

# LIGO Laboratory / LIGO Scientific Collaboration

LIGO- T1100607-v1

advanced LIGO

12/2/11

# EtherCAT Setup of Modbus Devices

Daniel Sigg

Distribution of this document: LIGO Scientific Collaboration

This is an internal working note of the LIGO Laboratory.

California Institute of Technology LIGO Project – MS 18-34 1200 E. California Blvd. Pasadena, CA 91125 Phone (626) 395-2129 Fax (626) 304-9834 E-mail: info@ligo.caltech.edu

**LIGO Hanford Observatory P.O. Box 159 Richland WA 99352** Phone 509-372-8106 Fax 509-372-8137 Massachusetts Institute of Technology LIGO Project – NW22-295 185 Albany St Cambridge, MA 02139 Phone (617) 253-4824 Fax (617) 253-7014 E-mail: info@ligo.mit.edu

LIGO Livingston Observatory P.O. Box 940 Livingston, LA 70754 Phone 225-686-3100 Fax 225-686-7189

http://www.ligo.caltech.edu/

### 1 Introduction

This document describes the setup of the <u>D1100251</u>, the 384 Channel Acromag Binary Output chassis, and the HMS AB9000, Anybus X-gateway Modbus-TCP for EtherCAT. The D1100251 contains 4 Acromag ES2113-0100 units. These are 96 channel binary input/output modules that are controlled through a Modbus-TCP interface. Combining these units with the Anybus X-gateway makes the IO channels transparently accessible through EtherCAT. Look for data sheets, manuals, application notes and setup software in C1107420.

### 2 Setting up the ES2113

The first step is to set up the IP address. Look up the available addresses in <u>E1101144</u>. Looking from the front into the D1100251 chassis, the PCB 1 is on the top left. The PCB 2 is on the top right, the PCB 3 is on the bottom left and PCB 4 is on the bottom right. Locate the manual and the application note for the ES2113 in <u>C1107420</u>.

By default each ES2113 has an IP address of 128.1.1.100. Hook up the Ethernet of the first unit to a computer and make sure its IP address is 128.1.1.111. Try to open a web page with http:// 128.1.1.100. If this doesn't work, one may have to factory reset the unit. For this turn it off, pull the toggle switch to the up position and turn on the power. Now hold the toggle switch in the up position for at least 10 seconds. After releasing the unit should have reset itself to 128.1.1.100.

No make sure the network configuration page looks like Figure 1. The default user name and password are User/password. Do not change this. Make sure you have the correct IP address and subnet mask. For the gateway use the "X.X.X.1" address. We are using a static IP address and ports are operated in hub configuration. You can use the wink on/off button to turn on/off a blinking green LED at the front of the Acromag unit. This is to make sure that the correct PCB is selected.

The IO configuration page should be left in the default position after factory reset as seen in Figure 2.

Continue this procedure for all 4 PCBs. Finally, daisy chain the Ethernet of all 4 PCBs and hook it to the rear panel CAT5 feedthroughs.

Add a label to the D1100251 that lists the IP addresses as configured.

advLIGO

Firefox Y		×
Network Configuration Page		*
( 10.80.32.11/configur.htm	- ← C 🛃 - acromag 👂 1	
🔧 Google	Bookma	irks
ES2113 Network Co	onfiguration Page	1
IP Address	Control	
Static IP Address 10.80.32.11	Number of Sockets	
Gateway	Subnet Mask	
Host NameES2	113M11	
Active IP Address10.80.32.11	MAC Address00:01:C3:00:52:9D	
© Use Static IF	Addressing	
C Use DHCP	Use BOOTP	
O USE DITET WITHOUT	KS OSE DOOTF WITAIDACKS	
Ethernet Por	rt Control	
Port 1	Port 2	
C Auto Negotiate	O Auto Negotiate	
C 10 Mbps/Half Duplex C 10 Mbps/Full Duplex	© 10 Mbps/Half Duplex © 10 Mbps/Full Duplex	
<ul> <li>100 Mbps/Half Duplex C 100 Mbps/Full Duplex</li> </ul>	100 Mbps/Half Duplex C 100 Mbps/Full Duplex	
Port 1 Status: 100M, Half Duplex	Port 2 Status: No Link On This Port	
Port 1 Fiber Optio	n Not Installed	
C Swi	tch	
• Hut	<b>)</b>	
subm	it	
Wink On	/Off	
		-
× Find: Next 🛊 Previous 🖍 Highlight all 🗖 Matgh ca	ase	

Figure 1: ES2113 network configuration.

advLIGO



Figure 2: ES2113 IO Configuration page.

