-11}$ Phil Willems

rhosteel -> 7800. gwinc/IFOModel v1.0

Csteel -> 486. gwinc/IFOModel v1.0

Ksteel -> 49. gwinc/IFOModel v1.0

Ysteel -> $2.119\ 10^{11}$ measured by MB, 11/18/05

alphasteel -> 0.000012 gwinc/IFOModel v1.0

betasteel -> -0.00025 gwinc/IFOModel v1.0

phisteel -> 0.0002 gwinc/IFOModel v1.0 = Geppo's value

rhomarag -> 7800. gwinc/IFOModel v1.0

Cmarag -> 460. gwinc/IFOModel v1.0

Kmarag -> 20. gwinc/IFOModel v1.0

Ymarag -> $1.87\ 10^{11}$ gwinc/IFOModel v1.0

alphamarag -> 0.000011 gwinc/IFOModel v1.0

betamarag -> -0.00025 Geppo's value - gwinc/IFOModel v1.0 is wrong

phimarag -> 0.0001 gwinc/IFOModel v1.0

tmU -> 0.0025 upper blade thickness, NAR 8/4/06

```
tmL -> 0.0017                                                    lower blade thickness, NAR
8/4/06

magicnumber -> 0.0737472                                         Zener, 1938, Phys. Rev.
53:90-99

deltabladeU -> 0.00182883                                        cf Bench delta_v1

deltabladeL -> 0.00182883                                        cf Bench delta_v3

deltawireU -> 0.00254266                                         cf Bench delta_h1

deltawireL -> 0.00255524                                         cf Bench delta_h3

deltafibre -> 0.00263381

taubladeU -> 0.113606

taubladeL -> 0.0525316

tauwireU -> 0.00072307                                           cf Bench tau_steel1

tauwireL -> 0.000228213                                          cf Bench tau_steel3

tausilica -> 0.00178564

                                        0.00130543 #1
damping[imag, bladeUtype] -> (0.0001 + ---------------- & )
                                                      2
                                       1 + 0.509525 #1

                                        0.000603633 #1
damping[imag, bladeLtype] -> (0.0001 + ---------------- & )
                                                      2
                                       1 + 0.108943 #1

damping[imag, wireUtype] -> (0.0002 & )

damping[imag, wireLtype] -> (0.0002 & )

                                         0.0000115518 #1
damping[imag, wireUatype] -> (0.0002 + -------------------- & )
                                                          2
                                       1 + 0.0000206405 #1

                                                   -6
                                         3.66397 10   #1
damping[imag, wireLatype] -> (0.0002 + -------------------- & )
                                                     -6   2
                                       1 + 2.05608 10   #1

damping[imag, fibretype] -> (0.0002 & )

                                                   -6
                                         1.34603 10   #1
damping[imag, fibreatype] -> (0.0002 + -------------------- & )
                                                     -7   2
                                       1 + 2.61179 10   #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0
```

```
kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

kconx3 -> 0

kcony3 -> 0

kconz3 -> 0

kconyaw3 -> 0

kconpitch3 -> 0

kconroll3 -> 0

lockedblades -> False                                        False for maximum realism.
True for compatibility with very old Matlab.

kw1usual -> 71498.8

kw2usual -> 39862.5

kw3usual -> 10784.7

kbuzusual -> 179.796

kblzusual -> 109.665

                  6
kbuy -> 1.79796 10

                  6
kbly -> 1.09665 10

kbux -> 17979.6

kblx -> 10966.5

ifs -> 0.13777

nb -> D0901792-v2

txMC -> 0.07435

nb -> SolidWorks model of D020234 (MC metal mass[0.5 deg wedge])

m3MC -> 2.889                                                SolidWorks model of D0901791
(MC metal mass [0.5 deg wedge] with metal prisms)

I3xMC -> 0.00832                                             SolidWorks model of D0901791
(MC metal mass [0.5 deg wedge] with metal prisms)

I3yMC -> 0.0054                                              SolidWorks model of D0901791
(MC metal mass [0.5 deg wedge] with metal prisms)

I3zMC -> 0.00557                                             SolidWorks model of D0901791
(MC metal mass [0.5 deg wedge] with metal prisms)

I3xyMC -> 0.                                                 SolidWorks model of D0901791
(MC metal mass [0.5 deg wedge] with metal prisms)

I3yzMC -> 0.                                                 SolidWorks model of D0901791
(MC metal mass [0.5 deg wedge] with metal prisms)

I3zxMC -> 0.                                                 SolidWorks model of D0901791
(MC metal mass [0.5 deg wedge] with metal prisms)

dl -> 0.                                                     correction to wire length to
offset wedge mass imbalance

bssteel -> 2000000000                                        breaking stress of steel
```

```
                  8
wssilica -> 7.7 10                                                  working stress of silica

r1opt -> 0.000145294

r2opt -> 0.0000840012

r3opt -> 0.00005412

dssilica -> 0.015                                                   gwinc/IFOModel v1.0

taufibre -> 0.0000813372
```

Back to Top

aLIGO SUS Operation Manual - Info on OMCS Suspensions

Back to Operation Manual main page

References

- D1002740 aLIGO SUS HAM 5-6 Wiring Diagrams
- T1000061 aLIGO HAM Triple Suspension Controls Design Description (HLTS, HSTS and OMCS)
- E1100109 HAM Suspensions Controls Arrangement Poster (HLTS, HSTS and OMCS)

Models

The OMCS suspension has been modelled. Key results are at Suspensions/OpsManual/OMCS/Models .

Screens

OMCS MEDM screens are at /Screens.

aLIGO: Suspensions/OpsManual/OMCS (last edited 2013-09-13 14:17:45 by MarkBarton)

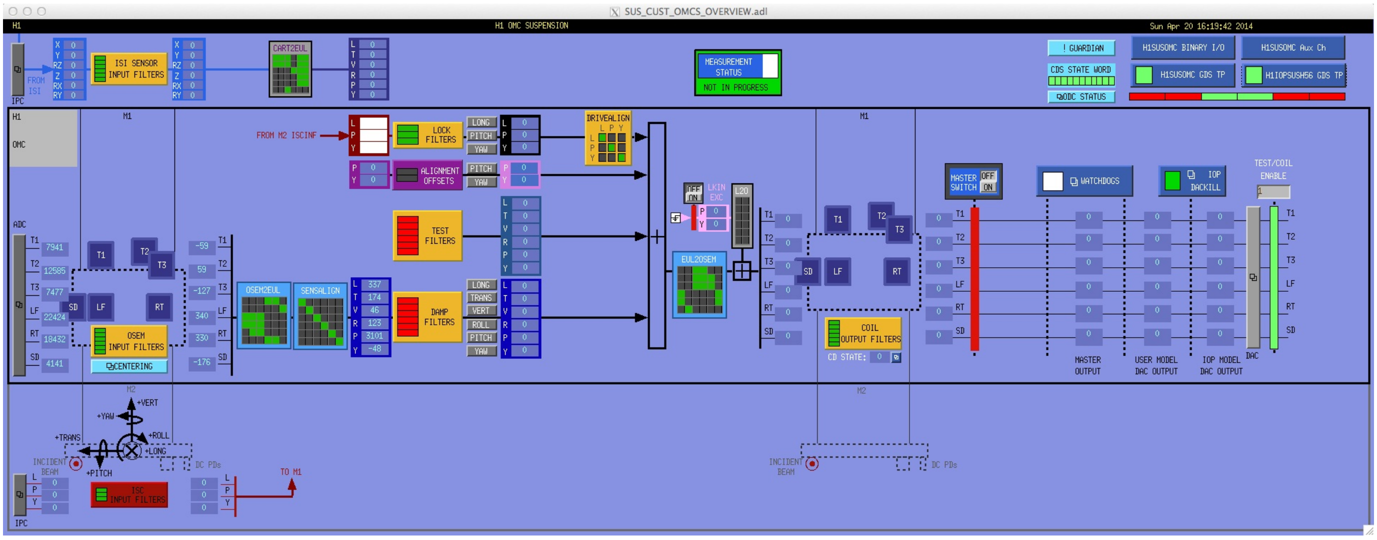
aLIGO SUS Operations Manual - Overview of OMCS MEDM screens

Back to Operation Manual main page
Back to OMCS main page

The OMCS screens described below live at `/opt/rtcds/userapps/release/sus/common/medm/omcs/`. They are all generic screens which rely for their execution on arguments passed in when they are called. Most of the arguments are defined in the macro file `/opt/rtcds/userapps/release/sus/common/medm/susomc_overview_macro.txt`. The generic screens can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_OMCS_OVERVIEW.adl

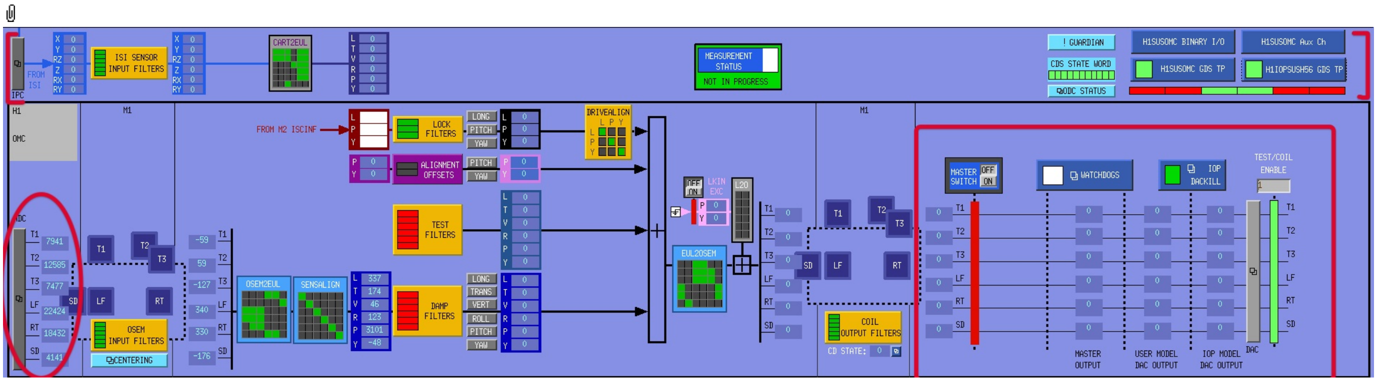


This is the overview screen and one of only a few specific to OMCS (with OMCS rather than OMCS in the name). There nominally two sensor-actuator groups, but only M1 is of much interest - there are no OSEMs on the suspended bench:

- Other - subscreens not associated with a sensor/actuator group
- M1 - 6 BOSEMs on the structure engaging the upper mass
- M2 - display of IPC channels from ISC.

M1 is used for local damping (relative to the structure). DC pitch and yaw offsets are also injected at M1.

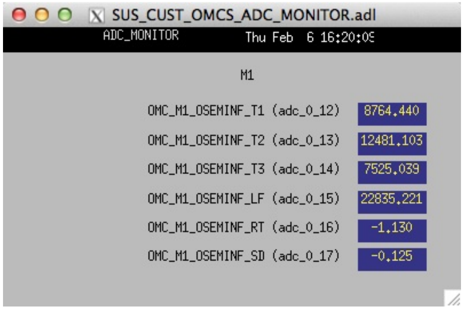
Other Screens



- MASTER SWITCH - a manual on/off switch for all actuation (common to all sensor-actuator groups).
- MEASUREMENT STATUS - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- HIERARCHY SWITCH - enables various ISC inputs and outputs.
- ADC - a monitor screen (common to all sensor-actuator groups) for the semi-raw ADC values as reported by the IOP process after downsampling.
- DAC - a popup allowing access to all the DAC outputs as reported by the IOP process.
- WATCHDOGS - a block implementing the watchdogs.
- IOP DACKILL - a watchdog that shuts off the IOP process (potentially other suspensions on the same front-end).
- H1IOPSUSH56 GDS TP (or the like; depends on the front-end computer name) - status of the IOP process for the front-end.
- H1SUSOMC GDS TP (or the like; depends on the suspension name) - status of the suspension process.
- H1SUSOMC BINARY I/O (or the like; depends on the suspension name) - binary input/output controls

- H1SUSOMC BINARY I/O (or the like; depends on the suspension name) - binary input/output controls.
- H1SUSOMC Aux Ch (or the like; depends on the suspension name) - readbacks from coil driver channels for all OSEMs as reported by the corresponding auxiliary front-end processor.
- ODC STATUS - ODC status
- !GUARDIAN Guardian controls
- ISIINF - ISI input filters.
- CART2EUL - transformation from the ISI basis to SUS.

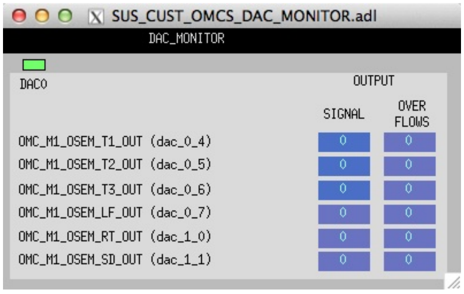
Screen SUS_CUST_OMCS_ADC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

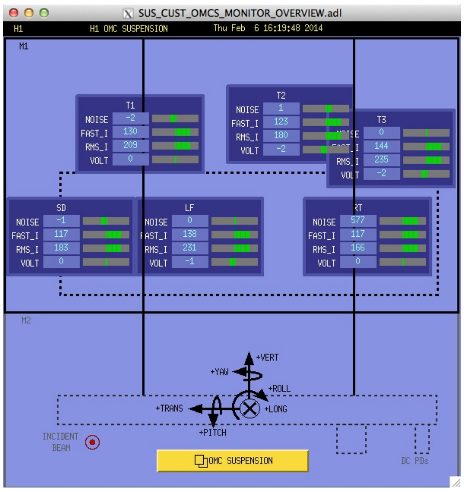
Screen SUS_CUST_OMCS_DAC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen SUS_CUST_OMCS_MONITOR_OVERVIEW.adl

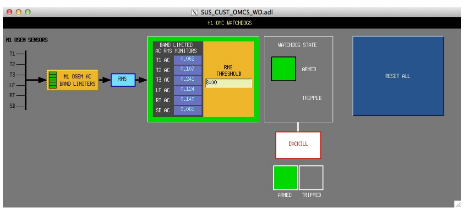


Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

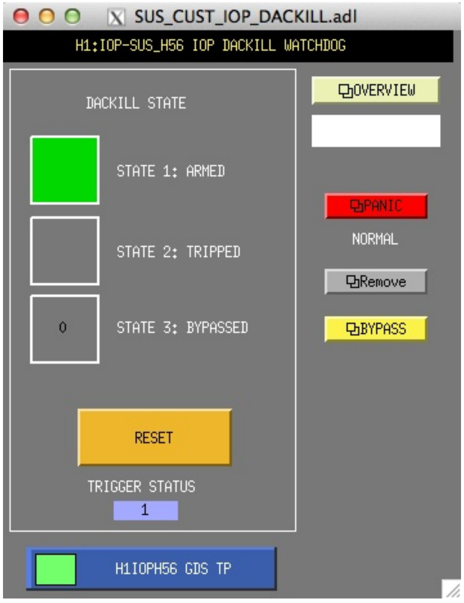
Screen SUS_CUST_OMCS_WD.adl



Suspensions/OpsManual/Boilerplate/WD:

This screen allows for setting the watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

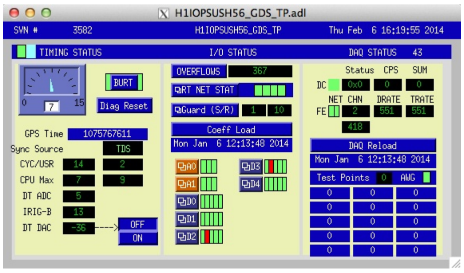
Screen SUS_CUST_IOP_DACKILL.adl



Suspensions/OpsManual/Boilerplate/IOP_DACKILL:

The watchdog controls for the IOP process. If this WD is tripped it disables IO to/from all user models on the front end. RESET resets from the tripped state if there are no outstanding triggers. BYPASS is a manual override that allows for operation of the suspension if there is a trigger condition that is known by the operator to be harmless.

Screen H1IOPSUSH56_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

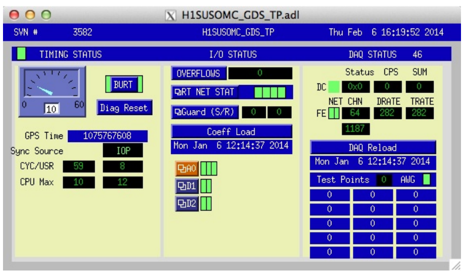
This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1SUSOMC_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

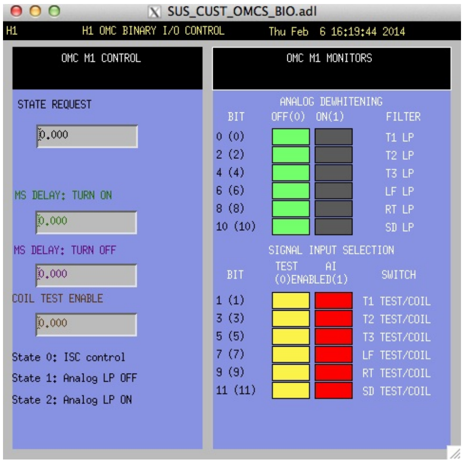
This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

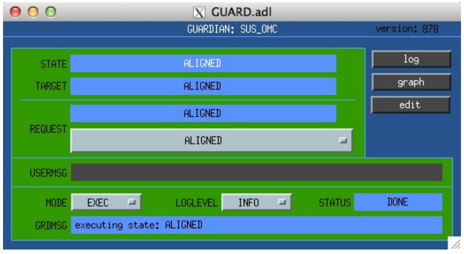
Screen SUS_CUST_OMCS_BIO.adl



Suspensions/OpsManual/Boilerplate/OMCS_BIO:

The Binary Input-Output control screen. State Requests for the various sensor-actuator groups can be input. Positive state values lock out manual control of the filters on the COILOUTF screen and set the digital dewhitening filters to match the analog whitening filters selected via the binary outputs. Negative state values allow the dewhitening filters to be manually overridden.

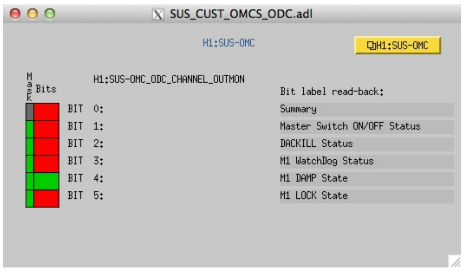
Screen GUARD.adl



Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

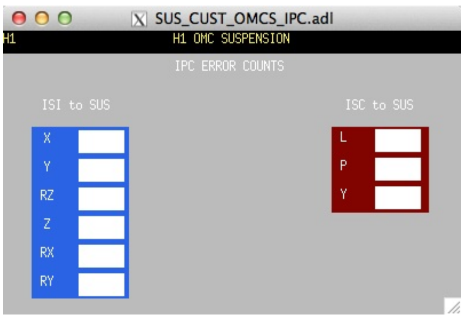
Screen SUS_CUST_OMCS_ODC.adl



Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

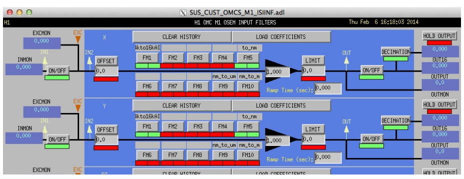
Screen SUS_CUST_OMCS_IPC.adl

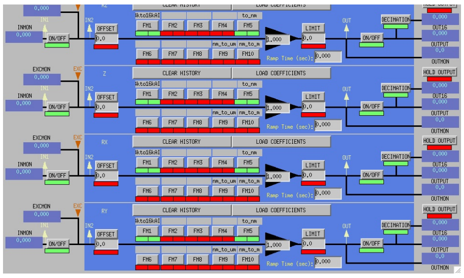


Suspensions/OpsManual/Boilerplate/IPC:

Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

Screen SUS_CUST_OMCS_M1_ISIINF.adl





Suspensions/OpsManual/Boilerplate/M1_ISIINF:

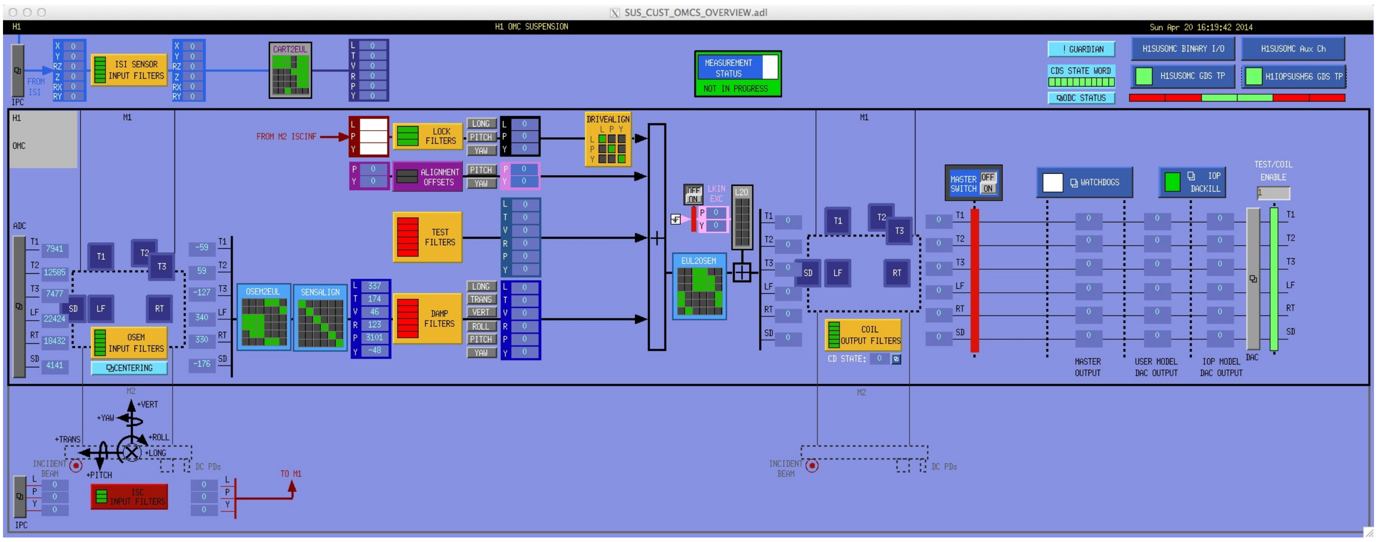
Filters for the diagnostic signals from the ISI.

Screen SUS_CUST_OMCS_M1_CART2EUL.adl

Suspensions/OpsManual/Boilerplate/M1_CART2EUL:

A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

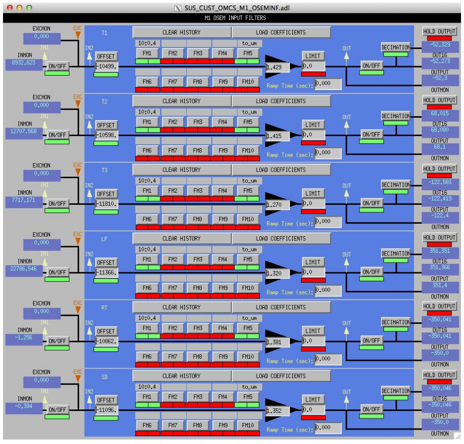
Sensor Actuator Group M1 - Upper Mass



Most of the items on the M1 section correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- DAMP FILTERS - a filter bank applying the damping filters.
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- OPTICALIGN ("ALIGNMENT OFFSETS") - a place to apply DC actuation in pitch and yaw.
- LOCK FILTERS - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

Screen SUS_CUST_OMCS_M1_OSEMINF.adl



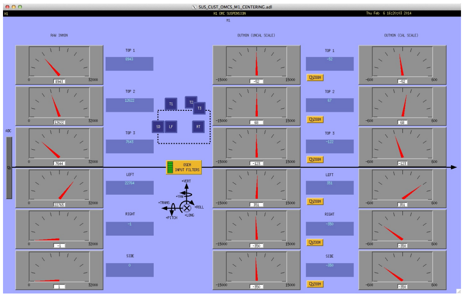
Suspensions/OpsManual/Boilerplate/M1_OSEMINF:

This block has 6 filter groups corresponding to the 6 M1 BOSEMs, T1/T2/T3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to μm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_OMCS_M1_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M1_CENTERING:

This screen gives various views of the M1 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_OMCS_M1_OSEM2EUL.adl

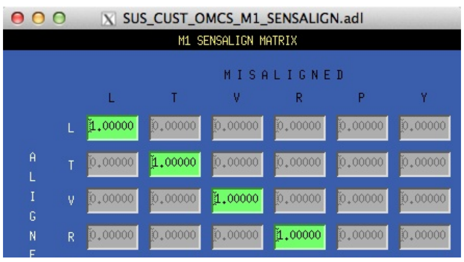
Suspensions/OpsManual/Boilerplate/M1_OSEM2EUL:

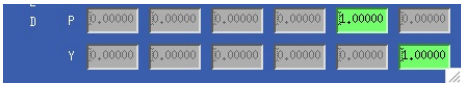
This screen allows entry of the matrix which converts from the M1 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in μm the outputs are then in μrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

Screen SUS_CUST_OMCS_M1_SENSALIGN.adl





Suspensions/OpsManual/Boilerplate/M1_SENSALIGN:

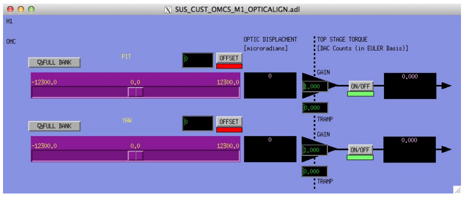
This screen is reserved for tweaking the M1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_OMCS_M1_DAMP.adl


Suspensions/OpsManual/Boilerplate/M1_DAMP:

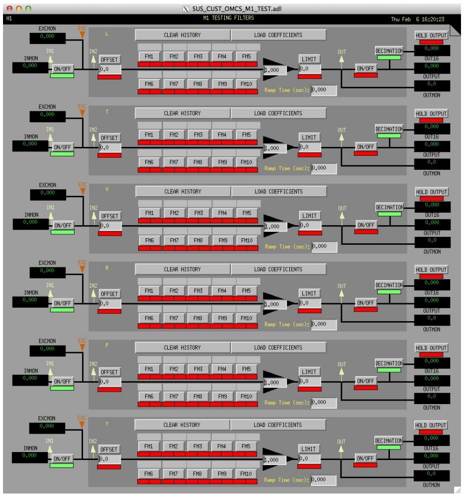
These filters implement the local damping. See Suspensions/OpsManual/Damping.

Screen SUS_CUST_OMCS_M1_OPTICALIGN.adl


Suspensions/OpsManual/Boilerplate/M1_OPTICALIGN:

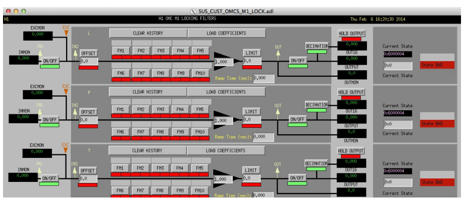
This screen has sliders for dialling in static pitch and yaw actuation offsets. See Suspensions/OpsManual/Alignments for more info.

Screen SUS_CUST_OMCS_M1_TEST.adl


Suspensions/OpsManual/Boilerplate/M1_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

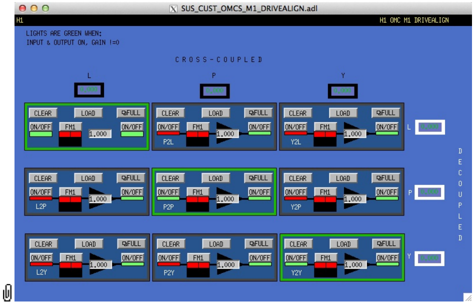
Screen SUS_CUST_OMCS_M1_LOCK.adl



Suspensions/OpsManual/Boilerplate/M1_LOCK:

Filters for the locking signals.

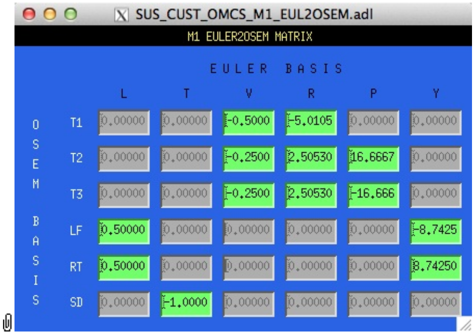
Screen SUS_CUST_OMCS_M1_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_DRIVEALIGN:

This screen is reserved for tweaking the M1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

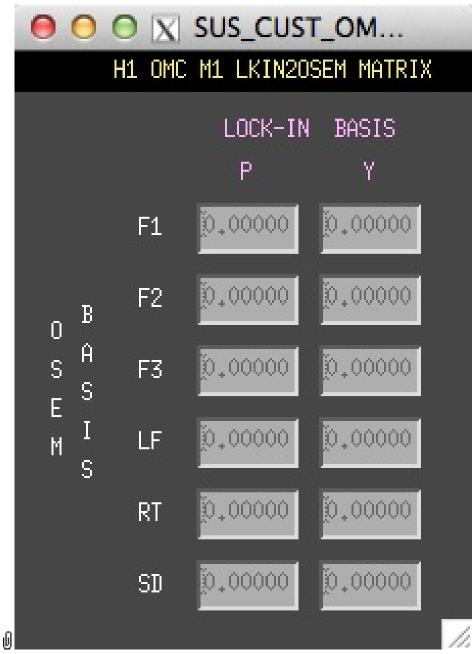
Screen SUS_CUST_OMCS_M1_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

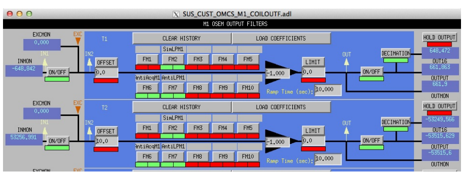
Screen SUS_CUST_OMCS_M1_LKIN2OSEM.adl

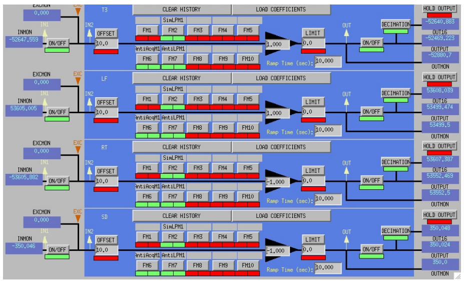


Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_OMCS_M1_COILOUTF.adl





Suspensions/OpsManual/Boilerplate/M1_COILOUTF:

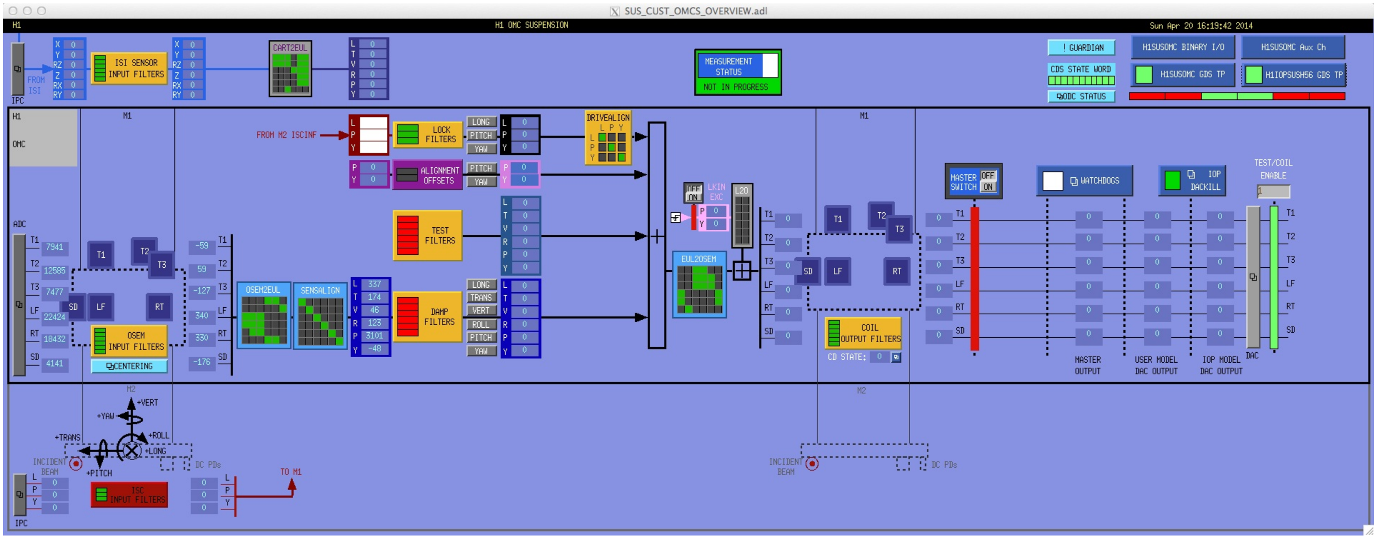
This screen applies compensation for the hardware filters in the M1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

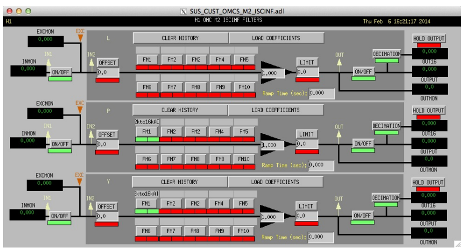
Sensor Actuator Group M2 - OMC Bench



The M2 level currently only has input filters for ISC inputs:

- ISCINF - ISI input filters.

Screen SUS_CUST_OMCS_M2_ISCINF.adl



Suspensions/OpsManual/Boilerplate/M2_ISCINF:

Filters for inputs from ISC.

aLIGO SUS Operations Manual - Models for OMCS Suspensions

Back to Operation Manual main page

There are Mathematica (T020205) and Matlab (T080188) programs that take a parameter set describing the OMCS and calculate the dynamics. The parameter set needs to be formatted differently for Mathematica vs Matlab, but the parameter names are the same as far as was practical.

The OMCS Mathematica model and parameter sets for it live in the SUS SVN at `^/trunk/Common/MathematicaModels/DualLite2`. Parameter sets for Matlab live at `^/trunk/QUAD/Common/MatlabTools/DoubleModel_Production`. Mark Barton maintains the Mathematica parameter sets, Jeff Kissel maintains the Matlab. Unfortunately two different naming conventions are in use and it's difficult to tell what corresponds to what - if in doubt, ask. References given below are to Mathematica versions as subdirectories of `^/trunk/Common/MathematicaModels/DualLite2DBLateral`.

