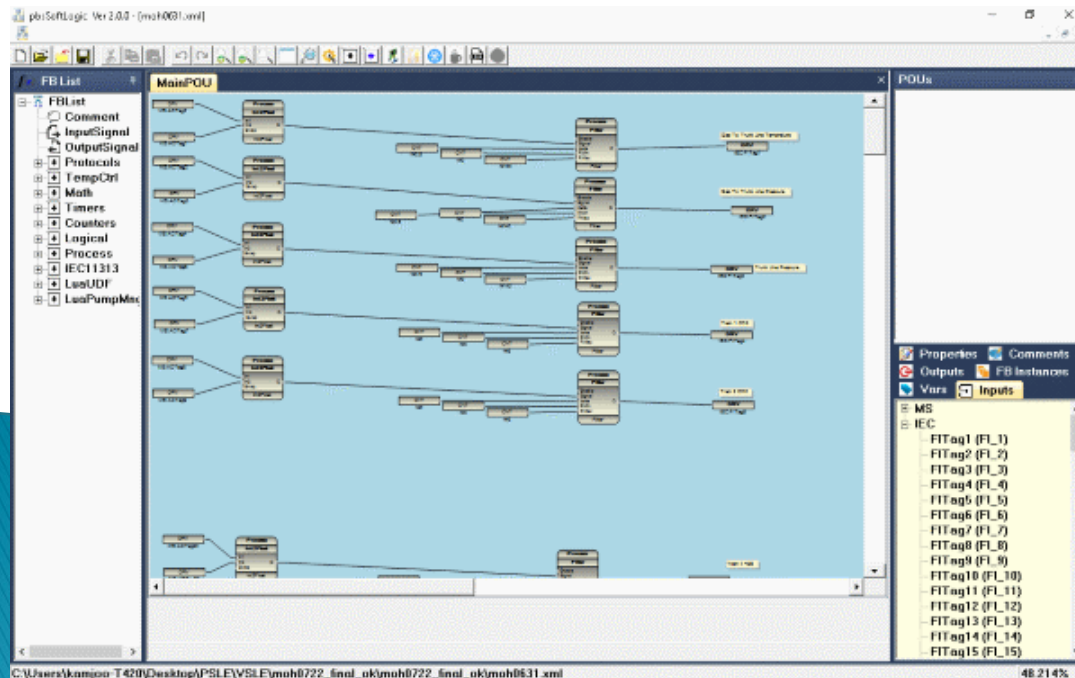


# pbsSoftLogic Specifications Version 2.0.0RC19



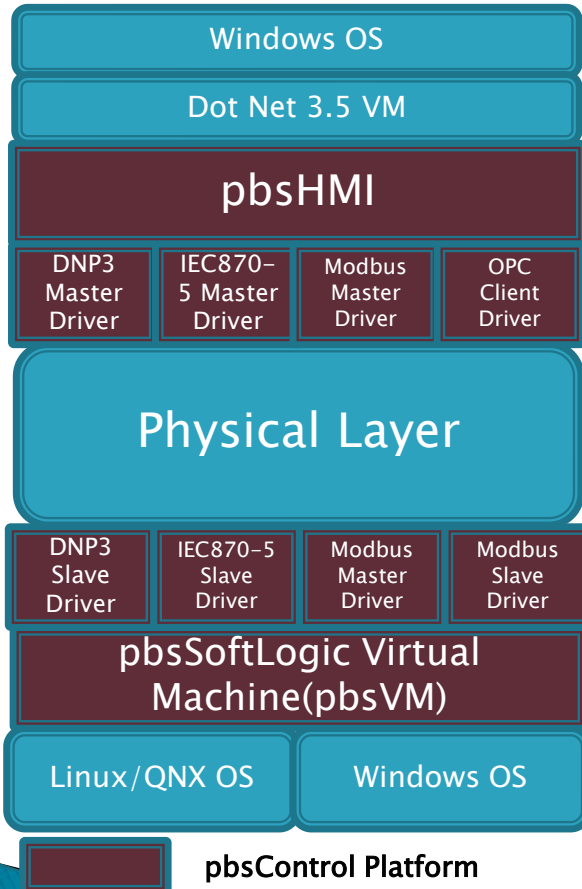
# pbsControl Systems

**P**rocess  
**B**uilding Automation  
**S**CADA

pbsControl is control solution from pbsControl Company for **P**rocess Control , **B**uilding Automation and **S**CADA projects. pbsControl has three major elements :