#### 3 Setting up the AB9000

Locate the user manual for the AB9000 in <u>C1107420</u>. Make sure the AB9000 contains a SD card. Then, hook up the Ethernet of the Modbus side of the AB9000 to a computer. By default its IP address should be set to 192.168.0.100. You can use the Anybus IPconfig utility, to locate this address (part of C1107420). This utility may also allow you to change the IP address of the unit temporarily. If so, set it to the same subnet as the ES2113 units but with the address ".1". If not set the computer network port to the same subnet as the AB9000 but a different address, say

192.168.0.111. Now, point a web browser to 192.168.0.100. It should somewhat similar to Figure 3.

0 0	Allyb	us X-gateway Modbus	-TCP - Eth	erCAT		
OVERVIEW Home Configuration	Anybus X-gateway configura the left side menu to navigate. Management page. 'Network' the controlled network, where	tion and status web pages. Welcome to Changes to the configuration do not take I' represents the controlling network, whe the X-gateway acts as a client.	the configuration into effect until the X-gat re the X-gateway act	erface of the A eway is restar s as a server.	nybus X-gate ted from the > 'Network 2' re	way. Use (-gateway epresents
Authentication		Identification	F	thernet link s	status	
Modbus Client	Product name:	Anybus X-gateway Modbus-TCP	Port 1:			
Modbus Servers	Firmware version:	1.03	Speed:	100 Mb	ps	
EtherCAT	Serial number:	A015214D	Duplex:	Half Du	plex	
TOOLS	MAC ID:	00:30:11:06:04:D2	Port 2:			
X-gateway	Uptime:	0 days, 0h:3m:0s	Speed:	100 Mb	ps	
Management	CPU Load:	35% (auto updated every 5s)	Duplex:	Full Du	plex	
Backup & Restore		Operation Mode	Eti	hernet link st	atistics	
Manning Overview	EtherCAT (Network 1):	I/O data exchanged	In pkts:	6596	Errors:	0
mapping Overview	Modbus-TCP (Network 2):	Run	Out pkts:	9344	Errors:	0
Monitor						

Figure 3: Home page of the AB9000.

Select the Modbus client page and make sure to change the IP address, subnet mask, DHCP setting, HICP, start-up operation mode and freeze setting so it looks like in Figure 4 but with the correct IP address. Save the settings.

Select the Modbus server page. Add the first Modbus server which should correspond to PCB 1 of D1100251. Leave the port and protocol at 502 and TCP, respectively. Now select the transaction link and add 3 transaction as shown in Figure 6. Go back to the server list and add the next server for PCB 2. Add the same transactions and repeat adding servers for PSB 3 and 4. The final page should look similar to Figure 5. Save the settings.

Select the EtherCAT page and enable the mapped live list. The setup should look like Figure 7.

Finally, go to the X-gateway management page and store the settings and reboot. If the IP address was changed, the new IP address will go into effect after the reboot of the AB9000. Make sure the computer network port is set accordingly. Now, connect the D1100251 Modbus Ethernet port to the second Modbus port of the AB9000. It should now be possible to look at the web pages of all 5 devices.

≥fox ▼	+			
10.80.32.1/conf.	/client.html		☆ - C 🚼 - acromag	۹ م
oogle				💽 Bookma
() Anybus'	Anybus X-ga	ateway Modbus-TC	P - EtherCAT	
OVERVIEW Home CONFIGURATION	Modbus client configuration (Network 2). (	Configure the Modbus-TCP (Netwo	rk 2) settings.	
Authentication		IP Configuration		
Modbus Client	IP address	10.80.32.1		
Modbus Servers	Subnet mask	255.255.255.0		
EtherCAT	Router IP address	0.0.0.0		
TOOLS	DHCP	Disabled	<u>.</u>	
X-gateway	Anybus IPconfig (HICF	P) Enabled		
Management		I/O settings		
Backup & Restore	Start-up operation mo	de Running	<u>.</u>	
Mapping Overview	When EtherCAT (Netw	ork1) is not Freeze data to !	Modbus servers 🔹	
Transaction	exchanging I/O data	Cancel Save setting	e	
Monitor		Cancer Save Setting	5	
© 2011 HMS Industrial Ne	tworks - All rights reserved		Con	necting Devices™
			ht	tp://10.80.32.1/conf/client.l
ind:	📕 Next 🎓 Previous 🖉 Highlight all	Match case		

Figure 4: AB9000 Modbus client setup.

