Created by Werner Krings, Tani GmbH
Date: 20.01.2017

Technical Data PLC Engine

At a Glance

The **PLC Engine** allows data processing in production lines. It connects controllers, devices, SCADA systems and databases from various manufacturers. It processes the data and dispatches them between the connected devices and other systems. It collects data, manages them and creates overviews for OEE applications.
Additional all the data can be handled over OPC. This are processed data and all data from the connected controllers, devices, OPC systems and databases.

PLC Engine is the perfect component for using it for communications in Industrie 4.0.

You will implement important use cases with **PLC Engine**

Boosts the PLC communication with logical optimizing of the requests. Often this will happen to error texts from the controllers. The messages need to be fetched normally if the corresponding error number is changing and a new error condition becomes true. Without any change to Your SCADA system the communication speeds up. The main reason is less load on the communication line.

Exchanges data between controllers and devices without PLC programming or SCADA functionality.
That's how it's done: PLC Engine is reading the data from a controller or device and writes them into another controller or device. The writing normally takes place if the data are changing. With configurable triggers data which are depending on other data conditions for the writing can be defined. The data will converted if necessary. This is important if both controllers are different: One Siemens S7, the other a Rockwell Control Logix.
More complex handling is possible, too: Collecting of data from one or multiple controllers, do some calculation with the data, and write the result to a controller or device. Of course, any OPC compatible system can use the same data also.

Collects and processes data
PLC Engine is reading data from one or more controllers.
The data are collected and possibly calculated or normalized.
If all data are in place they are ready to process for other systems or OPC access.
During the logical processing the data are blocked for access.
Syncronous data access are waiting until the collection is finished.
Configurable error conditions are configurable: If one of the controllers is not available constant data are used, or an error number is processed.

Exchanges data with common databases. You can read data from controllers and write them directly into common databases - with optional calculations. Data readed from databases can be written into controllers, too. All the data are available for OPC systems also.

Collects data over long time for OEE into a local or external database. So easily OEE applications can be build. The integrated websites are offering the collected and calculated data directly as curves or other. Additionally the data can be handled over OPC.

Functionality

With logic tables you define all the things your plant will need:

- Read and write data from industrial controllers and devices.
- Exchange data with databases. Read, write, update, delete.
- Read and write files, check for changes, create files, delete files.
- Read and write data over OPC UA and Classic OP.
- Sequence chains with conditions for changing the actual step.
- Calculate data. Round, basic calculations, constants, conversions, reinterpretations.
- Check data. AND OR XOR NOT. Comparisons for equal, less, greater. Float number plausability.
- Manage data structures. Create them, decode structures.
- Collect data. Combine and separate texts. Combine and separate binary data.
- Lots of triggers: Time trigger, data change trigger, bit trigger, file create trigger, file modify trigger, directory change trigger.
- Subroutines.

Online diagnostics for shortening the configuration time

- Watch connections and their state.
- Watch variables. Status, value, who needs them.
- Watch logic tables and their behaviour.
- Watch and modify single variables (Status variable).
- Diagnostics logger for long term watching.

Configuration

The configuration works in graphical style or with simple lists.

Lot of Wizards simplify this.

The example logs from barcode reader, adds a timestamp and writes both into a .csv file.

This example reads a barcode, gets the print data from a database and priths the result.

Diagnostics boost the startup in plants. Use Your smartphone or tablet and all of the status displays.

The diagnostics logger handles longer running information collecting.

OPC Interfaces

- **OPC Pipe** Open interface
- **OPC UA** (Unified Architecture)
- **OPC DA** (Classic OPC over DCOM, available under Windows only)

The maximum number of OPC clients is depending on used resources only. A PC from 2014 can handle multiple hundred connections.
All OPC interfaces are working locally in one PC or over network.
in case of Classic OPC Classic please do not use DCOM over networks, but it will be supported.
OPC UA supports the fast binary protocol. Security is supported in all variants. Multicast discovery is supported.
Data access data items are supported up to 200K each.

OPC UA functionality and limitations

The OPC UA implementation conforms to the specification 1.04.
An internal discovery server is active on standard, it supports multicast discovery also. It can be used as a global discovery server. Alternatively an external discovery server can be configured.

AddNodes is supported with the following restrictions:

- AddNodes is possible only in Objects.Topics.Memory tree
- Reference type must be OpcUaId_Organizes
- NodeId can't be specified
- BrowseName can't contain .
- NodeClass must be Variable or Object
- NodeAttributes for Variable:
  - DisplayName: unspecified or equal to BrowseName
  - Description: unspecified or any text
  - Value: is ignored; new variables will always be initialized to 0 (if numeric) or "" (if string type)
  - DataType:
    - OpcUaType_Boolean
    - OpcUaType_SByte, OpcUaType_Byte