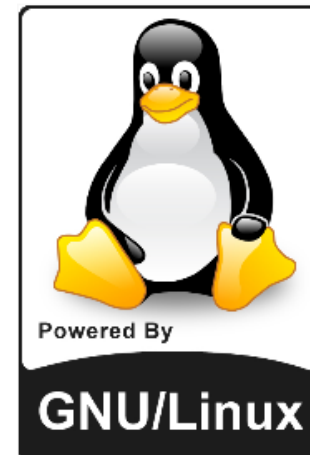
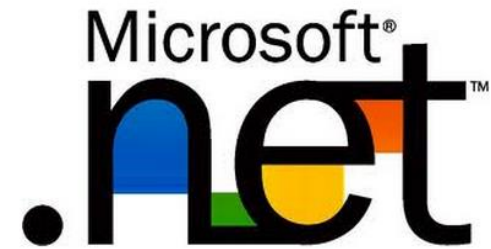
- 1 - pbsSoftLogic - PLC/RTU programming Environment
- 2 - pbsHMI - HMI /SCADA Platform
- 3 - pbsCOMM : Modbus , DNP3, IEC870-5-101/104 , GSP and OPC UA Protocols

[www.pbscontrol.com](http://www.pbscontrol.com)

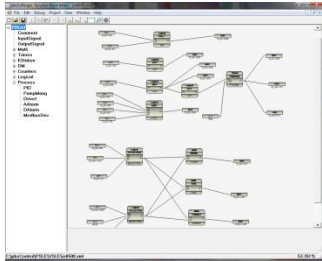


# Basic concepts

- ▶ pbsSoftLogic is RTU/PLC programming Environment from pbsControl Company based on IEC1131-3 Standard
- ▶ pbsSoftlogic target is running on Embedded Linux / QNX and Win32/WinCE Controllers
- ▶ pbsSoftLogic is developed based on Microsoft Dot Net and GNU/Linux technologies
- ▶ pbsSoftLogic supported Function block and Lua Language for developing control logic



# Logic Development



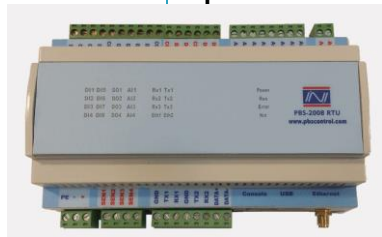
## pbsSoftLogic Engineering

- Develop control logic by Function Block language ,
- Develop User defined FB by Lua
- Simulate on Windows
- Transfer Configuration and logic to controllers
- Monitor Logic at runtime and update logic



