

Sauter novaNet Driver for Niagara AX

User Guide

Contents

1	Introduction	2
2	Requirements	2
3	Sauter EY3600 Specifics	3
4	Network	3
5	Devices	6
6	Points	6
6.1	Point Discovery	7
6.2	Polling	9
6.3	Change-of-value subscription	9
7	Schedules	10
7.1	Sauter Time Programs	11
7.2	Niagara Schedules	11
7.3	Niagara Calendars	14
8	Troubleshooting	14
9	Use Niagara and Sauter SCADA	15

1 Introduction

Sauter ¹ EY3600 system – also known as Sauter EY-modulo 2 – includes a wide range of programmable controllers for HVAC control and building automation. The controllers communicate with a proprietary protocol called novaNet using two wire twisted pair.

Niagara AX novaNet driver is designed to enable communication between Sauter controllers and Tridium Niagara AX powered devices. It provides an efficient solution for retrofit of building management systems, facilitates seamless integration of multiple protocols and allows convenient makeover of front-end software.

A single Niagara AX powered device with novaNet driver could serve as an integration platform for various Sauter controllers: EY-modulo 2 (novaNet), EY-modulo 4 (LON), EY-modulo 5 (BACnet) as well as 3rd party systems like Modbus, KNX, M-Bus, SNMP.

2 Requirements

- Niagara AX 3.7² or later powered device such as Jace 2 / 3 / 6, Supervisor AX or their OEM versions
- Communication options
 - Jace 2 / 3 / 6: serial RS-232 via EYZ291 or TCP/IP via EYZ292
 - Supervisor AX: TCP/IP via EYZ292
- Sauter driver license

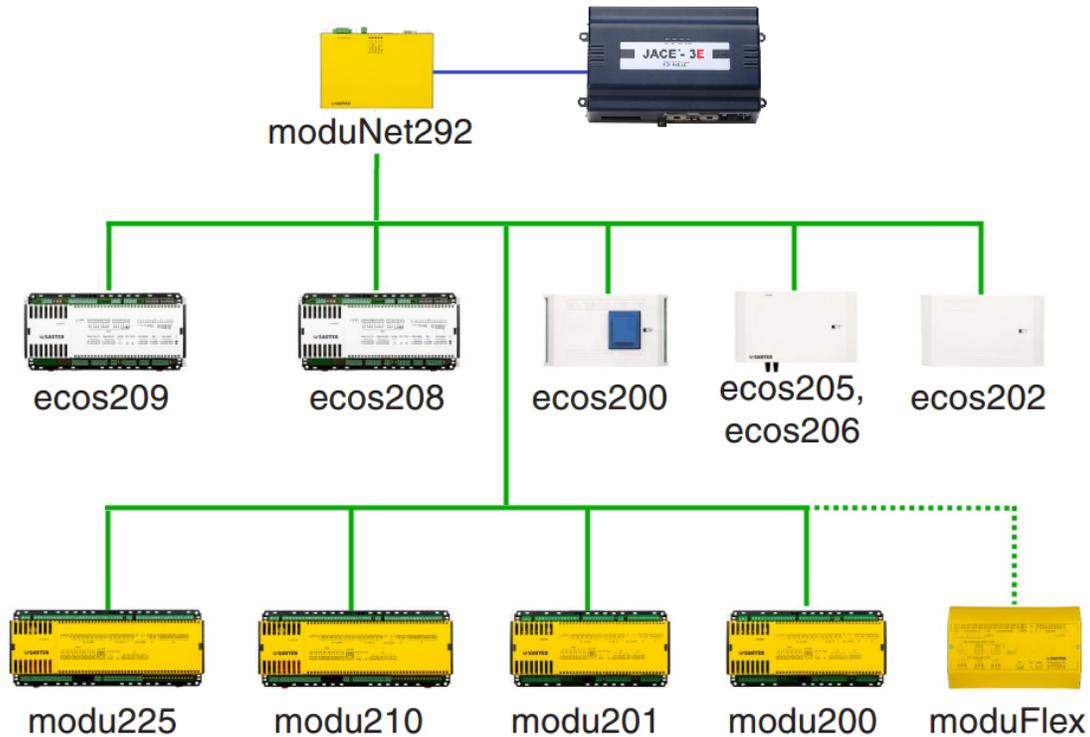
To enable driver functionality Sauter EY3600 system should be connected to Sauter novaNet router: serial EYZ291 or TCP/IP EYZ292 (same as EY-BU292).



Niagara Supervisor or Jace is connected to the novaNet network through the router as shown below.