Firefox 🔻									
Anybus X-gateway		+							*
♦ ♦ 🗍 10.80.3	2.1/conf/nodelist.html				ŕ	7 - C	谢 - acr	omag	۹ 🍙
Google								E	Bookmarks
() Anybu	s'	Anybus X	-gate	way Mod	lbus-TCP -	Ethe	rCAT		
OVERVIEW	Modbus server	configuration (Network	( 2). Add, (	edit or delete M	odbus connections f	or Netwo	rk <mark>2. On e</mark> a	ach connection several	
Home	server propertie	wards a modbus server ( and finally click the 'Tr	can be set ansactions	up. Press Add s' link to set up	transactions to call M	lodbus fi	ew connec unctions o	n the server. The globa	1
CONFIGURATIO	Iimit is 64 trans	actions and it is not pos	sible to ma	ap more proces	ss data than the X-ga	teway ca	n handle.	-	
Authentication									
Modbus Client	Transactions:	ation limits. 12/64 Input process dat	a: 112/256	bytes Output	process data: 48/256	i bytes l			
Modbus Server	S Alias	ID addrose	Dort	Protocol	Transactions	-,			
EtherCAT	ES2113M	10.80.32.11	502	TCP	3	Edit	Delete	Transactions	
TOOLS	ES2113M	12 10.80.32.12	502	TCP	3	Edit	Delete	Transactions	
X-gateway	ES2113M	13 10.80.32.13	502	TCP	3	Edit	Delete	Transactions	
Management	ES2113M	14 10.80.32.14	502	TCP	3	Edit	Delete	Transactions	
Backup & Rest	ore			Add nev	w server				
Mapping Overv	ew								
Transaction Monitor									
© 2011 HMS Ind	strial Networks - All rights	s reserved						Connecting Devices http://10.80.32.1/conf	™ /nodelist.html
× Find:	₩ <u>N</u> ext 1	🗈 Brevious 🔗 Highlight j	Mat	ch case					

Figure 5: AB9000 Modbus server setup.

Firefox 🔻												_	
Anybus X-gateway	+	•											*
♦ 10.80.32.1/0	onf/node.html							合-(	2) 🚺 - a	acromag		م	
Google												🖸 Bo	okmarks
() Anybus'		Any	/bus X	-gatew	ay M	odbu	us-TC	P - Eth	nerCA	г			
OVERVIEW Home CONFIGURATION Authentication	Transactions co transactions tow transaction proper handle.	nfiguration ards a Mod erties. The tion limits.	n. Add, edit o Ibus server o global limit i	or delete tran can be set uj is 64 transad	sactions ). Press ' tion and	used fo Add trar it is not	r a conne isaction' possible	ections on this button to add to map more	s page. On a new trans process da	each conn saction, the ata than th	ection : en edit i e gatew	several t to set vay can	
Modbus Servers	Transactions: 12	2/64   Minim	num allowed	l scan time: (	36 Input	process droce	s data: 11	2/256 bytes	Dutput proc	ess data: 4	48/256	bytes	
EtherCAT	E	ES2113M11	1	24	10.80	.32.11		502		TC	P	1	
TOOLS	#	Function	Encoding	Scan time	Timeou	UID	Address	Data Type	Elements	Register	S		
X-gateway Management	ReadBits WriteBits	3 16	BBEWLE	250 250	5000 5000	255 255	54 60	uint16 uint16	6 6	6 6	Edit Edit	Delete Delete	
Backup & Restore	ReadStatus	4	BBEWLE	1000	5000	255	1	uint16	7	7	Edit	Delete	
Mapping Overview		Hoverin	ig mouse ov	er an elemei Rack to co	nt where	the curs	or shows	s a question n	nark displaj	ys help.			
Transaction Monitor				Back to se	rveriisi			ransaction	J				
© 2011 HMS Industrial	Networks - All rights	reserved								Conne	ecting D	)evices™	
× Find:	<b>↓</b> <u>N</u> ext <b>↑</b>	Previous	🖉 Highlight g	Match	case								

Figure 6: AB9000 Modbus server transaction setup.