TCP/IP

pbs2008RTU



MAPCSR



AMS-4000GW



pbs2008RIO

Modbus Remote I/O



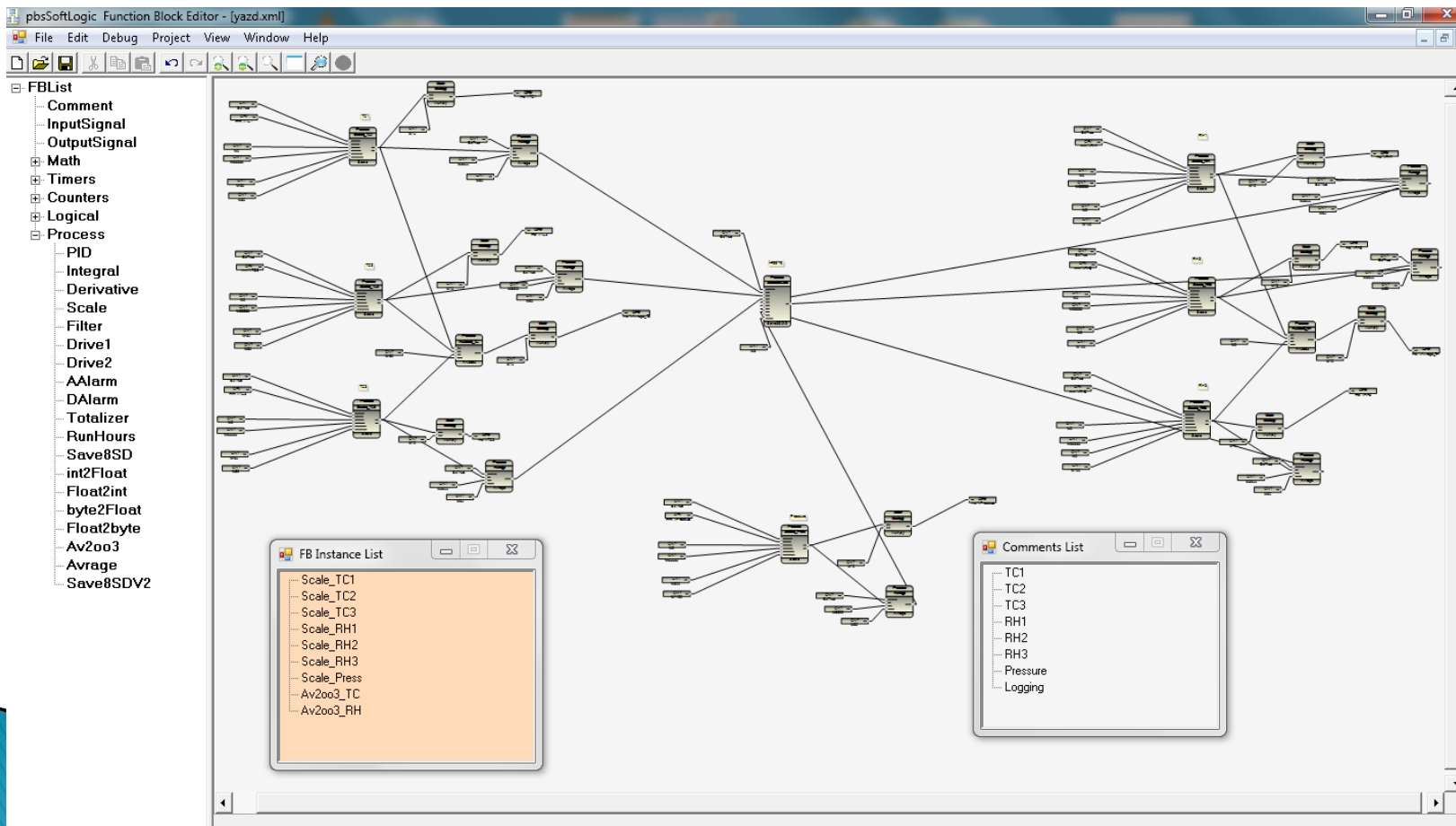
Power meter , Flow Computer

# Specifications

- ▶ Standard Function Block Language
- ▶ No Limitation in Number of Blocks and logic size
- ▶ No limitation on number of FB trees
- ▶ Lua Standard Scripting language for developing User define FB . No need for cross compiling
- ▶ More than 200 Ready and tested FB
- ▶ User Defined Communication protocol development
- ▶ Runtime kernel is ported for Embedded Linux , QNX and Win32.
- ▶ Runtime kernel is developed by ANSI C and easily can be ported to other real time OS
- ▶ Modbus , DNP3 , IEC870-5-101/103/104 , SOAP ,GSP (GPRS for SCADA Project), Siemens S7 Connect ,SQLite , MS SQL Client and OPC UA Server protocols are supported .
- ▶ Logic monitoring facility
- ▶ Tag Forcing Facility
- ▶ Offline Logic Simulation
- ▶ Cold & Warm Logic Update

# Development Environment

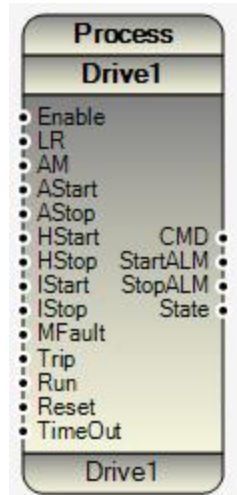
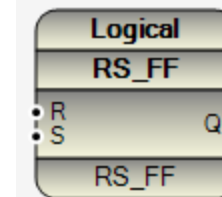
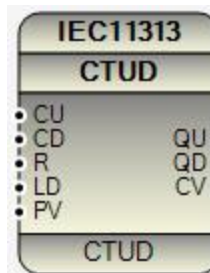
pbsSoftLogic supports standard Function block and Lua Scripting languages for development of control Logic