¹All trademarks or registered trademarks are property of their respective owners

²If support for older Niagara versions is required, please contact the vendor



3 Sauter EY3600 Specifics

Sauter controller memory is split into 256 or 128 MFA (Machine Fine Addresses) areas. Each MFA consists of 128 DW (datawords, 4 bytes each).

Each point variable (e.g. analog input, digital output, software analog point) is assigned to one MFA. This information is controller model specific and is available in Sauter datasheets. It might be particularly useful in order to find the specific hardware I/O in the controller. MFA contains a lot of information: current value, override flag, units, name, description, etc. Each of these values is stored in DW.

The hardware address range with its text block area is located between MFA 00-59 and DW 00-63 in the RAM-Map. One of DW contains Card Code, which identifies the type of point: hardware or software, analogue or digital, input or output. For each MFA (hardware and software address), the text block area could store point name and description; these areas are called House Address and Address Text.

4 Network

In order to start communication between Jace and Sauter devices, make sure Jace is connected to router (directly to COM port for EYZ291 and through Ethernet for EYZ292), router is connected to Sauter controllers and everything is powered.

1. Install **sauter.jar** and all dependent modules via Software Manager
2. Start the station and add new **Sauter Network** or **Sauter Serial Network**, depending on router model

3. In network **Properties** enter the license number and restart the station

SauterNetwork (Sauter Network)

<input type="checkbox"/> Status	{ok}
<input type="checkbox"/> Enabled	<input checked="" type="radio"/> true
<input type="checkbox"/> Fault Cause	
<input checked="" type="checkbox"/> Health	Fail [null]
<input checked="" type="checkbox"/> Alarm Source Info	Alarm Source Info
<input checked="" type="checkbox"/> Monitor	Ping Monitor
<input checked="" type="checkbox"/> Tuning Policies	Tuning Policy Map
<input checked="" type="checkbox"/> Poll Scheduler	N Poll Scheduler
<input checked="" type="checkbox"/> Comm	Sauter Comm
<input checked="" type="checkbox"/> Comm Config	Sauter Tcp Comm Config
<input checked="" type="checkbox"/> Timesync Interval	02:00 {Sun Mon Tue Wed Thu Fri Sat }
<input type="checkbox"/> License	Enter license code

4. For EYZ292 router enter its **IP address** and **port** in **Router Address** field (note, there are two fields for address: Address and Router Address, use the second one). Standard router port is 51807.

It is recommended to check if EYZ292 could be pinged from PC first.

SauterNetwork (Sauter Network)

<input type="checkbox"/> Status	{ok}
<input type="checkbox"/> Enabled	<input checked="" type="radio"/> true
<input type="checkbox"/> Fault Cause	
<input checked="" type="checkbox"/> Health	Ok [23-May-16 10:30 BST]
<input checked="" type="checkbox"/> Alarm Source Info	Alarm Source Info
<input checked="" type="checkbox"/> Monitor	Ping Monitor
<input checked="" type="checkbox"/> Tuning Policies	Tuning Policy Map
<input checked="" type="checkbox"/> Poll Scheduler	N Poll Scheduler
<input checked="" type="checkbox"/> Comm	Sauter Comm
<input checked="" type="checkbox"/> Comm Config	Sauter Tcp Comm Config
<input type="checkbox"/> Fault Cause	
<input checked="" type="checkbox"/> Address	local:0
<input type="checkbox"/> Send Socket Timeout	120 s
<input type="checkbox"/> Server Socket Timeout	0 s
<input type="checkbox"/> Router Ping Period	30 s [0 - 120]
<input checked="" type="checkbox"/> Router Address	baud.ddns.net:1911
<input type="checkbox"/> Ip Address	192.168.0.120
<input type="checkbox"/> Port	unspecified 51807 [-1 - 65536]
<input checked="" type="checkbox"/> Timesync Interval	02:00 {Sun Mon Tue Wed Thu Fri Sat }
<input type="checkbox"/> License	MCwCFAN2g2g+SSNF4rtE/HFFf1X+w80wAhQpnSt3

For EYZ291 router enter used COM port name and its baud rate. Baud rate is set with jumpers on EYZ291.