Key cases of the OMCS model are given below, with results such as mode frequencies and mode shapes.

Explanation of the standard result set is at .../Key. Instructions for generating wiki pages for new cases of Mathematica models is at .../HowTo.

Model 20130612OMCSmetal4904

/20130612OMCSmetal4904 corresponds to the omcsopt_metal.m r4904 of 6/12/13 in the SVN, with new values for the final bench design. Valid for metal or glass.

Model 20130526OMCSmetal4419

/20130526OMCSmetal4419 corresponds to the omcsopt_metal.m r4419 of 5/26/13 in the SVN.

Model 20130526OMCSglass4419

/20130526OMCSglass4419 corresponds to the omcsopt_glass.m r4419 of 5/26/13 in the SVN.

Model 20080601OMCasbuilt

The model /20080601OMCasbuilt corresponds to the eLIGO parameter set in T080138.

Case 20130612OMCSmetal4904 of Mathematica model DualLite2

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
"20130612OMCSmetal4904"
```

```
20130612OMCSmetal4904 - equivalent to Matlab case omcs_metal.m r4904 of 6/12/13. Also valid for
glass bench.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeY1	modeT1	modeL1	modeR1	modeV1	modeP1
modeT2	modeL2	modeP2	modeY2	modeV2	modeR2

Mode Summary

Back to Top

```
N    f          name     type

1    0.500869   modeY1   yaw2     yaw1

2    0.722761   modeT1   roll2    roll1

3    0.728969   modeL1   pitch1   pitch2

4    0.963506   modeR1   roll2    roll1

5    1.01526    modeV1   z2       z1

6    1.90754    modeP1   pitch1   pitch2

7    2.62587    modeT2   y1

8    2.72197    modeL2   pitch1

9    3.84991    modeP2   pitch1

10   3.85205    modeY2   yaw1

11   4.27468    modeV2   z1

12   6.76293    modeR2   roll1
```

Violin Modes	UM	Bench
f (Hz), n=1-4	511.805 1024.45 1538.78 2055.65	609.368 1219.1 1829.57 2441.13
Q, n=1-4	57709.8 81239.8 91050.1 90639.5	73379.7 78073.6 88003.2 93678.6

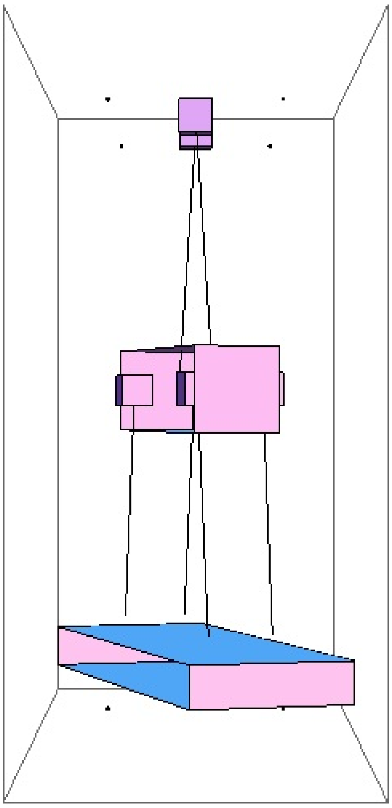
Mode Shapes

Mode #1 - modeY1

Back to Top

```
0.500869 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	0.605535	0	0
Mass I	0	0	0	0.795819	0	0

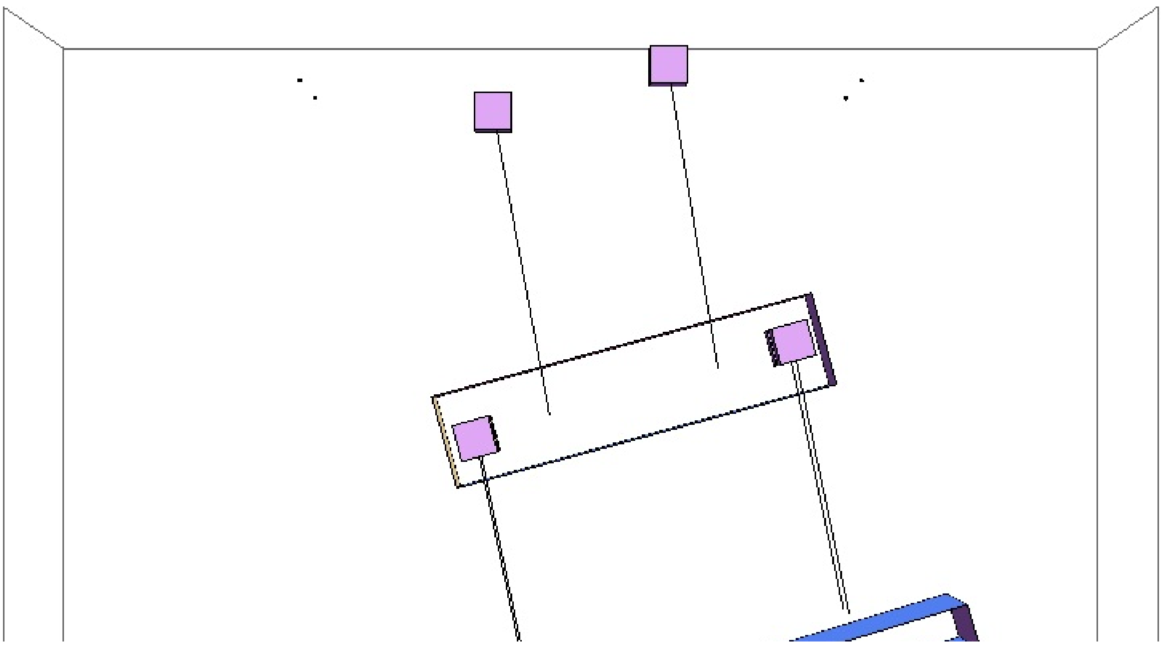


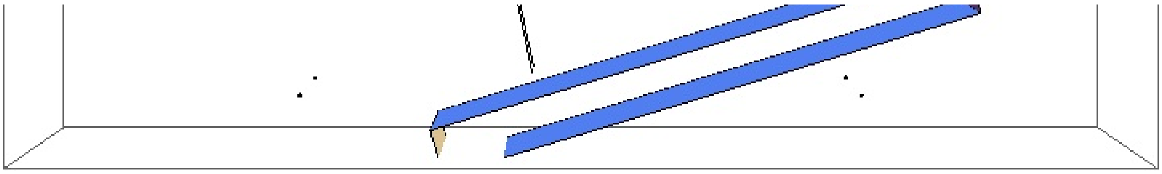
Mode #2 - modeT1

Back to Top

```
0.722761 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.105271	0	0	0	-0.647413
Mass I	0	-0.242892	0	0	0	-0.714687



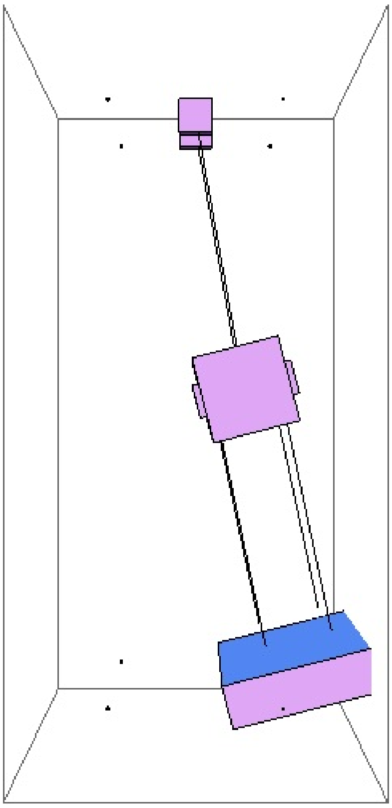


Mode #3 - modeL1

Back to Top

```
0.728969 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.118741	0	0	0	-0.684972	0
Mass I	0.268559	0	0	0	-0.666775	0

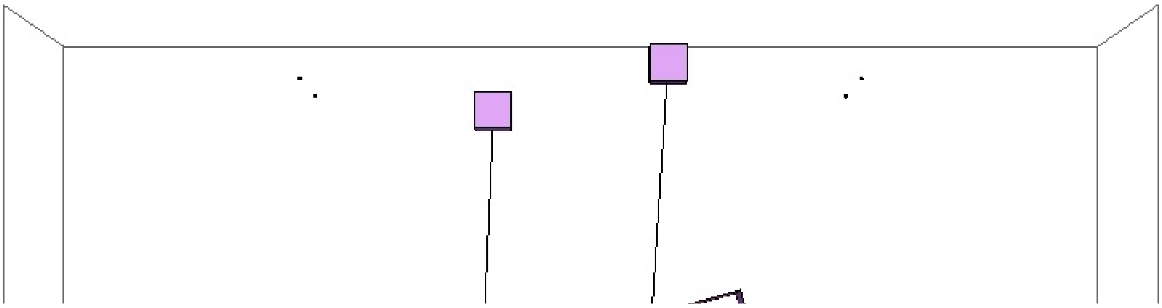


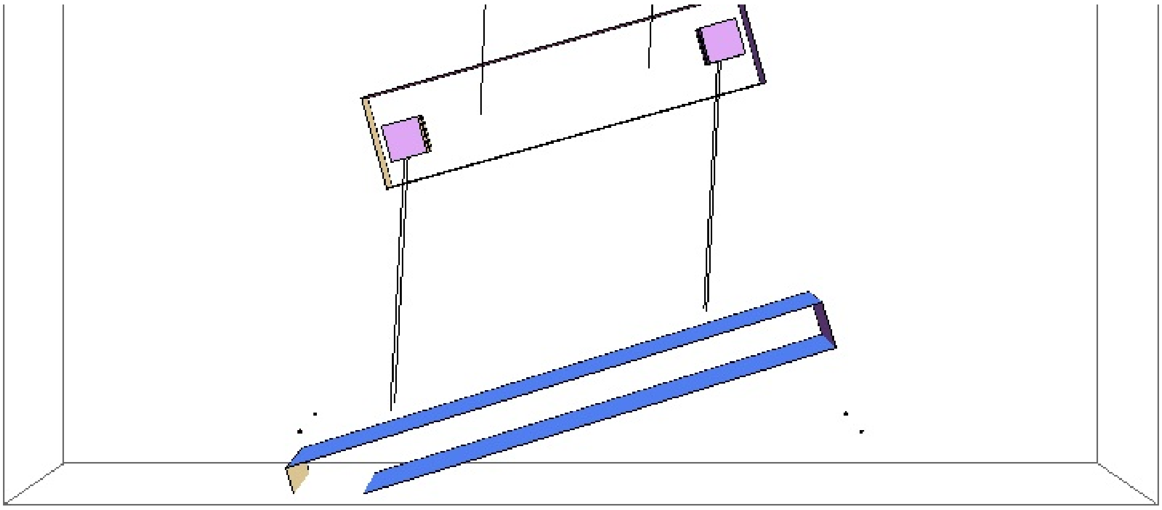
Mode #4 - modeR1

Back to Top

```
0.963506 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.0344247	0	0	0	-0.672564
Mass I	0	0.039167	0	0	0	-0.738199



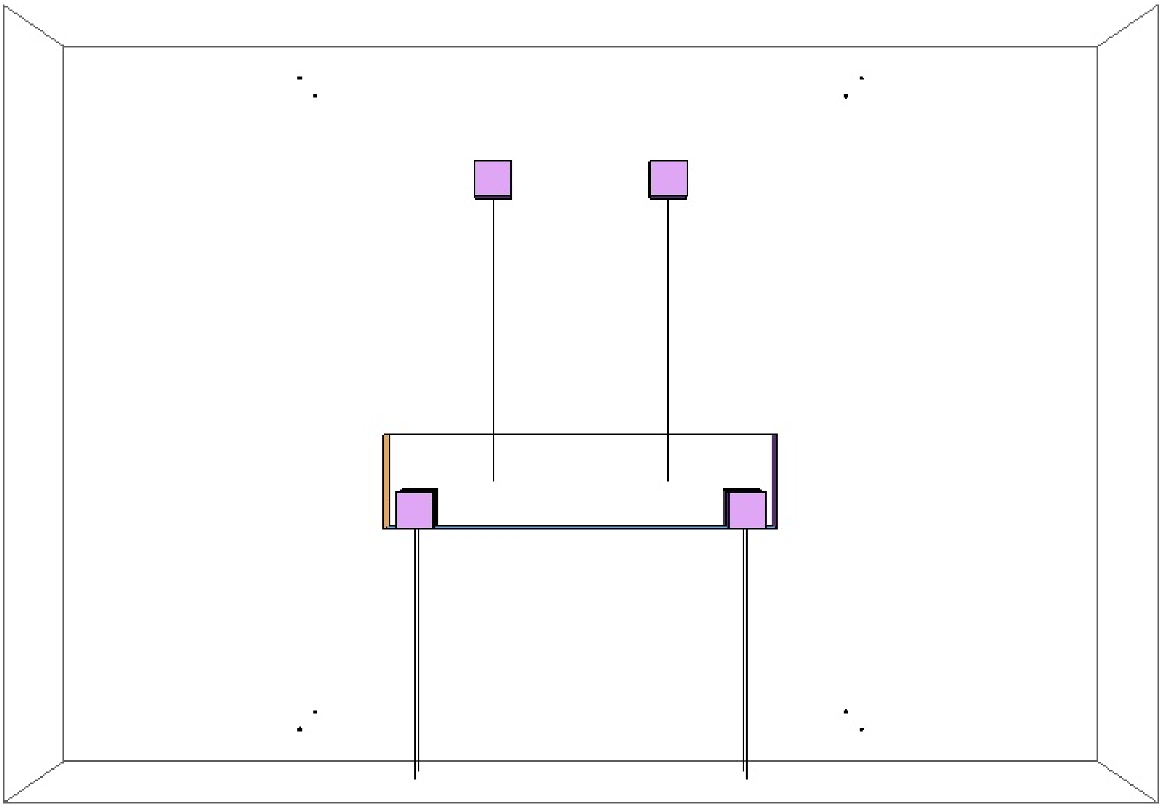


Mode #5 - modeV1

Back to Top

```
1.01526 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.605918	0	0	0
Mass I	0	0	-0.795527	0	0	0

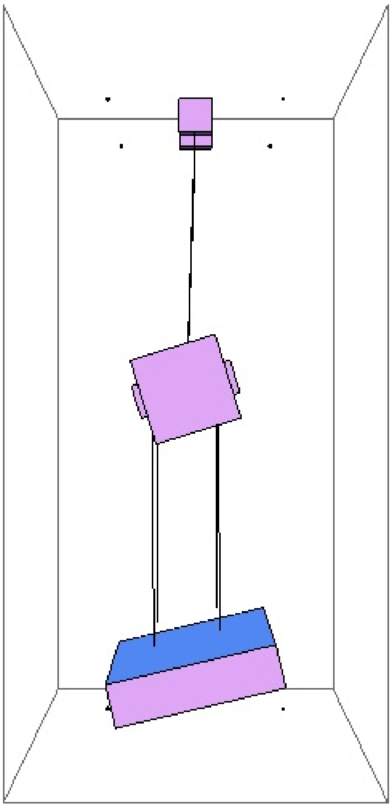


Mode #6 - modeP1

Back to Top

```
1.90754 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.0232808	0	0	0	-0.789996	0
Mass I	0.000243226	0	0	0	-0.61267	0

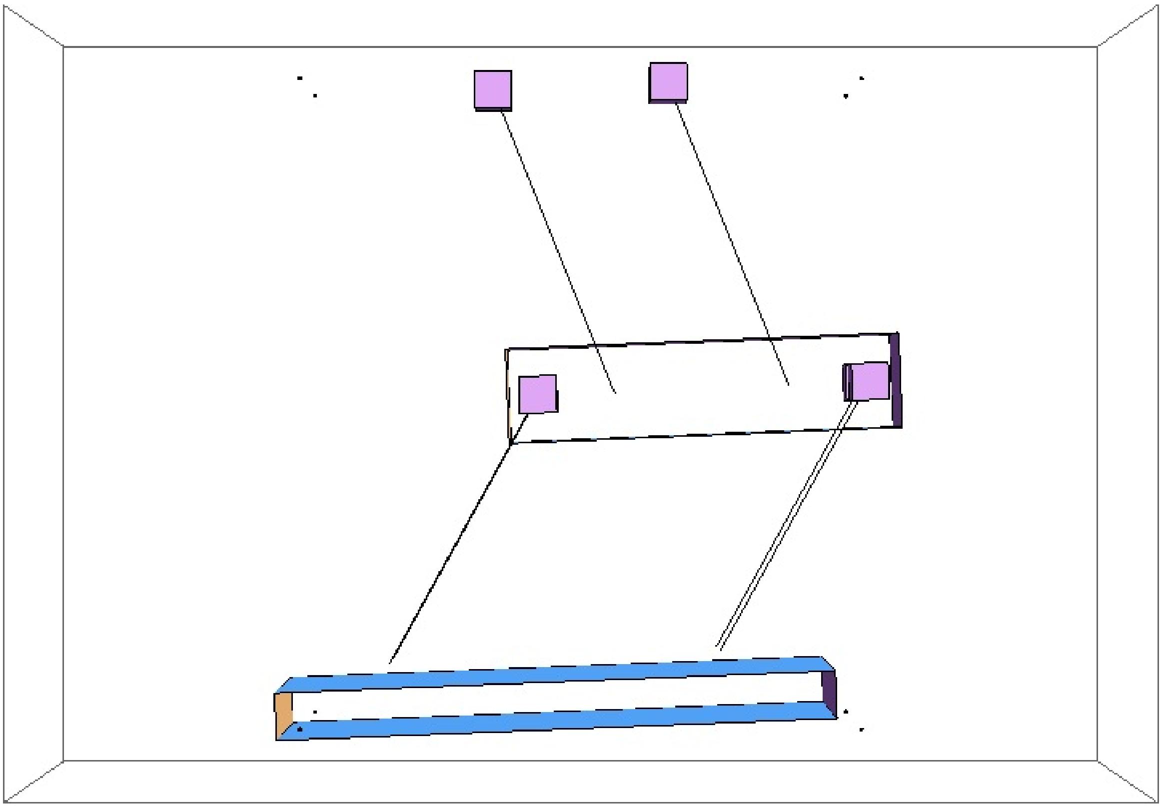


Mode #7 - modeT2

Back to Top

```
2.62587 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	-0.855455	0	0	0	-0.353207
Mass I	0	0.171867	0	0	0	-0.337495

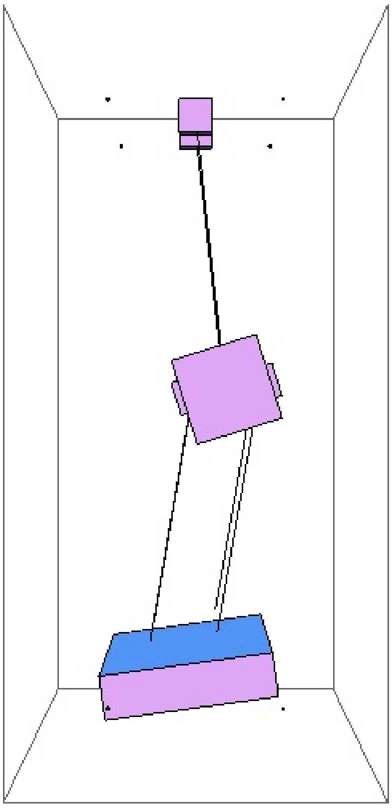


Mode #8 - modeL2

Back to Top

```
2.72197 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.0825194	0	0	0	-0.907918	0
Mass I	-0.0178333	0	0	0	-0.410558	0

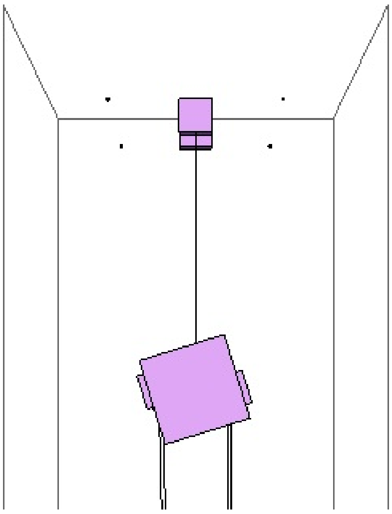


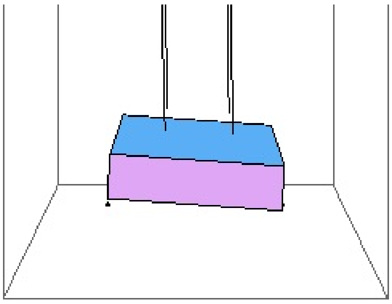
Mode #9 - modeP2

Back to Top

```
3.84991 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.0028352	0	0	0	-0.974201	0
Mass I	0.000840731	0	0	0	0.225664	0



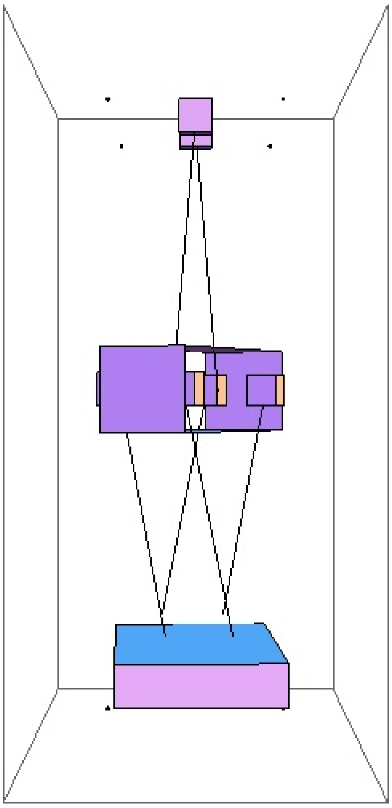


Mode #10 - modeY2

Back to Top

```
3.85205 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0	-0.997118	0	0
Mass I	0	0	0	0.0758702	0	0

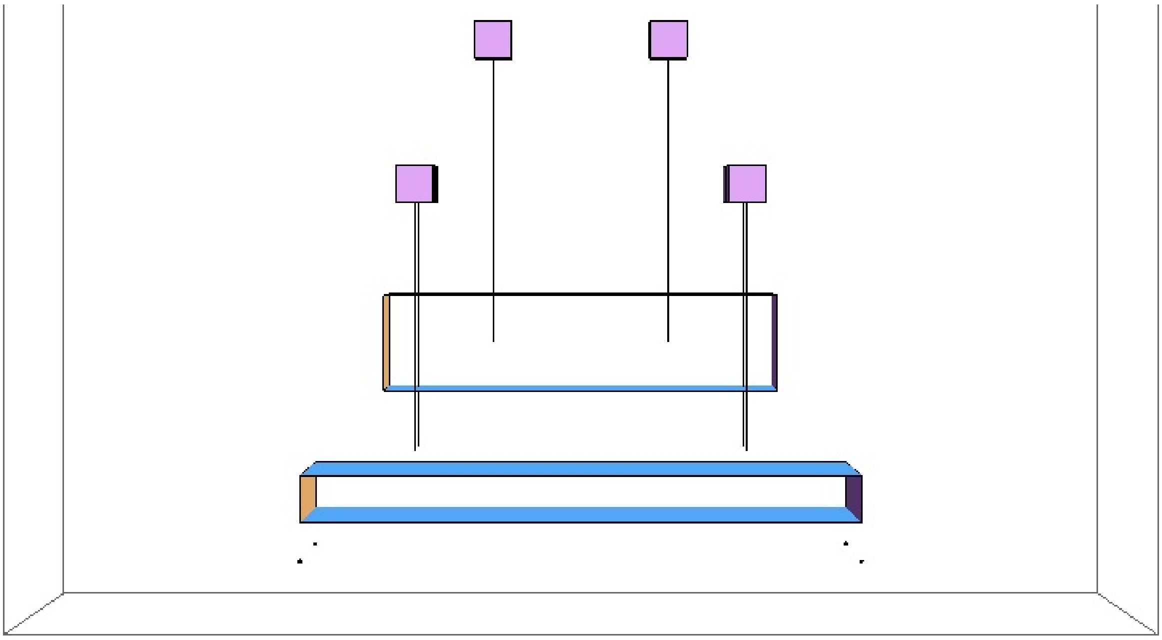


Mode #11 - modeV2

Back to Top

```
4.27468 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.955143	0	0	0
Mass I	0	0	0.296143	0	0	0

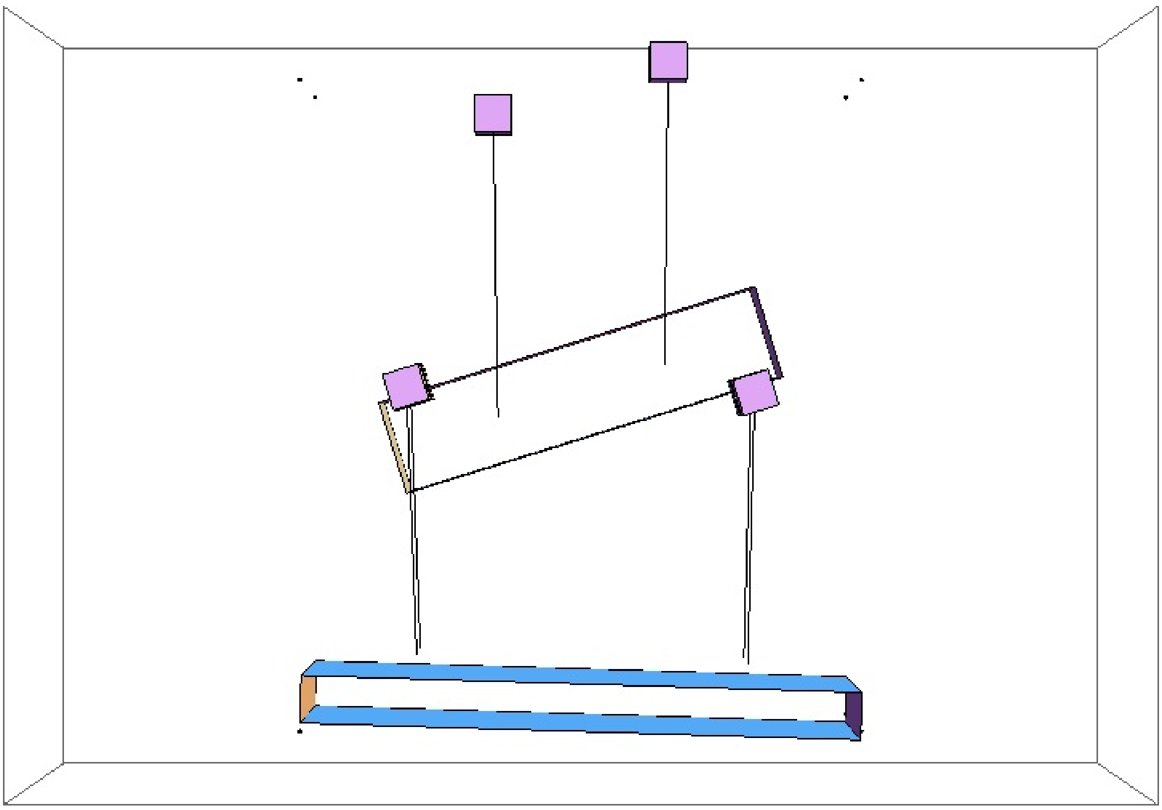


Mode #12 - modeR2

Back to Top

```
6.76293 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0.000237302	0	0	0	-0.994986
Mass I	0	0	0	0	0	0.10001



Parameters

Back to Top

```
g -> 9.81
```

```
ux -> 0.078                                                        D0900295, DISPLAY ONLY
display only

uy -> 0.32                                                         D0900295, DISPLAY ONLY
display only

uz -> 0.077                                                        D0900295, DISPLAY ONLY
display only

m1 -> 2.9                                                          As-built parameters, J.
Bartlett email, June 11 2013 4:34pm ET

I1x -> 0.015                                                       from Chris e-mail 25 April
2008, T080138

I1y -> 0.00248                                                     from Chris e-mail 25 April
2008, T080138

I1z -> 0.0148                                                      from Chris e-mail 25 April
2008, T080138

ix -> 0.15                                                         D0900295, DISPLAY ONLY
display only

iy -> 0.45                                                         D0900295, DISPLAY ONLY
display only

iz -> 0.0381                                                       D0900295, DISPLAY ONLY
display only

m2 -> 7.124                                                        As-built parameters, J.
Bartlett email, June 11 2013 4:34pm ET

I2x -> 0.136                                                       SolidWorks Assembly, D.
Bridges 2013-Mar-06

I2y -> 0.0152                                                      SolidWorks Assembly, D.
Bridges 2013-Mar-06

I2z -> 0.148                                                       SolidWorks Assembly, D.
Bridges 2013-Mar-06

l1 -> 0.2496                                                       D0900295 (verified 2013-06-10,
JSK)

l2 -> 0.22                                                         D0900295 (verified 2013-06-10,
JSK)

nw1 -> 2

nw2 -> 4

r1 -> 0.000178                                                     D0900295, 14 thou diameter
(verified 2013-06-10, JSK)

r2 -> 0.0001005                                                    D0900295, 8 thou diameter
(verified 2013-06-10, JSK)

              11                                                   -( (Confirmed in T080222-v3)
Barton Mark) + measured - Mark Barton ,                          measured (Confirmed in T080222 -
v3)
Y1 -> 2.119 10
Y2->2.119 10^11, nb[]->

             11
Y2 -> 2.12 10

ufc1 -> 2.09                                                       Fit to measured vertical
transfer functions, X2:SUS-OMC, 2013-06-11 using doub_spring_stiff_calc.m in
DoubleModel_Production directory.

ufc2 -> 2.09                                                       Fit to measured vertical
transfer functions, X2:SUS-OMC, 2013-06-11 using doub_spring_stiff_calc.m in
DoubleModel_Production directory.

d0 -> -0.0014                                                      Calculated via SolidWorks from
as-built blade tip heights, D. Bridges email 2013-06-12 1:24a

d1 -> -0.0001                                                      Calculated via SolidWorks from
as-built blade tip heights, D. Bridges email 2013-06-12 1:24a

d2 -> 0.035                                                        Fit to X2:SUS-OMC 2013-06-11
```

```
data

su -> 0.

sl -> 0.03

n0 -> 0.072                                                    D0900295

n1 -> 0.072                                                    D0900295

n2 -> 0.135                                                    D0900295

n3 -> 0.135                                                    D0900295

tl1 -> 0.2482

tl2 -> 0.2549

ltotal -> 0.5031

unstretched -> False

vertblades -> True

ul1 -> 0.249018

ul2 -> 0.219429

sl1 -> 0.2496

sl2 -> 0.22

si1 -> 0.

si2 -> 0.

c1 -> 1.

c2 -> 1.

pitchbul -> 0

pitchbur -> 0

pitchbll -> 0

pitchblr -> 0

pitchbllf -> 0

pitchblrf -> 0

pitchbllb -> 0

pitchblrb -> 0

rollbul -> 0

rollbur -> 0

rollbll -> 0

rollblr -> 0

rollbllf -> 0

rollblrf -> 0

rollbllb -> 0

rollblrb -> 0

                -8
A1 -> 9.95382 10

                -8
A2 -> 3.17309 10

kw1 -> 84503.8

kw2 -> 30577.
```

```
flex1 -> 0.00184336                                          Sqrt[nw1 M11
Y1/(m1+m2)/g]*c1^(3/2)

flex2 -> 0.000986004                                         Sqrt[nw2 M21
Y2/(m2)/g]*c2^(3/2)

kbuz -> 250.046

kblz -> 307.126

bdu -> 0.196635

bdl -> 0.0568875

m12 -> 10.024

I1xy -> 0

I1yz -> 0

I1zx -> 0

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

Ibtxyl -> 0

Ibtyzl -> 0

Ibtzxl -> 0

I2xy -> 0

I2yz -> 0

I2zx -> 0

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03

zpad -> 0.05

phib -> 0.001

                 -16
M11 -> 7.88442 10

                 -16
M12 -> 7.88442 10

                 -17
M21 -> 8.01224 10

                 -17
M22 -> 8.01224 10

temperature -> 290.
```

```
                       -23
boltzmann -> 1.38066 10

                     -7
alphasilica -> 5.1 10

betasilica -> 0.00015

rhosilica -> 2200.

Csilica -> 772.

Ksilica -> 1.38

                in
Ysilica -> 7. 10

                  -8
phisilica -> 2. 10

                    -11
phissilica -> 3.3 10

rhosteel -> 7800.

Csteel -> 486.

Ksteel -> 49.

                 11
Ysteel -> 2.12 10

alphasteel -> 0.000012

betasteel -> -0.00025

phisteel -> 0.0002

rhomarag -> 7800.

Cmarag -> 460.

Kmarag -> 20.

                 11
Ymarag -> 1.87 10

alphamarag -> 0.000011

betamarag -> -0.00025

phimarag -> 0.0001

tmU -> 0.0015                                                     MC blade size

tmL -> 0.001                                                      input modecleaner lower blades

magicnumber -> 0.0737472

deltabladeU -> 0.00182883                                         cf Bench delta_v1

deltabladeL -> 0.00182883                                         cf Bench delta_v3

deltawireU -> 0.00256766                                          cf Bench delta_h1

deltawireL -> 0.002595                                            cf Bench delta_h3

taubladeU -> 0.0408983

taubladeL -> 0.018177

tauwireU -> 0.00072307                                            cf Bench tau_steel1

tauwireL -> 0.000230501                                           cf Bench tau_steel3

                                        0.000469956 #1
damping[imag, bladeUtype] -> (0.0001 + ----------------- & )
                                                      2
                                       1 + 0.0660344 #1

                                         0.00020887 #1
```

```
damping[imag, bladeLtype] -> (0.0001 + ----------------- & )
                                                     2
                                       1 + 0.0130438 #1

damping[imag, wireUtype] -> (0.0002 & )

damping[imag, wireLtype] -> (0.0002 & )

                                         0.0000116653 #1
damping[imag, wireUatype] -> (0.0002 + -------------------- & )
                                                        2
                                       1 + 0.0000206405 #1

                                                   -6
                                         3.75828 10   #1
damping[imag, wireLatype] -> (0.0002 + -------------------- & )
                                                     -6   2
                                       1 + 2.09751 10   #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx1 -> 0

kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0

si -> 0.03                                                        D0900295
```

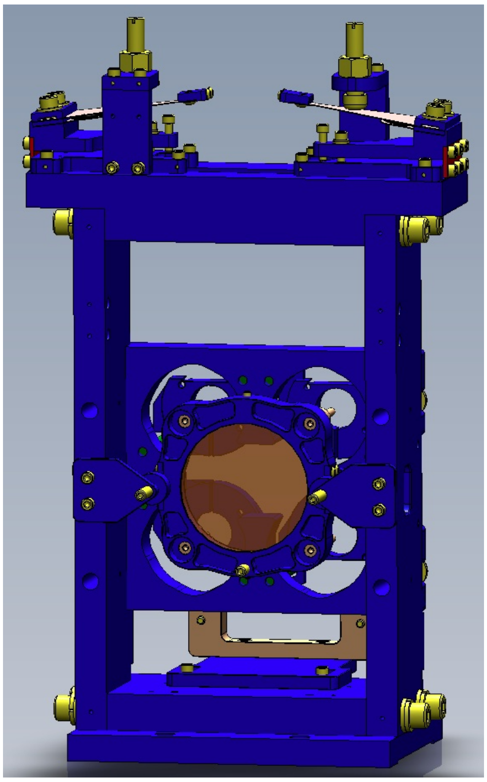
Back to Top

aLIGO SUS Operations Manual - Info on HTTS (Tip-Tilt)

Back to Operation Manual main page

The HTTS suspensions are the responsibility of the ASC group, but have been adopted by SUS to some extent. They hold RM1, RM2, OM1, OM2 and OM3.