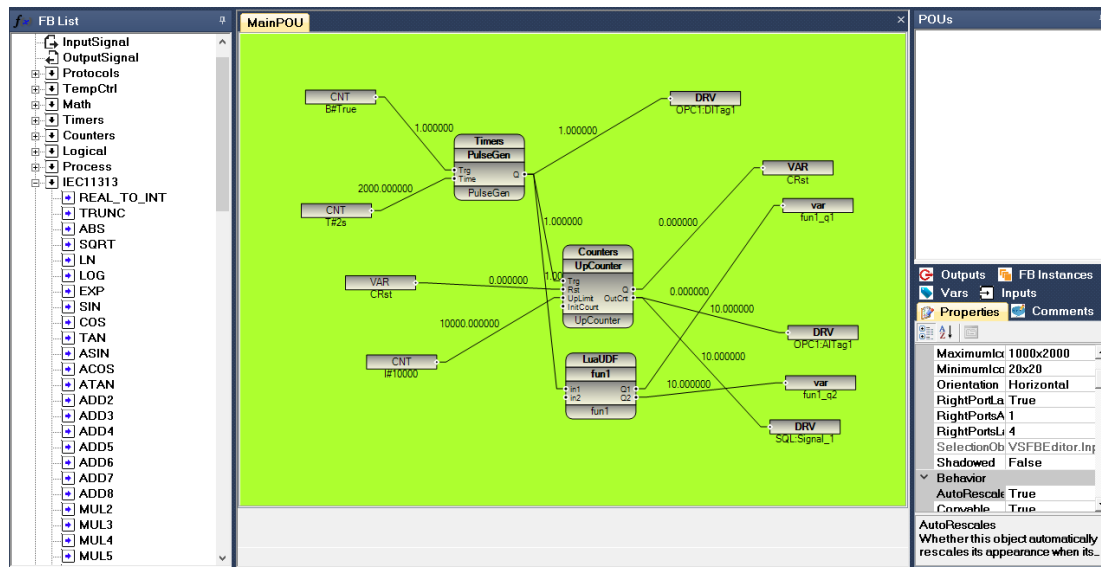


# Ready Function Blocks

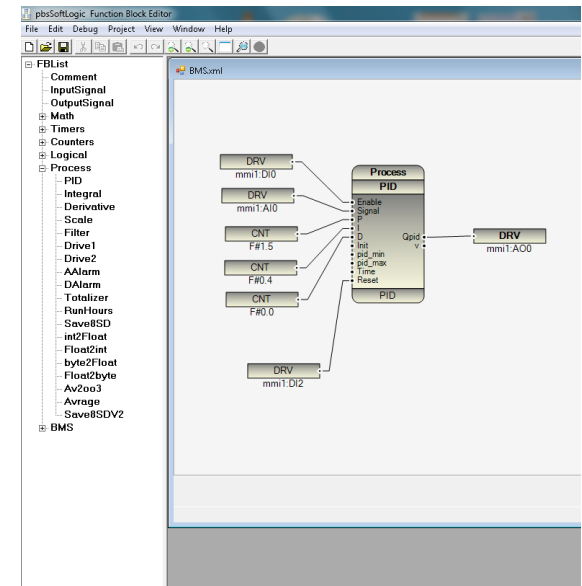
- ▶ pbsSoftLogic Supports more than 200 ready Function Blocks for easy and free bug programming
- ▶ **Math Group** : Mathematical functions – Add , Subtract , Multiply , Divide , Equal , LessThan , MoreThan , Sin, Cos ,tag , Log , Abs , ...
- ▶ **Timers Group** : OnTimer , OffTimer ,PulseGen,Rampgen,SinGen,...
- ▶ **Counter Group**: UpCounter , DownCounter ,...
- ▶ **Logical Group** : Latch , RSFF , JKFF,DFF, TFF , Pack16 , UnPack16 , AND , OR ,XOR ,MAP,Selector ,...
- ▶ **Process Group** : PID , Integral , Derivative , Scale , Filter , Drive1 , Drive2 , Aalarm , Dalarm , RunHours ,,,,... ..
- ▶ **IEC1131-3 Group** : TOF , TON , TP ,SR , RS ,MUX,MAX,MIN , ...