SauterSerialNetwork (Sauter Serial Network)	
<input type="checkbox"/> Status	{ok}
<input type="checkbox"/> Enabled	<input checked="" type="radio"/> true
<input type="checkbox"/> Fault Cause	
<input checked="" type="checkbox"/> Health	Fail [null]
<input checked="" type="checkbox"/> Alarm Source Info	Alarm Source Info
<input checked="" type="checkbox"/> Monitor	Ping Monitor
<input checked="" type="checkbox"/> Tuning Policies	Tuning Policy Map
<input checked="" type="checkbox"/> Poll Scheduler	N Poll Scheduler
<input checked="" type="checkbox"/> Comm	Sauter Comm
<input checked="" type="checkbox"/> Comm Config	Sauter Serial Comm Config
<input type="checkbox"/> Fault Cause	
<input type="checkbox"/> Port Name	none
<input type="checkbox"/> Baud Rate	Baud9600
<input type="checkbox"/> Data Bits	Data Bits8
<input type="checkbox"/> Stop Bits	Stop Bit1
<input type="checkbox"/> Parity	None
<input type="checkbox"/> Flow Control Mode	<input type="checkbox"/> RtsCtsOnInput <input type="checkbox"/> RtsCtsOnOutput <input type="checkbox"/> XonXoffOnInput <input type="checkbox"/> XonXoffOnOutput
<input type="checkbox"/> Receive Timeout	0 ms [0 - 20000]
<input type="checkbox"/> Inter Message Delay	000000h 00m 00.000s [0ms - 1sec]

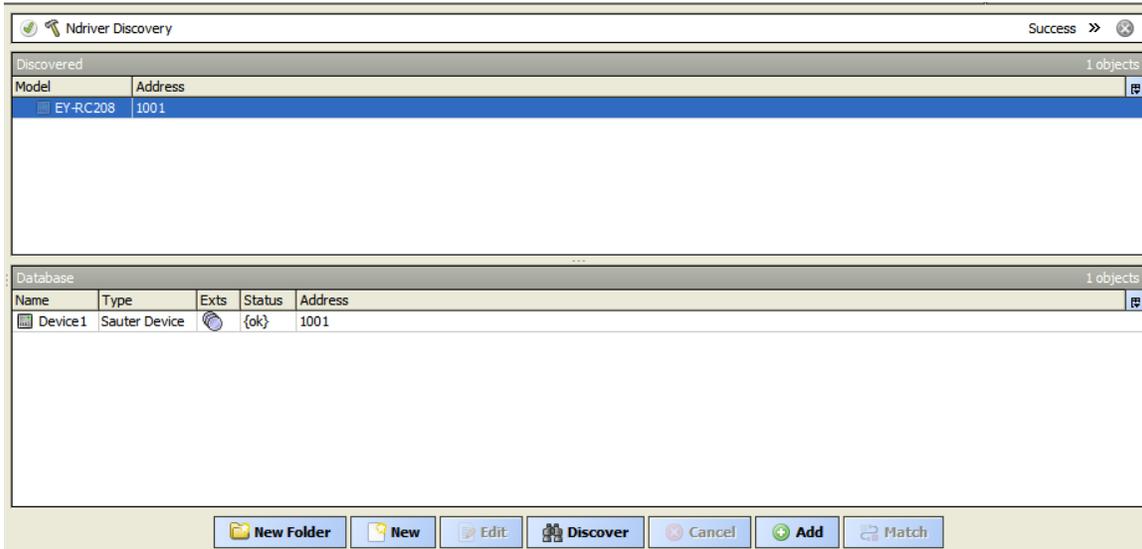
5. To check if Jace has established the communication with Sauter router, expand Comm component and check if **AS Counter Received** and **PC Counter Received** fields have changed their values from -1 and **Router Status** property is Running.

If counter fields display -1, try to ping router using network ping action (**Sauter Network** -> **Actions** -> **Ping**), then check port, baudrate or IP settings.

Comm Sauter Comm	
<input type="checkbox"/> Status	{ok}
<input type="checkbox"/> Router Status	Running
<input type="checkbox"/> As Counter Received	76
<input type="checkbox"/> Pc Counter Received	11
<input type="checkbox"/> Pc Counter Sent	11
<input type="checkbox"/> Address	31745
<input type="checkbox"/> Period	100 ms
<input type="checkbox"/> Timeout	5000 ms
<input checked="" type="checkbox"/> Cov	Cov Processor
<input type="checkbox"/> Add With Override Status	<input checked="" type="radio"/> false
<input type="checkbox"/> Point Name From	<input checked="" type="radio"/> Address Text
<input type="checkbox"/> Add Meta	<input checked="" type="radio"/> false
<input type="checkbox"/> Log Severity	Fine
<input checked="" type="checkbox"/> Comm Config	Sauter Tcp Comm Config
<input checked="" type="checkbox"/> Timesync Interval	02:00 {Sun Mon Tue Wed Thu Fri Sat }
<input type="checkbox"/> License	MCwCFAN2g2g+SSNF4rtE/HFFf1X+w80wAhQpnSt3