        - OpcUaType_Int16/32/64, OpcUaType_UInt16/32/64
        - OpcUaType_Float, OpcUaType_Double
        - OpcUaType_String
        - OpcUaType_DateTime
        - One of the structure types under Types -> DataTypes -> BaseDataType -> Structure -> UserStructures; these are the structures known to the PLC Engine core.
            - if the structure is given both here and via TypeDefinition, both settings must match
        - if unspecified, OpcUaType_Byte or the structure type of the TypeDefinition is used
    - ValueRank, ArrayDimensions: unspecified (= scalar), scalar or a one-dimensional array of any size
    - AccessLevel, UserAccessLevel: unspecified or (OpcUa_AccessLevels_CurrentRead | OpcUa_AccessLevels_CurrentWrite)
    - MinimumSamplingInterval: unspecified or 0
    - Historizing: unspecified or 0
    - WriteMask, UserWriteMask: unspecified or OpcUa_NodeAttributesMask_Value
- NodeAttributes for Object:
    - DisplayName: unspecified or equal to BrowseName
    - Description: unspecified or any text
    - EventNotifier, WriteMask, UserWriteMask: unspecified or 0
- TypeDefinition for Variable:
    - OpcUaId_BaseDataVariableType
    - one of the structure types under Types -> VariableTypes -> BaseVariableType -> BaseDataVariableType -> UserStructures; these are the structures known to the PLC Engine core.
- TypeDefinition for Object:
    - OpcUaId_FolderType

Traffic between different OPC interfaces (tunneling) is supported. It will be used for the OPC DA tunnels.

MQTT Interfaces

- **MQTT Client** if a station need to be a device
- **MQTT Broker**, the server

MQTT comes from the Internet of Things world. It is simple and fast.
A device can simultanously send data to multiple devices.
You can use the client and broker on the same device at the same time.

Controller Interfaces

All controllers will be connected over network. Often this is Ethernet, WLAN or other networks. All serial Ethernet and MPI Ethernet gateways for industrial controllers usage are supported.

Configuration Interfaces

The configuration can be done with the shipped configuration software or over OPC with the System topic.
The connection for the configuration is encrypted with TLS 1.2. The encryption can be switched off for usage in countries where encryption is forbidden.

Network Redundancy for connections to controllers and devices

Connections to devices and controllers are supporting network redundancy.
Double and triple redundancy can be selected.
Two redundancy operation modi are possible.
In **dynamic redundancy** any of the connections is working as master. If it breaks another connection becomes the master connection.
In **static redundancy** the first connection is the master. If it breaks another connection becomes the master. If the first connection works again it will become the master connection again.

The connections of the redundancy should work on different netaork adapters.The adapters need different IP subnets for proberly work..

Controller Types and Controller Protocols

- Siemens **S7 1200 and 1500** family. The **optimized data blocks** are supported, all symbols and comments are browsed online.
- Siemens S7 over RFC1006 and Sinec H1. Supported are **S7 200, 300 and 400, Logo 8**. Siemens CP or the Ethernet interface onboard the CPU can be used. Mainstream MPI Gateways as Hilscher Netlink, Helmholz Netlink, IBH Softec Netlink, INAT Echolink, Process Informatik S7Lan or Softing Netlink are supported, too. S7 compatible systems as VIPA Speed7 can be used, too.
- Siemens S5 over RFC1006, PLC Header, RAW or Sinec H1. Supported are Siemens CPs, INAT CPs, Helmholz CPs, IBH Softec S5Net, Process Informatik S5Lan.
- Rockwell Compact Logix and Control Logix, all firmware versions.
- Mitsubishi Melsec Q famils.
- BACnet devices. BACnet is common used in building automation
- Devices using the Modbus TCP protocol.
    - Modicon
    - Schneider
    - Wago
    - Beckhoff
    - Phoenix Contact
- All systems and devices which can be accessed with OPC UA or Classic OPC.
- Raw data.

Communicates via Ethernet.

Bacnet

Bacnet will be used over IP / UDP.

Databases

PLC Engine is a database client. It connects with user and password to the database.
The standard SQL statements INSERT INTO, UPDATE, SELECT, DELETE, FUNCTION and PROCEDURE will be used over the wizards. Other statements will be configured directly.
Supported are:

- My SQL (from version 1.9 not under Windows XP)
- PostGre Sql (not for Windows XP)
- Microsoft SQL
- Sybase SQL Server, Sybase ASE, SAP ASE (Adaptive Server Enterprise)
- ODBC
- Oracle can be used with ODBC

Multiple databases can be handled simultanously.
In one database multiple sub databases can be used.
During the configuration the databases will be browsed. This will require depending on the type of the database or the interface a username and a password, possibly more.
The database itself need to be configured that it can be accessed.
PLC-Engine will need an account on each database.

For using PLC Engine on a PC the locally installed databases can be used also.
On PLC Engine Device a My SQL database is installed. This database will be managed completely by PLC Engine.

Logger

The OPC Server contains a logger for diagnostics purposes during plant startup. The logger can be configured. The system load can be big if all controller ; data are logged.