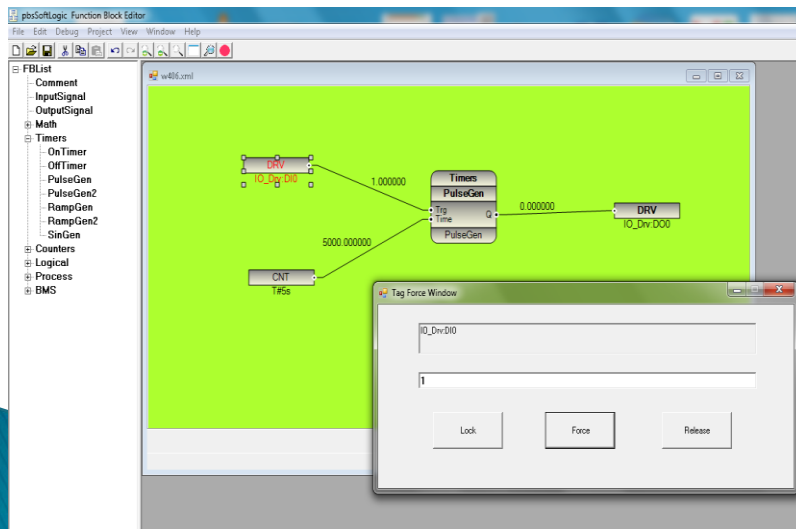
## Offline Logic Simulation



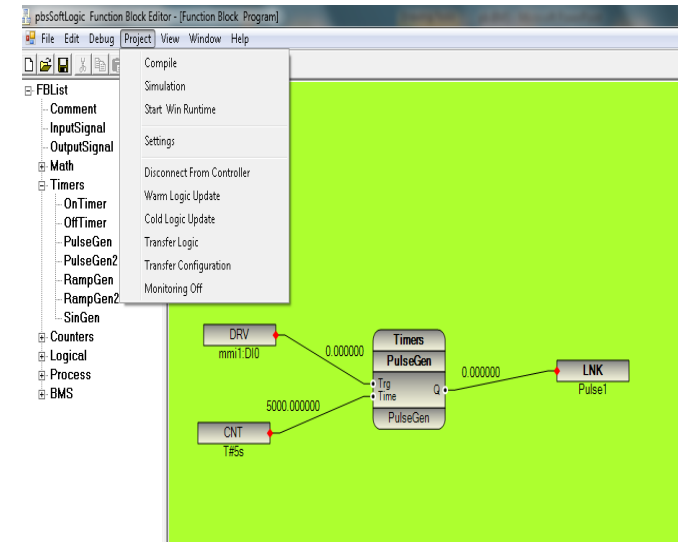
## Graphic Programming



## Tag Force Utility



## Logic Monitoring



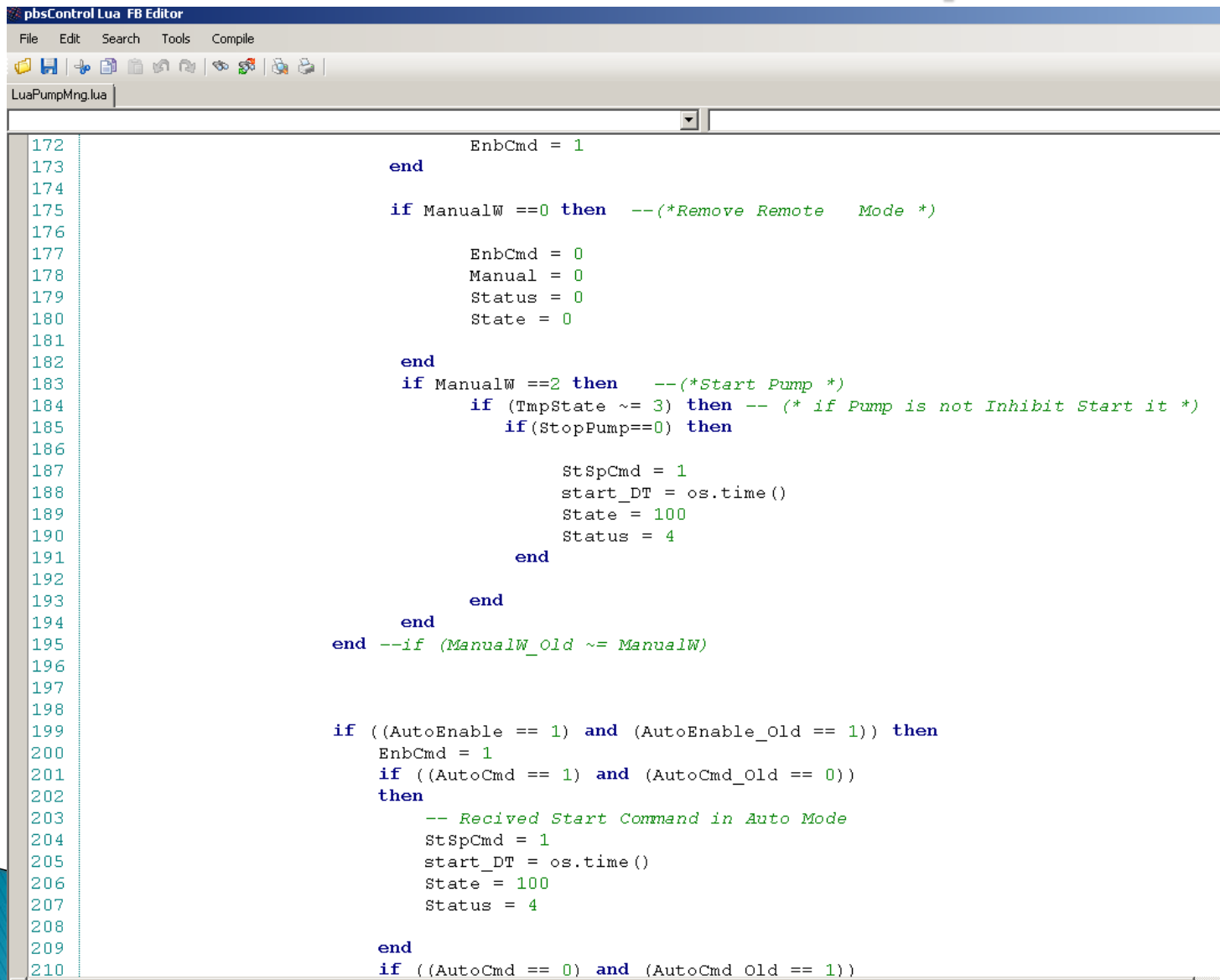


# User Defined Function Block Editor

- ▶ pbsSoftLogic is powered by Lua Scripting language for developing user defined FB
- ▶ Lua – [www.Lua.org](http://www.Lua.org) is one of the most famous scripting language for embedded systems
- ▶ pbsSoftLogic has built in Lua Editor and compiler for making UDF . UDF module will run on Windows ,Linux and QNX controller . ( No Cross compiling for Linux/WinCE/QNX Target )