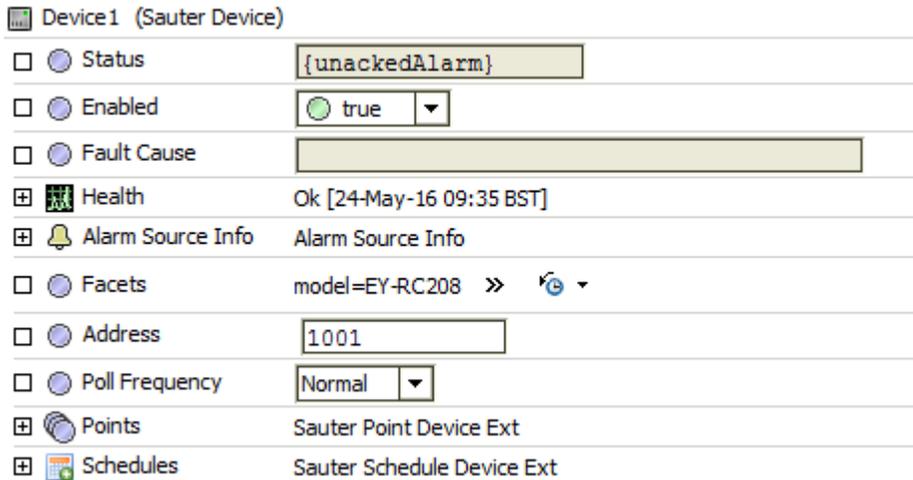
5 Devices

After connection with router is established, open the **Device Manager** and press **Discover** button.



Niagara will find all connected devices and will display their models and novaNet addresses. Press **Add** button to add them to Niagara station.

Sauter devices have address and poll frequency properties. Poll frequency specifies how often device points will be polled (if COV method is not used). Device facets contains controller model name.



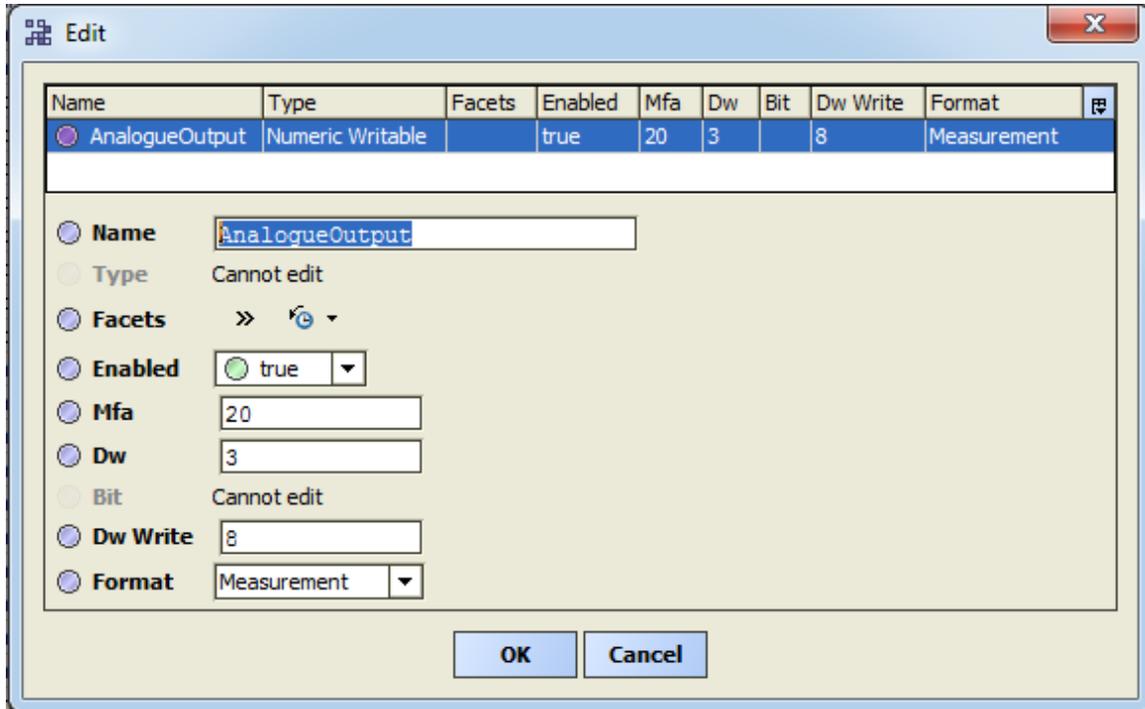
6 Points

Every Sauter point extension contain properties:

- MFA / DW – specify memory area and double word address in this area, which contains read-only data
- Format – specifies how raw value should be interpreted

There might be extra properties:

- DW Write – for writable points double word address, where data will be written
- Bit – in some cases Sauter boolean points (binary inputs or outputs) could be combined in a single MFA/DW. There could be up to 6 binary points stored together, e.g. fan speed control outputs. Point discovery manager will display them as a single point with configurable **bit** parameter. Duplicate this point and set **bit** parameter according to controller datasheet. Only one of these points could be overridden at the time to prohibit possible electrical faults with connected equipment.



There are two methods of reading point values: change-of-value (COV) subscription and polling. Polling is a simple time-based method, when necessary points are requested periodically with interval determined by poll frequency. This method allows to read any DW address. COV method is when driver asks controller to send point value when it changes. Using this method could be less traffic-intense and allows to see even fast changing point. Not all DW addresses could be COV subscribed.

6.1 Point Discovery

Open device **Sauter Point Manager** and click **Discover** to start point discovery. There are two discover modes available:

- Fast – discovers just point addresses (MFA/DW) and types (card codes)
- Full – also discovers point descriptors; takes longer to complete

The screenshot shows the 'Sauter Discovery' window with a 'Success' status. It displays a table of discovered points and an empty database table.

Card Code	Mfa	Dimension	House Address	Address Text
MV Ni/Pt1000 5s	4	C	Supply Temp	AHU 1
MV Ni/Pt1000 5s	5	C	Extract Temp	AHU1
MV UIR 1s	7	C	LTHW Inlet Temp	AHU1
MV Ni/Pt1000 5s	9	Pa	Supply Pressure	AHU1
AOut from ASIC	20	V	LTHW 3way actuator	AHU1
AOut from ASIC	21	V	Freeze Coil 3way Actuator	AHU1
AOut from ASIC	22	V	Supply Fan Speed	AHU1

Name	Type	Out	Mfa	Dw	Bit	Dw Write	Format

Niagara will find all available points, their types, units and attributes. Text descriptions will be available if programmers have included them into the program.

There are few settings under **Comm** component, which affect how discovered points are added to the station:

The screenshot shows the 'Sauter Comm' configuration window with various settings for communication and data handling.

Component	Setting	Value
Comm	Status	{ok}
	Router Status	Running
	As Counter Received	234
	Pc Counter Received	27
	Pc Counter Sent	27
	Address	31745
	Period	100 ms
	Timeout	5000 ms
Cov	Add With Override Status	false
	Point Name From	Address Text
	Add Meta	false
	Log Severity	Fine

- **Add with Override Status** – if true, driver will set DW, which contains point override status. The drawback is this DW is not suitable for change-of-value subscription and could only be polled
- **Point Name From** specifies how Niagara creates default point names, either from **Address Text** or from **House Address**.
- **Add Meta** – if true, then text fields and other meta-information will be saved in point facets. This might be useful to display points on graphical pages.

Add found points to Niagara station to start reading them. Note that all points are added as read-only by default. Change point types to writable before adding them to station if required.

If discovered point has dimension field, it will be automatically translated into Niagara units and assigned to the point.

Sauter points respond differently to writing commands, depending on their type and program configuration.

Some of points (hardware outputs, AIA_Soft, DIA_Soft) could be overridden: point will display "overridden" status and written value, but will return to original value if set to AUTO.

Other points (AI_Soft, BI_Soft) could only be written only if the program does not write to them already. In this case they would not retain their original value and if written value is null, point value will become nan (not a number).

6.2 Polling

Point polling method is used if

- Point **Tuning Policy** property **Use Cov** is set to false or
- Point DW property is not equal to 2 (binary value), 3 (measurement) or 6 (counter)

All polled points are grouped together and polled in groups to optimize network traffic. Group polling frequency could be set with **Device -> Poll Frequency** and **Network -> Poll Scheduler** settings.

Polled points could be read with deterministic frequency, but if point value changes more often some changes could be missed.

6.3 Change-of-value subscription

COV method is used if

- Point **Tuning Policy** property **Use Cov** is set to true and
- Point DW property is equal to 2 (binary value), 3 (measurement) or 6 (counter)

When COV point value is requested for the first time, driver subscribes this point, i.e. asks controller to send point value every time a substantial value change is detected. Controller instantly sends back point value as the first update. Then it sends value only after the point is changed.