Firefox 🔻				_	
Anybus X-gateway	+				-
(+) 10.80.32.1/c	onf/fb.html		☆ - C	🛃 – acromag 🛛 🔎	
Google				🖸 Bo	okmarks
() Anybus'	Anybus X-gat	eway Modbus-T	CP - Ether	CAT	
OVERVIEW Home	EtherCAT configuration (Network 1). Configure control/status word or the live list affects the pro-	re the Network 1 side of the X ocess data size.	(-gateway, Enabling o	or disabling the mapping of the	
CONFIGURATION Authentication					
Modbus Client	Global configuration limits. Transactions: 12/64 Input process data: 112/	256 bytes   Output process da	ata: 48/256 bytes		
Modbus Servers	Setti	ng	Configured		
EtherCAT	Station alias	0			
TOOLS	When Modbus-TCP (Net	work 2) error Freez	e data to master	•	
X-gateway Management	I/O mapped control/statu	s word Disab	led		
Backup & Restore	I/O mapped live list	Concol Cove cotti	led	×	
Mapping Overview	1	Cancer Save Setti	ngs		
Transaction Monitor	]				
© 2011 HMS Industria	Networks - All rights reserved			Connecting Devices™	
× Find:	🐺 🕸 Next 👚 Previous 🖉 Highlight all 🗖	Mat <u>c</u> h case			

Figure 7: AB9000 EtherCAT setup.