# Lua Editor and compiler



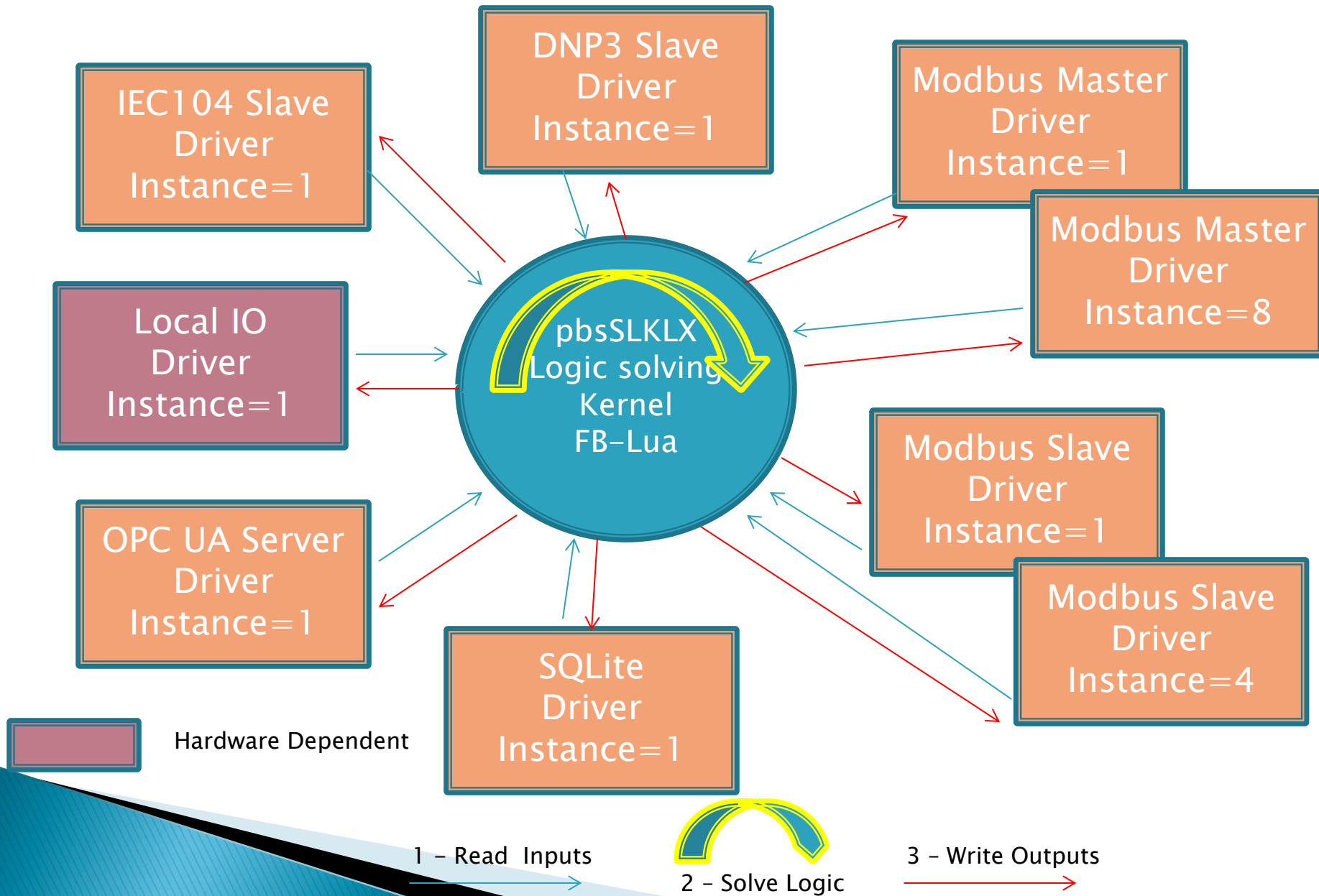
The screenshot shows the 'pbsControl Lua FB Editor' window. The menu bar includes 'File', 'Edit', 'Search', 'Tools', and 'Compile'. The toolbar contains icons for file operations and execution. The file 'LuaPumpMng.lua' is open. The editor displays a Lua script with line numbers 172 to 210 on the left. The script logic includes setting 'EnbCmd = 1', handling manual mode transitions, and starting a pump under specific conditions.

```
172         EnbCmd = 1
173     end
174
175     if ManualW == 0 then --(*Remove Remote Mode *)
176
177         EnbCmd = 0
178         Manual = 0
179         Status = 0
180         State = 0
181
182     end
183     if ManualW == 2 then --(*Start Pump *)
184         if (TmpState ~= 3) then -- (* if Pump is not Inhibit Start it *)
185             if (StopPump == 0) then
186
187                 StSpCmd = 1
188                 start_DT = os.time()
189                 State = 100
190                 Status = 4
191             end
192         end
193     end
194 end
195 --if (ManualW_Old ~= ManualW)
196
197
198
199 if ((AutoEnable == 1) and (AutoEnable_Old == 1)) then
200     EnbCmd = 1
201     if ((AutoCmd == 1) and (AutoCmd_Old == 0))
202     then
203         -- Received Start Command in Auto Mode
204         StSpCmd = 1
205         start_DT = os.time()
206         State = 100
207         Status = 4
208
209     end
210     if ((AutoCmd == 0) and (AutoCmd_Old == 1))
```