For binary points (DW=2) every change is substantial. For analogue points (DW=3 or DW=6) substantial change is the one, when it exceeds threshold value. Threshold values could be set individually for every point. If controllers are programmed, all points of interest should have thresholds set already and no further actions are required. If COV subscribed point does not update its value, then thresholds might not be set for some reason. There are two ways to do it.

The default way is to set properties in **Comm -> Cov** component. Measurement threshold applies to measured values (DW=3) and counter threshold applies to counter values (DW=6). If the value is set to *nan* (not a number), then threshold will not be set. For any other value threshold will be written into point MFA during the first subscription.

Comm		Sauter Comm
<input type="checkbox"/>	Status	{ok}
<input type="checkbox"/>	Router Status	Running
<input type="checkbox"/>	As Counter Received	1030
<input type="checkbox"/>	Pc Counter Received	30
<input type="checkbox"/>	Pc Counter Sent	30
<input type="checkbox"/>	Address	31745
<input type="checkbox"/>	Period	100 ms
<input type="checkbox"/>	Timeout	5000 ms
Cov		Cov Processor
<input type="checkbox"/>	Points	28
<input type="checkbox"/>	Measurement Threshold	0.10
<input type="checkbox"/>	Counter Threshold	nan

As this method sets one value for all measured points, it might be necessary to set other thresholds for some specific points. This could be accomplished by creating a new facet *threshold* of type *double* under **Device Facets** component. Then after controller is restarted this value will be written to threshold during the first subscription.

<input type="checkbox"/>	Device Facets	threshold=0.50	>>	↻
<input type="checkbox"/>	Conversion			
<input type="checkbox"/>	Tuning Policy Name			
<input type="checkbox"/>	Read Value			
<input type="checkbox"/>	Write Value			
<input type="checkbox"/>	Mfa			
<input type="checkbox"/>	Dw			
<input type="checkbox"/>	Format			
<input type="checkbox"/>	Out	912.7		

Config Facets

Key	Type	Value
threshold	Double	0.50

Once a day every controller performs *garbage collection* – deletes all subscriptions from every MFA – and then reports it to subscribers. When Niagara receives such message, it will re-subscribe all points, which are currently subscribed.

7 Schedules

Sauter controllers could store and execute complex time programs to implement various control strategies.

The driver provides full control over Sauter time programs from Niagara schedules. Schedules could

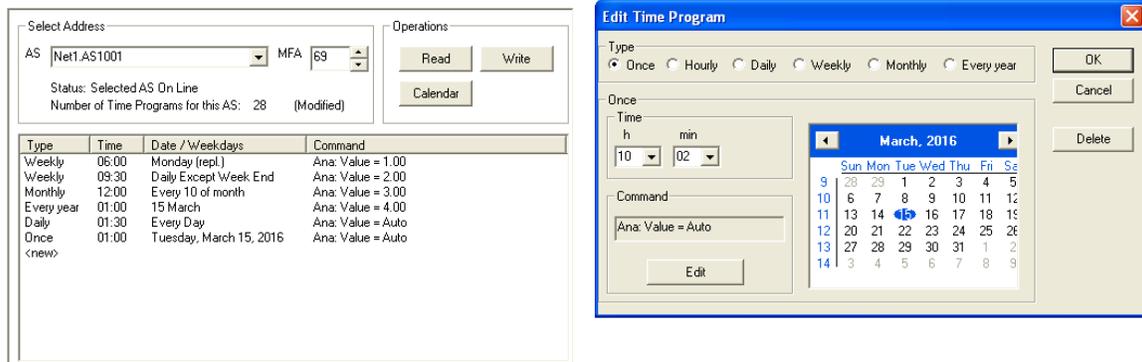
be imported, modified and exported back into controllers. Schedules could be edited with standard Niagara graphical controls from Workbench, PX or HTML graphics.

It is highly recommended to fully read this chapter and try to read and write schedules with test bench equipment before moving to working site.