Unlike other suspensions (QUAD, BSFM etc), the software for HTTS is filed under ASC, e.g., the Simulink model for the H1 TipTilts lives in the `cds_user_apps` (`userapps`) repository at `^/trunk/asc/h1/models/h1asctt`.


(BOSEMs not shown)

References

- Tip-tilt group's wiki page
- D1001396 HTTS Top Level Assembly
- Stuff on HAUX and HTTS actuation ranges from LLO alog
- T1400030: aLIGO HTTS Controls Design Description

Models

The HTTS suspension has been modeled in Mathematica/Matlab. Key results are post at Suspensions/OpsManual/HTTS/Models.

Screens

HTTS MEDM screens are at /Screens.

aLIGO: Suspensions/OpsManual/HTTS (last edited 2014-01-30 11:55:34 by MarkBarton)

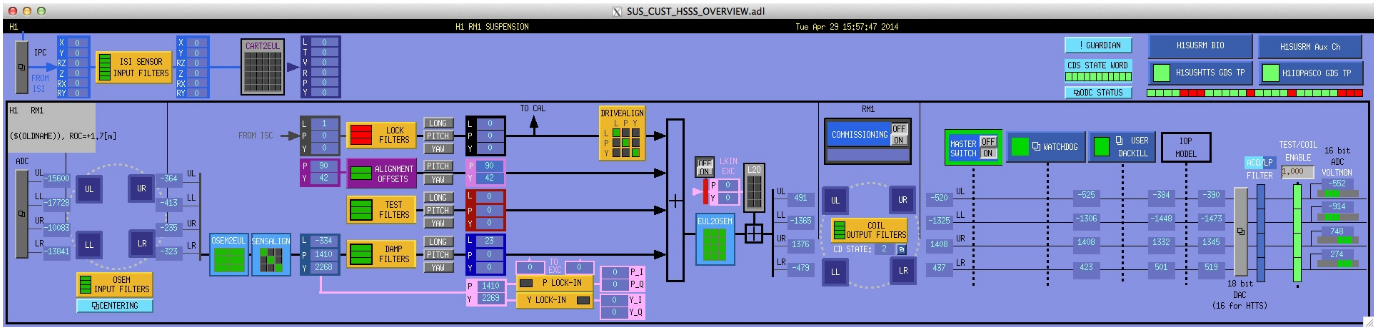
aLIGO SUS Operations Manual - Overview of HTTS MEDM screens

Back to Operation Manual main page
Back to HTTS main page

The HTTS screens described below are common to all HTTS (RM1-2, OM1-3) and live at `/opt/rtcds/userapps/release/sus/common/medm/htts/`. They are all generic screens which rely for their execution on site-, interferometer- and suspension-specific arguments passed in when they are called. They can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_HSSS_OVERVIEW.adl



This is the overview screen . There is only one sensor-actuator group - M1:

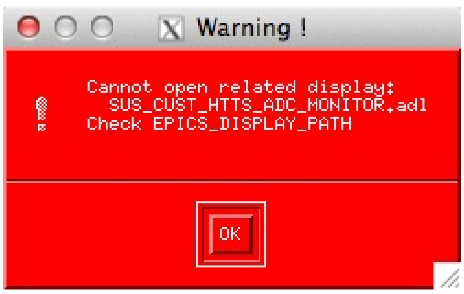
- Other
- M1 - 4 AOSEMs on the structure engaging the optic

M1 is used for local damping (relative to the structure). DC pitch and yaw offsets are also injected at M1.

Other Screens

- MASTER SWITCH - a manual on/off switch for all actuation
- COMMISSIONING - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- ADC - a monitor screen (common to all sensor-actuator groups) for the semi-raw ADC values as reported by the IOP process after downsampling.
- DAC - a popup allowing access to all the DAC outputs as reported by the IOP process.
- USER DACKILL - a watchdog that shuts off all sensor-actuator groups for the individual suspension but not the IOP process (potentially other suspensions on the same front-end).
- Note: the HTTSs do not have an IOP watchdog
- ODC STATUS - ODC status.
- IPC - interprocess control signals.
- H1IOPASC0 GDS TP (or the like; depends on the front-end computer name) - status of the IOP process for the front-end.
- H1SUSHTTS GDS TP (or the like; depends on the suspension name) - status of the suspension process.
- H1SUSHTTS BIO dummy warning screen - there are no binary input/outputs to control.
- H1SUSRM Aux Ch (or the like; depends on the suspension name) - readbacks from coil driver channels for all OSEMs as reported by the corresponding auxiliary front-end processor.
- GUARDIAN - Guardian.
- ISIINF - ISI input filters.
- CART2EUL - transformation from the ISI basis to SUS.

Screen SUS_CUST_HTTS_ADC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

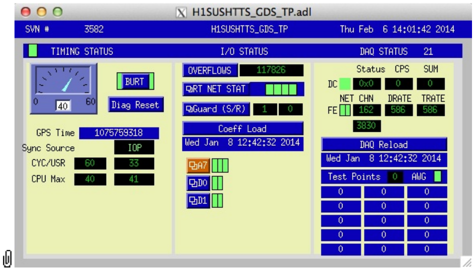
Screen SUS_CUST_HTTS_DAC_MONITOR.adl

Screen SUS_CUST_HTTS_DAC_MONITOR.adl

Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen H1SUSHTTS_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

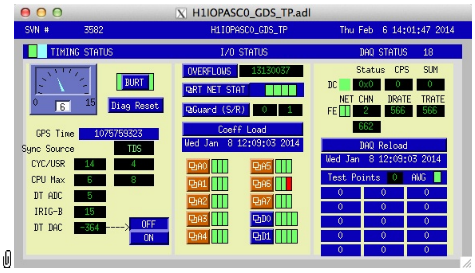
This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1IOPASC0_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

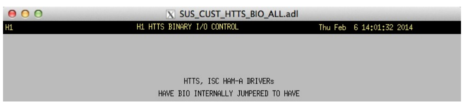
Screen SUS_CUST_HSSS_MONITOR_OVERVIEW.adl

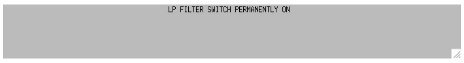
Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

Screen SUS_CUST_HTTS_BIO_ALL.adl

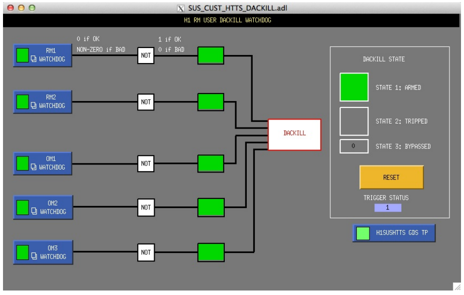




Suspensions/OpsManual/Boilerplate/SUS_CUST_HTTS_BIO_ALL:

This is a dummy screen that explains that the binary input output settings have been jumpered in hardware and are not under CDS control.

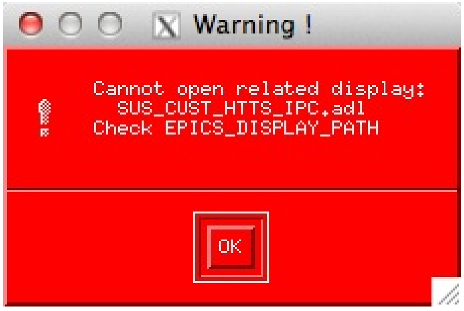
Screen SUS_CUST_HTTS_DACKILL.adl



Suspensions/OpsManual/Boilerplate/SUS_CUST_HTTS_DACKILL:

A single user model runs all the HTTSs so there is a joint USER DACKILL screen. See Suspensions/OpsManual/Watchdogs for more info.

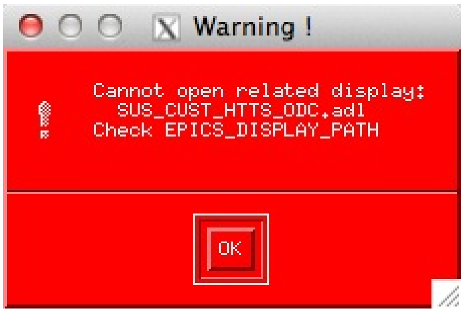
Screen SUS_CUST_HTTS_IPC.adl



Suspensions/OpsManual/Boilerplate/IPC:

Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

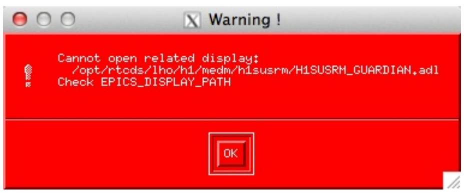
Screen SUS_CUST_HTTS_ODC.adl



Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

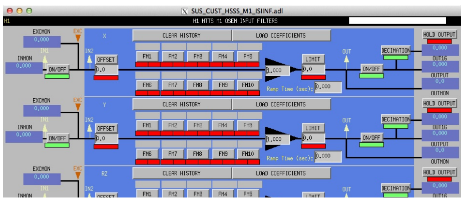
Screen H1SUSRM_GUARDIAN.adl

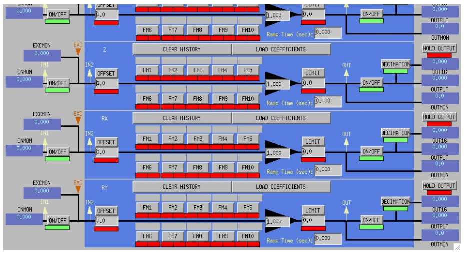


Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

Screen SUS_CUST_HSSS_M1_ISIINF.adl

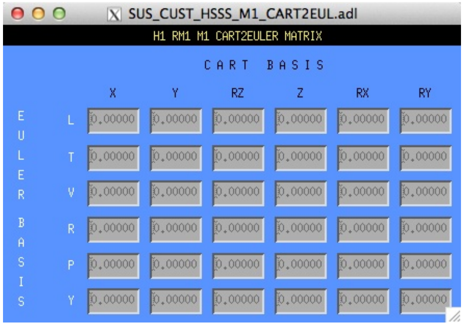




Suspensions/OpsManual/Boilerplate/M1_ISIINF:

Filters for the diagnostic signals from the ISI.

Screen SUS_CUST_HSSS_M1_CART2EUL.adl



Suspensions/OpsManual/Boilerplate/M1_CART2EUL:

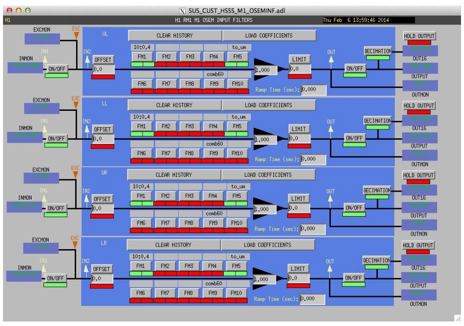
A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

Sensor Actuator Group M1 - Optic

Most of the items on the M1 section correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- LOCKIN P and Y - filters for global control signals.
- DAMP FILTERS - a filter bank applying the damping filters.
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.
- LOCK FILTERS - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.
- WATCHDOG - a block implementing the watchdog on the M1 sensor actuator group.

Screen SUS_CUST_HSSS_M1_OSEMINF.adl



Suspensions/OpsManual/Boilerplate/M1_OSEMINF:

This block has 6 filter groups corresponding to the 6 M1 BOSEMs, T1/T2/T3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

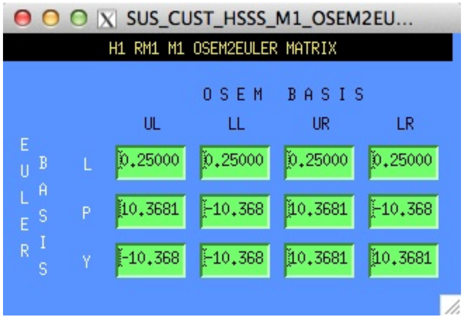
The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_HSSS_M1_OSEM_ALIGN.adl

Suspensions/OpsManual/Boilerplate/M1_OSEM_ALIGN:

These screens give various views of the OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_HSSS_M1_OSEM2EUL.adl



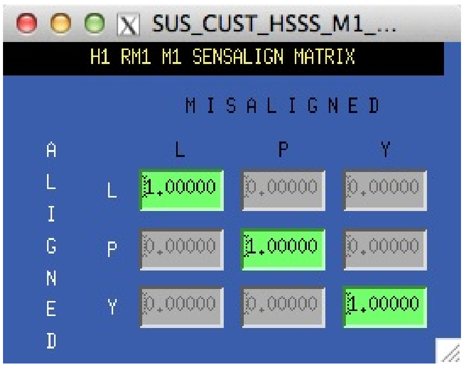
Suspensions/OpsManual/Boilerplate/M1_OSEM2EUL:

This screen allows entry of the matrix which converts from the M1 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

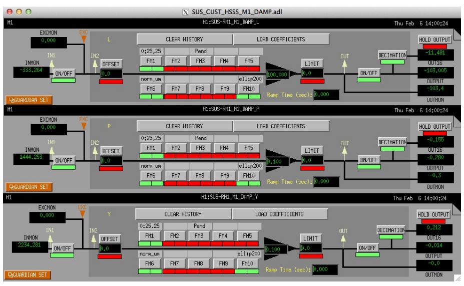
Screen SUS_CUST_HSSS_M1_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_SENSALIGN:

This screen is reserved for tweaking the M1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_HSSS_M1_DAMP.adl

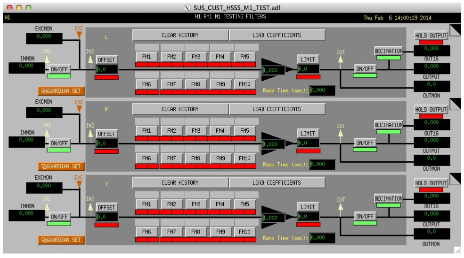


Suspensions/OpsManual/Boilerplate/M1_DAMP:

These filters implement the local damping. See Suspensions/OpsManual/Damping

These filters implement the local damping. See Suspensions/OpsManual/Damping.

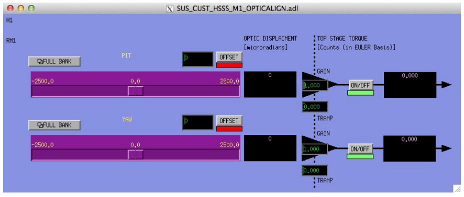
Screen SUS_CUST_HSSS_M1_TEST.adl



Suspensions/OpsManual/Boilerplate/M1_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

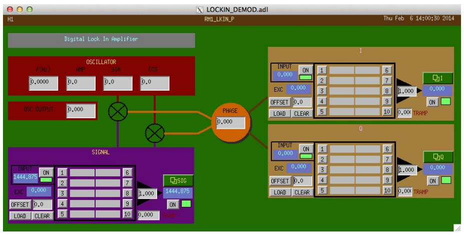
Screen SUS_CUST_HSSS_M1_OPTICALIGN.adl

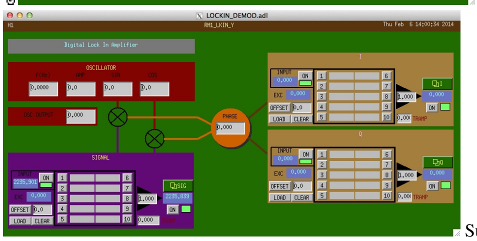


Suspensions/OpsManual/Boilerplate/M1_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets. See Suspensions/OpsManual/Alignments for more info.

Screen LOCKIN_DEMOD.adl (two versions: P and Y)

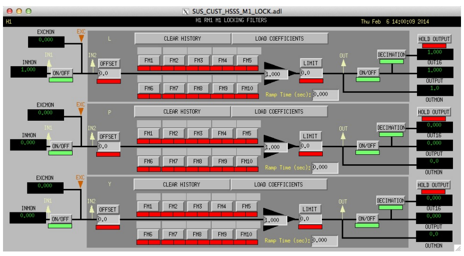




Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

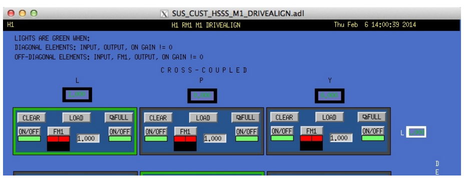
Screen SUS_CUST_HSSS_M1_LOCK.adl



Suspensions/OpsManual/Boilerplate/M1_LOCK:

Filters for the locking signals.

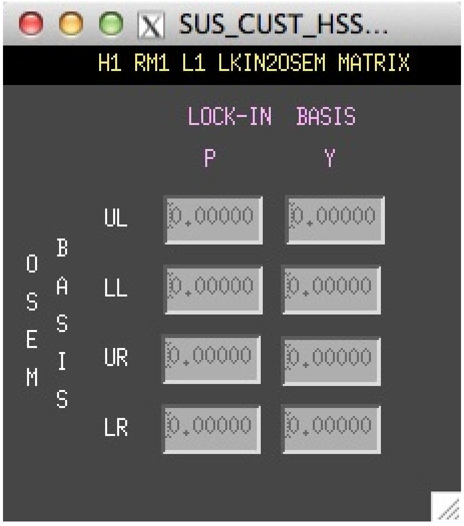
Screen SUS_CUST_HSSS_M1_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_DRIVEALIGN:

This screen is reserved for tweaking the M1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

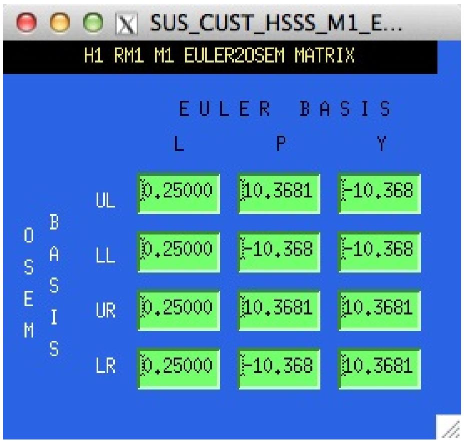
Screen SUS_CUST_HSSS_M1_LKIN2OSEM.adl



Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

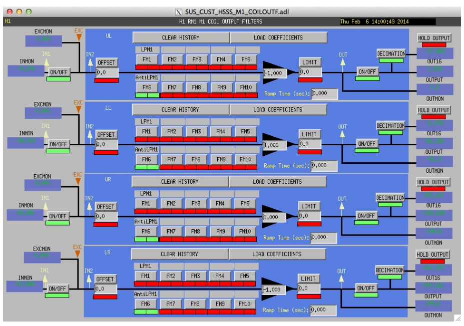
Screen SUS_CUST_HSSS_M1_EUL2OSEM.adl



Suspensions/OpsManual/Boilerplate/M1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_HSSS_M1_COILOUTF.adl



Suspensions/OpsManual/Boilerplate/M1_COILOUTF:

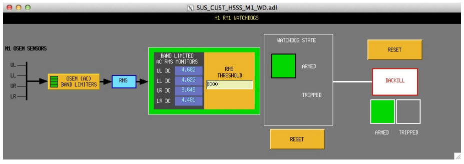
This screen applies compensation for the hardware filters in the M1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

Screen SUS_CUST_HSSS_M1_WD.adl



Suspensions/OpsManual/Boilerplate/M1_WD:

This screen allows for setting the M1 watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

aLIOG SUS Operations Manual - HTTS (Tip-Tilt) Models

Back to Operation Manual main page

The Tip-Tilt suspension has been modeled in Mathematica. See Suspensions/MathematicaModels. Results will go here. There is also a Matlab equivalent. See `^/trunk/Common/MatlabTools/SingleModel_Production` in the SUS SVN.

Production Model

20140123HTTSdamp has new, correct data from D1001396-v3 for the MOIs. (The previous values from P1100090-v5 = Rev. Sci. Instrum. 82, 125108, were off by 10 in I0x, I0y and I0z and slightly out of date.) The final numbers are still an excellent match to the observed mode frequencies. Corresponds to httsopt_damp.m r6162 of 1/23/14.

Old Models

20130723TipTilt has data from P1100090-v5 = Rev. Sci. Instrum. 82, 125108, but with I0x, I0y and I0z increased by factors of 10, 10 and 11.5 to match observed mode frequencies. Corresponds to httsopt_test.m r5233 of 7/23/13.

20130628TipTilt has data from P1100090-v5 = Rev. Sci. Instrum. 82, 125108. The relationship between tl0 and 'pendulum length' has been fixed - tl0 is inclusive of dpitch whereas Bram has clarified that 'pendulum length' was not. Corresponds to httsopt_test.m r5013 of 6/28/13.

20130627TipTilt is a first cut at the HTTS, with data from P1100090-v5 = Rev. Sci. Instrum. 82, 125108. Corresponds to httsopt_test.m of 6/27/13.

Data

Corrected info on masses and MOIs from Bram Slagmolen. D1001396

aLIGO: Suspensions/OpsManual/HTTS/Models (last edited 2014-05-27 09:14:27 by MarkBarton)

Case 20140123HTTSdamp of Mathematica model TwoWireSimpleBlades

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
"20140123HTTSdamp"
```

```
20140123HTTSdamp. Uses tweaked model with support for ECD. Corresponds to httsopt_damp.m r6162
of 1/23/14, including damping. Non-ECD damping disabled. New as-built MOI data from D1001396-v3
relative to 20130723TipTilt.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeL1	modeT1	modeY1	modeP1	modeV1	modeR1

Mode Summary

Back to Top

```
N   f         name     type

1   1.30721   modeL1   pitch0

2   1.33756   modeT1   roll0

3   1.58336   modeY1   yaw0

4   1.69967   modeP1   pitch0

5   6.12635   modeV1   z0

6   8.73955   modeR1   roll0
```

Violin Modes	Optic
f (Hz), n=1-4	243.907 490.052 740.742 998.426
Q, n=1-4	58299.7 31376. 18723.3 12129.5

Mode Shapes

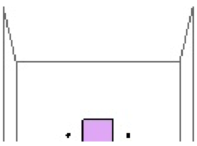
Mode #1 - modeL1

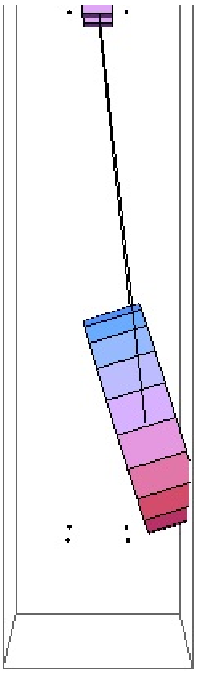
Back to Top

```
1.30721 Hz
```

```
        x           y    z    yaw    pitch       roll

optic   0.0527637   0    0    0      -0.998607   0
```



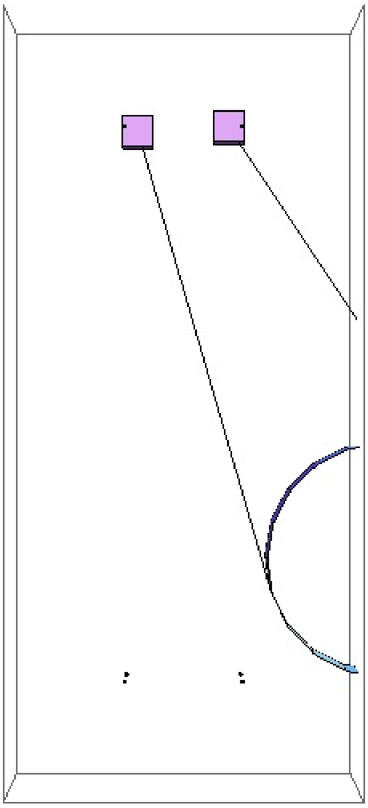


Mode #2 - modeT1

Back to Top

```
1.33756 Hz
```

	x	y	z	yaw	pitch	roll
optic	0	-0.212819	0	0	0	-0.977092



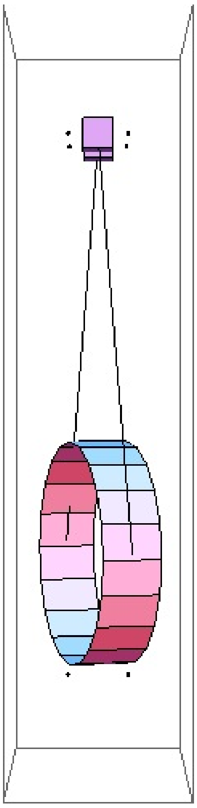
Mode #3 - modeY1

Back to Top

```
1.58336 Hz
```

	x	y	z	yaw	pitch	roll

	x	y	z	yaw	pitch	roll
optic	0	0	0	1.	0	0

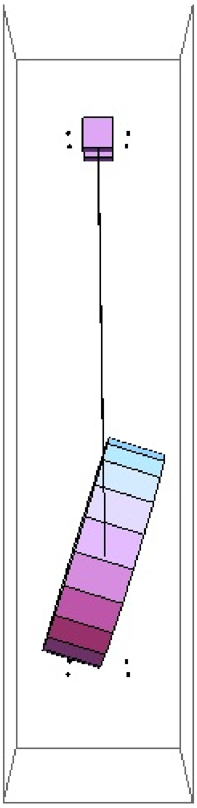


Mode #4 - modeP1

Back to Top

```
1.69967 Hz
```

	x	y	z	yaw	pitch	roll
optic	0.00656515	0	0	0	0.999978	0

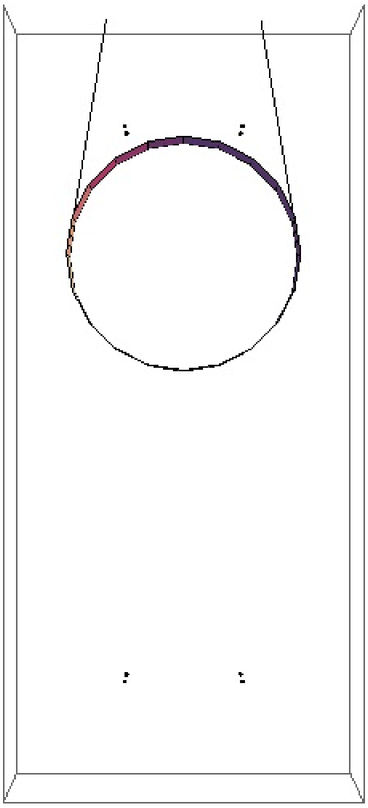


Mode #5 - modeV1

Back to Top

```
6.12635 Hz
```

	x	y	z	yaw	pitch	roll
optic	0	0	1.	0	0	0

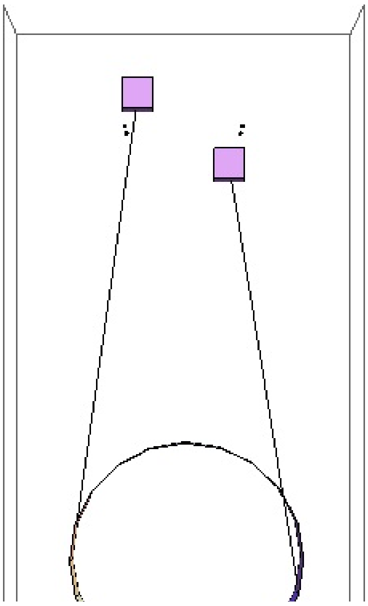


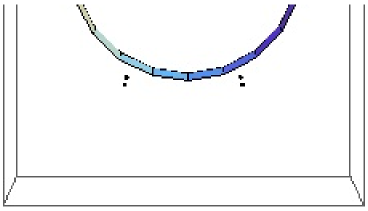
Mode #6 - modeR1

Back to Top

```
8.73955 Hz
```

	x	y	z	yaw	pitch	roll
optic	0	-0.00332535	0	0	0	0.999994





Parameters

Back to Top

```
g -> 9.81

tx -> 0.02                              P1100090, USED ONLY FOR DRAWING

tr -> 0.0381                            P1100090, USED ONLY FOR DRAWING

den0 -> 4000.

m0 -> 0.0885                            D1001396-v3

COMx -> 0

COMy -> 0

COMz -> 0

I0x -> 0.0000641                        D1001396-v3

I0y -> 0.0000307                        D1001396-v3

I0z -> 0.0000377                        D1001396-v3

I0xy -> 0

I0yz -> 0

I0zx -> 0

                                                             2                       2
l0 -> 0.141893                          Sqrt[(tl0 - dpitch)  + (dyaw2/2 - dyaw1/2) ]

nw0 -> 2

        127
r0 -> -------
      2000000                           P1100090

               11
Y0 -> 2.119 10                          measured, MB

dpitch -> 0.0013                        P1100090

dyaw1 -> 0.03                           P1100090

dyaw2 -> 0.0762                         P1100090

tl0 -> 0.1413                           P1100090

ltotal -> 0.1413

dpend -> 0.1413

ul0 -> 0.14187

sl0 -> 0.141893

si0 -> 0.162799

c0 -> 0.986659

                 -8
A0 -> 1.26677 10

kw0 -> 18917.7

flex0 -> 0.00244691
```

```
                 -17
M01 -> 1.27698 10

                 -17
M02 -> 1.27698 10

vertblades -> True

pitchbl -> 0

pitchbr -> 0

rollbl -> 0

rollbr -> 0

kbz -> 65.8                              P1100090

bd -> 0.00659715

btx -> 0.01                              display size of blade tip

bty -> 0.01                              display size of blade tip

btz -> 0.01                              display size of blade tip

temperature -> 290.

                       -23
boltzmann -> 1.38066 10

                     -7
alphasilica -> 5.1 10

betasilica -> 0.0002

rhosilica -> 2200.

Csilica -> 772.

Ksilica -> 1.38

                10
Ysilica -> 7. 10

                  -8
phisilica -> 2. 10

                    -11
phissilica -> 3.3 10

rhosteel -> 7800.

Csteel -> 486.

Ksteel -> 49.

                 11
Ysteel -> 1.65 10

alphasteel -> 0.000012

betasteel -> -0.00025

phisteel -> 0.0002

rhomarag -> 7800.

Cmarag -> 460.

Kmarag -> 20.

                 11
Ymarag -> 1.65 10

alphamarag -> 0.000011

betamarag -> 0.

phimarag -> 0.0001
```

```
tmU -> 0.0045

tmI -> 0.0055

tmL -> 0.005

deltablade -> 0.00161367

taublade -> 0.368085

damping[imag, bladetype] -> (0 & )

magicnumber -> 0.0732

deltafibre -> 0.00183364

taufibre -> 0.0000913384

damping[imag, fibretype] -> (0 & )

damping[imag, fibreatype] -> (0 & )

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx0 -> 0

kcony0 -> 0

kconz0 -> 0

kconyaw0 -> 0

kconpitch0 -> 0

kconroll0 -> 0

damping[real, ECDtype] -> (0 & )

damping[imag, ECDtype] -> (2 Pi #1 & )

bx1 -> 0.00885

by1 -> 0.00885

bz1 -> 0.00885

byaw1 -> 0.00001885

                  -6
bpitch1 -> 3.07 10

                 -6
broll1 -> 6.41 10
```

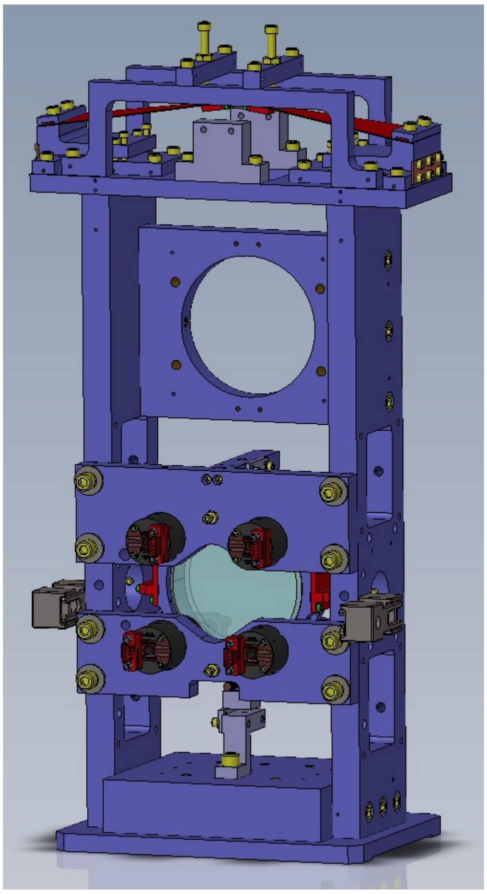
Back to Top

aLIGO SUS Operation Manual - Info on HAUX

Back to Operation Manual main page

Overview

The HAUX suspensions are the responsibility of the IOO group, but to some extent have been adopted by SUS. They hold the input optics IM1, IM2, IM3 and IM4.



References

- https://dcc.ligo.org/LIGO-D1000120 HAUX Top Assembly
- Stuff on HAUX and HTTS actuation ranges from LLO alog
- T1400029 aLIGO HAUX Controls Design Description

Models

The HAUX suspension has been modeled in Mathematica/Matlab. Key results are posted at Suspensions/OpsManual/HAUX/Models .