# Supported Protocols

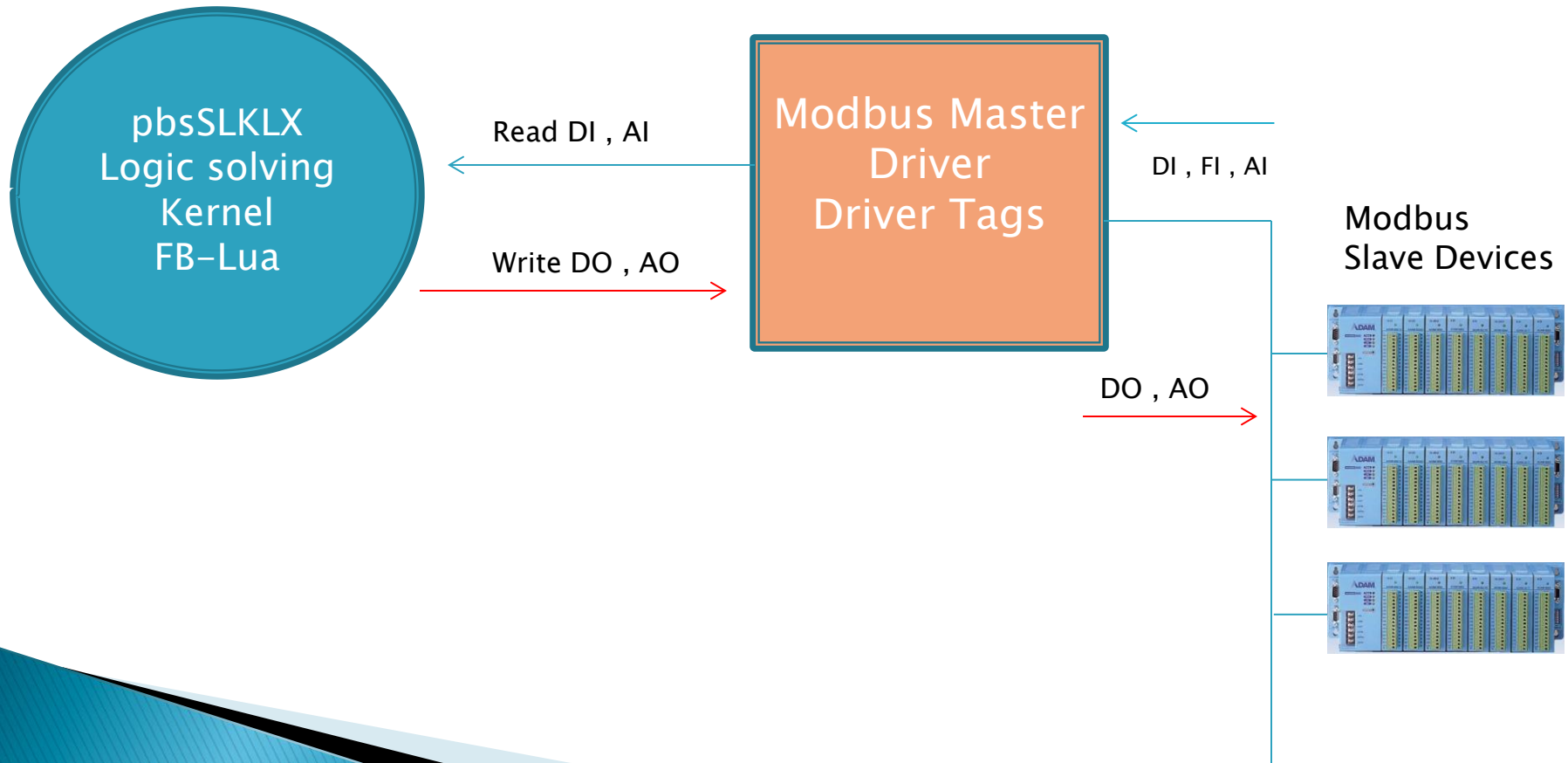
- ▶ pbssoftLogic supports following protocols :
  - Modbus Master TCP/RTU , 8 different instances
  - Modbus Slave TCP/RTU , 4 different instance
  - DNP3 Slave level3 , 4 different instance
  - IEC870-5-101 , Slave 4 different instance
  - IEC870-5-104 , Slave 4 different instance
  - IEC870-5-103 , Master 4 different instance
  - GSP ( GPRS for SCADA Project) low profile Protocol for SCADA systems based on GPRS
  - OPC DA2 client ( Just for win32 Target)
  - Siemens S7-Connect
  - OPC UA DA Server (for ADAM-3600 & pbs2008RTU ,UNO-1252)
  - SQLite Server with MS SQL Server Direct Client Driver
  - Advantech TagLink Interface for ADAM-3600
  - User Defined C API for custom Communication protocol development
- ▶ Each Communication instance accept 1024 signals
- ▶ For getting DNP and IEC870-5-104 device profile , visit [www.pbscontrol.com](http://www.pbscontrol.com)