Impute Section 1         Overview: Anybus X-gateway Modbus-TCP - EtherCAT         OVERVIEW         Data Mapping Overview. Shows how configured transactions, control word, status word and live list are mapped from Network 2 to Network 1 in the X-gateway and vice versa.         OVERVIEW         Modbus Client       Object Transaction alias       Element size (bytes)       Elements         Modbus Servers       0x2102       1       8       0x2102       6         EtherCAT       Object View       Ox2102       ReadBits       2       6         Management       0x210a       ReadStatus       2       7       0x2102       ReadBits       2       6         Backup & Restore       Monitor       Ox2101       ReadStatus       2       7       0x2101       ReadStatus       2       7       0x2101<	→ 10.80.32.1/1	ool/mapping.html			ź	- C 🛃	<ul> <li>acromag</li> </ul>	2
Overwiew       Data Mapping Overview. Shows how configured transactions, control word, status word and live list are mapped from Network 2 to Network 1 in the X-gateway and vice versa.         ONFIGURATION       Authentication         Modbus Client       Object       Transaction alias       Element size (bytes)       Elements         Kagteway       Object       Transaction alias       2       7         Management       0x2102       Live-List       1       8         Backup & Restore       0x2102       ReadBits       2       6         Monitor       0x2101       ReadBits       2       6         OUTPUT DATA       Object       Transaction alias       Element size (bytes)       Elements         0x210b       ReadBits       2       6       0x2102       ReadBits       2       6         0x2101       ReadBits       2       6       0x2101       ReadBits       2       6         0x2110       ReadBits       2       6       0x2111       ReadStatus       2       7         0x2102       ReadBits       2       6       0x2111       ReadStatus       2       7         0x2105       WriteBits       2       6       0x2185       1       6 <td< th=""><th>oogle</th><th></th><th></th><th></th><th></th><th></th><th></th><th>Bool</th></td<>	oogle							Bool
OVERVIEW         Bate Mapping Overview. Shows how configured transactions, control word, status word and live list are mapped from Network 2 to Network 1 in the X-gateway and vice versa.         CONFIGURATION       Authentication         Modbus Client       Object Transaction alias       Elements         Modbus Servers       0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/0/	() Anybus'	An	ybus X-	gateway M	odbus-TCP -	EtherCA	т	
Authentication       INPUT DATA       Object view         Modbus Servers       Object       Transaction alias       Element size (bytes)       Elements         EtherCAT       0x2102       Live-List       1       8         TOOLS       0x2102       Live-List       1       8         0x2102       ReadBits       2       6         0x2100       ReadBits       2       6         0x21010       ReadStatus       2       7         0x2102       ReadStatus       2       7         0x2102       ReadStatus       2       7         0x21010       ReadStatus       2       7         0x2102       ReadStatus       2       7         0x21010       ReadStatus       2       7         0x21010       ReadStatus       2       7         0x2101       ReadStatus       2       7         0x2110       ReadStatus       2       7         0x2111       ReadStatus       2       7         0x2112       ReadStatus       2       7         0x2111       ReadStatus       2       7         0x2185       WriteBits       2       6	OVERVIEW Home Configuration	Data Mapping Overview. to Network 1 in the X-gate	Shows how cor way and vice ve	nfigured transaction 'sa.	s, control word, status wo	ord and live list	are mapped from Network 2	
Modbus Client       Object view         Modbus Servers       Object       Transaction alias       Element size (bytes)       Elements         EtherCAT       0x2102       Live-List       1       8         TOOLS       0x210a       ReadBits       2       6         X-gateway       0x210c       ReadBits       2       6         Management       0x210c       ReadBits       2       7         Backup & Restore       0x210d       ReadStatus       2       7         Mapping Overview       0x2107       ReadBits       2       6         0x2101       ReadStatus       2       7       0x2101       ReadStatus       2       7         Mapping Overview       0x2101       ReadStatus       2       7       0       0x2110       ReadStatus       2       7         OUTPUT DATA       0x2110       ReadStatus       2       7       0       0       0x2111       ReadStatus       2       7       0         OUTPUT DATA       Object view       Object view       Object view       Object view       Object view       Output Bits       2       6       0       0       218       WriteBits       2       6	Authentication							T I
Modbus ServersObjectTransaction aliasElement size (bytes)ElementsEtherCAT0x2102Live-List18TOOL S0x210aReadBits26X-gateway Management0x210cReadStatus27Backup & Restore Mapping Overview0x210cReadBits26Mapping Overview Monitor0x2101ReadStatus27OUTPUT DATA0x2101ReadStatus27OUTPUT DATAObjectTransaction aliasElement size (bytes)Elements0x2185WriteBits260x21850x21020x2186WriteBits260x21860x218520x2188WriteBits260x21880x21880x21880x2188WriteBits260x21880x21880x21850	Modbus Client			0	biect view			
EtherCAT         0x2102         Live-List         1         8           TOOL S         0x210a         ReadBits         2         6           X-gateway Management         0x210c         ReadStatus         2         7           Backup & Restore         0x210c         ReadBits         2         6           Mapping Overview         0x210f         ReadStatus         2         7           Transaction Monitor         0x2101         ReadStatus         2         7           OUTPUT DATA         0x2101         ReadStatus         2         7           Outreut DATA         0x2101         ReadStatus         2         7           Outreut DATA         0x2111         ReadStatus         2         7	Modbus Servers		Object	Transaction alias	Element size (bytes)	Elements		
TOOLS       0x210a       ReadBits       2       6         X-gateway Management       0x210b       ReadStatus       2       7         Backup & Restore       0x210c       ReadBits       2       6         Mapping Overview       0x210e       ReadBits       2       6         Transaction Monitor       0x2110       ReadStatus       2       7         OUTPUT DATA       0x2111       ReadStatus       2       7         OUTPUT DATA       Object       Transaction alias       Element size (bytes)       Elements         0x2185       WriteBits       2       6       6         0x2185       WriteBits       2       6       6         0x2186       WriteBits       2       6       6         0x2188       WriteBits       2       6       6	EtherCAT		0x2102 Li	ve-List	1	8		
X-gateway Management       0x210b       ReadStatus       2       7         Backup & Restore       0x210c       ReadStatus       2       6         Mapping Overview       0x210f       ReadStatus       2       7         Transaction Monitor       0x2110       ReadStatus       2       7         OUTPUT DATA       0x2111       ReadStatus       2       7         OutPUT DATA       Object       Transaction alias       Element size (bytes)       Elements         0x2185       WriteBits       2       6       6         0x2186       WriteBits       2       6       6         0x2188       WriteBits       2       6       6	TOOLS		0x210a R	eadBits	2	6		
Management       0x210c       ReadBits       2       6         Backup & Restore       0x210d       ReadStatus       2       7         0x210e       ReadBits       2       6         Mapping Overview       0x210f       ReadStatus       2       7         Transaction       0x2110       ReadBits       2       6         Monitor       0x2111       ReadStatus       2       7         OUTPUT DATA       Object       Transaction alias       Element size (bytes)       Elements         0x2185       WriteBits       2       6       6         0x2186       WriteBits       2       6       6         0x2188       WriteBits       2       6       6	X-gateway		0x210b R	eadStatus	2	7		
Backup & Restore       0x210d       ReadStatus       2       7         Mapping Overview       0x210e       ReadStatus       2       6         0x2101       ReadStatus       2       7         Overview       0x2101       ReadStatus       2       7         0x2101       ReadStatus       2       7         0x2101       ReadStatus       2       6         0x2111       ReadStatus       2       7         OUTPUT DATA       Object       Transaction alias       Element size (bytes)       Elements         0x2185       WriteBits       2       6       6         0x2186       WriteBits       2       6       6         0x2188       WriteBits       2       6       6	Management		0x210c R	eadBits	2	6		
Mapping Overview       0x210e       ReadBits       2       6         0x210f       ReadStatus       2       7         0x2110       ReadBits       2       6         0x2111       ReadBits       2       6         0x2111       ReadStatus       2       7         OUTPUT DATA         Object view         Object View     <	Backup & Restore	6	0x210d R	eadStatus	2	7		
Ox210F       ReadStatus       2       7         Transaction       0x2110       ReadBits       2       6         0x2111       ReadStatus       2       7         OUTPUT DATA       Object       Transaction alias       Element size (bytes)       Elements         0x2185       WriteBits       2       6       6         0x2187       WriteBits       2       6       6         0x2188       WriteBits       2       6       6	Manning Overview		0x210e R	eadBits	2	6		
Output DATA     Object     Transaction       0x2111     ReadBits     2     6       0x2111     ReadStatus     2     7         Output DATA       Object     Transaction alias       Element size (bytes)     Elements       0x2185     WriteBits     2     6       0x2187     WriteBits     2     6       0x2188     WriteBits     2     6	Transportion		0x210f R	eadStatus	2	7		
OUTPUT DATA         Object       Transaction alias       Element size (bytes)       Elements         0x2185       WriteBits       2       6         0x2186       WriteBits       2       6         0x2187       WriteBits       2       6         0x2188       WriteBits       2       6         0x2188       WriteBits       2       6	Monitor		0x2110 R 0x2111 R	eadBits eadStatus	2	7		
Object ViewObjectTransaction aliasElement size (bytes)Elements0x2185WriteBits260x2186WriteBits260x2187WriteBits260x2188WriteBits26								1
ObjectTransaction aliasElement size (bytes)Elements0x2185WriteBits260x2186WriteBits260x2187WriteBits260x2188WriteBits26				0	bject view			
0x2185         WriteBits         2         6           0x2186         WriteBits         2         6           0x2187         WriteBits         2         6           0x2188         WriteBits         2         6           0x2188         WriteBits         2         6			Object	Transaction alias	Element size (bytes)	Elements		
0x2186         WriteBits         2         6           0x2187         WriteBits         2         6           0x2188         WriteBits         2         6			0x2185 W	riteBits	2	6		
0x2187 WriteBits 2 6 0x2188 WriteBits 2 6			0x2186 W	riteBits	2	6		
UX2188 WIREBITS Z D			0x2187 W	riteBits	2	6		
			0X2188 W	ntebits	2	0		