Screens

HAUX MEDM screens are at /Screens.

aLIGO: Suspensions/OpsManual/HAUX (last edited 2014-01-30 11:56:41 by MarkBarton)

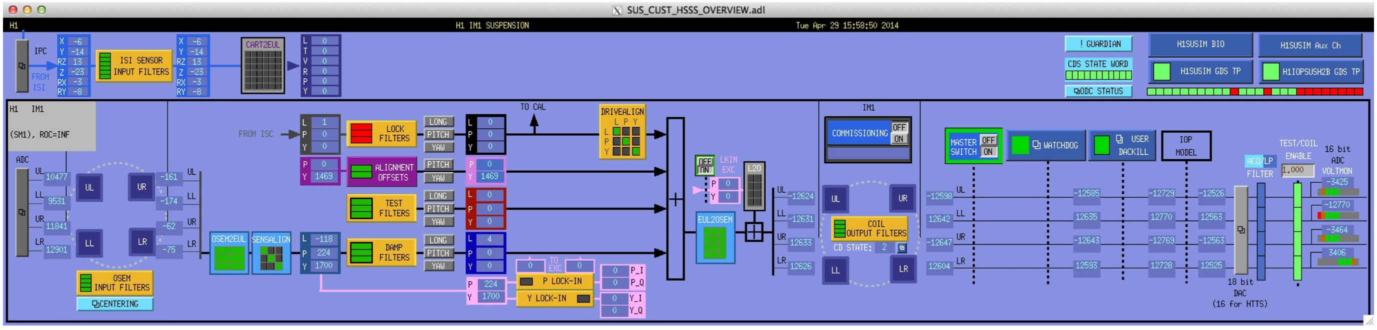
aLIGO SUS Operations Manual - Overview of HAUX MEDM screens

Back to Operation Manual main page
Back to HAUX main page

The HAUX screens described below are common to all HAUX (IM1-4) and live at `/opt/rtcds/userapps/release/sus/common/medm/haux/`. They are all generic screens which rely for their execution on site-, interferometer- and suspension-specific arguments passed in when they are called. They can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_HSSS_OVERVIEW.adl



This is the overview screen . There is only one sensor-actuator group - M1:

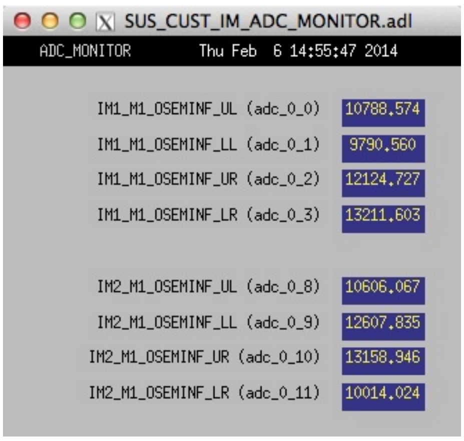
- Other
- M1 - 4 AOSEMs on the structure engaging the optic

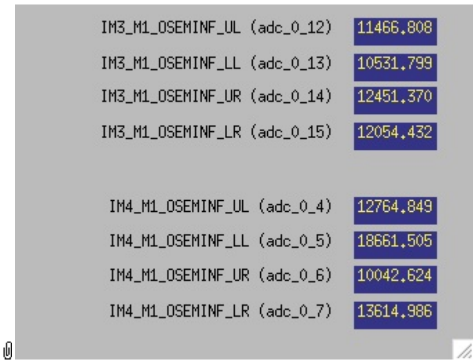
M1 is used for local damping (relative to the structure). DC pitch and yaw offsets are also injected at M1.

Other Screens

- MASTER SWITCH - a manual on/off switch for all actuation
- COMMISSIONING - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- ADC - a monitor screen (common to all sensor-actuator groups) for the semi-raw ADC values as reported by the IOP process after downsampling.
- DAC - a popup allowing access to all the DAC outputs as reported by the IOP process.
- USER DACKILL - a watchdog that shuts off all sensor-actuator groups for the individual suspension but not the IOP process (potentially other suspensions on the same front-end).
- Note: the IMs do not have an IOP watchdog
- ODC STATUS - ODC status.
- IPC - interprocess control signals.
- H1SUSIM GDS TP (or the like) - status of the suspension process.
- H1IOPSUSH2B GDS TP (or the like) - status of the IOP process for the front-end.
- H1SUSIM BIO (or the like) - binary input/output controls.
- H1SUSIM Aux Ch (or the like) - readbacks from coil driver channels for all OSEMs as reported by the corresponding auxiliary front-end processor.
- GUARDIAN - Guardian.
- ISIINF - ISI input filters.
- CART2EUL - transformation from the ISI basis to SUS.

Screen SUS_CUST_IM_ADC_MONITOR.adl

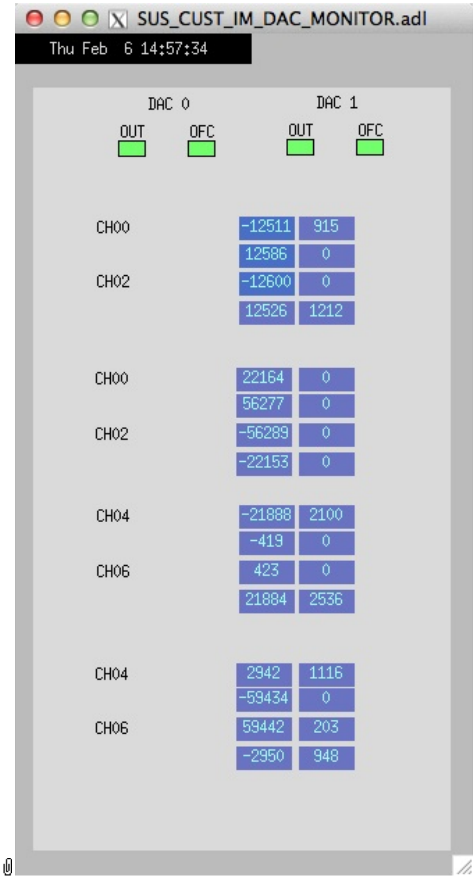




Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

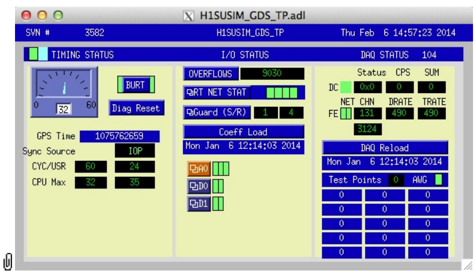
Screen SUS_CUST_IM_DAC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen H1SUSIM_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1IOPSUSH2B_GDS_TP.adl

Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

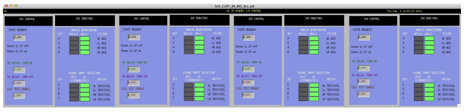
Screen SUS_CUST_HSSS_MONITOR_OVERVIEW.adl

Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

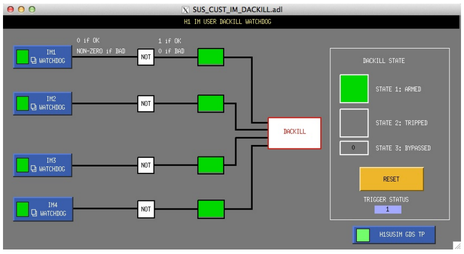
- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

Screen SUS_CUST_IM_BIO_ALL.adl


Suspensions/OpsManual/Boilerplate/SUS_CUST_IM_BIO_ALL:

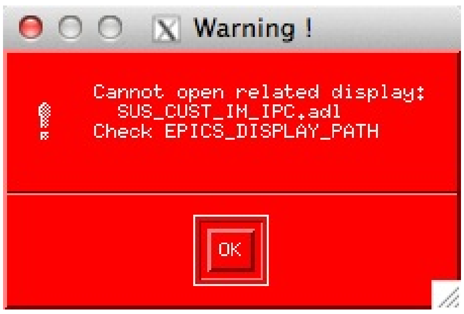
The binary input/output (BIO) settings for all the IM suspensions.

Screen SUS_CUST_IM_DACKILL.adl


Suspensions/OpsManual/Boilerplate/SUS_CUST_IM_DACKILL:

One user model runs all the IM suspensions, so there is a joint USER DACKILL. See Suspensions/OpsManual/Watchdogs for more info.

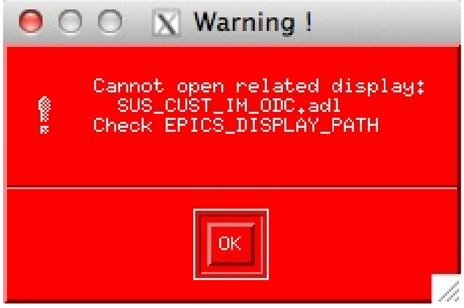
Screen SUS_CUST_IM_IPC.adl



Suspensions/OpsManual/Boilerplate/IPC:

Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

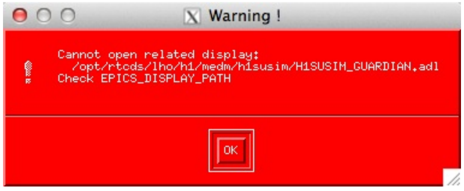
Screen SUS_CUST_IM_ODC.adl



Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

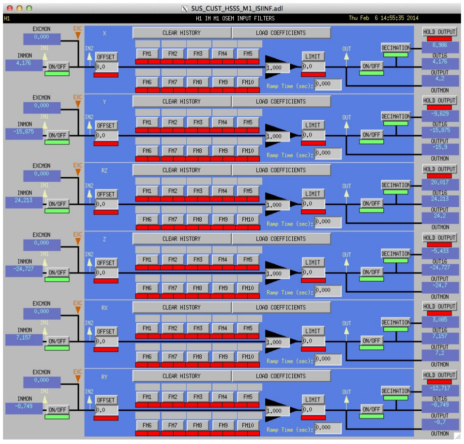
Screen H1SUSIM_GUARDIAN.adl



Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

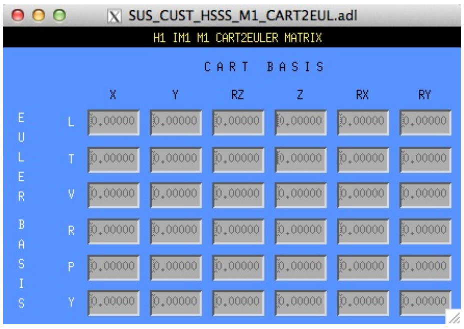
Screen SUS_CUST_HSSS_M1_ISIINF.adl



Suspensions/OpsManual/Boilerplate/M1_ISIINF:

Filters for the diagnostic signals from the ISI.

Screen SUS_CUST_HSSS_M1_CART2EUL.adl



Suspensions/OpsManual/Boilerplate/M1_CART2EUL:

A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

Sensor Actuator Group M1 - Optic

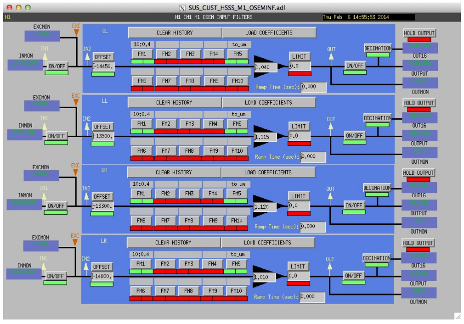
Most of the items on the M1 section correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by

default).

- LOCKIN P and Y - filters for global control signals.
- DAMP FILTERS - a filter bank applying the damping filters.
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.
- LOCK FILTERS - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ???.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.
- WATCHDOG - a block implementing the watchdog on the M1 sensor actuator group.

Screen SUS_CUST_HSSS_M1_OSEMINF.adl



Suspensions/OpsManual/Boilerplate/M1_OSEMINF:

This block has 6 filter groups corresponding to the 6 M1 BOSEMs, T1/T2/T3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_HSSS_M1_OSEM_ALIGN.adl

Suspensions/OpsManual/Boilerplate/M1_OSEM_ALIGN:

These screens give various views of the OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_HSSS_M1_OSEM2EUL.adl

Suspensions/OpsManual/Boilerplate/M1_OSEM2EUL:

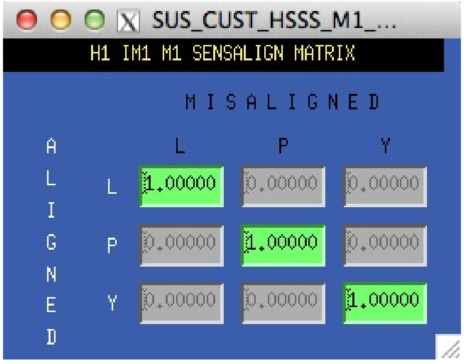
This screen allows entry of the matrix which converts from the M1 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the

AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

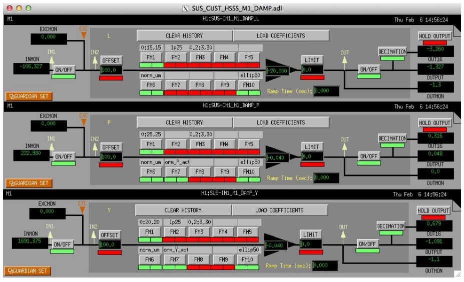
Screen SUS_CUST_HSSS_M1_SENSALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_SENSALIGN:

This screen is reserved for tweaking the M1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

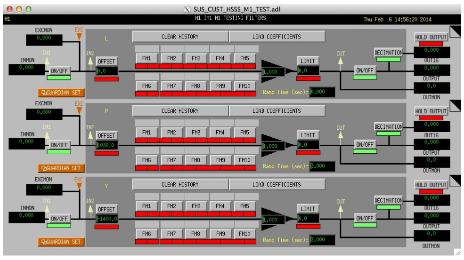
Screen SUS_CUST_HSSS_M1_DAMP.adl



Suspensions/OpsManual/Boilerplate/M1_DAMP:

These filters implement the local damping. See Suspensions/OpsManual/Damping.

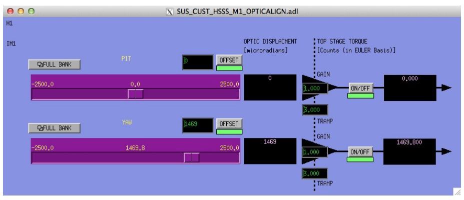
Screen SUS_CUST_HSSS_M1_TEST.adl



Suspensions/OpsManual/Boilerplate/M1_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

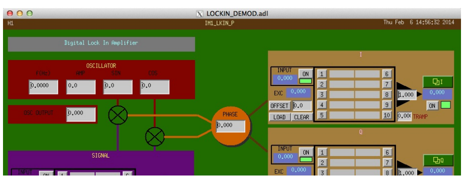
Screen SUS_CUST_HSSS_M1_OPTICALIGN.adl

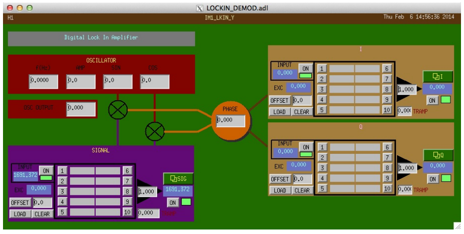


Suspensions/OpsManual/Boilerplate/M1_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets. See Suspensions/OpsManual/Alignments for more info.

Screen LOCKIN_DEMOD.adl (two versions: P and Y)

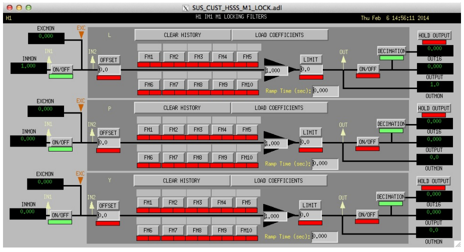




Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

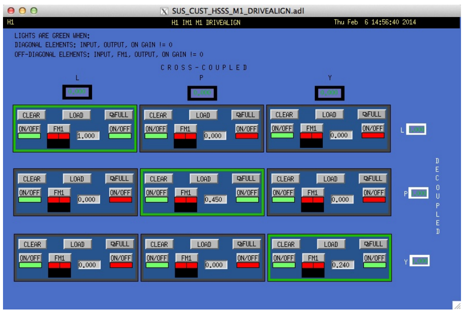
Screen SUS_CUST_HSSS_M1_LOCK.adl



Suspensions/OpsManual/Boilerplate/M1_LOCK:

Filters for the locking signals.

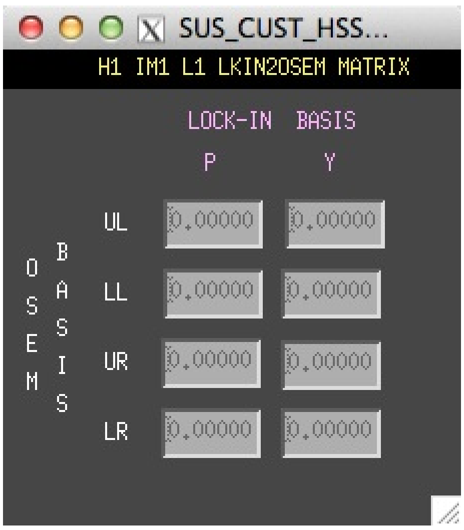
Screen SUS_CUST_HSSS_M1_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_DRIVEALIGN:

This screen is reserved for tweaking the M1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

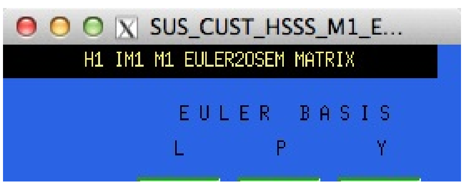
Screen SUS_CUST_HSSS_M1_LKIN2OSEM.adl

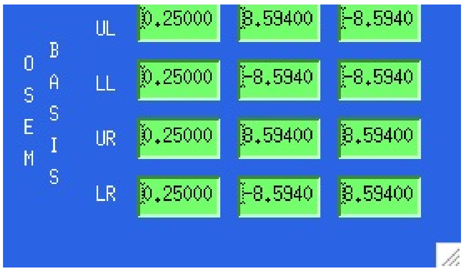


Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_HSSS_M1_EUL2OSEM.adl

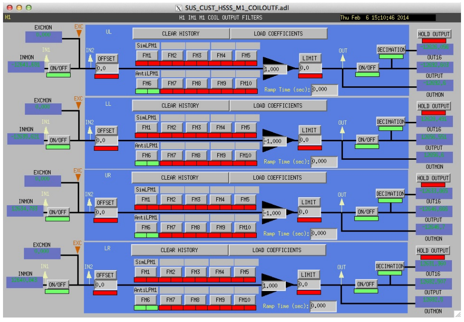




Suspensions/OpsManual/Boilerplate/M1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

Screen SUS_CUST_HSSS_M1_COILOUTF.adl



Suspensions/OpsManual/Boilerplate/M1_COILOUTF:

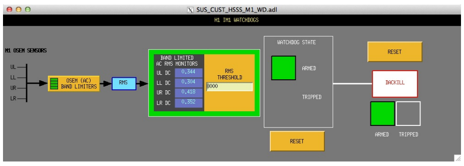
This screen applies compensation for the hardware filters in the M1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

Screen SUS_CUST_HSSS_M1_WD.adl



Suspensions/OpsManual/Boilerplate/M1_WD:

This screen allows for setting the M1 watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

aLIGO SUS Operations Manual - HAUX Models

Back to Operation Manual main page

The HAUX suspension has been modeled in Mathematica and Matlab. See Suspensions/MathematicaModels.

HAUX Production Model

The current best HAUX model is /20131231HAUXdamp. It includes the built-in eddy-current damping in both the Matlab and Mathematica versions.

HAUX_Design Model

/HAUX_Design

aLIGO: Suspensions/OpsManual/HAUX/Models (last edited 2014-05-27 09:14:00 by MarkBarton)

Case 20131231HAUXdamp of Mathematica model TwoWireSimpleBlades

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
"20131231HAUXdamp"
```

```
20131231HAUXdamp. Uses tweaked model with support for ECD. Corresponds to hauxopt_damp.m rTBD of TBD, including damping. Non-ECD damping disabled.
```

Links to Result Sections

Pendulum Mode Summary	Violin Mode Summary	Pendulum Mode Shapes	Parameters

modeY1	modeL1	modeT1	modeP1	modeV1	modeR1

Pendulum Mode Summary

Back to Top

```
N   f          name     type

1   0.754235   modeY1   yaw0

2   0.976308   modeL1   pitch0

3   0.99325    modeT1   roll0

4   1.11873    modeP1   pitch0

5   7.09832    modeV1   z0

6   10.4881    modeR1   roll0
```

Violin Mode Summary

Back to Top

Violin Modes	Optic
f (Hz), n=1-4	228.297 456.904 686.131 916.292
Q, n=1-4	131563. 79259.2 56340.4 43486.7

Mode Shapes

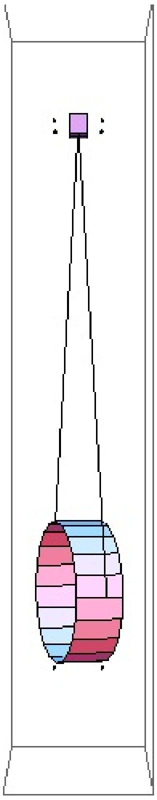
Mode #1 - modeY1

Back to Top

```
0.754235 Hz
```

```
        x   y   z   yaw   pitch   roll

optic   0   0   0   1.    0       0
```

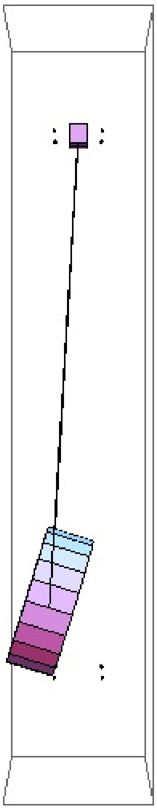


Mode #2 - modeL1

Back to Top

```
0.976308 Hz
```

	x	y	z	yaw	pitch	roll
optic	-0.050974	0	0	0	0.9987	0

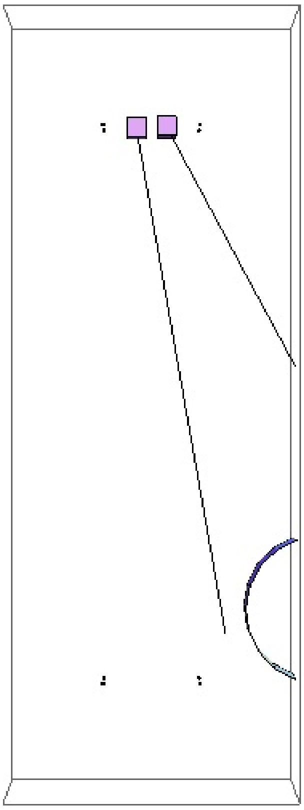


Mode #3 - modeT1

Back to Top

```
0.99325 Hz
```

	x	y	z	yaw	pitch	roll
optic	0	-0.27852	0	0	0	-0.96043

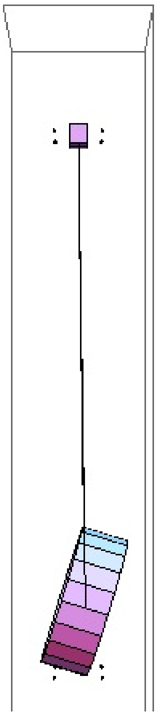


Mode #4 - modeP1

Back to Top

```
1.11873 Hz
```

	x	y	z	yaw	pitch	roll
optic	0.0111109	0	0	0	0.999938	0

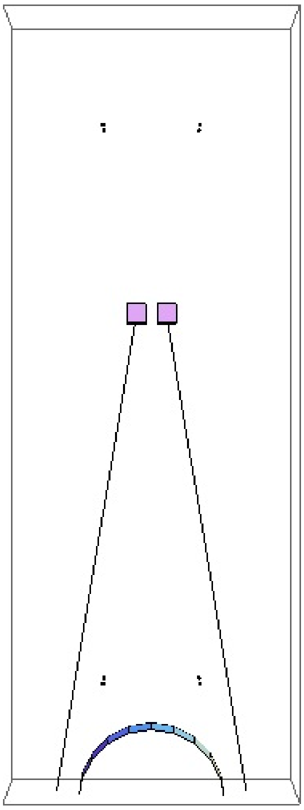


Mode #5 - modeV1

Back to Top

```
7.09832 Hz
```

	x	y	z	yaw	pitch	roll
optic	0	0	-1.	0	0	0

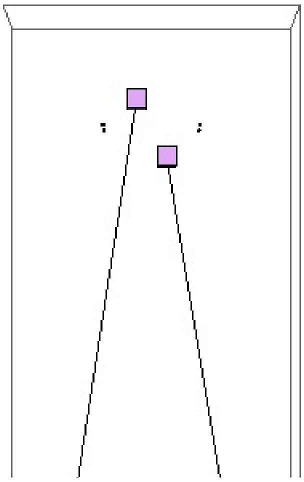


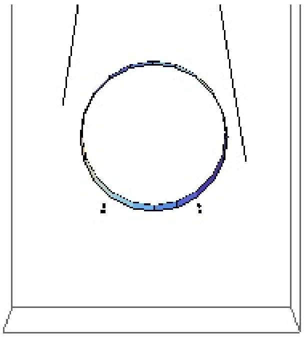
Mode #6 - modeR1

Back to Top

```
10.4881 Hz
```

	x	y	z	yaw	pitch	roll
optic	0	-0.00401153	0	0	0	0.999992





Parameters

Back to Top

```
g -> 9.81

tx -> 0.0254                            thickness of optic, USED ONLY FOR DRAWING

tr -> 0.0381                            radius of optic, USED ONLY FOR DRAWING

den0 -> 4000.

m0 -> 0.3738                            mass of the suspended assembly

COMx -> 0                               suspended assembly COM x coordinate (in local
coordinates)

COMy -> 0                               suspended assembly COM y coordinate (in local
coordinates)

COMz -> 0                               suspended assembly COM z coordinate (in local
coordinates)

I0x -> 0.000434854                      moment of inertia (roll) in body coordinates

I0y -> 0.000211996                      moment of inertia (pitch) in body coordinates

I0z -> 0.00026846                       moment of inertia (yaw) in body coordinates

I0xy -> 0

I0yz -> 0

I0zx -> 0

l0 -> 0.253842                          wire length

nw0 -> 2

r0 -> 0.0000762                         radius of wire

            11
Y0 -> 2.1 10                            wires' Young's modulus

dpitch -> 0.001                         height of wire break-off above optic local coordinate
origin

dyaw1 -> 0.0157                         y-separation of wires at structure

dyaw2 -> 0.10033                        y-separation of wires at optic

tl0 -> 0.25029                          vertical distance between wire attachment points

ltotal -> 0.25029

dpend -> 0.25029

ul0 -> 0.253719

sl0 -> 0.253842

si0 -> 0.166698

c0 -> 0.986008

                -8
```

```
A0 -> 1.82415 10

kw0 -> 15090.9

flex0 -> 0.00170508

                 -17
M01 -> 2.64795 10

                 -17
M02 -> 2.64795 10

vertblades -> True

pitchbl -> 0

pitchbr -> 0

rollbl -> 0

rollbr -> 0


2    2 m0
2    2 m0
                                         GUESSTIMATE-blade strength to give measured bounce
frequency (including wires' effects). The value coming from the blade modeling would be 381 (or
7.19  4 π  --). To match a resonace frequency (calculated for the blades, thus not accounting
for wires) use: f  4 π  --)
kbz -> 381.441
2
2

bd -> 0.00480675

btx -> 0.01                              display size of blade tip

bty -> 0.01                              display size of blade tip

btz -> 0.01                              display size of blade tip

temperature -> 290.

                       -23
boltzmann -> 1.38066 10

                     -7
alphasilica -> 5.1 10

betasilica -> 0.0002

rhosilica -> 2200.

Csilica -> 772.

Ksilica -> 1.38

                10
Ysilica -> 7. 10

                  -8
phisilica -> 2. 10

                    -11
phissilica -> 3.3 10

rhosteel -> 7800.

Csteel -> 486.

Ksteel -> 49.

                 11
Ysteel -> 1.65 10

alphasteel -> 0.000012

betasteel -> -0.00025

phisteel -> 0.0002
```

```
rhomarag -> 7800.

Cmarag -> 460.

Kmarag -> 20.

                  11
Ymarag -> 1.65 10

alphamarag -> 0.000011

betamarag -> 0.

phimarag -> 0.0001

tmU -> 0.0045

tmI -> 0.0055

tmL -> 0.005

deltablade -> 0.00161367

taublade -> 0.368085

damping[imag, bladetype] -> (0 & )

magicnumber -> 0.0732

deltafibre -> 0.00186476

taufibre -> 0.000131527

damping[imag, fibretype] -> (0 & )

damping[imag, fibreatype] -> (0 & )

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx0 -> 0

kcony0 -> 0

kconz0 -> 0

kconyaw0 -> 0

kconpitch0 -> 0

kconroll0 -> 0

damping[real, ECDtype] -> (0 & )

damping[imag, ECDtype] -> (2 Pi #1 & )

