Amtiri

REFERENCE MANUAL

Module catalogue

Generated on November 25, 2026

Modules connect an instance's variables to the world: they read devices and peers into variables, write variables out to them, keep history and send notices. This catalogue lists every module type, grouped by family, with what each one reads and what it writes, and links to its section in this manual.

The **code** is what a module's record stores as its `moduleType`. The **name** is how the editor lists the type, in the **Module type** list that opens when you add a module and in the diagram's list of types, in English in both of the editor's languages. What every type has in common, from **Enable** to the status variables, is in Settings every module shares.

Choosing a module

To	Use
Exchange live values with another Data Orchester instance	Orchester Push or Orchester Pull
Keep the history of variables	Logger
Copy that history to another instance	Logger Replicator
Read and write a device that speaks Modbus	Modbus TCP Master or Modbus Serial Master
Read and write a PLC or gateway with an OPC UA server	OPC UA Client
Read and write a Siemens PLC without OPC UA	Siemens S7 Client
Read and write a Rockwell Logix controller's tags	EtherNet/IP Client
Interrogate and command a telecontrol station	IEC 60870-5-104 Client
Read a GNSS receiver, a boat's instruments or AIS over NMEA 0183	NMEA 0183 Listener
Read and write an NMEA 2000 backbone	NMEA 2000 Bus
Read and write a vehicle's or a machine's CAN bus	CAN Signals with the machine's DBC file, or CAN Raw Frames without one
Let a SCADA, HMI or PLC read and write this instance's variables	Modbus TCP Slave, Modbus Serial Slave, OPC UA Server or IEC 60870-5-104 Server
Feed a chart plotter or navigation software with this instance's values	NMEA 0183 Talker
Send an e-mail when something happens	Mail Sender
Call a web address when something happens	WebService Out
Recalculate formulas on a clock	Scheduler
Take the latest of several writers into one variable	Merger
Act on a value a fixed time after it changed	Delay
Keep the recent changes of a variable, for a dashboard	FIFO
Hold a value at its setpoint	PID
Drive a relay or a pump on and off from a demand	PWM
Smooth a noisy value	Kalman filter

Orchester links

Between two Data Orchester instances, over the signed peer link. Page: Orchester links.

Code	Name	Reads	Writes
`DOPUSH`	Orchester Push	Local variables, when they change and again every few minutes	The same variables on a peer, optionally under a prefix
`DOPULL`	Orchester Pull	A peer's variables, on a schedule	Local variables, optionally under a prefix
`DBREPLICATOR`	Logger Replicator	The rows of a local LOGGER, on a schedule	The rows the far side lacks, into a LOGGER on a peer

Modbus

Modbus TCP and RTU, as master and as slave. Page: Modbus.

Code	Name	Reads	Writes
`MB-TCP-Master`	Modbus TCP Master	Coils, discrete inputs, holding and input registers of a device over TCP, on schedules, into variables	A coil (FC05) or a holding register (FC06) when a forwarded variable changes
`MB-SER-Master`	Modbus Serial Master	The same, from the devices on one serial line	The same, to the devices on the line
`MB-TCP-Slave`	Modbus TCP Slave	Four list variables, to answer a remote master over TCP	What the master writes to coils and holding registers, into those variables
`MB-SER-Slave`	Modbus Serial Slave	The same, to answer a master on a serial line	The same

OPC UA

Client and server. Page: OPC UA.

Code	Name	Reads	Writes
`OPCUA-Client`	OPC UA Client	Nodes on a server, by subscription or on schedules, into variables	A node when a forwarded variable changes
`OPCUA-Server`	OPC UA Server	The variables it publishes, as nodes for clients to read	The writes a client is permitted, into those variables

Siemens S7

Page: Siemens S7.

Code	Name	Reads	Writes
`S7-Client`	Siemens S7 Client	PLC memory over S7comm, on schedules, into variables	A PLC address when a forwarded variable changes

EtherNet/IP

Page: EtherNet/IP.

Code	Name	Reads	Writes
`EIP-Client`	EtherNet/IP Client	Logix controller tags by name, on schedules, into variables	A tag when a forwarded variable changes

IEC 60870-5-104

Both telecontrol roles. Page: IEC 60870-5-104.

Code	Name	Reads	Writes
`IEC104-Client`	IEC 60870-5-104 Client	A station's reports, after interrogating it, into variables	A command when a forwarded variable changes
`IEC104-Server`	IEC 60870-5-104 Server	Variables, as points for the masters it allows	The commands those masters send to writable points, into variables

NMEA

Both NMEA protocols. Page: NMEA.

Code	Name	Reads	Writes
`NMEA-0183-Listener`	NMEA 0183 Listener	A talker's sentences and AIS messages, as they arrive, into variables	Nothing
`NMEA-0183-Talker`	NMEA 0183 Talker	Variables, built into sentences served on TCP, UDP or a serial port	What a client sends back, into variables
`NMEA-2000`	NMEA 2000 Bus	A backbone's parameter groups, as they arrive, into variables	A parameter group, sent as a node, on change or on a clock

CAN

A bus read by frame or by name. Page: CAN.

Code	Name	Reads	Writes
`CAN-Raw`	CAN Raw Frames	Slices of frames, by identifier and bits, as they arrive, into variables	A frame, on change or on a clock
`CAN-Signals`	CAN Signals	Signals of a DBC file, in engineering units, as they arrive, into variables	A DBC message, on change or on a clock

History

Page: Logger.

Code	Name	Reads	Writes
`LOGGER`	Logger	Its variables, at each change	One row per change into `loggers/<code>.sqlite`

Notices

Pages: Mailer and WebService Out.

Code	Name	Reads	Writes
`MAILER`	Mail Sender	A change of each trigger variable, and the variables its messages name	An e-mail through the mail server it is given
`WEB_SERVICE_OUT`	WebService Out	A change of its trigger variable, and the variables its expressions read	An HTTP GET or POST to the address it builds

Clock

Page: Scheduler.

Code	Name	Reads	Writes
`SCHEDULER`	Scheduler	Nothing: it runs on its schedules	No value: it makes formula variables recalculate

Logic

Page: Logic modules.

Code	Name	Reads	Writes
`MERGER`	Merger	The changes of its inputs, and what its filter reads	The latest accepted change into its output
`DELAY`	Delay	The changes of its source	Each change into its destination, a fixed time later
`FIFO`	FIFO	The changes of its source	The list of its source's recent changes into its destination

Control

Control loops that run on their own clock. Page: Logic modules.

Code	Name	Reads	Writes
`PID`	PID	What its input, setpoint, gains and limits read, every period	Its output, and the term outputs that are set
`PWM`	PWM	What its demand reads, at the start of every cycle	On and off edges into its output
`KALMAN`	Kalman filter	What its input and noises read, every period	Its estimate into its output

Every type can also write its status and messages into four variables of its own: see Status and messages.

Next steps

- Settings every module shares
- Module messages
- Connectivity

This page describes Data Orchester engine 6.20.0.