Figure 8: AB9000 Mapping.

Check the mapping on the AB9000 using the mapping overview page. It should look like Figure 8.

Up to 2 D1100251 can be connected to a single AB9000. Repeat the setup for the D1100251 but make sure to use a different set of IP addresses on the same subnet. Then, move to the Modbus server page on the AB9000 and add 4 corresponding server entries.

## 4 Setting up TwinCAT

Locate the EtherCAT slave TwinCAT application note as well as the ABXS\_ECT file in <u>C1107420</u>. The ABXS\_ECT file can also be downloaded from the HMS web page. This may be necessary if an AB9000 with a more recent firmware release is used. Copy the ABXS\_ECT file to "C:\Program files\TwinCAT\Io\EtherCAT\" before you start the TwinCAT system manager. Start the TwinCAT system manager.

Connect a EtherCAT output port of the computer to the EtherCAT input port of the AB9000. Make sure the EtherCAT NIC is recognized by the TwinCAT system manager. Now you should be able to scan for new devices and recognize the AB9000. Go to its process data tab and load the PDO information from the device. Now the input and output variables should be available. They have intuitive names likes SubIndex 001, etc. Try to go to active run. This should start the data transfer.



Figure 9: TwinCAT system manager.

Starting (active) run mode sometimes fails after the AB9000 has been reconfigured. The easiest remedy is to delete the X-gateway box and rescan the EtherCAT chain.

Finally, go back to the AB9000 web page and check the home page for errors. When working, it should look like Figure 3. You can also look the transaction monitor page and check for errors, see Figure 10.



Figure 10: AB9000 transaction monitor.