bx0 -> 0.1869

by0 -> 0.3738

bz0 -> 0.3738

byaw0 -> 0.000013423

bpitch0 -> 0.000423992

broll0 -> 0.000434854
```

Back to Top

aLIGO SUS Operation Manual - Info on TMTS Suspensions

Back to Operation Manual main page

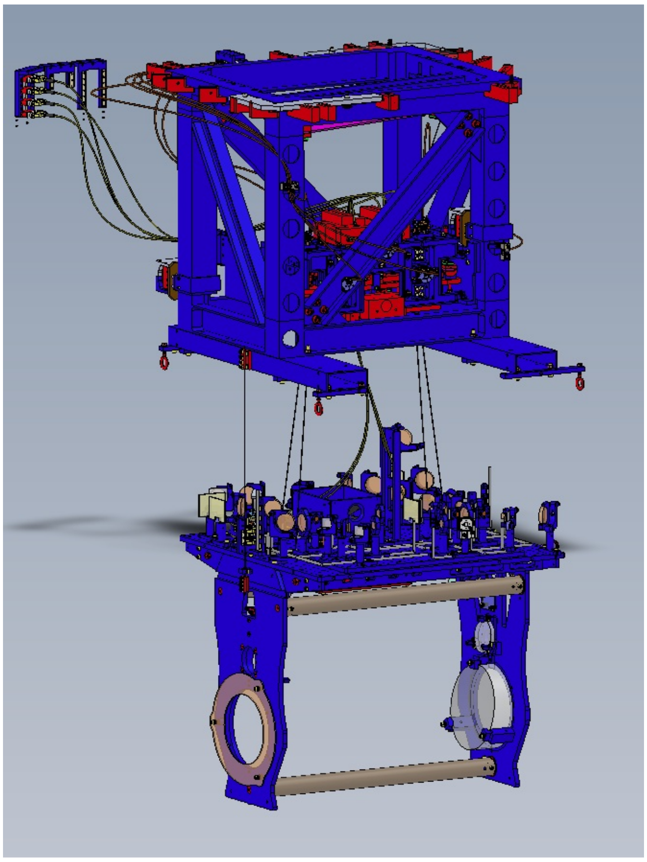
Background

The Transmon Telescope proper is suspended as the bottom mass of a double pendulum, with two blades in the structure, and two blades at the top mass. The telescope proper has a mass of ≈80 kg, equivalent to the bottom two masses of a quad. To save design effort a number of parts were taken over from the quad:

- The TMTS top mass is based on a QUAD UIM (upper intermediate mass; ≈22 kg), heavily ballasted with extra weight equivalent to that of a quad top mass (≈22 kg).
- The two TMTS top blades are QUAD top blades rated for a total of 22+22+40+40=124 kg, and are used with quad blade mounts in a quad upper structure weldment.

There are left- and right-handed versions of the TMTS (the one in the picture below is a "right"):

Left	H1:TMSY, H2:TMSX, L1:TMSY
Right	H1:TMSX, H2:TMSY, L1:TMSX



References

- D0901880: aLIGO AOS Transmission Monitor System Assembly
- E1100537: TRANSMISSION MONITOR TELESCOPE SUSPENSION (TMS) FINAL DESIGN
- T1200364: Double Pendulum Parameter Descriptions and Naming Convention (Note -v1 is inaccurate in several respects for TMTS.)
- D0900419: AdvLIGO SUS BSC6-H2, XYZ Local CS for TMS ETM Tel Assy (First Article in-chamber drawing and eDrawing)
- E1200045: TMTS Suspension Controls Arrangement Poster

- https://dcc.ligo.org/LIGO-T1300537 aLIGO TMTS Controls Design Description

Models

The TMTS suspension has been modelled. Key results are at Suspensions/OpsManual/TMTS/Models .

Screens

TMTS MEDM screens are at /Screens.

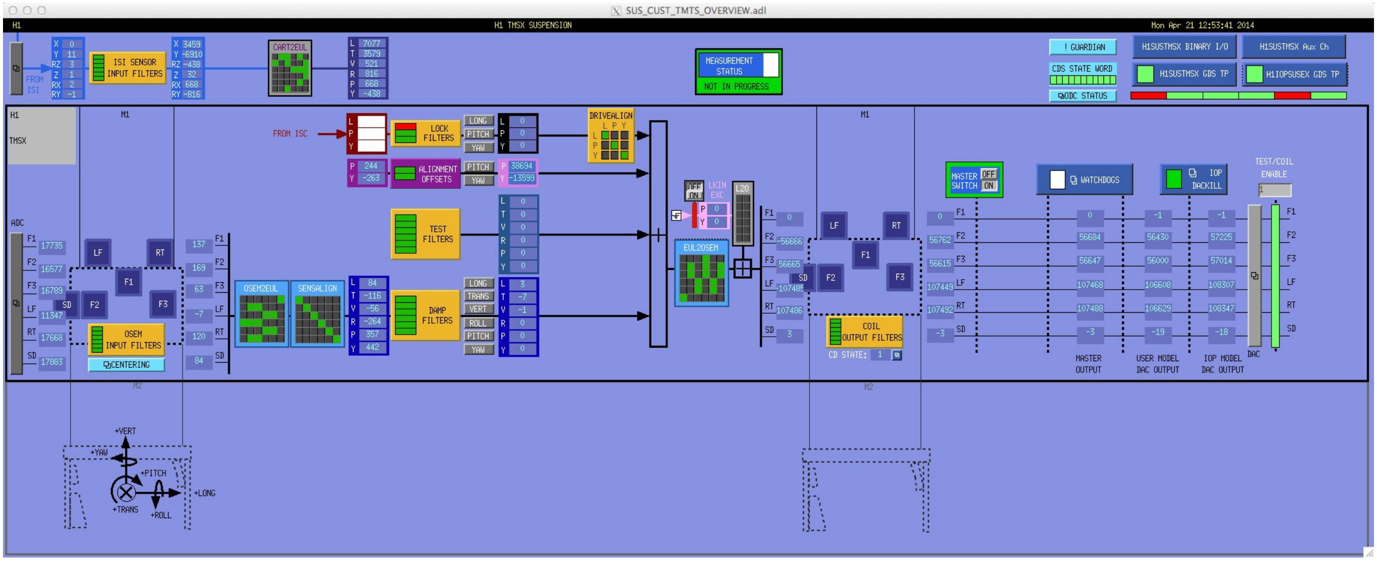
aLIGO SUS Operations Manual - Overview of TMTS MEDM screens

Back to Operation Manual main page
Back to TMTS main page

The TMTS screens described below live at `/opt/rtcds/userapps/release/sus/common/medm/tmts/`. They are all generic screens which rely for their execution on arguments passed in when they are called. Most of the arguments are defined in suspension specific files `/opt/rtcds/userapps/release/sus/common/medm/sustms*_overview_macro.txt` (*=x/y). The generic screens can be edited in MEDM but can't usefully be executed directly from there. Rather they need to be opened directly or indirectly from the SITEMAP screen `/opt/rtcds/${site}/${ifo}/medm/SITEMAP.adl`. See the MEDM page for further information.

Overview Screen

Screen SUS_CUST_TMTS_OVERVIEW.adl

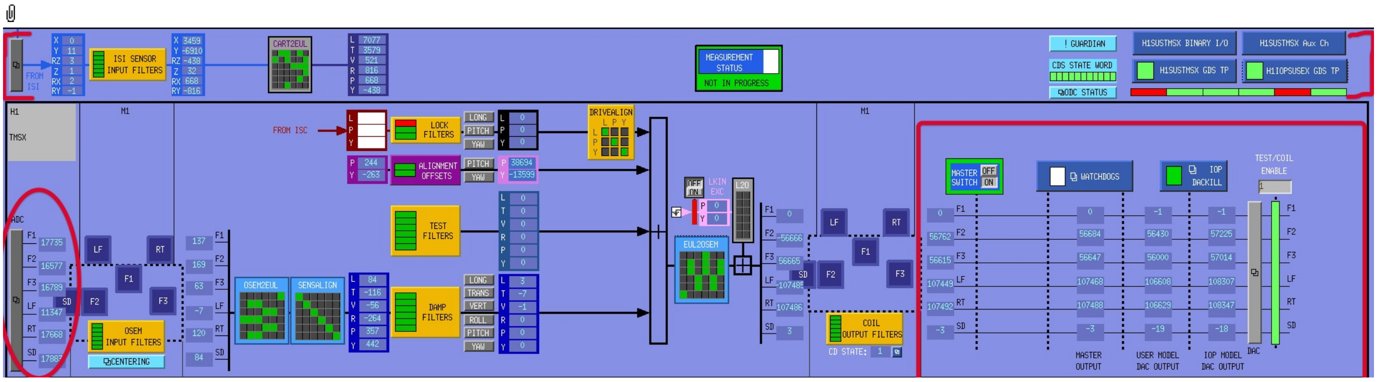


This is the overview screen. There is one sensor-actuator group, M1 - there are no OSEMs on the telescope:

- Other - subscreens not associated with a sensor/actuator group
- M1 - 6 BOSEMs on the structure engaging the upper mass

M1 is used for local damping (relative to the structure). DC pitch and yaw offsets are also injected at M1.

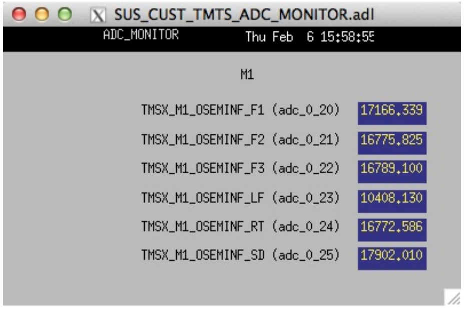
Other Screens



- MASTER SWITCH - a manual on/off switch for all actuation (common to all sensor-actuator groups).
- MEASUREMENT STATUS - flashes when manually set to ON, alerting other users to the fact that someone has claimed the suspension for testing.
- HIERARCHY SWITCH - enables various ISC inputs and outputs.
- ADC - a monitor screen (common to all sensor-actuator groups) for the semi-raw ADC values as reported by the IOP process after downsampling.
- DAC DAC - a popup allowing access to all the DAC outputs as reported by the IOP process.
- WATCHDOGS - a block implementing the watchdogs.
- IOP DACKILL - a watchdog that shuts off the IOP process (potentially other suspensions on the same front-end).
- H1IOPSUSEX GDS TP (or the like; depends on the front-end computer name) - status of the IOP process for the front-end.
- H1SUSTMSX GDS TP (or the like; depends on the suspension name) - status of the suspension process.
- H1SUSTMSX BIO (or the like; depends on the suspension name) - binary input/output controls.
- H1SUSTMSX Aux Ch (or the like; depends on the suspension name) - readbacks from coil driver channels for all

- H1SUSTMSX Aux Ch (or the like; depends on the suspension name) - readbacks from coil driver channels for all OSEMs as reported by the corresponding auxiliary front-end processor.
- ODC STATUS - ODC status
- !GUARDIAN - Guardian
- ISIINF - ISI input filters.
- CART2EUL - transformation from the ISI basis to SUS.

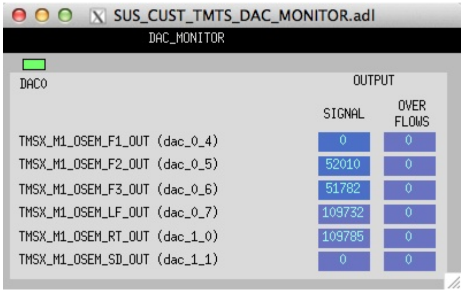
Screen SUS_CUST_TMTS_ADC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/ADC_MONITOR:

Shows the raw OSEM etc signals as reported by the IOP process. The ADC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

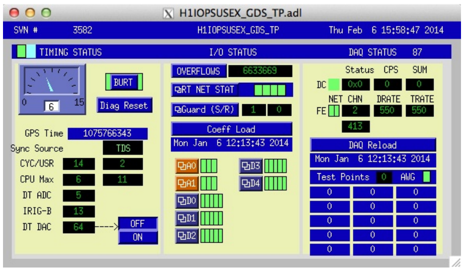
Screen SUS_CUST_TMTS_DAC_MONITOR.adl



Suspensions/OpsManual/Boilerplate/DAC_MONITOR:

Actuation signals just before the DAC as reported by the IOP process. The DAC channel numbers are hard-coded in suspension-specific macro files read by MEDM when the screen is displayed - see Suspensions/OpsManual/MEDM.

Screen H1IOPSUSEX_GDS_TP.adl



Suspensions/OpsManual/Boilerplate/IOP_GDS_TP:

This screen reports the status of the CPU running the IOP model for the suspension, which runs on a different core of the same front-end computer running the user model and is responsible for ADC and DAC access. Of particular importance:

The number of the CPU running the IOP process, e.g., "33", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the Test Points will be non-zero. If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen H1SUSTMSX_GDS_TP.adl

Suspensions/OpsManual/Boilerplate/USER_GDS_TP:

This screen reports the status of the CPU running the user model for the suspension. Of particular importance:

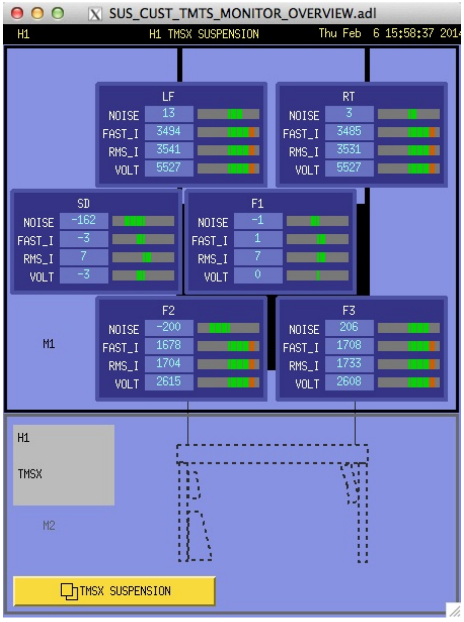
The number of the CPU, e.g., "37", can be read from this screen just to the right of the "DAQ STATUS" heading

The number of the CPU, e.g., "97", can be read from this screen just to the right of the "DAQ STATUS" heading.

The Coeff Load button reloads all the filters for the model.

If an excitation is active, one or more of the test points will be showing non-zero numbers. See Suspensions/OpsManual/diag for instructions on clearing test points.

Screen SUS_CUST_TMTS_MONITOR_OVERVIEW.adl

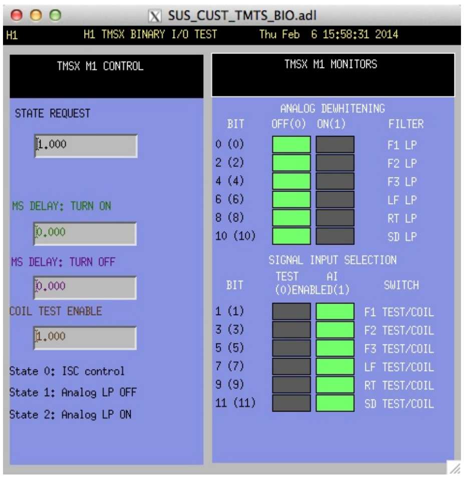


Suspensions/OpsManual/Boilerplate/MONITOR_OVERVIEW:

This screen shows the OSEM readback channels as reported by the AUX model for the suspension.

- VOLTS is a measure of the analog voltage at the coil driver.
- FAST_I is a measure of the current output by the coil driver.
- RMS_I is a low-passed measure of the current.
- NOISE is a high-passed measure of the current.

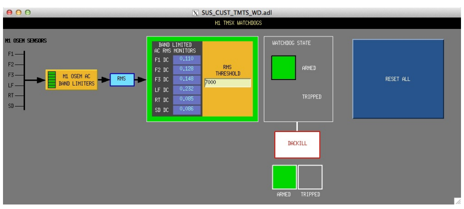
Screen SUS_CUST_TMTS_BIO.adl



Suspensions/OpsManual/Boilerplate/TMTS_BIO:

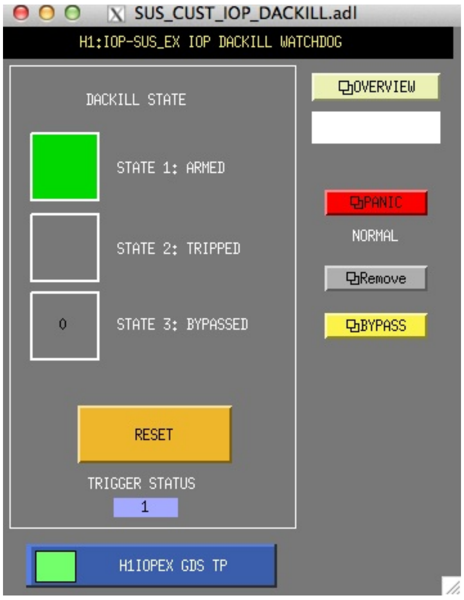
The Binary Input-Output control screen. State Requests for the various sensor-actuator groups can be input. Positive state values lock out manual control of the filters on the COILOUTF screen and set the digital dewhitening filters to match the analog whitening filters selected via the binary outputs. Negative state values allow the dewhitening filters to be manually overridden.

Screen SUS_CUST_TMTS_WD.adl


Suspensions/OpsManual/Boilerplate/WD:

This screen allows for setting the watchdog thresholds and for resetting the watchdog should it trip. See Watchdogs for more information.

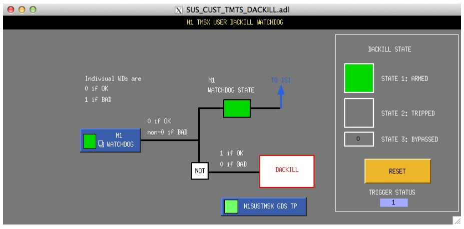
Screen SUS_CUST_IOP_DACKILL.adl



Suspensions/OpsManual/Boilerplate/IOP_DACKILL:

The watchdog controls for the IOP process. If this WD is tripped it disables IO to/from all user models on the front end. RESET resets from the tripped state if there are no outstanding triggers. BYPASS is a manual override that allows for operation of the suspension if there is a trigger condition that is known by the operator to be harmless.

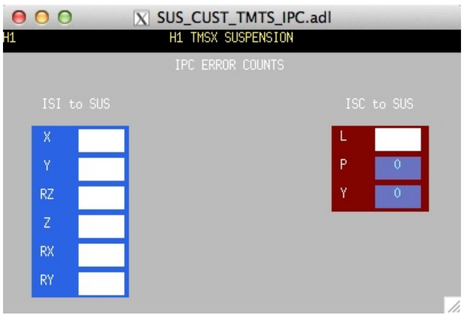
Screen SUS_CUST_TMTS_DACKILL.adl



Suspensions/OpsManual/Boilerplate/USER_DACKILL:

The watchdog controls for the user process. If this WD is tripped it disables IO to/from all sensor/actuator levels of the suspension. RESET resets from the tripped state if there are no outstanding triggers.

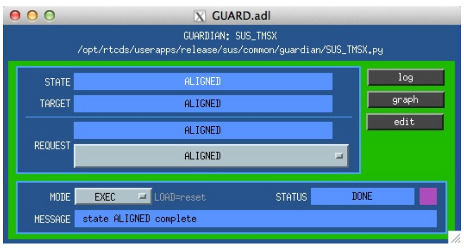
Screen SUS_CUST_TMTS_IPC.adl



Suspensions/OpsManual/Boilerplate/IPC:

Monitor screen for the interprocess communication (IPC) signals from the associated SEI platform (ISI) and from the IFO sensing and control system (ISC).

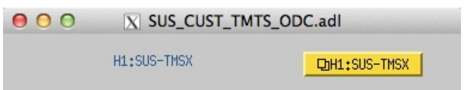
Screen GUARD.adl

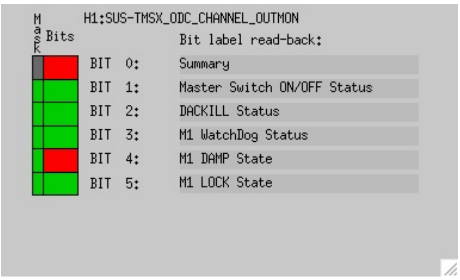


Suspensions/OpsManual/Boilerplate/GUARD:

The Guardian screen. See Suspensions/OpsManual/Guardian and Suspensions/OpsManual/OperatingInstructions.

Screen SUS_CUST_TMTS_ODC.adl

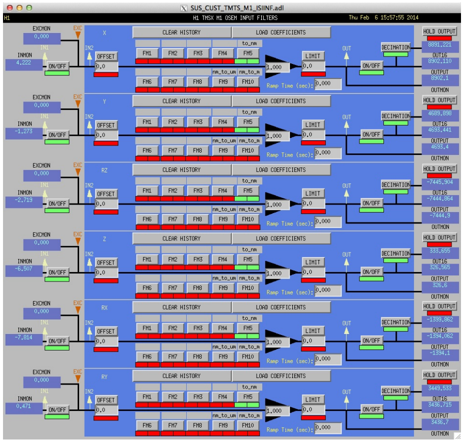




Suspensions/OpsManual/Boilerplate/ODC:

ODC Status.

Screen SUS_CUST_TMTS_M1_ISIINF.adl


Suspensions/OpsManual/Boilerplate/M1_ISIINF:

Filters for the diagnostic signals from the ISI.

Screen SUS_CUST_TMTS_M1_CART2EUL.adl

Suspensions/OpsManual/Boilerplate/M1_CART2EUL:

A matrix to diagonalize/rotate the ISI signals into the SUS basis (with +L from AR looking towards the HR face, +T left and +V up). See Suspensions/OpsManual/Projections and T1100617.

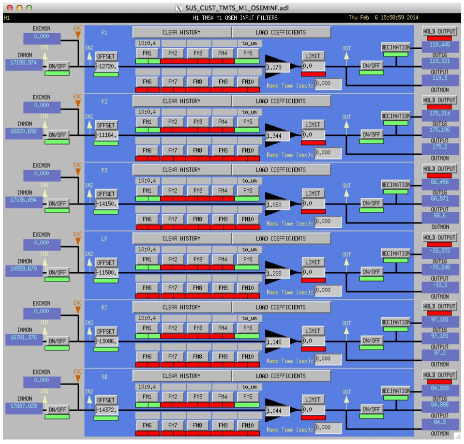
Sensor Actuator Group M1 - Upper Mass

Most of the items on the M1 section correspond to blocks in the underlying Simulink model and can be clicked on to access subscreens with settings for those block. It is laid out to reflect the flow of the local control loop, which involves the following elements:

- OSEM INPUT FILTERS - a block that accepts OSEM inputs from 0 to roughly 20K-30K counts (depending on the OSEM), applies dewhitening and normalizes to µm.
- CENTERING - a utility screen which displays the OSEM readouts as large on-screen meters that can be read at a distance, for convenience while mechanically adjusting OSEMs.
- OSEM2EUL - a block that diagonalizes the OSEM signal to signals in a longitudinal/transverse/vertical/roll/pitch/yaw (L/T/V/R/P/Y) basis.
- SENSALIGN - a block reserved for applying corrections if the sensors are not well-matched (the identity matrix by default).
- DAMP FILTERS - a filter bank applying the damping filters.
- TEST FILTERS - a filter bank group (with no filters!) serving as a place to apply AWG actuation in the L/T/V/R/P/Y basis without it having to go through the damping filters.
- ALIGNMENT OFFSETS - a place to apply DC actuation in pitch and yaw.
- LOCK FILTERS - filters for global control signals.
- DRIVEALIGN - a block reserved for applying corrections if the actuators are not well-matched (the identity matrix by default).
- EUL2OSEM - a block that de-diagonalizes the damping signals back to per-OSEM signals.
- L2O - ????.
- COIL OUTPUT FILTERS - a filter bank that corrects for hardware run/acquisition mode filters and for magnet polarity.

There are the following auxiliary inputs:

Screen SUS_CUST_TMTS_M1_OSEMINF.adl



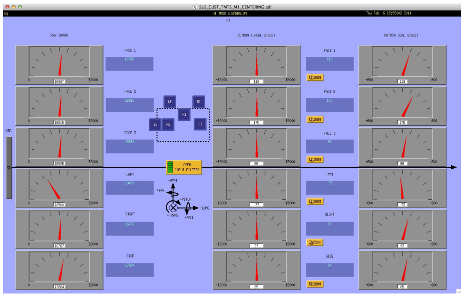
Suspensions/OpsManual/Boilerplate/M1_OSEMINF:

This block has 6 filter groups corresponding to the 6 M1 BOSEMs, T1/T2/T3/LF/RT/SD. See OSEMs and Calibration for more information on appropriate gain/offset settings.

The filter in slot FM1 corrects for the hardware whitening filter and should always be on. The filter in slot FM5 converts normalized counts to µm.

The Limit and Ramp Time fields should be 0 and both ON/OFF switches should be ON.

Screen SUS_CUST_TMTS_M1_CENTERING.adl



Suspensions/OpsManual/Boilerplate/M1_CENTERING:

This screen gives various views of the M1 OSEM signals as meter panels for ease of viewing when mechanically adjusting OSEMs.

Screen SUS_CUST_TMTS_M1_OSEM2EUL.adl

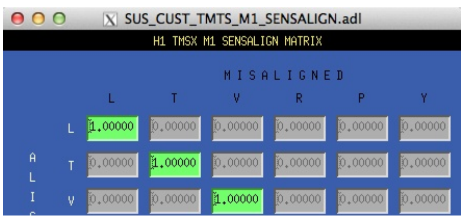
Suspensions/OpsManual/Boilerplate/M1_OSEM2EUL:

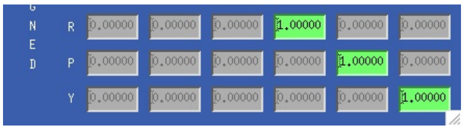
This screen allows entry of the matrix which converts from the M1 OSEM basis to the Euler basis. See Suspensions/OpsManual/Projections for more info. The entries for the linear DOFs L, T and V are dimensionless and implement appropriate averages of the OSEMS.

The entries for the angular DOFs R, P and Y have dimensions meter^-1 and incorporate the lever arms from the COM to the OSEM positions. Since the inputs are in µm the outputs are then in µrad.

The sign convention is the one that has been used consistently in SUS modelling efforts: +L is forward (looking from the AR side towards the HR side), +T is left (looking forward) and +V is up. +R, +P and +Y are right-handed about the corresponding axes.

Screen SUS_CUST_TMTS_M1_SENSALIGN.adl





Suspensions/OpsManual/Boilerplate/M1_SENSALIGN:

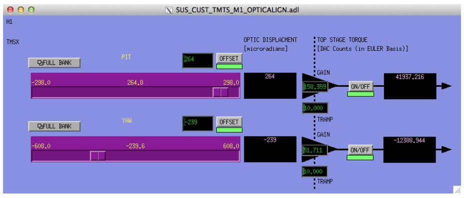
This screen is reserved for tweaking the M1 sensing diagonalization. It defaults to the identity matrix. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_TMTS_M1_DAMP.adl

Suspensions/OpsManual/Boilerplate/M1_DAMP:

These filters implement the local damping. See Suspensions/OpsManual/Damping.

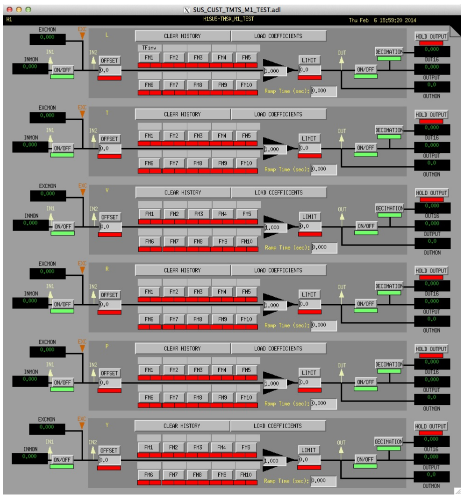
Screen SUS_CUST_TMTS_M1_OPTICALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_OPTICALIGN:

This screen has sliders for dialling in static pitch and yaw actuation offsets. See Suspensions/OpsManual/Alignments for more info.

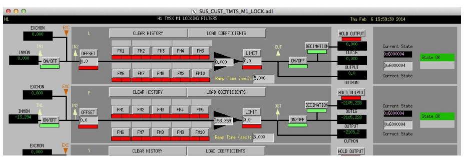
Screen SUS_CUST_TMTS_M1_TEST.adl

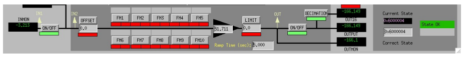


Suspensions/OpsManual/Boilerplate/M1_TEST:

This is a deliberately empty set of 6 filter banks corresponding to the Euler DOFs. It allows AWG excitations to be applied at the EXC test points without passing through the damping filters. Offsets should normally be 0 and gains should be 1. However offsets of ≈10000 in L/T/V or ≈1000 in R/P/Y can be applied temporarily to check that actuation is taking effect (and with the right sign). In such cases a Ramp Time of 10 s or so is often helpful. The button on the main screen which brings up this screen has a series of indicators, one for each filter, that are green when both the left and right ON/OFF switches are ON. Thus although the left switch doesn't do anything useful (there are no inputs before it and excitations are injected after it), it should be left ON so that the indicator tracks the right switch.

Screen SUS_CUST_TMTS_M1_LOCK.adl

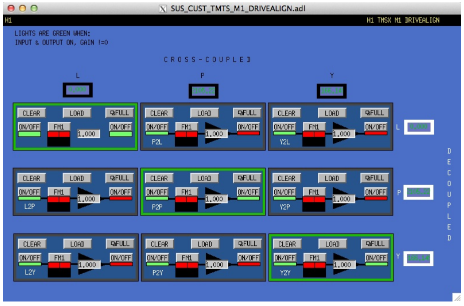




Suspensions/OpsManual/Boilerplate/M1_LOCK:

Filters for the locking signals.

Screen SUS_CUST_TMTS_M1_DRIVEALIGN.adl



Suspensions/OpsManual/Boilerplate/M1_DRIVEALIGN:

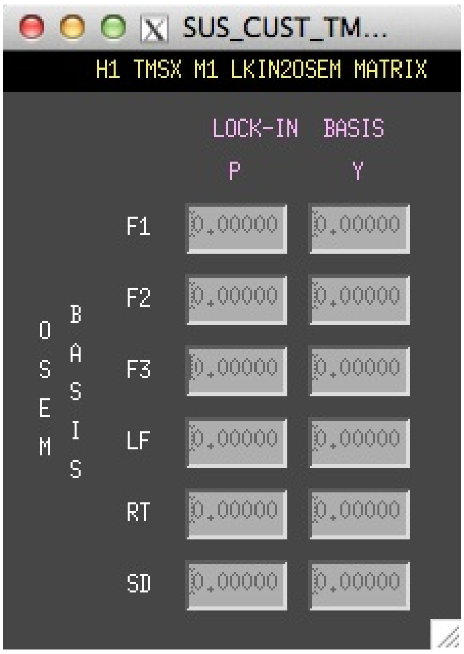
This screen is reserved for tweaking the M1 actuator diagonalization and frequency response. Unlike SENSALIGN there is also a single-slot filter bank associated with each entry. See Suspensions/OpsManual/Projections for more info.

Screen SUS_CUST_TMTS_M1_EUL2OSEM.adl

Suspensions/OpsManual/Boilerplate/M1_EUL2OSEM:

This screen allows entry of the matrix which converts force/torque requests from the Euler basis to the OSEM basis. See Suspensions/OpsManual/Projections for more info. The sign convention for the input forces/torques is the same as for linear/angular displacements in the Euler basis. The sign convention for the outputs is that positive is a request for the actuator to push. Lever arms have been taken into account but not any any calibration of the actuation train.

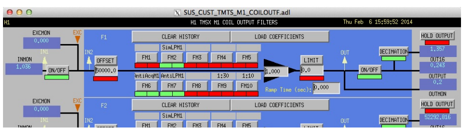
Screen SUS_CUST_TMTS_M1_LKIN2OSEM.adl

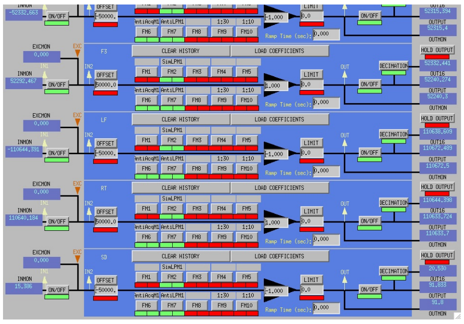


Suspensions/OpsManual/Boilerplate/M1_LKIN2OSEM:

Need to find out what this does.

Screen SUS_CUST_TMTS_M1_COILOUTF.adl





Suspensions/OpsManual/Boilerplate/M1_COILOUTF:

This screen applies compensation for the hardware filters in the M1 actuation electronics and also for the magnet sign. In normal use, the filters cannot be set manually but are enabled or disabled automatically in sync with the hardware filters. (If it should be necessary for testing purposes, this automatic control can be disabled in the BIO screen by entering a negative state request value.)

See the Calibration page for information on setting the gain fields.

The Offset, Limit and Ramp Time fields should normally be 0 and both ON/OFF switches should be ON. However offsets of ≈10000 [TBC] can be applied temporarily to check that actuation is taking effect (and with the right sign). A positive offset applied here (i.e., before the ±1 gain applied later on the same screen) should give a positive displacement in the corresponding OSEM signal, otherwise the sign of the gain and the magnet don't match.

The sign convention at the output is positive for a positive current (rather than force).

aLIGO: Suspensions/OpsManual/TMTS/Screens (last edited 2014-05-27 14:34:18 by MarkBarton)

aLIGO SUS Operations Manual - Models for TMTS Suspensions

Back to Operation Manual main page

There are Mathematica (T020205) and Matlab (T080188) programs that take a parameter set describing the TMTS and calculate the dynamics. The parameter set needs to be formatted differently for Mathematica vs Matlab, but the parameter names are the same as far as was practical.

The Mathematica version of the TMTS model (DualLite2DBLateral) was based on the OMCS model (DualLite2) with three main modifications:

- There are only two blades at the top mass rather than four (two wires attach to each blade rather than one).
- The lower wires toe out in both directions rather than being parallel in one.
- Blade lateral compliance was included for the blades in the top mass, because it is a modified quad UIM and lateral compliance has been found important for the quad.

The Matlab version of the TMTS has been included as a special case of that for the OMCS. If the flag `pend.db` is set to 1 (true) in the data structure defining the pendulum, matrix elements appropriate for TMTS are used.

The +x direction is defined normal to the plan of the top wires, as for OMCS. However since the telescope axis (+y) is the one of most interest to the operators, this has been defined as L in the EPICS/MEDM. Thus unlike every other model where L=x and T=y, for TMTS, L=y and T=-x. Similarly, R=pitch and P=-roll.

The TMTS Mathematica model and parameter sets for it live in the SUS SVN at `^/trunk/Common/MathematicaModels/DualLite2DBLateral`. Parameter sets for Matlab live at `^/trunk/QUAD/Common/MatlabTools/DoubleModel_Production`. Mark Barton maintains the Mathematica parameter sets, Jeff Kissel maintains the Matlab. Unfortunately two different naming conventions are in use and it's difficult to tell what corresponds to what - if in doubt, ask. References given below are to Mathematica versions as subdirectories of `^/trunk/Common/MathematicaModels/DualLite2DBLateral`.

Key cases of the TMTS model are given below, with results such as mode frequencies and mode shapes.

Explanation of the standard result set is at .../Key. Instructions for generating wiki pages for new cases of Mathematica models is at .../HowTo.

First Article Model

Current best model for the first article (as installed as TMSy at LHO) is 20131224TMTS_FirstArticle. It has all as-built numbers from D0902773-v8 except for a tweak of +2.7 mm to `d1` to better match `modeR1`. It is equivalent to Matlab parameter set tmtsopt_firstarticle.m r6273 of 2/14/14.

Production Model

Current best model for the production TMTS is 20131224TMTS_Production. It has all as-built numbers from D0902773-v8. It is equivalent to Matlab parameter set tmtsopt_production.m r6273 of 2/14/14.

Old

20130501TMTS_FirstArticle has all as-built numbers as far as they could be determined except for a tweak of +4.3 mm to `d1` to better match `modeR1`. 20130426TMTS_FirstArticle has all as-built numbers as far as they could be determined but didn't match modeR1 very well.

Data

Some screenshots supplied by Ken Mailand showing SolidWorks MOI values are attached for future reference.

First Article

The TMTS first article is installed as TMSy at LHO (see D0900419). It is slightly different from the production build

The TMTS first article is installed as TMSy at LHO (see D0900419). It is slightly different from the production build.

- TMTS Top Mass Properties - First Article.pdf
- TMTS Telescope Mass Properties - First Article.pdf

Production Build

- TMTS Top Mass Properties - Production.pdf
- TMTS Telescope Mass Properties - Production.pdf

Case 20131224TMTS_FirstArticle of Mathematica model DualLite2DBLat

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
"20131224TMTS_FirstArticle"
```

```
First article case of TMS with data from D0902773-v8 of 12/24/13, with +2.7 mm tweak to d1 to
better match fundamental pitch ('R') mode of 0.679 Hz, equivalent to tmtsopt_firstarticle r6273
of 2/14/14.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeY1	modeT1	modeL1	modeR1	modeV1	modeP1
modeT2	modeL2	modeV2	modeP2	modeY2	modeR2

Mode Summary

Back to Top

```
N    f          name     type

1    0.357487   modeY1   yaw2

2    0.44897    modeT1   pitch1   x2

3    0.474697   modeL1   y2       roll2

4    0.679192   modeR1   pitch1

5    0.73404    modeV1   z2

6    0.754609   modeP1   roll1    roll2

7    1.36581    modeT2   pitch2   x1

8    1.72179    modeL2   y1       roll1

9    2.40182    modeV2   z1

10   2.44429    modeP2   roll1

11   2.58853    modeY2   yaw1

12   4.14866    modeR2   pitch2   pitch1
```

Violin Modes	UM	Telescope
f (Hz), n=1-4	330.376 661.737 995.079 1331.42	103.424 207.22 311.767 417.452
Q, n=1-4	115715. 132493. 121822. 103683.	59792.1 79433.8 80124.9 72435.1

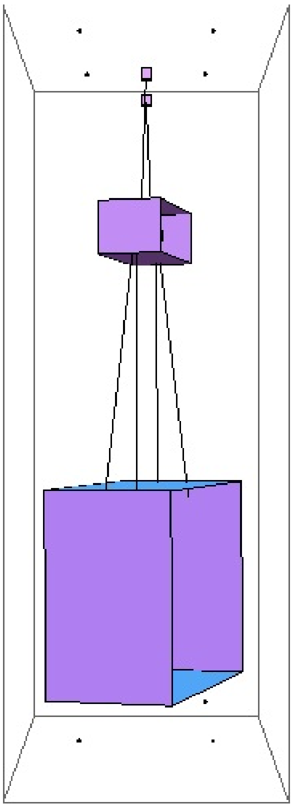
Mode Shapes

Mode #1 - modeY1

Back to Top

0.357487 Hz

	x	y	z	yaw	pitch	roll
Mass U	0.00251884	0	0	-0.539942	-0.0517532	-0.000535574
Mass I	0.0150708	-0.000100741	0	-0.839774	-0.0181643	-0.00046231

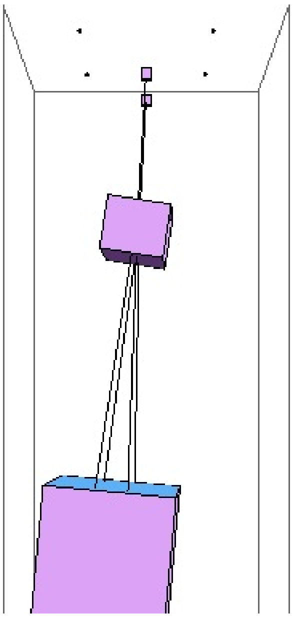


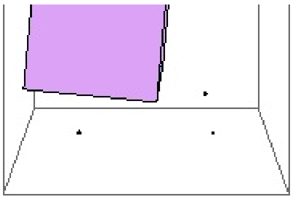
Mode #2 - modeT1

Back to Top

0.44897 Hz

	x	y	z	yaw	pitch	roll
Mass U	-0.119878	0.000763099	0	-0.0929112	0.74162	0.00721203
Mass I	-0.470044	0.00543945	0	-0.142829	0.430686	0.00906475



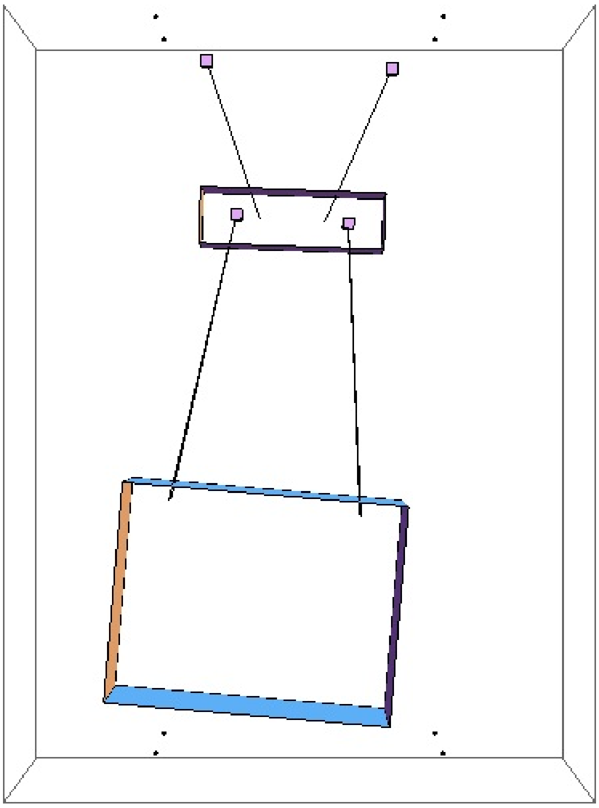


Mode #3 - modeL1

Back to Top