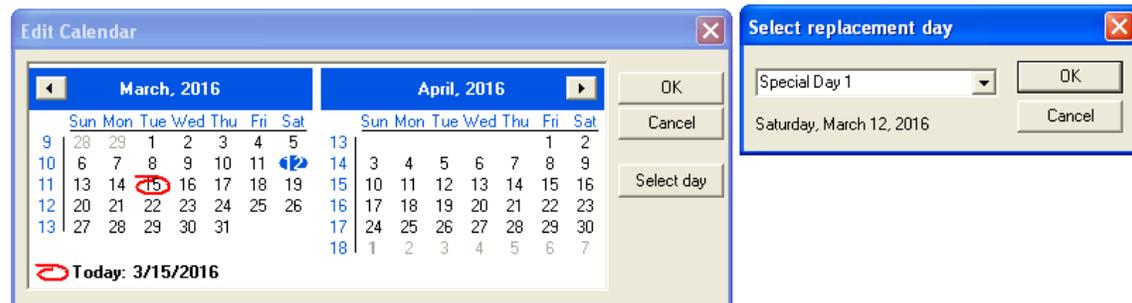
7.1 Sauter Time Programs

The time programs issue automatic, time-dependent commands with which it is possible to implement digital values, setpoints, analogue values (absolute or relative) and entries in the HDB on an autonomous basis in each AS. ECOS controllers could have up to 64 commands, others models – up to 320 time commands.

The year, month, day, hour, minute and MFA are linked to the time command; and month and day codes can also be linked to a weekly program. Each command issue a specific value on specific time as demonstrated below with Sauter software.



In addition to individual time programs, there are two year calendars for exception dates: one for even and one for odd years. In them each date could be overridden with specified week day (Mon-Sun) or special day 1-8.



7.2 Niagara Schedules

Every Sauter device in Niagara has Schedules extension, which contains Schedule Exports – components, which connect generic Niagara schedules with Sauter time programs.

In **Sauter Schedule Export Manager** press **Discover** button to initiate time program discovery. The driver automatically groups found time programs in MFA groups. Each MFA group contains all time programs affecting it. Both exception year calendars are displayed as well.

If no time programs are discovered, then controller might not use them and time control is done on SCADA side by writing into some Sauter points.

The screenshot displays two main sections: 'Discovered' and 'Database'.

Discovered (11 objects):

Name	Timestamp	Value
MFA66 Analog		
event0	Mon Tue Wed Thu Fri 06:00	21.0
event1	Sat 09:00	20.0
event2	Sat 15:00	AUTO
event3	Mon Tue Wed Thu Fri 18:00	AUTO
event4	Special day 1 06:00	19.0
event5	Special day 1 21:00	AUTO
MFA67 Analog Relative		
MFA68 Command		
Even Year Calendar		
Odd Year Calendar		

Database (3 objects):

Name	Mfa	Schedule Type	Supervisor Ord	State	Execution Time	Last Success
ScheduleExportMFA66	66	Analog	station: slot:/Drivers/SauterNetwork/Device1/NumericSchedule	Idle	Manual	15-Mar-16 15:40 GMT
ScheduleExportMFA67	67	Analog Relative	station: slot:/Drivers/SauterNetwork/Device1/EnumSchedule	Idle	Manual	15-Mar-16 15:40 GMT
ScheduleExportMFA68	68	Command	station: slot:/Drivers/SauterNetwork/Device1/BooleanSchedule	Idle	Manual	15-Mar-16 15:40 GMT

At the bottom, there are buttons for 'New', 'Edit', 'Discover', 'Cancel', 'Add', 'Match', and 'Export'.

To configure schedule export, create a regular Niagara schedule: BooleanSchedule for Command type, Numeric or Enum for Analog type. Then select discovered MFA groups and **Add** them to station. Schedule Export component will be created for every MFA group.

The 'Edit' dialog box shows the configuration for a schedule export component. It includes a table at the top and several input fields below.

Name	Mfa	Schedule Type	Supervisor Ord	Execution Time
ScheduleExportMFA66	66	Analog	station: slot:/Drivers/SauterNetwork/Device1/NumericSchedule	Manual

Below the table, the following fields are visible:

- Name:** ScheduleExportMFA66
- Mfa:** 66
- Schedule Type:** Analog
- Supervisor Ord:** station:|slot:/Drivers/SauterNetwork/Device1/NumericSchedule
- Execution Time:** Manual

Buttons for 'OK' and 'Cancel' are at the bottom.

Select the exported schedule in **Supervisor Ord**. Do not set execution time to periodical export for now to avoid writing an empty schedule.

After all schedule exports are set, it is possible to read existing time programs from controller into schedules. Use **Read Schedule Exports** in **Schedule Device Ext**. The driver will once again read all time programs, parse them into Niagara format and sets the referenced schedule.

	Sun	Mon	Tue	Wed	Thu	Fri	Sat
03:00	Null	Null	Null	Null	Null	Null	Null
06:00		21.00	21.00	21.00	21.00	21.00	
09:00							20.00
12:00							
15:00							
18:00							
21:00							

Event Start
 Event Finish
 Event Output null

Because Sauter time program and Niagara schedule formats are not fully compatible, the driver ignores all non-weekly time programs, “every hour” and “every X minutes” time programs. Exception dates should be created in separate calendars, see below.