## 5 PLC Programming with TwinCAT

A TwinCAT example project with a single AB9000 and a single  $\underline{D1100251}$  can be found in the zip associated with this document. The example code uses three main structures for the hardware input, the hardware outputs and the user interface, respectively.

```
TYPE IscWhiteningIn :
STRUCT
     LiveList:
                        ARRAY[1..8] OF BYTE;
                       ARRAY[1..4,1..13] OF WORD;
      PCB:
      InfoDataState: WORD;
END STRUCT
END TYPE
TYPE IscWhiteningOut :
STRUCT
                        ARRAY[1..4,1..6] OF WORD;
     PCB:
END STRUCT
END TYPE
TYPE IscWhitening :
STRUCT
                 ARRAY[1..6] OF IscWhiteningChassis;
      Chassis:
END STRUCT
END TYPE
TYPE IscWhiteningChannel :
STRUCT
      Valid: BOOL; (* read only *)
Switches: BYTE; (* read only *)
     Toggle:
                       BYTE;
      SetVal:
                       BYTE;
END STRUCT
END TYPE
TYPE IscWhiteningChassis :
STRUCT
      Channels:
                       ARRAY[1..8] OF IscWhiteningChannel;
END STRUCT
END TYPE
```

Figure 11: Program structures and sub structures.

A function block is used to transfer data from the input and output structures to the user structure. The mapping of IO channels to the variables is described in  $\underline{T1100195}$ -v1. Two functions are used to obtain the channel and chassis index, respectively. The main program is straight forward and just calls the function block using global variables for the main structures.

```
FUNCTION BLOCK IscWhiteningFB
VAR INPUT
      In:
                  IscWhiteningIn;
END VAR
VAR OUTPUT
                  IscWhiteningOut;
     Out:
END VAR
VAR IN OUT
      Val:
                 IscWhitening;
END VAR
VAR TEMP
                  INT;
     pcb:
     port:
                 INT;
                 INT;
     chassis:
      chn:
                  INT;
     LiveVal:
                 WORD;
     OutVal:
                WORD;
END VAR
FOR pcb := 1 TO 4 DO
      FOR port := 1 TO 6 DO
            OutVal := 0;
            LiveVal := SHR (BYTE TO WORD (In.LiveList[1]) OR
                          SHL (BYTE TO WORD (In.LiveList[2]), 8),
                          3*(pcb-1)) AND 16#0007;
            chassis := IscWhiteningChassisIndex (pcb, port, TRUE);
            chn := IscWhiteningChannelIndex (pcb, port, TRUE);
            Val.Chassis[chassis].Channels[chn].Switches :=
                  WORD TO BYTE (In.PCB[pcb,port] AND 16#00FF);
            Val.Chassis[chassis].Channels[chn].Valid :=
                  (LiveVal = 16\#0007) AND
                  ((In.InfoDataState AND 16#3F) = 8) AND
                  ((In.PCB[pcb,port+7] AND 16#0002) = 0);
            Val.Chassis[chassis].Channels[chn].SetVal :=
                  Val.Chassis[chassis].Channels[chn].SetVal XOR
                  Val.Chassis[chassis].Channels[chn].Toggle;
            Val.Chassis[chassis].Channels[chn].Toggle := 0;
            OutVal := BYTE TO WORD
                  (Val.Chassis[chassis].Channels[chn].SetVal);
            chassis := IscWhiteningChassisIndex (pcb, port, FALSE);
            chn := IscWhiteningChannelIndex (pcb, port, FALSE);
            Val.Chassis[chassis].Channels[chn].Switches :=
                  WORD TO BYTE (SHR (In.PCB[pcb,port] AND 16#FF00, 8));
            Val.Chassis[chassis].Channels[chn].Valid :=
                  (LiveVal = 16\#0007) AND
                  ((In.InfoDataState AND 16#3F) = 8) AND
                  ((In.PCB[pcb,port+7] AND 16#0002) = 0);
            Val.Chassis[chassis].Channels[chn].SetVal :=
                  Val.Chassis[chassis].Channels[chn].SetVal XOR
                  Val.Chassis[chassis].Channels[chn].Toggle;
            Val.Chassis[chassis].Channels[chn].Toggle := 0;
            OutVal := OutVal OR SHL (BYTE TO WORD
                  (Val.Chassis[chassis].Channels[chn].SetVal), 8);
            Out.PCB[pcb,port] := OutVal;
     END FOR;
END FOR;
END FUNCTION BLOCK
```