```
0.474697 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00283183	0.139576	0	0.000532582	0.00443452	0.260558
Mass I	0.0102213	0.726303	0	0.000700188	-0.00778284	0.620418

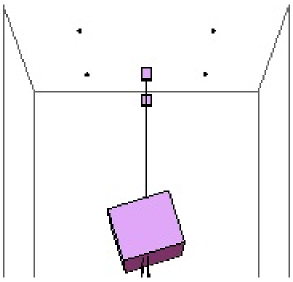


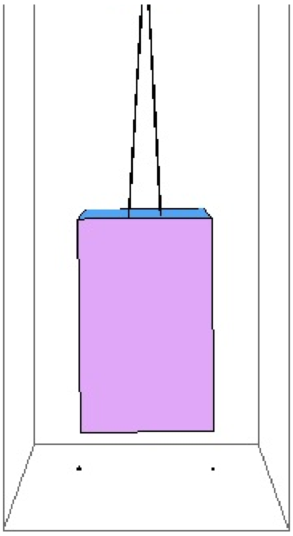
Mode #4 - modeR1

Back to Top

```
0.679192 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.000678194	0.0012376	0	0.00458067	-0.998873	-0.0165986
Mass I	-0.0111506	0.00143269	0	0.00677182	-0.0403708	-0.012351



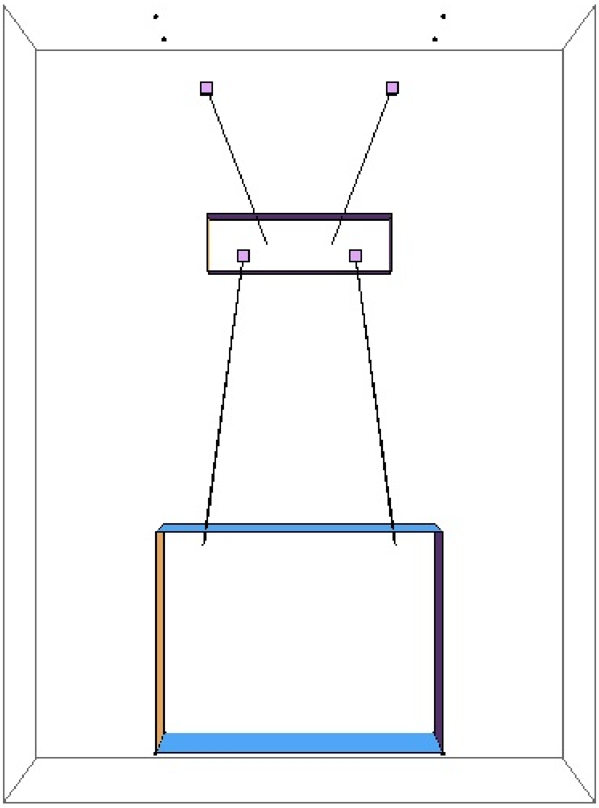


Mode #5 - modeV1

Back to Top

```
0.73404 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.546978	0	0	0
Mass I	0	0	-0.837147	0	0	0



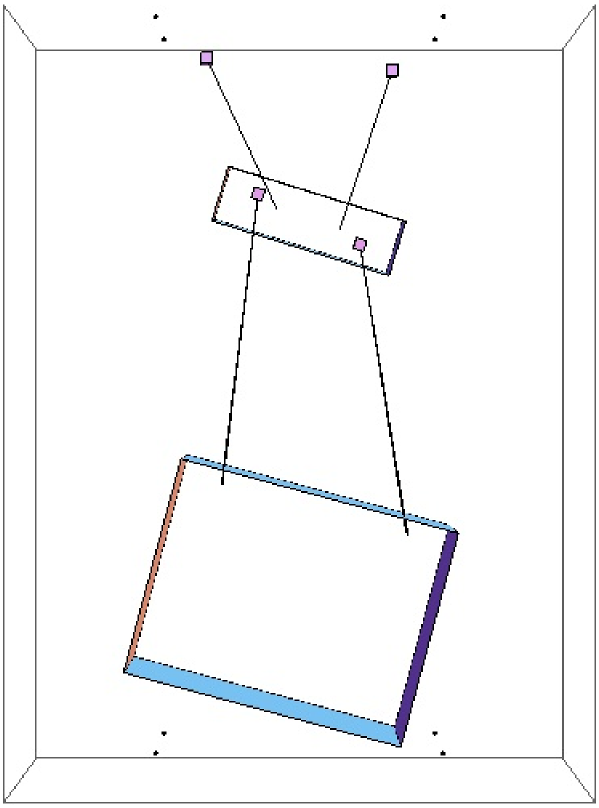
Mode #6 - modeP1

Back to Top

```
0.754609 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00120088	-0.0612507	0	0.000855226	-0.246071	0.726837

Mass I	-0.000737829	-0.0563724	0	0.000426317	-0.00999259	0.635708

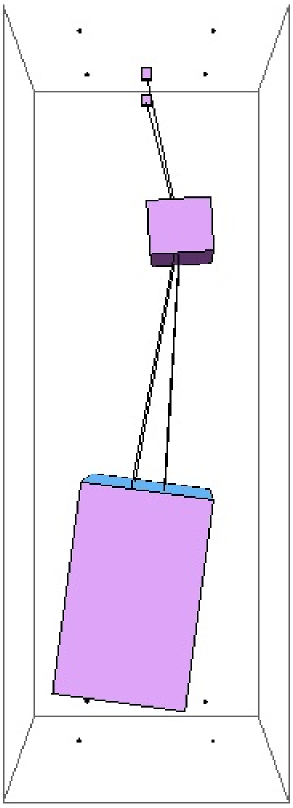


Mode #7 - modeT2

Back to Top

```
1.36581 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.559138	0.00214459	0	-0.0846648	-0.335605	-0.0245928
Mass I	-0.0367902	-0.000223457	0	-0.0954877	0.745868	-0.0129017

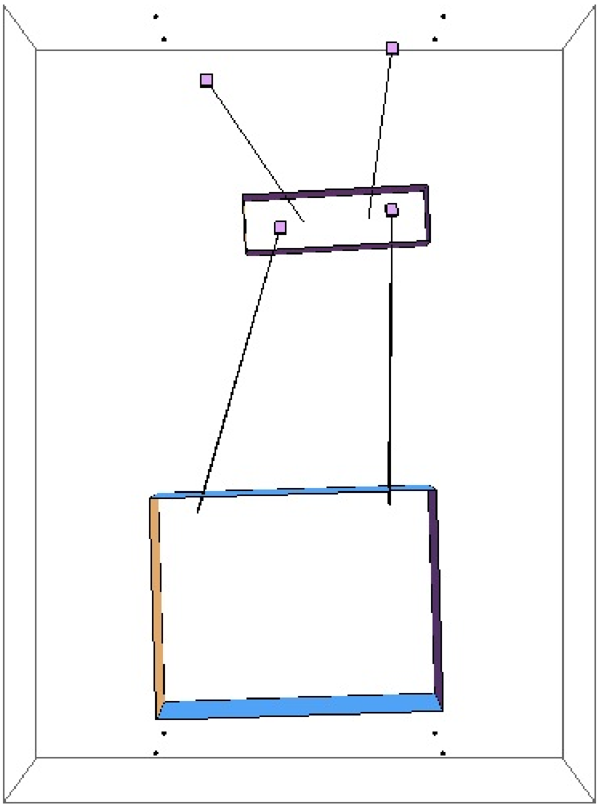


Mode #8 - modeL2

Back to Top

```
1.72179 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00218559	-0.831405	0	-0.00307276	0.0333112	-0.474583
Mass I	-0.000493154	0.11687	0	0.000122685	0.0039364	-0.262181

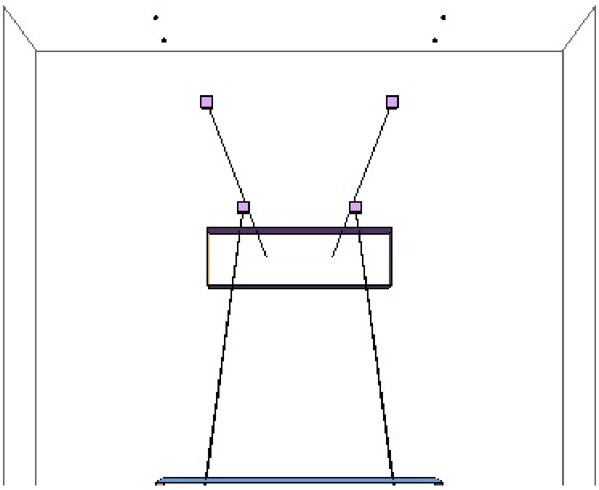


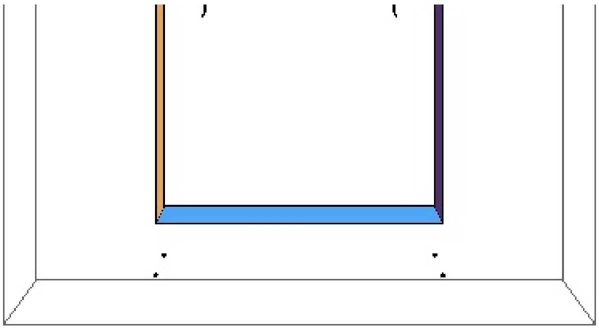
Mode #9 - modeV2

Back to Top

```
2.40182 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	-0.938207	0	0	0
Mass I	0	0	0.346076	0	0	0



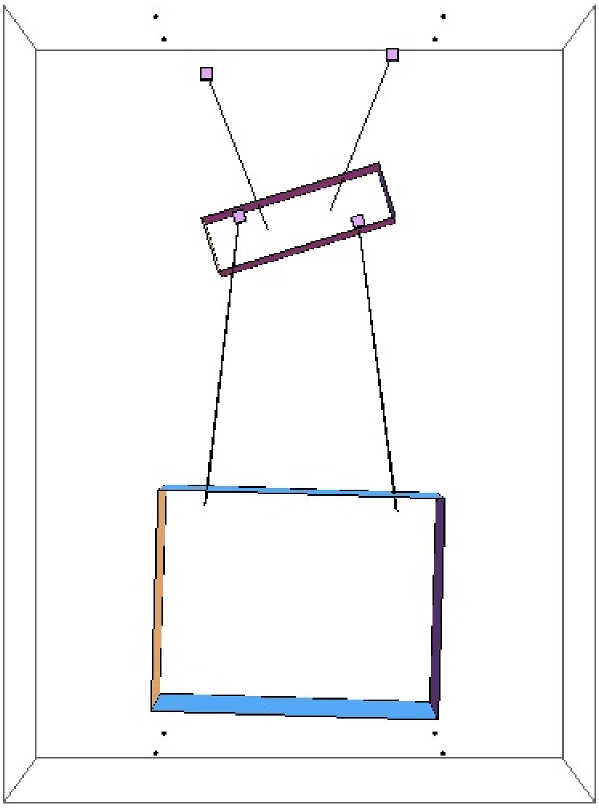


Mode #10 - modeP2

Back to Top

```
2.44429 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.0014845	0.00254942	0	-0.0659853	0.0527661	-0.990937
Mass I	0.000647649	-0.0063968	0	0.00365049	0.00352797	0.104069

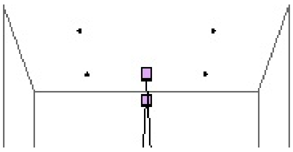


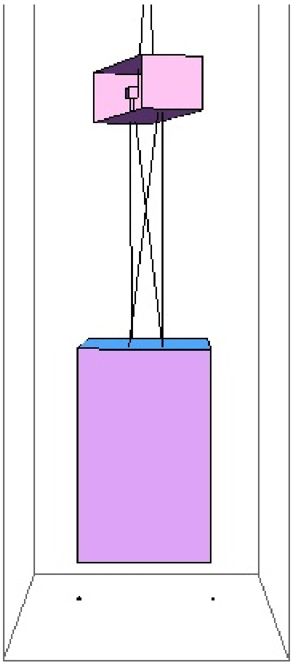
Mode #11 - modeY2

Back to Top

```
2.58853 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00246406	0	0	0.996598	0.0115421	-0.0588346
Mass I	-0.00103983	-0.000322964	0	-0.0562088	-0.00150423	0.00544294



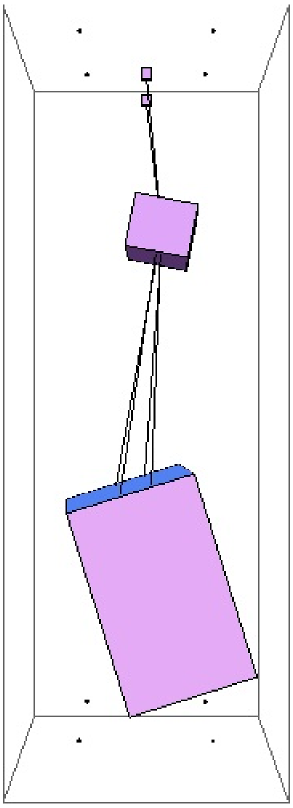


Mode #12 - modeR2

Back to Top

```
4.14866 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.127633	0.000144401	0	-0.0399531	0.536705	-0.0283866
Mass I	-0.0649891	-0.000125685	0	0.106124	-0.823118	0.0157009



Parameters

Back to Top

```
g -> 9.81
```

```
ux -> 0.18                                                        D0900419 measured; Display only

uy -> 0.5                                                         D0900419 measured; Display only

uz -> 0.152                                                       D0900419 measured; Display only

uzz -> 0

m1 -> 44.371                                                      D0902773-v8

I1x -> 0.832304                                                   D0902773-v8

I1y -> 0.4312                                                     D0902773-v8

I1z -> 0.698457                                                   D0902773-v8

I1xy -> 0.0231722                                                 D0902773-v8

I1yz -> 0.000044516                                               D0902773-v8

I1zx -> 0.00559483                                                D0902773-v8

lx -> 0.368                                                       D0900419 measured; Display only

ly -> 0.762                                                       D0900419 measured; Display only

lz -> 0.588                                                       D0900419 measured; Display only

lzz -> -0.176                                                     D0900419 measured; Display only

m2 -> 78.595                                                      D0902773-v8

I2x -> 8.73011                                                    D0902773-v8

I2y -> 5.46882                                                    D0902773-v8

I2z -> 7.93248                                                    D0902773-v8

I2xy -> 0.147437                                                  D0902773-v8

I2yz -> 0.993508                                                  D0902773-v8

I2zx -> 0.00692386                                                D0902773-v8

l1 -> 0.45621                                                     D0902773-v8, 16.76"

l2 -> 0.80185                                                     D0902773-v8

nw1 -> 2

nw2 -> 4

       11
r1 -> -----
      20000                                                       D1101166 (item 1 in parts list)

       11
r2 -> -----
      20000                                                       D1101163 (item 2 in parts list)

              11
Y1 -> 2.119 10                                                    measured

              11
Y2 -> 2.119 10                                                    measured

d0 -> 0.0005                                                      D0902773-v8

d1 -> -0.00409                                                    D0902773-v8 with fudge of +2.7
mm, breakoff is above COM, so negative

d2 -> 0.07031                                                     D0902773-v8

su -> 0.

slu -> 0.00236                                                    D0902773-v8

sll -> 0.049022                                                   D0902773-v8 (1.93")

n0 -> 0.25302                                                     D0902773-v8
```

n1 -> 0.08888 D0902773-v8

n2 -> 0.15323 D0902773-v8

n3 -> 0.26194 D0902773-v8

tl1 -> 0.426159

tl2 -> 0.859295

ltotal -> 1.28545

unstretched -> False

vertblades -> True

ul1 -> 0.454746

ul2 -> 0.801074

sl1 -> 0.45621

sl2 -> 0.80185

si1 -> -0.35979

si2 -> 0.147535

c1 -> 0.933033

c2 -> 0.989057

pitchbul -> 0

pitchbur -> 0

pitchbll -> 0

pitchblr -> 0

rollbul -> 0

rollbur -> 0

rollbll -> 0

rollblr -> 0

A1 -> $9.50332\ 10^{-7}$

A2 -> $9.50332\ 10^{-7}$

kw1 -> 441409.

kw2 -> 251138.

flex1 -> 0.00452866

flex2 -> 0.0087431

kbuz -> 1759.3 doub_spring_stiff_calc.m using 2013-03-14 H1TMSY Measured Data, [k1_est k2_est] = doub_spring_stiff_calc(0.7031,2.391,1.242,44,88)

kblz -> 2423.55 doub_spring_stiff_calc.m using 2013-03-14 H1TMSY Measured Data, [k1_est k2_est] = doub_spring_stiff_calc(0.7031,2.391,1.242,44,88)

kblx -> 146988. Same ratio as for the quad suspension (kbix/kbiz)

bdu -> 0.342834

bdl -> 0.159068

m12 -> 122.966

COM1x -> 0

COM1y -> 0

COM1z -> 0

FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

Ibtxyl -> 0

Ibtyzl -> 0

Ibtzxl -> 0

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03

zpad -> 0.1

phib -> 0.001

M11 -> 7.18688 10^{-14}

M12 -> 7.18688 10^{-14}

M21 -> 7.18688 10^{-14}

M22 -> 7.18688 10^{-14}

temperature -> 290.

boltzmann -> 1.38066 10^{-23}

alphasilica -> 5.1 10^{-7}

betasilica -> 0.00015

rhosilica -> 2200.

Csilica -> 772.

Ksilica -> 1.38

Ysilica -> 7. 10^{10}

phisilica -> 2. 10^{-8}

phissilica -> 3.3 10^{-11}

rhosteel -> 7800.

Csteel -> 486.

Ksteel -> 49.

Ysteel -> 2.119 10^{11}

```
Ysteel -> 2.119 10

alphasteel -> 0.000012

betasteel -> -0.00025

phisteel -> 0.0002

rhomarag -> 7800.

Cmarag -> 460.

Kmarag -> 20.

                 11
Ymarag -> 1.65 10

alphamarag -> 0.000011

betamarag -> 0.

phimarag -> 0.0001

tmU -> 0.0045

tmL -> 0.005

magicnumber -> 0.0732

deltabladeU -> 0.00161367

deltabladeL -> 0.00161367

deltawireU -> 0.00265699

deltawireL -> 0.0024294

taubladeU -> 0.368085

taubladeL -> 0.454426

tauwireU -> 0.00685222

tauwireL -> 0.00685222

                                        0.00373201 #1
damping[imag, bladeUtype] -> (0.0001 + --------------- & )
                                                     2
                                       1 + 5.34879 #1

                                        0.00460742 #1
damping[imag, bladeLtype] -> (0.0001 + --------------- & )
                                                     2
                                       1 + 8.15239 #1

damping[imag, wireUtype] -> (0.0002 & )

damping[imag, wireLtype] -> (0.0002 & )

                                         0.000114393 #1
damping[imag, wireUatype] -> (0.0002 + ------------------ & )
                                                        2
                                       1 + 0.00185363 #1

                                         0.000104595 #1
damping[imag, wireLatype] -> (0.0002 + ------------------ & )
                                                        2
                                       1 + 0.00185363 #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx1 -> 0
```

```
kcony1 -> 0

kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0
```

Back to Top

Case 20131224TMTS_Production of Mathematica model DualLite2DBLat

This page is automatically generated and manual edits may be lost. Put page-specific notes in subpage /Notes. Guidance on how to interpret the results is in ../Key.

Formal name and description

```
"20131224TMTS_Production"
```

```
Production case of TMS with data from D0902773-v8 of 12/24/13, equivalent to tmtsopt_production r6273 of 2/14/14.
```

Links to Result Sections

Mode Summary	Mode Shapes	Parameters

modeY1	modeT1	modeL1	modeR1	modeP1	modeV1
modeT2	modeL2	modeV2	modeP2	modeY2	modeR2

Mode Summary

Back to Top

```
N    f          name     type

1    0.356438   modeY1   yaw2     yaw1

2    0.44903    modeT1   pitch1   x2

3    0.474545   modeL1   y2       roll2

4    0.650001   modeR1   pitch1

5    0.729786   modeP1   z2

6    0.750216   modeV1   roll1    roll2

7    1.37346    modeT2   pitch2   x1

8    1.73388    modeL2   y1

9    2.40454    modeV2   z1

10   2.56147    modeP2   roll1

11   2.71571    modeY2   yaw1

12   4.31193    modeR2   pitch2   pitch1
```

Violin Modes	UM	Telescope
f (Hz), n=1-4	331.653 664.287 998.893 1336.48	104.241 208.852 314.207 420.689
Q, n=1-4	116353. 133227. 122540. 104334.	60584.1 80515. 81258.9 73503.5

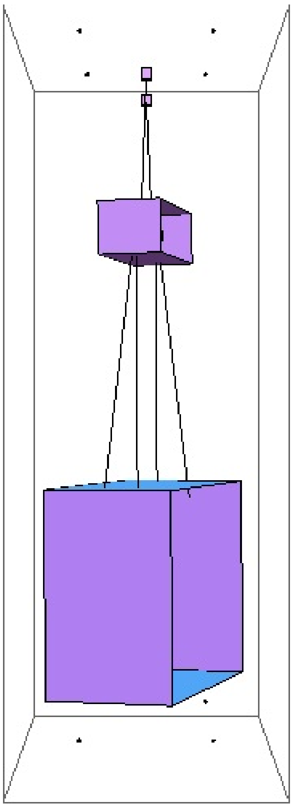
Mode Shapes

Mode #1 - modeY1

Back to Top

0.356438 Hz

	x	y	z	yaw	pitch	roll
Mass U	0.00214618	0	0	-0.548084	-0.0566326	0
Mass I	0.0124007	0	0	-0.834263	-0.0155783	0.000106431

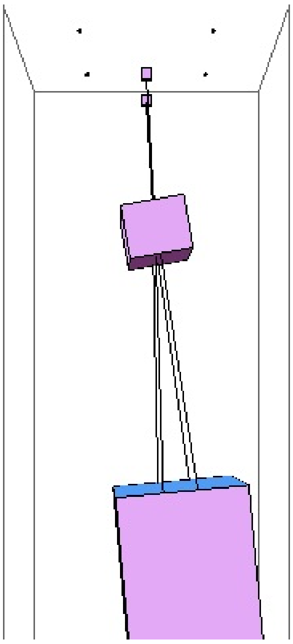


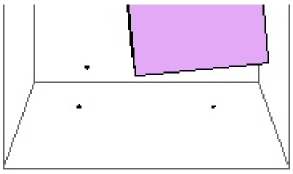
Mode #2 - modeT1

Back to Top

0.44903 Hz

	x	y	z	yaw	pitch	roll
Mass U	0.114052	0.000231643	0	0.0735923	-0.777863	-0.0006473
Mass I	0.442885	0.00117691	0	0.110837	-0.409965	0.000978118



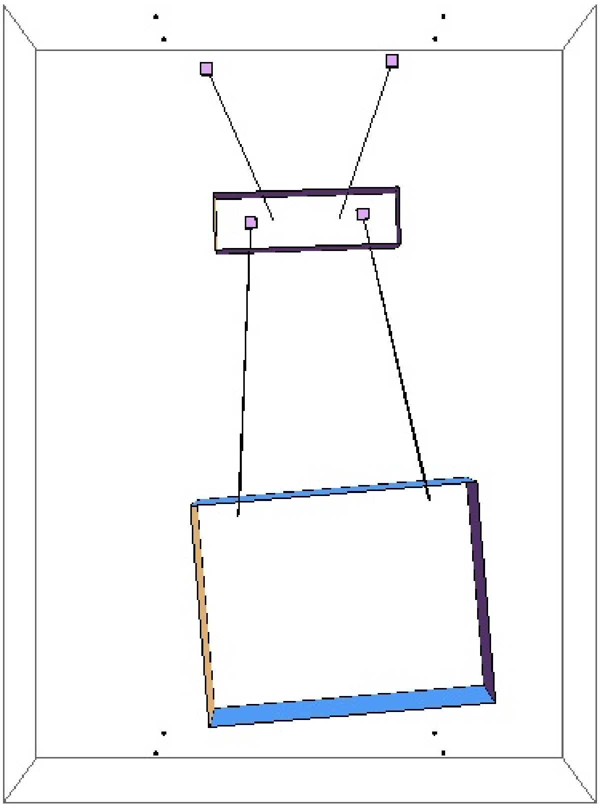


Mode #3 - modeL1

Back to Top

```
0.474545 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.000678823	-0.141594	0	-0.000149739	-0.0221134	-0.250844
Mass I	0.00196371	-0.72946	0	-0.000188325	-0.00250301	-0.620013

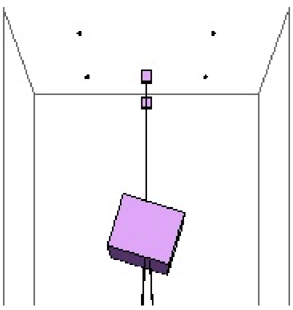


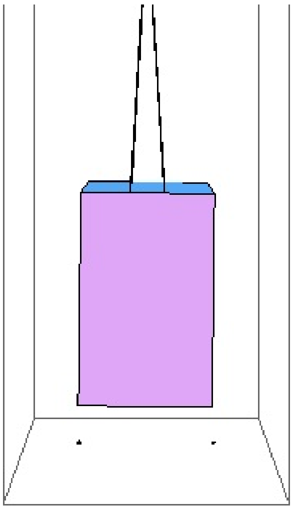
Mode #4 - modeR1

Back to Top

```
0.650001 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.000446624	-0.000779975	0	-0.00400759	0.998975	0.0118903
Mass I	0.010147	-0.000917391	0	-0.00587751	0.0411041	0.00798482



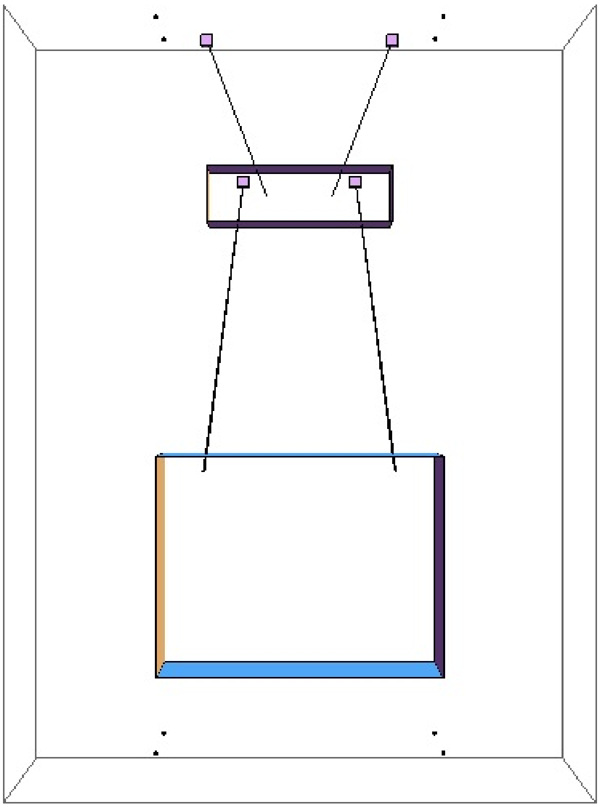


Mode #5 - modeP1

Back to Top

```
0.729786 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0.546094	0	0	0
Mass I	0	0	0.837724	0	0	0

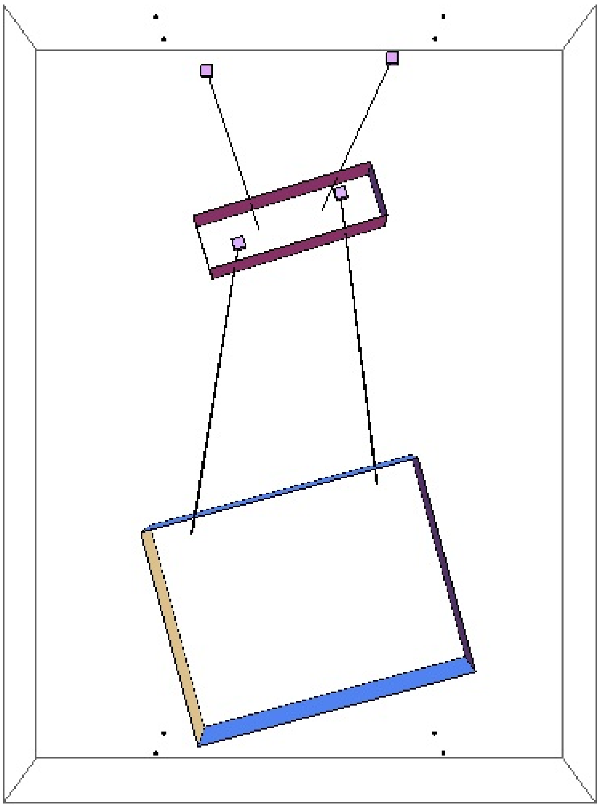


Mode #6 - modeV1

Back to Top

```
0.750216 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.000344524	0.0606695	0	-0.00131059	0.217234	-0.731098
Mass I	0.00215234	0.0564335	0	-0.00161743	0.00987833	-0.641352

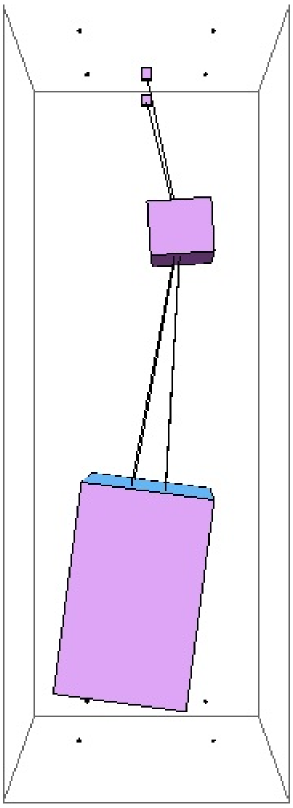


Mode #7 - modeT2

Back to Top

```
1.37346 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.556436	-0.00124418	0	-0.0679998	-0.375888	0.00448965
Mass I	-0.0350134	0	0	-0.0774597	0.73291	0.0076415

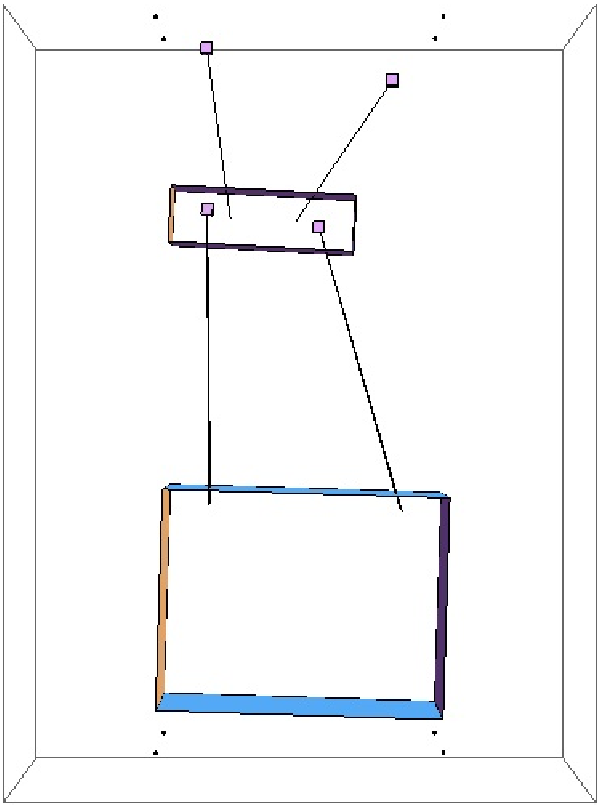


Mode #8 - modeL2

Back to Top

```
1.73388 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00156555	0.842152	0	0.00107783	-0.0409143	0.456129
Mass I	-0.000444717	-0.116251	0	0.000100647	-0.000588995	0.259873

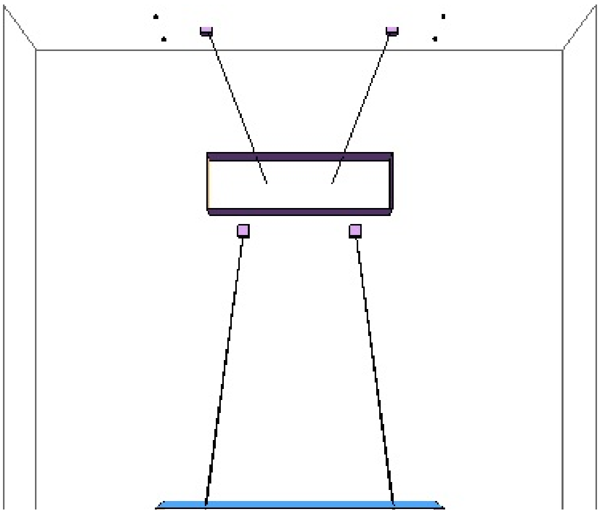


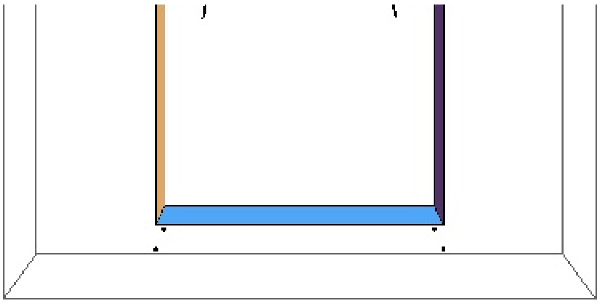
Mode #9 - modeV2

Back to Top