Limits

Maximum number of configurable connections: 1000.
Maximum length of a single item: 4GB.
Maximum number of elements each connection: 4 Million.
Maximum number of elements (Items): 16 Million.
Maximum OPC groups each connection: 100.

Maximum number of passive connection for each port is 999.
The OPC synchronous functions returning a bad quality immediately if the PLC connection is not established.
Changes in controller configuration will be checked all 10 seconds.
Fields can be up to 64K in length each.
Multi dimensional arrays can have up to seven dimensions.

Fields can be up to 64K in length each.The maximum amount of configurable logic tables is 60000.
The maximum length of one logic table is 4GB.
Maximum number of entries on a logic element 60000.
File operations can handle up to 64K in each request. Adding data to a file is limited only by the space on the disk.
Multi dimensional arrays can have up to seven dimensions.

Depending on the license the limits can be less.

Speed

The throughput will be mainly limited by the controller speed or the reaction time of OPC applications.
Read requests to the controller will be optimized as much the controller is supporting this. For that elements will be collected to blocks reading more than requested, but not for inputs and outputs. These optimizing can be affected by configuration separately for each connection. Optimizing can be switched off, too.
Write requests to the controller are collected or handled in that order the application did called the system.
On OPC all optimizing the individual OPC uses is supported.
The normal time in cyclic controller requests is 50ms.It can be faster if the controller polling interval is set to zero.
Only data are sent to OPC which did change in the controller between two read requests.
The throughput of one logic table normally will be handled below 10ms. Lot amounts of logic tables containing much of mathematics functions may rise the 10ms. This is true especially on embedded devices without a mathematical co processor.
No handling can be faster than the controller and device data acquisition. The same is true for database access.

Functions

Logic tables

Logic tables are constructed for linear logic. Loops are not possible.

Sequence chains

A sequence chain supports the maximum of 65535 steps each.

Error Handling in Logic Tables

All logic elements which functions can fail are supporting an OK bit. The user need handle this preventing unexpected run-time behavior.

Variables and Structures

Structures can not contain loops.
A structure or variable can be up to 4GB in its size.

Status Diagnostics Lists

The status diagnostics lists are supporting arrays up to 100 values each. If an array is longer than 100 values the first 100 elements are shown only. Writing this array is not possible.

Field and text optimizings

The from version 1.8 existing field optimizings will prevent reading the long fields too often, the index is requested on standard only.This optimizing bases of the fact that the index does net changed too frequently.

Usage of memory

- Program code: A minimum of 6MB is used. The exactly memory usage is depending of the internal behavior of the operating systems. So dynamic libraries are loaded once for all running instances using them. Example: If the standard library is not loaded already it will use additional 4MB of memory.

- User data: The minimum data usage is 2MB internally. Additional the controller data are held in memory for comparing new data. Each item uses the length of data and additional 64 bytes. Each configured connection occupies 4KB.

- Fields can be up to 64K in length each.The maximum amount of configurable logic tables is 60000.
- The maximum length of one logic table is 4GB.
- Maximum number of entries on a logic element 60000.
- Depending on the license the limits can be less.

Usage of computation time

The consumed computation time is depending on the load with communication. Most the time it will be waited for controller data or OPC application reaction.
All software is working with events. This maximizes the throughput and minimizes the usage of computation time.
Multiple CPU are supported. Up to ten CPU will be used, the main work will be handled by three CPU.

Installation

The installation does depending on the product install multiple parts separately. On uninstall not all products are deleted automatically. But all installed products can be deleted over the menu or the software part in the system control manager. The user settings will be preserved and not deleted during uninstall.

Operating systems

- Windows 7, 8, 8.1, 10 - anniversary update and creators update also. Older versions as XP and Vista also. 64 and 32 bit.
- Windows Server 2008, 2012 and 2016.
- Linux on the Raspberry and Odroid computers (64 and 32 bit).
- Linux on many Phytec devices.
- Linux on the Wiesemann & Theis pure.box 3, pure.box 5.
- Linux on a PC with Debian, Ubuntu, Suse, Arch, Centos, Redhat and other Distributions.
- Linux 64 Bit as Docker Container.

- OPC DA will require Microsoft Windows. All from Microsoft supported operating systems for Intel and all user languages will be supported. The latest service pack must be present.
- Under Windows the OPC server are working as service, Linux runs them as daemon.
- The Raspberry version supports all Linux distributions offered for this platform.
- All other will run under lot of operation systems also, mostly Linux based.
- Under Linux the OPC Server needs a POSIX compatible System. The Standard Library needs V2.2 as minimum. The configuration software is bases on KDE 5 and is needing the kdelibs. Please use actual distributions like Debian, Ubuntu, Suse, Redhat or similar.
- Tested is: Windows Intel 32 and 64 bit, Linux Intel 32 and 64 Bit, Linux MIPS CPU, Linux ARM 32 and 64 Bit CPU.
- Running in virtual machines is supported. Docker containers are supported, too.
- Windows 7 needs as minimum service pack 1 for using the drivers.
- All configurations are compatible to all OPC servers, also over operating systems.