Figure 12: Function block.

```
FUNCTION IscWhiteningChannelIndex : INT
VAR INPUT
      PCB:
                 INT;
      Port:
                 INT;
      LSB:
                 BOOL;
END VAR
VAR
END VAR
CASE PCB OF
      1:
           CASE Port OF
                  1 : IscWhiteningChannelIndex := 1;
                  2 : IscWhiteningChannelIndex := 3;
                  3 : IscWhiteningChannelIndex := 5;
                  4 : IscWhiteningChannelIndex := 1;
                  5 : IscWhiteningChannelIndex := 3;
                  6 : IscWhiteningChannelIndex := 5;
            ELSE
                  IscWhiteningChannelIndex := 0;
            END CASE;
      2:
            CASE Port OF
                  1 : IscWhiteningChannelIndex := 5;
                  2 : IscWhiteningChannelIndex := 7;
                  3 : IscWhiteningChannelIndex := 1;
                  4 : IscWhiteningChannelIndex := 5;
                  5 : IscWhiteningChannelIndex := 7;
                  6 : IscWhiteningChannelIndex := 1;
            ELSE
                  IscWhiteningChannelIndex := 0;
            END CASE;
            CASE Port OF
      3:
                  1 : IscWhiteningChannelIndex := 7;
                  2 : IscWhiteningChannelIndex := 1;
                  3 : IscWhiteningChannelIndex := 3;
                  4 : IscWhiteningChannelIndex := 7;
                  5 : IscWhiteningChannelIndex := 1;
                  6 : IscWhiteningChannelIndex := 3;
            ELSE
                  IscWhiteningChannelIndex := 0;
            END CASE;
      4:
            CASE Port OF
                  1 : IscWhiteningChannelIndex := 3;
                  2 : IscWhiteningChannelIndex := 5;
                  3 : IscWhiteningChannelIndex := 7;
                  4 : IscWhiteningChannelIndex := 3;
                  5 : IscWhiteningChannelIndex := 5;
                  6 : IscWhiteningChannelIndex := 7;
            ELSE
                  IscWhiteningChannelIndex := 0;
            END CASE;
      ELSE
            IscWhiteningChannelIndex := 0;
END CASE;
IF (NOT LSB AND (IscWhiteningChannelIndex > 0)) THEN
      IscWhiteningChannelIndex := IscWhiteningChannelIndex + 1;
END IF;
END FUNCTION
```

Figure 13: Channel index function.

```
FUNCTION IscWhiteningChassisIndex : INT
VAR INPUT
     PCB:
                 INT;
     Port:
                 INT;
     LSB:
                 BOOL;
END VAR
VAR
END VAR
CASE PCB OF
     1 : CASE Port OF
                 1..3 : IscWhiteningChassisIndex := 4;
                  4..6 : IscWhiteningChassisIndex := 1;
            ELSE
                 IscWhiteningChassisIndex := 0;
           END CASE;
      2 :
           CASE Port OF
                  1..2 : IscWhiteningChassisIndex := 5;
                  3 : IscWhiteningChassisIndex := 6;
                  4..5 : IscWhiteningChassisIndex := 2;
                        IscWhiteningChassisIndex := 3;
                  6 :
            ELSE
                  IscWhiteningChassisIndex := 0;
           END CASE;
      3 :
           CASE Port OF
                 1 : IscWhiteningChassisIndex := 4;
                  2..3 : IscWhiteningChassisIndex := 5;
                  4 : IscWhiteningChassisIndex := 1;
                  5..6 : IscWhiteningChassisIndex := 2;
            ELSE
                  IscWhiteningChassisIndex := 0;
           END CASE;
      4:
            CASE Port OF
                  1..3 : IscWhiteningChassisIndex := 6;
                  4..6 : IscWhiteningChassisIndex := 3;
            ELSE
                  IscWhiteningChassisIndex := 0;
            END CASE;
     ELSE
            IscWhiteningChassisIndex := 0;
END CASE;
END FUNCTION
```

Figure 14: Chassis index function.

```
VAR GLOBAL
     Whitening1In AT %IB0: IscWhiteningIn;
Whitening1Out AT %QB0: IscWhiteningOut;
     Whitening1:
                                   IscWhitening;
END_VAR
PROGRAM MAIN
VAR
     Whitening1FB: IscWhiteningFB;
Counter: INT := 200;
END_VAR
(* do some testing *)
IF (Counter <= 0) THEN
     Whitening1.Chassis[1].Channels[1].Toggle := 1;
     Counter := 200;
ELSIF (Counter = 100) THEN
      Whitening1.Chassis[5].Channels[7].Toggle := 16#F0;
      Counter := Counter - 1;
ELSE
     Counter := Counter - 1;
END IF;
END_PROGRAM
```

Figure 15: Main program.