```
2.40454 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0	0	0.940944	0	0	0
Mass I	0	0	-0.338562	0	0	0



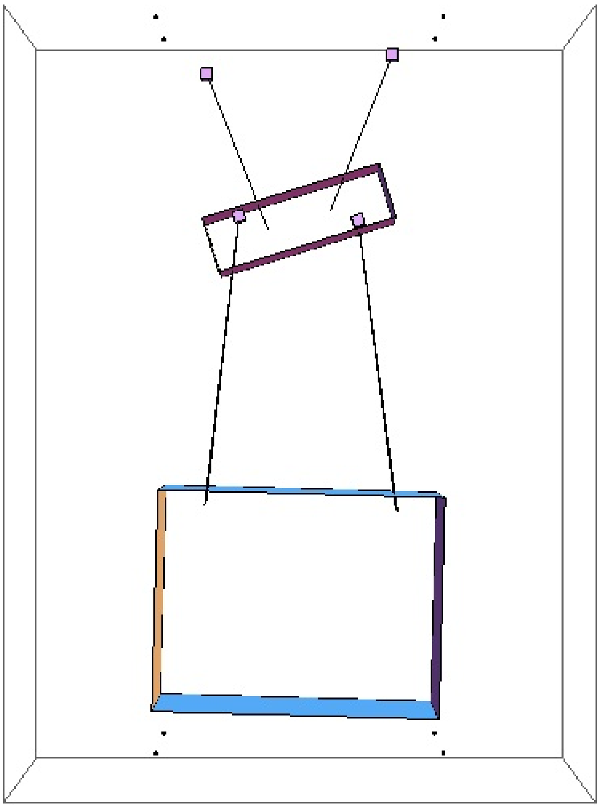


Mode #10 - modeP2

Back to Top

```
2.56147 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	-0.000798488	0.00197712	0	-0.025076	0.0851658	-0.99185
Mass I	0.000387809	-0.00564603	0	0.00100205	0.005129	0.0910412

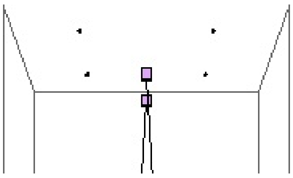


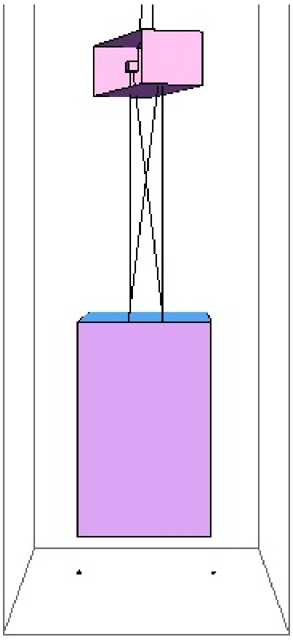
Mode #11 - modeY2

Back to Top

```
2.71571 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.00197368	0	0	0.998259	0.0102511	-0.0241291
Mass I	-0.000835285	-0.000116255	0	-0.0527468	-0.00130051	0.00187436



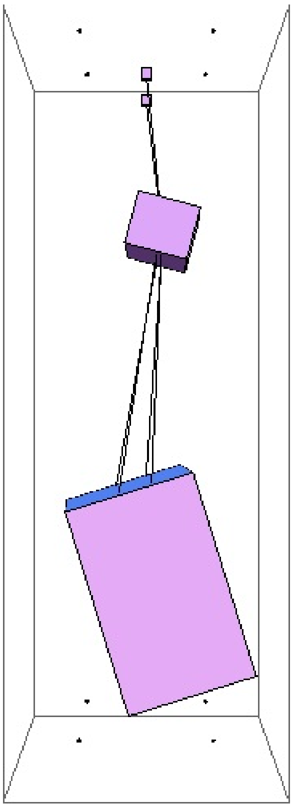


Mode #12 - modeR2

Back to Top

```
4.31193 Hz
```

	x	y	z	yaw	pitch	roll
Mass U	0.118649	0	0	-0.0318432	0.649881	-0.0345202
Mass I	-0.0594217	0	0	0.0788201	-0.742705	-0.00411072



Parameters

Back to Top

```
g -> 9.81
```

```
ux -> 0.18                                                     D0900419 measured; Display only

uy -> 0.5                                                      D0900419 measured; Display only

uz -> 0.152                                                    D0900419 measured; Display only

uzz -> 0

m1 -> 44.079                                                   D0902773-v8

I1x -> 0.744504                                                D0902773-v8

I1y -> 0.338025                                                D0902773-v8

I1z -> 0.665289                                                D0902773-v8

I1xy -> 0.0283922                                              D0902773-v8

I1yz -> 0.000228387                                            D0902773-v8

I1zx -> 0.0019529                                              D0902773-v8

lx -> 0.368                                                    D0900419 measured; Display only

ly -> 0.762                                                    D0900419 measured; Display only

lz -> 0.588                                                    D0900419 measured; Display only

lzz -> -0.176                                                  D0900419 measured; Display only

m2 -> 79.859                                                   D0902773-v8

I2x -> 8.88186                                                 D0902773-v8

I2y -> 5.59111                                                 D0902773-v8

I2z -> 8.25                                                    D0902773-v8

I2xy -> -0.0576386                                             D0902773-v8

I2yz -> 0.851031                                               D0902773-v8

I2zx -> -0.00509547                                            D0902773-v8

l1 -> 0.45621                                                  D0902773-v8

l2 -> 0.80185                                                  D0902773-v8

nw1 -> 2

nw2 -> 4

       11
r1 -> -----
      20000                                                    D1101166 (item 1 in parts list)

       11
r2 -> -----
      20000                                                    D1101163 (item 2 in parts list)

              11
Y1 -> 2.119 10                                                 measured

              11
Y2 -> 2.119 10                                                 measured

d0 -> 0.00046                                                  D0902773-v8

d1 -> -0.00675                                                 D0902773-v8

d2 -> 0.07031                                                  D0902773-v8

su -> 0.

slu -> 0.00236                                                 D0902773-v8

sll -> 0.05142                                                 D0902773-v8

n0 -> 0.25302                                                  D0902773-v8

n1 -> 0.08887                                                  D0902773-v8
```

```
n2 -> 0.15323                                                    D0902773-v8

n3 -> 0.26194                                                    D0902773-v8

tl1 -> 0.426115

tl2 -> 0.85649

ltotal -> 1.28261

unstretched -> False

vertblades -> True

ul1 -> 0.454734

ul2 -> 0.801061

sl1 -> 0.45621

sl2 -> 0.80185

si1 -> -0.359812

si2 -> 0.14874

c1 -> 0.933025

c2 -> 0.988876

pitchbul -> 0

pitchbur -> 0

pitchbll -> 0

pitchblr -> 0

rollbul -> 0

rollbur -> 0

rollbll -> 0

rollblr -> 0

                -7
A1 -> 9.50332 10

                -7
A2 -> 9.50332 10

kw1 -> 441409.

kw2 -> 251138.

flex1 -> 0.0045108

flex2 -> 0.00867126

kbuz -> 1759.3                                                   doub_spring_stiff_calc.m using
2013-03-14 H1TMSY Measured Data, [k1_est k2_est] =
doub_spring_stiff_calc(0.7031,2.391,1.242,44,88)

kblz -> 2423.55                                                  doub_spring_stiff_calc.m using
2013-03-14 H1TMSY Measured Data, [k1_est k2_est] =
doub_spring_stiff_calc(0.7031,2.391,1.242,44,88)

kblx -> 146988.                                                  Same ratio as for the quad
suspension (kbix/kbiz)

bdu -> 0.345544

bdl -> 0.161626

m12 -> 123.938

COM1x -> 0

COM1y -> 0

COM1z -> 0
```

```
FRP1x -> 0

FRP1y -> 0

FRP1z -> 0

Ibtxyl -> 0

Ibtyzl -> 0

Ibtzxl -> 0

COM2x -> 0

COM2y -> 0

COM2z -> 0

FRP2x -> 0

FRP2y -> 0

FRP2z -> 0

btx -> 0.03

bty -> 0.03

btz -> 0.03

zpad -> 0.1

phib -> 0.001

                 -14
M11 -> 7.18688 10

                 -14
M12 -> 7.18688 10

                 -14
M21 -> 7.18688 10

                 -14
M22 -> 7.18688 10

temperature -> 290.

                       -23
boltzmann -> 1.38066 10

                     -7
alphasilica -> 5.1 10

betasilica -> 0.00015

rhosilica -> 2200.

Csilica -> 772.

Ksilica -> 1.38

                 10
Ysilica -> 7. 10

                   -8
phisilica -> 2. 10

                     -11
phissilica -> 3.3 10

rhosteel -> 7800.

Csteel -> 486.

Ksteel -> 49.

                  11
Ysteel -> 2.119 10

alphasteel -> 0.000012
```

```
alphasteel -> 0.000012

betasteel -> -0.00025

phisteel -> 0.0002

rhomarag -> 7800.

Cmarag -> 460.

Kmarag -> 20.

                 11
Ymarag -> 1.65 10

alphamarag -> 0.000011

betamarag -> 0.

phimarag -> 0.0001

tmU -> 0.0045

tmL -> 0.005

magicnumber -> 0.0732

deltabladeU -> 0.00161367

deltabladeL -> 0.00161367

deltawireU -> 0.00265962

deltawireL -> 0.00243096

taubladeU -> 0.368085

taubladeL -> 0.454426

tauwireU -> 0.00685222

tauwireL -> 0.00685222

                                       0.00373201 #1
damping[imag, bladeUtype] -> (0.0001 + --------------- & )
                                                    2
                                      1 + 5.34879 #1

                                       0.00460742 #1
damping[imag, bladeLtype] -> (0.0001 + --------------- & )
                                                    2
                                      1 + 8.15239 #1

damping[imag, wireUtype] -> (0.0002 & )

damping[imag, wireLtype] -> (0.0002 & )

                                        0.000114507 #1
damping[imag, wireUatype] -> (0.0002 + ------------------ & )
                                                       2
                                       1 + 0.00185363 #1

                                        0.000104662 #1
damping[imag, wireLatype] -> (0.0002 + ------------------ & )
                                                       2
                                       1 + 0.00185363 #1

x00 -> 0

y00 -> 0

z00 -> 0

yaw00 -> 0

pitch00 -> 0

roll00 -> 0

kconx1 -> 0

kcony1 -> 0
```

```
kconz1 -> 0

kconyaw1 -> 0

kconpitch1 -> 0

kconroll1 -> 0

kconx2 -> 0

kcony2 -> 0

kconz2 -> 0

kconyaw2 -> 0

kconpitch2 -> 0

kconroll2 -> 0
```

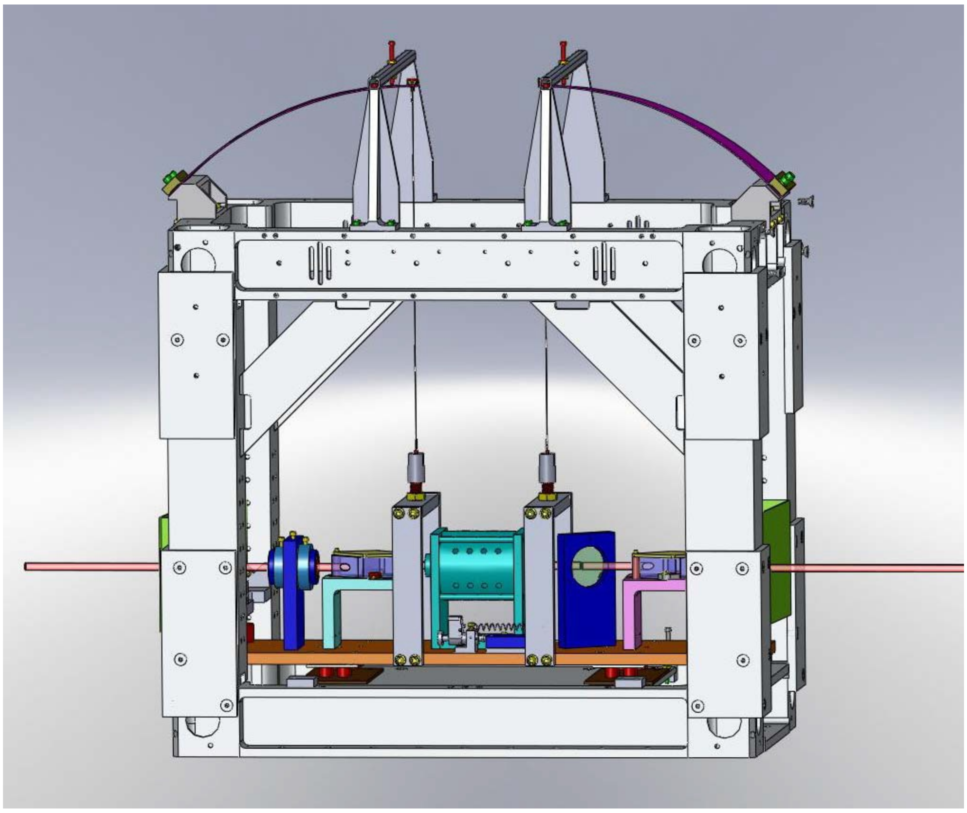
Back to Top

aLIGO SUS Operation Manual - Info on OFIS

Back to Operation Manual main page

Background

The Output Faraday Isolator (OFI) is suspended in an Output Faraday Isolator Suspension (OFIS). The suspension is a passive design with only eddy current damping and no electronic damping or CDS interface. Both OFI and OFIS are the responsibility of the AOS group. See Auxillary_Optics_Stuff for more information.



Models

The OFIS suspension has not been modelled in Mathematica/Matlab. If this ever gets done, results will be put at Suspensions/OpsManual/OFIS/Models .

Screens

There are no OFIS screens.

aLIGO: Suspensions/OpsManual/OFIS (last edited 2013-09-11 12:44:54 by MarkBarton)

Suspension Models in Mathematica

This page is now an honorary part of the SUS Operations Manual.

Introduction

Welcome to Mark Barton's LIGO suspension modeling wiki page. Generalizing on some work I did in Japan, I've developed a Mathematica toolkit for modeling mass-wire-spring systems of the sort commonly used for vibration isolation, as well as models of many specific systems of interest to LIGO.

The toolkit and selected models were formerly published at http://www.ligo.caltech.edu/~e2e/SUSmodels/. The explanatory content will gradually be moved here. The models are being uploaded to the SUS SVN repository, browsable at https://redoubt.ligo-wa.caltech.edu/websvn/listing.php?repname=sus& and accessible with a subversion client at https://redoubt.ligo-wa.caltech.edu/svn/sus . If you're viewing this wiki page you presumably already have LIGO.ORG credentials, which are what you need for browse or checkout access to the SVN. However for commit access to the SVN you need to apply to David Barker of LHO.

Requirements

To use any of this, you need a tolerably recent version of Mathematica, preferably v6 or later - v5 at a pinch, but not v4. (*1/7/12: v9 has been tested and doesn't seem to have any issues.*) See Compatibility for more info. You also need an SVN client to download stuff from the repository. See http://subversion.apache.org/ for software and http://svnbook.red-bean.com/ for documentation.

Directory Structure

The toolkit and models assume a particular directory structure (see DirectoryStructure) which you should review carefully because it's fairly complicated and has recently been updated to be more SVN-friendly (see SVNInstructions). Considerable thought has been given to backward compatibility, so if you have a working setup with models downloaded from the old website, they should continue to work. You can even checkout the new to the same directory as the old and have them co-exist. However you probably want to delete your old toolkit directory (or rename it out of the way) because old-format models will give it precedence over the new SVN-based one, and you won't get the benefit of easy updates.

Toolkit

The toolkit is a set of Mathematica packages which provide support for the following:

- rigid body masses with x, y, z, yaw, pitch and roll DOFs
- wires or ribbons, without distributed mass (no true violin modes) but with a detailed elasticity model including longitudinal, torsional, and bending near the attachment points
- spring elements with a 6*6 stiffness matrix and a 6 component vector of preload forces/torques
- optional massless connecting elements between elastic elements
- arbitrary frequency-dependent damping on all sources of elasticity
- dissipation dilution
- fully symbolic expressions for the potential and kinetic energy
- numerical solutions for the minimum of the potential and the eigenvalues and eigenvectors
- ::usage tags defined for all symbols (for use with the Mathematica help operator "?")
- utility functions to plot mode shapes, transfer functions and thermal noise
- automatic saving of results of time-consuming calculations for easy reloading later
- a default parameter set which is easily overridable to reflect design variations
- export of state-space matrices to Matlab and/or E2E
- export of parameter files in `quadopt.m` or `triplep.m` format to Matlab, with comments.

T020205 describes how to create models with the toolkit and the method of calculation. However -v1 is out of date with respect to the new SVN-friendly directory structure and installation method. See the Toolkit page on this wiki for more information, and SVNInstructions#Toolkit for instructions on download and installation from the SVN.

Models

Using this toolkit, I've developed models of a generic GEO-style triple pendulum, and a generic quad pendulum as proposed for Advanced LIGO, as well as many toy models.

Quad Lateral Model

This reflects the conceptual design of the test-mass suspension for Advanced LIGO, with 4 masses, 6 blade springs and 14 wires. The "Lateral" in the name reflects the fact that blade lateral compliance is allowed for.

See QuadLite2LateralModel.

Triple Model

This reflects the conceptual design of the GEO suspension (or BS or HLTS or HSTS), with 3 masses, 6 blade springs and 10 wires. Blade lateral compliance is not included because it turns out to be negligible for all the aLIGO triples. (But a version with it exists if anyone needs it.)

See TripleLite2Model.

Dual Model

This reflects the conceptual design of the OMC suspension with 2 masses, 6 blade springs and 6 wires.

See DualLite2Model.

Dual-Blade Dual Model

This reflects the conceptual design of the TMTS suspension with 2 masses, 4 blade springs and 6 wires.

See DualLite2DBLateralModel.

Two Wire Simple Pendulum

This is a two-wire (one loop) single mass suspension with blades as for the iLIGO LOS and SOS.

See TwoWireSimpleModel.

Two Wire Simple Pendulum with Blades

This is a two-wire (one loop) single mass suspension with blades as for the aLIGO TipTilts and aLIGO HAM Aux.

See TwoWireSimpleBladesModel.

Other Models

Other models that have been developed include:

- A triple model with blade lateral compliance.
- Quad and Triple models with beads in the final stage fibres/wires to approximate violin modes.
- Lots more.

If there there's a model you'd like to see posted in the new format, please hassle Mark Barton.

Directory Structure for Mathematica Suspension Models

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page

This page explains the new SVN-friendly directory structure supported by the Mathematica pendulum modelling toolkit and how to work with it. However the toolkit doesn't absolutely require an SVN (in principle, models and cases could be exchanged by any method) and the scope of this page is non-SVN-specific. For information (especially etiquette) on using it with the aLIGO SUS SVN, see SVNInstructions

Old-School Directory Structure (Historical Interest Only - Not Recommended)

The formerly recommended directory structure is explained here because some features of the new layout are a bit puzzling unless you know what they're trying to be backward compatible with. It is rather unsatisfactory because paths to two key directories need to be hard-coded into the calculation notebook for each case, but it seemed to be the least bad system achievable under Mathematica v4, for which the toolkit was first developed. For a single user it was not too bad - since most models and cases were derived by cut and paste from existing ones, the hard-coded paths rarely needed to be touched.

Unfortunately since the hard-coded paths normally need to be different for different users, this made collaboration difficult - one couldn't send another user a case of a model and have it be immediately usable. With an SVN it would have been intolerable, because different users would be continually overwriting files for non-substantive changes. Fortunately new features introduced in Mathematica v5 made it possible to do better - see below.

- `toolkit/` (Support files are installed here - see note below.)
  - `PendUtil.nb` (Mathematica package with functions for modelling suspensions.)
  - `PendUtil.m` (Autosave file with executable code from `PendUtil.nb`.)
  - ... (Other toolkit files.)
- `quadlite2lateral/` (A directory for quad model stuff but not "the" model directory! See note below.)
  - `old/` (Old versions of the quad model.)
  - `current/` ("The" model directory - see notes.)
    - `ASUS4L2LateralModelDefn.nb` (The "model definition notebook" for the quad model.)
    - `ASUS4L2LateralModelDefn.m` (Autosave file with executable code from model definition notebook.)
    - `default/` (The "case directory" for the `default` case of the quad model)
      - `ASUS4XLLateralModelCalcDefault.nb` (The "case calculation notebook" for the `default` case.)
      - `precomputed/` (Precomputed results for the `default` case.)
    - `anothercase/` (Another case, with different numerical parameters.)
      - `ASUS4XLLateralModelCalcAnotherCase.nb`
      - `precomputed/`
    - ... (Other cases.)
- ... (Other models.)

Notes

- The exact name `toolkit` is arbitrary, but the Mathematica variable `modelsupportdirectory` needs to be set to point there by every calculation notebook for every model.
- The `old/current` level of the hierarchy was useful to Mark Barton but dispensable for other users.
- The exact name `current` doesn't matter, but the Mathematica variable `modeldirectory` needs to be set to point here by the calculation notebook for every case of the particular model.
- The name of the case directory (e.g., `default`, `anothercase`) matters: it has to agree with the Mathematica variable `modelcase`, set in the case calculation notebook.

New-Style Directory Structure (Recommended)

New hierarchy:

- `$BaseDirectory/Autoload/PendulumToolkit/`
  or `$UserBaseDirectory/Autoload/PendulumToolkit/` (Toolkit files are checked out here from the SVN - see notes below.)
  - `PendUtil.nb` (Mathematica package with functions for modelling suspensions.)
  - `PendUtil.m` (Autosave file with executable code from `PendUtil.nb`.)
  - `init.nb` (Source file for `init.m`.)
  - `init.m` (Discovered and run by Mathematica; does path setup - see note below.)
  - ... (Lots of other support files.)
- `$BaseDirectory/Kernel/`
  or `$UserBaseDirectory/Kernel/` (A patch needed for Mathematica v7 and earlier lives here - see note below.)
  - `init.nb` (Source notebook for patch.)
  - `init.m` (Executable for patch.)
- `QuadLite2Lateral/` (The model directory, checked out from the SVN.)
  - `ASUS4L2LateralModelDefn.nb` (Model definition notebook for the quad model.)
  - `ASUS4L2LateralModelDefn.m` (Autosave version with executable code from above.)
  - `default/` (The "case directory" for the default case of the quad model)
    - `ASUS4XLLateralCaseDefn.nb` (The case definition notebook for the default case.)
    - `ASUS4XLLateralCaseDefn.m` (Autosave version with executable code from above.)
    - `precomputed/` (Precomputed results for the default case.)
    - `stdcalc` (Pseudo-user directory with prototype version of calculation.)
      - `ASUS4XLLateralModelCalcPlots.nb` (User calculation notebook for pseudo-user stdcalc with representative collection of plots.)
      - `ASUS4XLLateralModelCalcExport.nb` (User calculation notebook for pseudo-user stdcalc that generates standard set of exported result files.)
    - `quickcalc` (Pseudo-user directory with prototype version of quick calculation.)
      - `ASUS4XLLateralModelCalcPlots.nb` (User calculation notebook for pseudo-user quickcalc with representative collection of plots.)
      - `ASUS4XLLateralModelCalcExport.nb` (User calculation notebook for pseudo-user stdcalc that generates standard set of exported result files.).)
    - `albert.einstein` (User directory for A. Einstein with custom calculation.)
      - `ASUS4XLLateralModelCalcDefault.nb` (User calculation notebook for A. Einstein.)
  - `anothercase/` (Another case, with different numerical parameters.)
    - `ASUS4XLLateralCaseDefn.nb` (The case definition notebook for `anothercase`.)
    - `precomputed/`
    - `stdcalc/`
    - `quickcalc`
  - `SuchAndSuchCases` (A folder for a particular type of case.)
    - `asuchandsuchcase` (A case in a folder.)
    - `SoAndSoCases` (A subfolder for a particular subtype of case.)
      - `asoandsocase` (A case in a subfolder.)
      - ... (And so on to arbitrary depth.)
  - ... (Other cases at the top level. Note however that `default` is the only top-level case allowed in the SVN.)
- ... (Other models.)

See the Toolkit page and pages for individual models (QuadLite2LateralModel, TripleLite2Model) for help setting up this structure.

Notes

The need to make the toolkit and models SVN-friendly was the impetus to revise the old scheme. The key changes are as follows:

- We now insist that the toolkit files are stored in one of the Autoload directories that are on the Mathematica path by default, either `$BaseDirectory/Autoload/PendulumToolkit` or `$UserBaseDirectory/Autoload/PendulumToolkit` . (These path specifiers are a hybrid of Mathematica (`$BaseDirectory`) and Unix (`.../Autoload/PendulumToolkit/`) syntaxes, but the idea is that they're relative to the system and user preferences areas for the particular OS.) The exact name `PendulumToolkit` is convenient because it matches the SVN but is not significant to the toolkit.
- The `init.m` in `PendulumToolkit` does most of the path set up. It adds `PendulumToolkit` to the Mathematica path (`$Path`) (eliminating the need for the `modelsupportdirectory` variable), and defines

functions to make it easy to define `modeldirectory` relative to the calculation notebook.

- In Mathematica v7 and earlier, there is no function by which the `init.m` file in `PendulumToolkit` can tell the name of the directory it is in, which it needs to set the path to include itself. Thus we use a supplementary `init.m` file in `$BaseDirectory/Kernel/` or `$UserBaseDirectory/Kernel/` to provide that information via one hard-coded assignment. Templates for creating one of these files are provided, but only need be installed by users with older versions.
- The `old/current` level of the hierarchy is gone. "The" model directory (the one pointed to by `modeldirectory`) can now have a name reflecting the model. It's easiest to have it be the same as on the SVN (e.g., `QuadLite2Lateral`) but you can check it out under another name if you prefer. Models can be checked out in arbitrary places - different models don't have to be siblings.
- There can now be an arbitrary number of levels of intermediate subdirectories. **Directories used for the purpose of grouping cases will be called folders.** For cases in folders, the Mathematica symbol `modelcase` should be a list of strings locating the case directory as a sequence of folders below the model directory plus the case directory itself.
- A new "user" level has been added at the bottom of the hierarchy. Each person using a case of a model from the SVN is strongly encouraged to have their own calculation notebook in their own user subdirectory, with their preferred set of plots etc. There are two standard pseudo-users, `stdcalc` and `quickcalc`, with template calculations that can be customized. `stdcalc` is the full calculation as of old. `quickcalc` is currently only available for quad and triple model and uses the package `QuickQuadLateral` or `QuickTriple` to do a slightly approximate but quicker version of the calculation.
- The case calculation notebook has been split into a case definition notebook that contains only definitional stuff and lives on the case level, and a user calculation notebook that lives on the user level.
- The case definition notebook now has all cells set to be initialization cells and has an associated autosave `.m` file like the model definition notebook.
- The user calculation notebook is responsible for loading the case definition, which in turn is responsible for loading the model definition. Functions are provided to make this easy.

How It All Works (Gory Detail)

The following sequence illustrates how everything gets loaded for a case `{"myfolder","mysubfolder","mycase"}` of the quad model when the toolkit has been installed in `$UserBaseDirectory/Autoload/PendulumToolkit`.

1. When the Mathematica kernel launches, `$UserBaseDirectory/Kernel/init.m` file is discovered and run (assuming it has been created as recommended for Mathematica v7 or earlier). If it is actually Mathematica v7 or earlier that is running, it adds `$UserBaseDirectory/Autoload/PendulumToolkit` to `$Path`.
2. The `$UserBaseDirectory/Autoload/PendulumToolkit/init.m` file is discovered and run, and for Mathematica v8 or later adds `$UserBaseDirectory/Autoload/PendulumToolkit` to `$Path`.
3. Regardless of version, `$UserBaseDirectory/Autoload/PendulumToolkit/init.m` also defines a number of useful functions for managing the hierarchy, all based on `notebookdirectory[]` which returns the directory of the notebook from which it is executed (typically a user calculation notebook). The key ones are `loadcasefromuser[]` which attempts to load a case definition, and `loadmodelfromuser[]` which attempts to load a model definition.
4. The user calculation notebook, `QuadLite2Lateral/myfolder/mysubfolder/mycase/stdcalc/ASUS4XLLateralModelCalc.nb` is responsible for calling `loadcasefromuser["ASUS4XLLateralCaseDefn.m"]` to run the `.m` version of the case definition notebook in the directory one level above (`QuadLite2Lateral/myfolder/mysubfolder/mycase/ASUS4XLLateralCaseDefn.m`).
5. The case definition notebook is responsible for setting `modelcase = {"myfolder","mysubfolder","mycase"}` and calling `loadmodelfromuser["ASUS4L2LateralModelDefn.m"]` to load the model. `loadmodelfromuser[]` searches upward from the case directory (`QuadLite2Lateral/myfolder/mysubfolder/mycase`) to the model directory (`QuadLite2Lateral`).

Notes

- The name of the function `loadmodelfromuser[]` is potentially confusing because it reflects the anticipated situation at runtime rather than anything about the source code. The call to the function will normally appear in the case definition notebook, and will be copied to the associated `.m` file by the autosave process. The `.m` file will be run by a call to `loadcasefromuser[]` in the user calculation notebook, so the user calculation notebook will be the current notebook for the execution of both calls. If you evaluate `loadmodelfromuser[]` directly in the case definition notebook with Shift-Return or the like, it will fail. Normally this should not be a problem because there's so little non-trivial code in most case definition files that it can be debugged simply by saving changes and evaluating the `loadcasefromuser[]` line in the user calculation notebook. If you do ever have a particularly complicated case definition notebook that you would like to debug a cell at a time you can change `loadmodelfromuser[]` to `loadmodelfromcase[]` temporarily. In Mathematica v8 there's a variable `$InputFileName` by which the `.m` version of the case definition notebook could know where it is, but this has been avoided for backward compatibility.

- In older versions of the toolkit, some functions like `saveprecomputed[]` assumed that the Mathematica working directory was set to the case directory, and the case notebook was responsible for setting it. That requirement has been eliminated from the toolkit and the setup for it has been eliminated from the `default` template cases, but if it matters to your own code you can always reinstate it with `SetDirectory[casedirectory[]]` after the model has been loaded.
- The search behaviour of `loadmodelfromuser[]` is a natural generalization of the old system. Formerly the standard code for loading a model would search the case directory and the model directory in that order. That way, a customised model (say with added asymmetry) in the case directory would be given precedence. Now the search goes from the case directory via all intermediate levels to the model directory. If you have a group of cases with the same type of asymmetry, consider taking advantage of this by putting them in a subfolder with a shared custom model definition.

aLIGO: Suspensions/MathematicaModels/DirectoryStructure (last edited 2014-05-27 09:20:53 by MarkBarton)

aLIGO SUS SVN Structure and Installation Instructions for Mathematica Suspension Models

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page

This page describes how the generic directory structure described at DirectoryStructure works in practice with the aLIGO SUS SVN:

- How to install the toolkit from the SVN.
- How to install models from the SVN.
- How to work with existing cases.
- How to create new cases.
- How to adapt old-format cases.

SVN 101

SVN is variously short for Apache Subversion (a particular open-source version control system) or Subversion repository, a virtual filesystem that is hosted on a server computer by Subversion server software and contains current and past versions of files that have been committed to it. Files on the server are identified by browser-style URLs and can be viewed by a web browser in a pinch, but are intended to be accessed by special Subversion client software. It is common to run a parallel WebSVN server that gives a more browser-friendly view of the same files. For the aLIGO SUS repository the root URLs for the two views are

- https://redoubt.ligo-wa.caltech.edu/svn/sus (SVN-native)
- https://redoubt.ligo-wa.caltech.edu/websvn/listing.php?repname=sus& (WebSVN)

LIGO.ORG credentials are required for read access. For write access, additional authorization from David Barker of LHO is required. Hereafter, all URLs will be given in the SVN-native form, or with ^ used as shorthand for the root, e.g., `^/trunk/Common/MathematicaModels`.

Between a server and client, a number of useful operations are supported:

- Importing a set of files from the client to populate the repository or create a new subtree.
- Checking out a working copy of some or all of the repository to a client machine. A working copy includes extra information in hidden directories (`.svn` or the like) that allows the client to keep track efficiently of which files are under version control, which have been modified, etc, etc, while minimizing network operations.
- Exporting a clean copy of some or all of the repository with no hidden directories.
- Committing changes from some or all of the working copy to create a new revision on the server.
- Resolving conflicts when multiple users attempt to commit different changes to the same file.
- Updating some or all of the working copy to the latest revision on the server, or some historic revision.
- Adding, deleting, moving and copying files and directories.

See http://subversion.apache.org/ for software and http://svnbook.red-bean.com/ for documentation.

aLIGO SUS SVN Structure

To make stuff easy to find and prevent different users from trampling each other, the following rules are ordained for the Mathematica models section of the aLIGO SUS SVN:

- The only case of a model that should ever exist in the model directory is the template case, `default`.
- All model cases relevant to aLIGO production hardware should exist in a folder named according to Jeff Kissel's four-letter scheme used throughout the SVN:
  - QUAD, for QUADruple suspension
  - BSFM, for Beam Splitter / Folding Mirror
  - HSTS, for Ham Small Triple Suspension
  - HLTS, for Ham Large Triple Suspension
  - OMCS, for Output Mode Cleaner Suspension
  - TMTS, for Transmission Monitor and Telescope Suspension
  - HAUX, for Ham AUXiliary Suspension
  - OFIS, for Output Faraday Isolation and Suspension
  - HTTS, for Ham Tip-Tilt Suspension
- For suspensions under the scope of the aLIGO SUS group there will be more rules - stay tuned.
- Folders for non-SUS suspensions can be organized according to the discretion of the respective groups (but please do use some discretion!).
- Personal stuff should exist in folders with the LIGO.ORG name of the respective user (e.g., `albert.einstein`). These can be at the top level and/or wherever else it makes sense.
- What users do on their own computers is their business. However if they don't put personal work in personal folders, they may not add it to the SVN, and it may be deleted on sight if it shows up.