Normally schedule reading is only required during initial set-up. After that schedules are modified in Niagara and exported (written) into Sauter controllers. As Sauter time programs are stored in one continuous list, Niagara schedules could not be read or written individually, only all device schedules at once.

When schedules are exported to Sauter controller, all existing time programs and calendars for all MFA in Sauter RAM will be rewritten. Make sure schedule exports are configured for all MFA points before initiating writing.

Schedule could be exported in multiple ways:

Type	Actions
Manually	Use Export button in Sauter Schedule Export Manager Or trigger Write Schedule Exports action in Schedules extension
Periodically	Set Execution Time in Schedule Export for periodical export
By demand	Link master Schedule property Last Modified to Schedule Export object action Execute . This link initiates exporting process as soon as the master schedule is modified and saved in Scheduler

As all device schedules could only be written together, it is enough to export one of schedules – the rest will be exported as well.

Writing schedules is a “heavy” task from traffic perspective, so it is better not to do it exceedingly.

After the schedule is exported into controller, time programs could be discovered again to check how Niagara schedule is mapped into Sauter time programs.

7.3 Niagara Calendars

To set schedule exception dates (e.g. holidays) add a regular **Calendar Schedule** to the station from **schedule** palette and then reference it from schedule **Special Events** tab.

Set daily time schedule for all special dates listed in the calendar.

The screenshot displays the Niagara AX software interface for managing calendars. At the top, there are navigation buttons: 'Prev Page', 'Prev Month', 'Today', 'Next Month', and 'Next Page'. Below these are three calendar grids for March 2016, April 2016, and May 2016. The March 2016 calendar shows the 15th highlighted in blue. Below the calendars is a table with columns 'Name' and 'Summary'. The table contains one entry: 'Event' with the summary 'Reference: slot:/Folder/CalendarSchedule'. To the right of the table is a time slot grid with columns for times from 03:00 to 21:00. The 06:00 slot is highlighted in green and contains the value '19.00'. Below the time slot grid are input fields for 'Event Start' (15:40), 'Event Finish' (15:40), and 'Event Output' (null, 19.00). At the bottom of the interface are buttons for 'Add', 'Edit', 'Priority', 'Rename', and 'Delete'. Below these are tabs for 'Weekly Schedule', 'Special Events', 'Properties', and 'Summary'. At the very bottom are 'Save' and 'Refresh' buttons.

Now after the schedule is exported into controller, a special day time commands will be created in individual schedules and this special day will be referenced in both yearly calendars.

One calendar could be referenced from various schedules in one or multiple devices. This way it will be easy to modify site-wide date exception and keep individual time programs for every point. Multiple calendars could be used for every point, but there should be up to 8 different calendars per device (limited by 8 special days in Sauter exception calendars).

Exception dates could not be read from controllers into calendars, because normally one calendar will be shared with multiple points. Sauter ability to override exception dates with a particular weekday is not implemented, special days are used instead. Niagara calendars should only contain Date type events. Schedule Special Events tab should only contain Calendar references.

8 Troubleshooting

Issue	To do
EYZ292 router is not answering to the ping message	Check if Ethernet cable is connected and both PC and EYZ292 router have Ethernet connection LED active Check if your PC is in the same IP range as router Check if only one Sauter client is connected to the router
Device discovery does not find any devices	Check that novaNet network is connected to the router and novaNet LED is active
EYZ292 router could be pinged from PC but not by the driver	Try to restart the router and the station

EYZ292 router LED lamp diagnostic

Name	Colour status	Meaning
Power	Green on	Power supply present
Activity	Green on	Receiving and sending Ethernet telegram
Link	Yellow on	Ethernet connection established
novaNet	Yellow on	novaNet telegram transmitted
Error	Red off	Normal state
	Red on	Startup 30 seconds
	Red flash 1 sec	novaVPort communication missing
	Red flash 1/4 sec	No novaNet telegrams received

9 Use Niagara and Sauter SCADA

When refurbishing an existing BMS with Sauter SCADA (e.g. novaPro32), it is important to make the transition comfortable for users, so they would not lose control over their system in any time period. Niagara could successfully coexist with other SCADA, if few configuration rules are applied properly.

Niagara should have a separate Sauter router: neither EYZ291, nor EYZ292 could accept two clients connected simultaneously.

Point polling interval should be selected considering the total traffic to not overload the system.