Toolkit Installation Instructions

The overall goal here is (i) get the files onto your system, and (ii) make sure that all the .m files are on the Mathematica directory search path (`$Path`). Feel free to adapt the suggestions to suit the setup of your computer, but avoid changing any of the existing files, lest you cause problems for other users if your changes ever get committed. (Implicit in that of course is, please don't commit changes to toolkit files! Rather, please send requests for, or trial implementations of, suggested bug fixes or cool new features to Mark Barton, who will make every effort to incorporate them without breaking anything.)

The example terminal sessions assume a Mac OS X terminal session and file layout, but the commentary is written so as to make it as obvious as possible how to adapt it to other flavours of Unix, as well as Windows.

1. Decide whether you want to make the toolkit available to all users on your system, or just yourself.
2. If all users, then in Mathematica, see what the value of `$BaseDirectory` is. If just yourself, see what `$UserBaseDirectory` is:
   ```
   In[1]:= $BaseDirectory
   Out[1]= /Library/Mathematica

   In[2]:= $UserBaseDirectory
   Out[2]= /Users/aeinstein/Library/Mathematica
   ```
3. Check that the directory you just identified exists and has a subdirectory `Autoload`, creating them if necessary.
4. Check out `^/trunk/Common/MathematicaModels/PendulumTookit` into `Autoload`
   ```
   # go to the Autoload directory
   $ cd /Users/aeinstein/Library/Mathematica/Autoload
   # do checkout - svn will prompt for username and password
   $ svn checkout https://redoubt.ligo-
   wa.caltech.edu/svn/sus/trunk/Common/MathematicaModels/PendulumToolkit
   # now have /Users/aeinstein/Library/Mathematica/Autoload/PendulumToolkit with files
   ```

5. If you're running Mathematica v8, that's it. The `init.m` file that is part of the checked-out collection will be automatically found and read when Mathematica starts up, and contains a command to add its enclosing directory to `$Path`. Otherwise continue to the next step.
6. If you're running Mathematica v7 or earlier, then depending on whether you took the `$BaseDirectory` or the `$UserBaseDirectory` route, open `kernel_init_template.nb` or `user_kernel_init_template.nb` and follow the instructions therein. Briefly, you will be told to make a copy of the file under the name init.nb in a certain nearby directory and edit it to point back to where the files are. (The reason for having you create the patch elsewhere is to avoid it accidentally getting committed back to the SVN.)

Notes:

- In Unix (including Mac OS X), Mathematica follows symlinks, so if you would much rather keep the toolkit directory with your LIGO files, you can do that and put a symlink to it in one of the Autoload directories. Symlinks in Windows Vista and 7 probably work as well. However Mac OS X aliases definitely don't work and Windows shortcuts probably don't.
- If at a later time you want to get the latest versions of the toolkit files, just do an update:

```
$ cd /Users/aeinstein/Library/Mathematica/Autoload/PendulumToolkit
$ svn update
```

Generic Model Installation Instructions

The instructions assume a Mac OS X terminal session but should be readily adaptable to other flavours of Unix, and Windows). Change `albert.einstein` to your own LIGO.ORG name and fill in the other placeholders such as `<modelname>` according to the following table:

Suspension Type	<modeldirectory>	<modeldefnfilename>	<casedefnfilename>	<usercalcfilename>	<userquickcalcfilename>
Quad: QUAD	QuadLite2Lateral	ASUS4L2LateralModelDefn	ASUS4XLLateralCaseDefn	ASUS4XLLateralModelCalc	ASUS4XLLateralModelQuickCalc
Triple: BSFM, HLTS, HSTS	TripleLite2	ASUS3L2ModelDefn	ASUS3L2ModelCaseDefn	ASUS3L2ModelCalc	ASUS3L2ModelQuickCalc
Double: OMCS, TMTS	TBD	TBD	TBD	TBD	TBD
Single with blades: HAUX, HTTS	TwoWireSimpleBlades	TwoWireSimpleBladesDefn	TwoWireSimpleBladesCaseDefn	TwoWireSimpleBladesCalc	TwoWireSimpleBladesQuickCalc
Single: OFIS?	TwoWireSimple	TwoWireSimpleDefn	TwoWireSimpleCaseDefn	TwoWireSimpleCalc	TwoWireSimpleQuickCalc

Clean Installation

1. If you've never worked with a particular model before, or don't care to blend old and new, do a shallow checkout of the appropriate model directory of the repository to a new local directory:

```
# go to a convenient place where there is no existing directory <modeldirectory>
$ cd /Users/aeinstein/ligo
# do checkout - svn will prompt for username and password and create a new directory - note
use of --depth switch
$ svn checkout --depth immediates https://redoubt.ligo-
wa.caltech.edu/svn/sus/trunk/Common/MathematicaModels/<modelname>
# now have /Users/aeinstein/ligo/<modeldirectory> with model definition files, empty
default case directory, and empty versions of any folders
```

2. Fully check out the `default` case:

```
# still in /Users/aeinstein/ligo
$ cd <modeldirectory>
$ svn update --set-depth infinity default
```

3. Make yourself a user directory in the `default` case by copying either the `stdcalc` or `quickcalc` pseudo-user template directories within `default`. An invisible `.svn` directory may come along for the ride, but the svn client is smart enough to realize that it is inapplicable in the copy and ignore it.)

```
# still in /Users/aeinstein/ligo/<modeldirectory>
$ cd default
$ cp -R stdcalc albert.einstein # or, cp -R quickcalc albert.einstein
```

4. Open the "plots" calculation notebook in your user directory (`albert.einstein/<usercalcfilename>Plots.nb` or `albert.einstein/<userquickcalcfilename>Plots.nb`), customize as you see fit, and check that if you select and evaluate all the cells, a status window opens, precomputed results are read (or the quick calculation is done), and plots are regenerated.
5. Repeat the previous step for the "export" calculation notebook (`albert.einstein/<usercalcfilename>Export.nb` or `albert.einstein/<userquickcalcfilename>Export.nb`) and check that results are exported to your user directory.
6. Selectively extend checkout to include folders of interest, e.g, `albert.einstein` or `XYZW`, hereafter `<interestingfolder>`. If you think you may want to pick and choose among the subfolders of the folder, use `--set-depth immediates`, otherwise `--set-depth infinity`.

```
# still in /Users/aeinstein/ligo/<modeldirectory>
$ svn update --set-depth immediates <interestingfolder> # or --set-depth infinity
# repeat for other folders of interest
```

7. Move down a level, inside `<interestingfolder>`, and check out additional cases and subfolders of interest.

Parallel Installation

If you have an existing installation of a particular model, you can checkout on top of it and have old and new cases coexisting:

1. Review what files and directories exist in the model directory on the SVN (should be just the model definition files, the case directory for `default`, some user folders, and some aLIGO folders).

```
$ svn list https://redoubt.ligo-
wa.caltech.edu/svn/sus/trunk/Common/MathematicaModels/<modelname>
```

2. Identify the model directory of your existing installation (where you set all the instances of `modeldirectory` to point to, hereafter `<modeldirectoryofexisting>`).

3. Delete or rename out of the way any top-level subdirectories that conflict with ones on the server (typically just default).

```
# go to existing model directory
$ cd /Users/aeinstein/ligo/<modeldirectoryofexisting>
$ mv default defaultold
```

4. Delete or rename out of the way the .nb and .m versions of the model definition file.

```
# still in /Users/aeinstein/ligo/<modeldirectoryofexisting>
$ mv <modeldefnfilename>.nb <modeldefnfilename>old.nb
$ mv <modeldefnfilename>.m <modeldefnfilename>old.m
```

5. Do the checkout.

```
# still in /Users/aeinstein/ligo/<modeldirectoryofexisting>
$ svn checkout --depth immediates https://redoubt.ligo-
wa.caltech.edu/svn/sus/trunk/Common/MathematicaModels/<modelname> . # note "." = current
directory
```

6. If you have missed any directory conflicts the operation will abort leaving everything unchanged - just fix them and retry. If you missed any file conflicts, nothing will be overwritten but the old, local versions will be marked as version-controlled and modified - consider reverting them to the new versions (with svn revert) to avoid accidentally committing them. Non-conflicting files will also be left in place but will be left as unversioned.
7. Continue from Step 2 of the clean installation instructions, substituting <modeldirectoryofexisting> for <modeldirectory>.

Other Operations

Making yourself a user folder

Any personal work that's not narrowly related to aLIGO production hardware and that you want to commit to the SVN must be in a personal folder.

1. Make yourself a user *folder* (directories used for grouping different cases of a model are called folders):

```
# go to model directory
$ cd /Users/aeinstein/ligo/<modelname>
$ mkdir albert.einstein
```

2. (Optional) Add it to the SVN:

```
# still in /Users/aeinstein/ligo/<modelname>
$ svn add albert.einstein
$ svn commit
```

Note that as you subsequently add cases of a model to your user folder, you need to add them in the analogous way:
$ cd albert.einstein; svn add interestingcase; svn commit.

Creating New Cases

The easiest way to proceed is to copy the case directory for an existing case and customize it. The following steps illustrate the process for a case mycase of the quad model, to be based on the default case:

1. Duplicate the default subdirectory and rename the copy mycase. Unless you are *quite* sure that you will never add this case to the SVN, you should put it in a personal folder (e.g., albert.einstein) or other appropriate SVN-legal folder.
2. Look in mycase/stdcalc and mycase/quickcalc and delete any output files such as QuadLite2Lateral_default_MatlabSS.m left over from the default calculation.
3. Open mycase/ASUS4XLLateralCaseDefn.nb (the case definition notebook).
4. Find the modelcase = "default" statement in the source and change default to mycase (i.e., to match the new subdirectory name). (Or, if you're putting it in a folder or subfolder, make it a list of strings as appropriate: modelcase = {"albert.einstein","mysubfolder","mycase"}
5. Find the modelcasecomment = "..." statement in the source and edit the text in the quotes as appropriate for the new case.
6. Find the overrides = {} statement and add substitutions as appropriate to define the new case. Be sure to comment the substitutions liberally using the new-style commenting provisions.
7. Save and close mycase/<casedefnfilename>.
8. Open mycase/stdcalc/<userplotsfilename>.
9. Find the assignment to useprecomputed in the destination and change it to useprecomputed = False.
10. Select and evaluate the whole notebook, debugging the case definition notebook if any errors are noticed. Be sure to save the case definition notebook after any edits so that the associated <casedefnfilename> file is generated. It's also a good idea to quit the Mathematica kernel each time (with the Evaluation->Quit Kernel->Local menu item) so that you're starting from a known state.
11. Change the assignment to useprecomputed back to useprecomputed = False and save the notebook. Once you as the model author are satisfied that the results in precomputed are sensible, nobody else should evaluate anything with useprecomputed = False ever again.
12. When you have mycase/stdcalc/<userplotsfilename> working, open and evaluate all the cells in mycase/stdcalc/<userexportfilename> to evaluate the standard set of exported results. (Feel free to customize this as appropriate.)
13. Even if you are the model author, once you have the stdcalc subdirectory set up, create your own user directory based on it and use that to be going on with.
14. If there are any other user directories, feel free to delete them. The affected users are responsible for recreating them if they have an interest in your new case.

Converting Old Cases

If you have a case in the old format that you want to modernize, the easiest way to proceed is probably to copy the directory for the default case and transplant the customized bits. The following steps illustrate the process for a case myoldcase of the quad model:

1. Rename the mycase subdirectory to mycasebackup or the like.
2. Duplicate the default subdirectory and rename the copy mycase. Unless you're *quite* sure you will never want to upload it to the SVN, move it to a folder or subfolder.
3. Delete any output files such as QuadLite2Lateral_default.txt left over from the default calculation.
4. Open mycasebackup/ASUS4XLLateralModelCalcMyCase.nb (hereafter "the source") and mycase/ASUS4XLLateralModelCaseDefn.nb (hereafter, at least for the next few steps, "the destination").
5. Find any introductory comments that are specific to mycase and copy them over comments mentioning default in the destination.
6. Find the modelcase = "..." statement in the source and copy the contents of the quotes to the equivalent spot in the destination. If you moved the case to a folder or subfolder, convert the RHS to a list and prepend the sequence of subfolders: modelcase = "mycase" becomes modelcase = {"myfolder","mysubfolder","mycase"} or the like.

`modelcase = {"myfolder", "mysubfolder", "mycase"}` or the like.

7. Find the `modelcasecomment = "..."` statement in the source and copy the contents of the quotes to the equivalent spot in the destination.
8. Find the `overrides = {...}` statement in the source and copy the contents of the braces to the equivalent spot in the destination. Update any old-style comments to use the new commenting mechanism (see section "Adding New-Style Comments" below).
9. Find any cells that implement modifications to the structure of the model and copy them to the destination. Make sure they have the Initialization Cell attribute checked (`Cell->Cell Properties->Initialization Cell` menu item). (The same applies to any complete cells you copy into new-style case definition notebooks from old-style case calculation notebooks.)
10. Save and close `mycase/ASUS4XLLateralCaseDefn.nb`.
11. Open `mycase/stdcalc/ASUS4XLLateralModelCalcPlots.nb`, hereafter the (new) destination.
12. Find any useful custom plots or tables in the source and copy them to the destination. Delete any of the standard plots or tables in the destination that are not of interest.
13. Find the assignment to `useprecomputed` in the destination and change it to `useprecomputed = False`.
14. Select and evaluate the whole notebook, debugging if any errors are noticed.
15. Change the assignment to `useprecomputed` back to `useprecomputed = True`, and save and close the notebook.
16. Open `mycase/stdcalc/ASUS4XLLateralModelCalcExport.nb` and do Select All, Evaluate, Save and Close. (You don't need to touch the assignment to `precomputed`.) Check that the standard export files are generated and look plausible.
17. If the quick version of the calculation may be useful (i.e., if your case is of a supported type, and is symmetrical and you don't need damping or thermal noise results), repeat the previous step with `mycase/quickcalc/ASUS4XLLateralModelCalcQuickPlots.nb` and `mycase/quickcalc/ASUS4XLLateralModelCalcQuickExport.nb`. Otherwise delete the `quickcalc` subdirectory.

Adding New-Style Comments

Older models will have an `overrides` list where the entries which define values for quantities are commented with ordinary Mathematica comments:

```
overrides = {
    ...
    m3 -> 41, (* extra heavy optic per email from Norna, 5/4/11 *)
    ...
};
```

The problem with this is that Mathematica totally ignores the comment text, making it unavailable to uses such as summary reports and export to Matlab. Thus the processing mechanism has been extended to accept comment strings via a function `nb[]`:

```
overrides = {
    ...
    m3 -> 41, nb[]->"extra heavy optic per email from Norna, 5/4/11",
    ...
};
```

Note the usage of commas: the substitution for `nb[]` is an entry in the list and needs to be set off from any adjacent entries by commas. (It's easy to end up with one missing comma and one spurious comma at the end of the list with a search and replace on `(*` and `*)`.) For your convenience, considerable processing is done on the comments between `overrides` and the fully numeric form `constval`. Any substitutions to an argumentless `nb[]` will be filled in based on the immediately preceding non-comment substitution (i.e., `nb[]` becomes `nb[m3]` in the above example). Then (if it's not already the case) all comments are sorted to appear in `constval` immediately after the substitution they relate to. The expression `findcomment[x,constval]` can then be used to find the comment string for x and the function `tablecomments[constval]//TableForm` can be used to print a neat table of substitutions and comments.

Mathematica Suspension Modeling Toolkit

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page

Introduction

The toolkit is a collection of Mathematica packages. As is conventional for packages, each has a .nb file and a .m file. The .nb file is the source and contains the comments. The .m file is the version actually loaded. The .nb file has the `AutoGeneratedPackage` option set so that the .m file is automatically regenerated whenever the .nb file is edited.

- `PendUtil.*`: The pendulum-specific utilities.
- `StatusWindow.*`: Utilities to display progress and timing information in a status window. Loaded directly by `PendUtil`.
- `RotationsXYZ.*`: A package similar to the standard Mathematica package `Geometry\Rotations` except that it uses a yaw, pitch and roll convention for angles. Loaded directly by `PendUtil`.
- `MyShapes.*`: This package is obsolete. It had two functions: (i) in versions of Mathematica prior to v6, some 3D shapes in the standard Mathematica package GraphicsShapes did not respond properly to `Translate[]` and `Rotate[]` commands, so better-behaving equivalents were defined, and (ii) some additional shapes of interest, like open-ended boxes and cylinders, were defined. Since Mathematica v6 introduced radical changes to the graphics system, new versions of all the routines needed to be written, and it was decided to move all the functionality into `PendUtil`. `PendUtil` checks which version of Mathematica is running and loads old or new definitions appropriately. However `MyShapes` is still left in the distribution for the benefit of very old code.
- `IFOModel.*`: *Badly out of date.* A Mathematica implementation of key parts of the Bench model of a generic LIGO interferometer, including all the physical and interferometer specific constants. Loaded by all the standard models, but otherwise optional.
- `MatlabExport.*`: Utilities to export numbers, numerical vectors and matrices as Matlab ".m" files. Loaded by many calculation notebooks.
- `E2EExport.*`: Utilities to export numbers, numerical vectors and matrices as text files suitable for importing into the E2E model. Loaded by many calculation notebooks.
- `QuickQuadLateral.*` and `QuickTriple.*`: Utilities to do a slightly approximate but quick (seconds rather than minutes) version of the calculation for the quad and triple models. Off-diagonal MOI terms are supported, but other asymmetries, damping and thermal noise are not.
- `QuadLateralMatlabParamExport.*` and `TripleMatlabParamExport.*`: Utilities to export numerical values defining a case of the quad or triple models as a `quadopt.m` or `triplep.m` file for use by the corresponding Matlab models.
- `ViolinModes.*`: Utilities contributed by Ben Lee, formerly of UWA, which add violin modes.
- `init.*`: A one-liner which, in Mathematica v8 at least, identifies the name of the directory it is run from and adds it to the Mathematica path.
- `kernel_init_template.nb` and `user_kernel_init_template.nb`: Templates for init.* files that support Mathematica v7 and earlier.

Installation

See SVNInstructions.

Usage

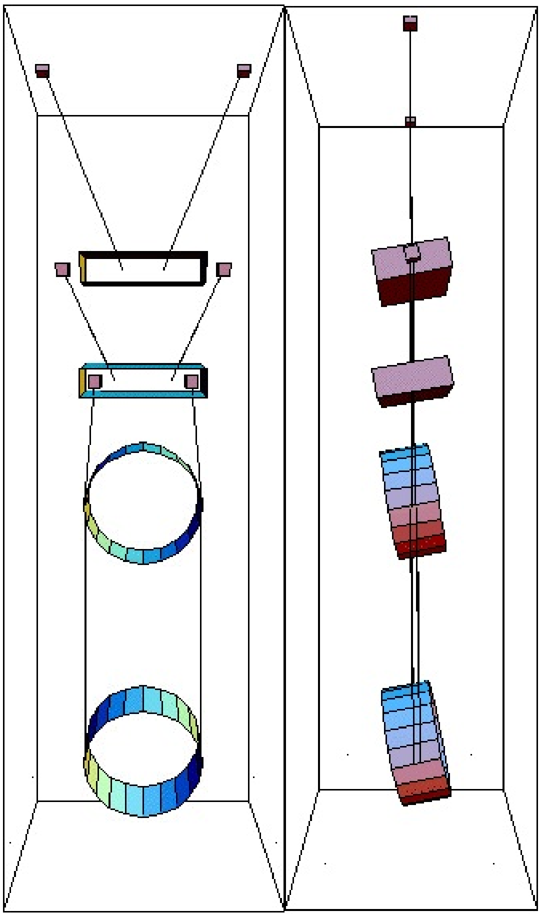
[Add lots more here.]

aLIGO: Suspensions/MathematicaModels/Toolkit (last edited 2014-05-27 09:20:35 by MarkBarton)

Quad Lateral Model

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page.

This reflects the conceptual design of the test-mass suspension for Advanced LIGO, with 4 masses, 6 blade springs and 14 wires. The "Lite2" in the name reflects the fact that blades are modelled as massless. The "Lateral" reflects the fact that blade lateral compliance is allowed for.



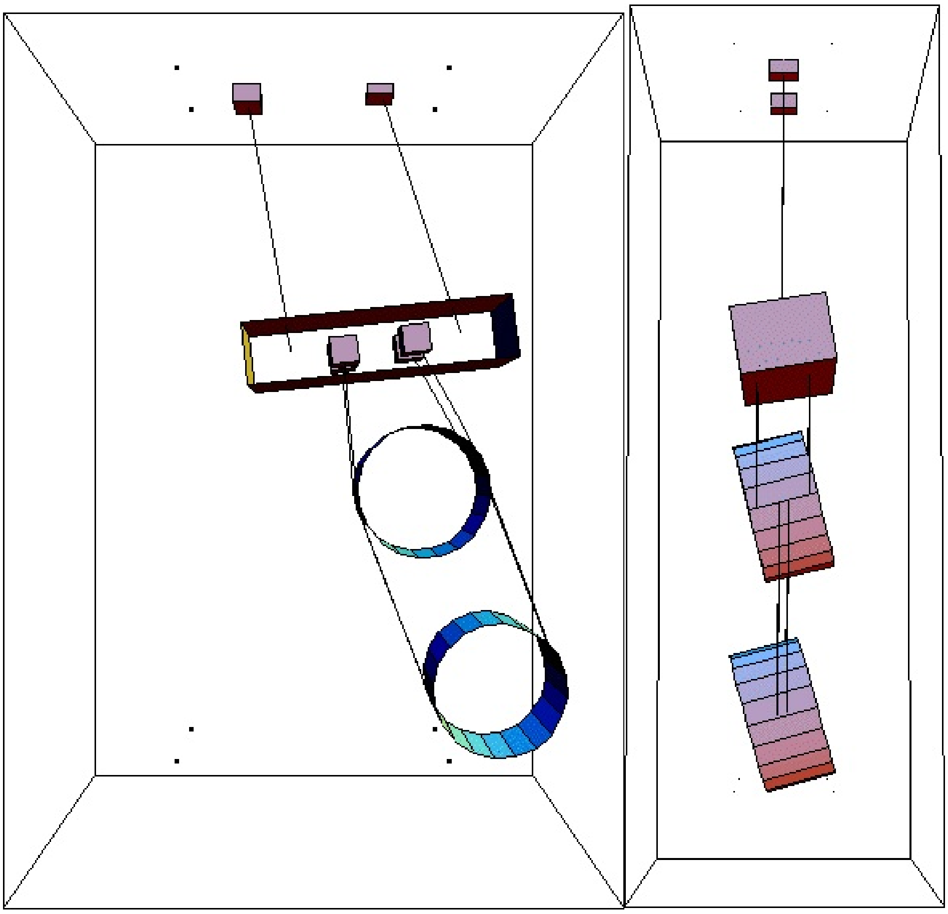
Installation

See SVNInstructions for instructions on download and installation from the SVN, as well as other common operations, such as creating a new case.

Triple Model

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page.

This reflects the conceptual design of the GEO suspension (or aLIGO BS or HLTS or HSTS), with 3 masses, 6 blade springs and 10 wires. The "Lite2" in the name reflects the fact that the blades are modelled as massless. Blade lateral compliance is not included because it turns out to be negligible for all the aLIGO triples. (But a version with it exists if anyone needs it.)



Installation

See SVNInstructions for instructions on download and installation from the SVN, as well as other common operations, such as creating a new case.

Cases

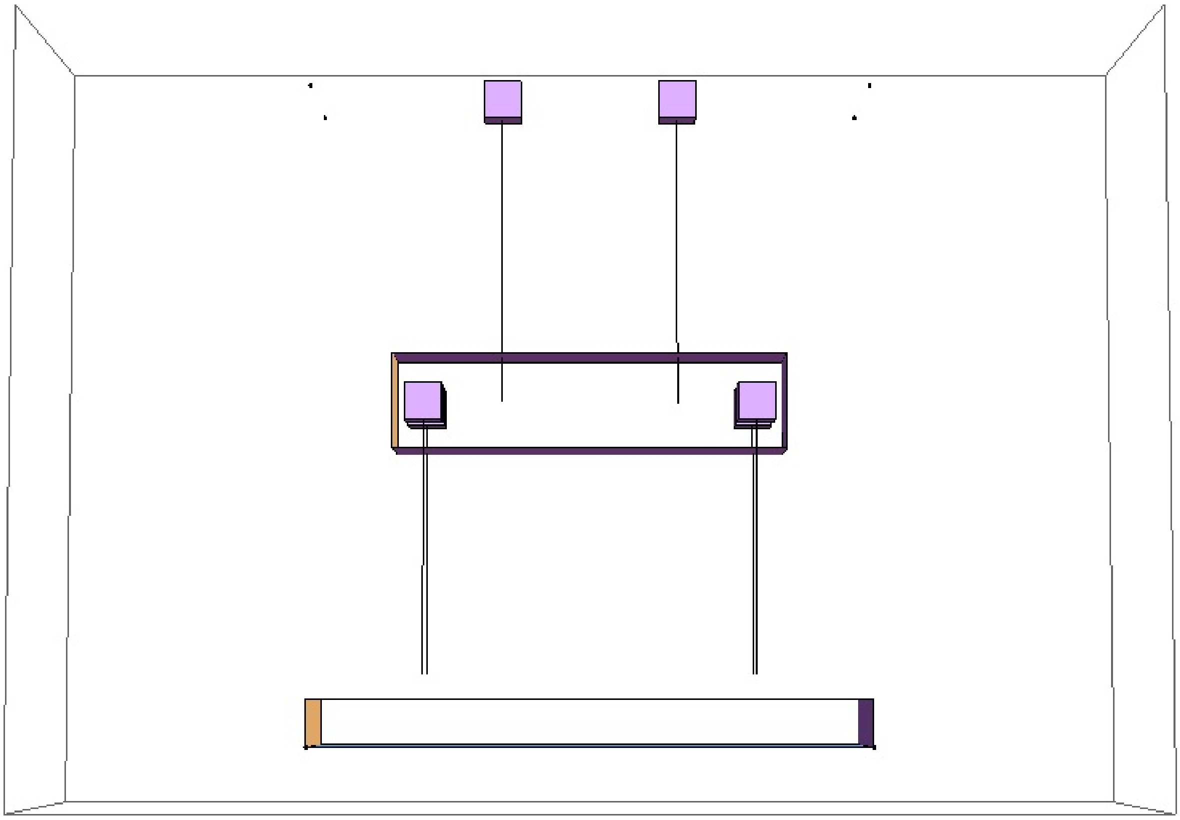
See the BSFM, HLTS and HSTS model pages for cases of particular interest to aLIGO.

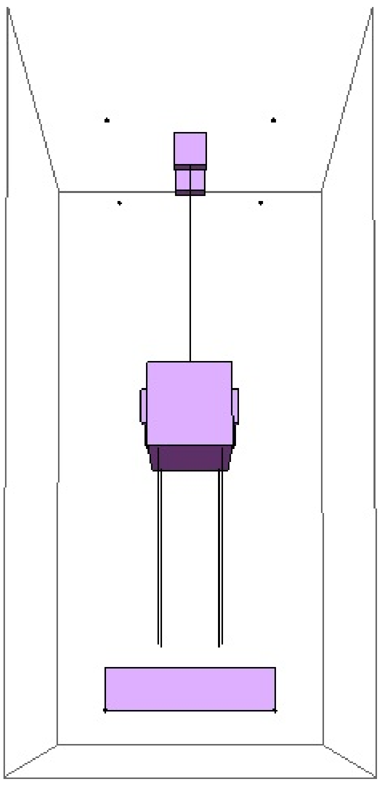
- BSFM: Suspensions/OpsManual/BSFM/Models
- HLTS: Suspensions/OpsManual/HLTS/Models
- HSTS: Suspensions/OpsManual/HSTS/Models

(Plain) DualLite2 Model

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page.

This reflects the conceptual design of the OMC suspension, with 2 masses, 6 blade springs and 6 wires. The "Lite2" in the name reflects the fact that the blades are modelled as massless. Blade lateral compliance is not included because it turns out to be negligible for all the aLIGO triples.





Installation

See SVNInstructions for instructions on download and installation from the SVN, as well as other common operations, such as creating a new case.

Cases

See the OMCS models page for cases of particular interest to aLIGO.

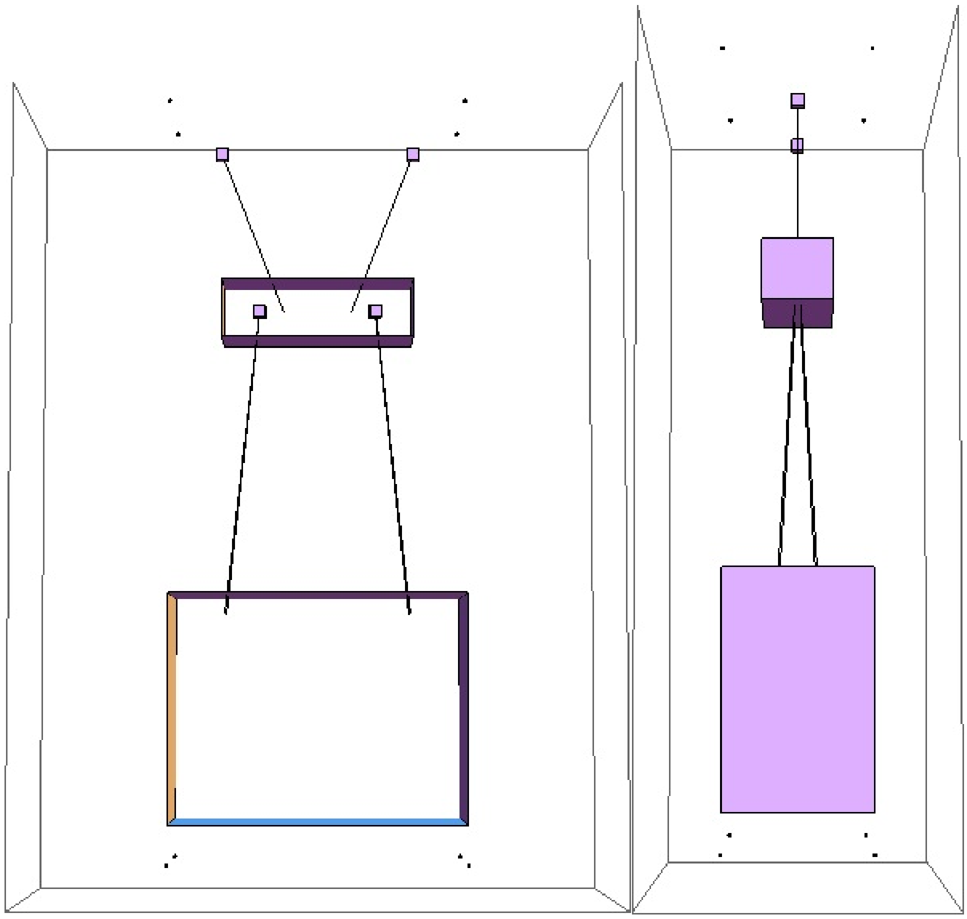
- OMCS: Suspensions/OpsManual/OMCS/Models

aLIGO: Suspensions/MathematicaModels/DualLite2Model (last edited 2014-05-27 09:17:27 by MarkBarton)

DualLite2DBLateral Model

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page.

This reflects the conceptual design of the TMTS suspension, with 2 masses, 4 blade springs and 6 wires. "DB" in the name reflects taht there are two blades at the top mass rather than four as for OMC. The "Lite2" reflects the fact that the blades are modelled as massless. Blade lateral compliance is included because it is important in the quad suspensions which the top mass was adapted from. Toe-out of the wires in both the frnt-back and side-side directions is allowed for to accommodate the TMTS (most models, including the plain DualLite2 for the OMCS, only have side-side toe-out).



Installation

See SVNInstructions for instructions on download and installation from the SVN, as well as other common operations, such as creating a new case.

Cases

See the TMTS models page for cases of particular interest to aLIGO.

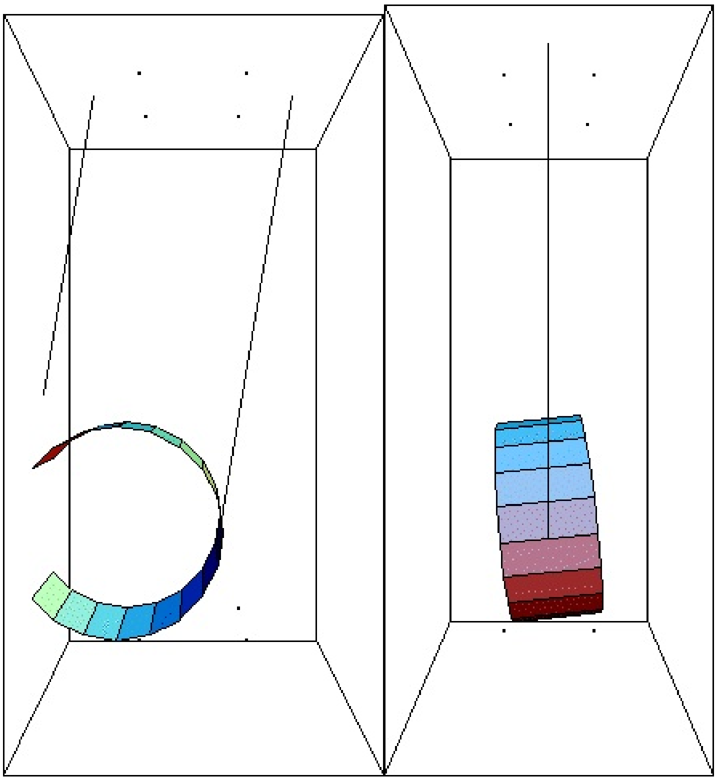
- TMTS: Suspensions/OpsManual/TMTS/Models

aLIGO: Suspensions/MathematicaModels/DualLite2DBLateralModel (last edited 2014-05-27 09:17:43 by MarkBarton)

Two Wire Simple Pendulum with Blades

Mathematica Suspension Models Main Page
SUS Operations Manual Main Page.

This is a two-wire (one loop) single mass suspension with blades as for the aLIGO TipTilts and aLIGO HAM Aux.



Installation

See SVNInstructions for instructions on download and installation from the SVN, as well as other common operations, such as creating a new case.

aLIGO: Suspensions/MathematicaModels/TwoWireSimpleBladesModel (last edited 2014-05-27 09:18:48 by MarkBarton)