# Runtime Structure



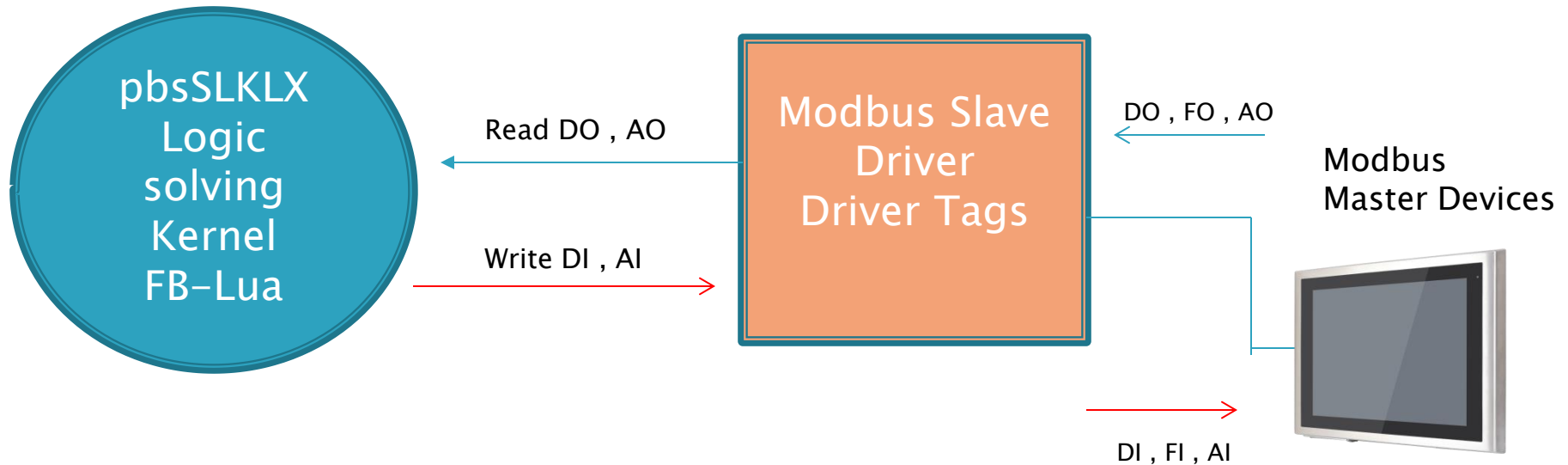
# Driver Structure ( Master )

- Each Driver has it Thread and Tags
- Logic is reading Inputs and Writing Outputs at each cycle
- Driver is independent of other module , communicates with Third Party and update Driver Tags



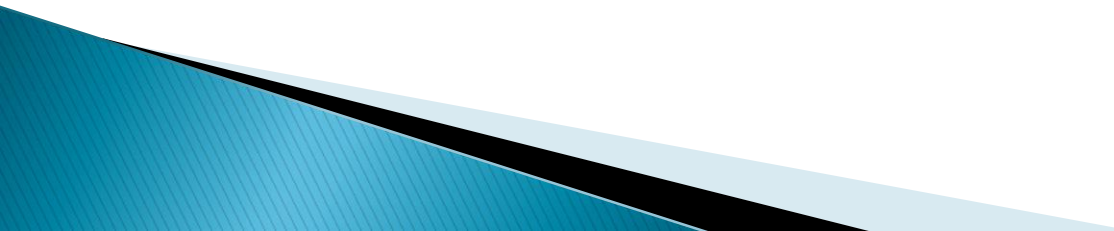
# Driver Structure Slave

- Slave drivers are reverse
  - Logic will read DO , AO from Driver
  - Logic will write DI , AI , FI to Driver
  - Master Device Like Will read DI , AI , FI from Driver
  - Master Device will Write DO , AO , FO to Driver





# Runtime Structure Specifications

- ▶ 100% isolation between Logic and communication Driver
  - ▶ Never use Logic to handling communication
  - ▶ Communication Driver has Built in Tags that should be Read/Write by Logic
  - ▶ You will do integration between Drivers by RTU Logic by Read/Write tags between different Communication drivers
- 

# Local IO Driver

- ▶ Local IO driver is isolating Hardware from pbsSoftLogic Runtime Kernel
- ▶ All Local IO , Watch Dog , LED , ... which is related to a specific hardware will define In Local IO Driver
- ▶ You can reset RTU , Make Integrated Modem Off/On , Read/Write Local IO , Make LED Off/On , Enable Watch Dog , Set AI /AO Range ... By Local IO Driver , ...

```
<OPCSrvTags>
```

```
<Version>1.0.0</Version>
```

```
<Tag Name="SYS.Reset" Type="SYS" Init="0" Address="0" />
```

```
<Tag Name="SYS.3GModemON" Type="SYS" Init="0" Address="1" />
```

```
<Tag Name="SYS.3GModemSignallevel" Type="SYS" Init="0" Address="2" />
```

```
<Tag Name="SYS.Temp1" Type="SYS" Init="0" Address="3" />
```

```
<Tag Name="SYS.Temp2" Type="SYS" Init="0" Address="4" />
```

```
<Tag Name="SYS.CNTTimer" Type="SYS" Init="200" Address="5" />
```

```
<Tag Name="SYS.AORange" Type="SYS" Init="1" Address="6" />
```

```
<Tag Name="SYS.Buzzer" Type="SYS" Init="0" Address="7" />
```

```
<Tag Name="SYS.IOScan" Type="SYS" Init="100" Address="8" />
```

```
<Tag Name="SYS.Total1" Type="SYS" Init="0" Address="9" />
```

```
<Tag Name="SYS.Total2" Type="SYS" Init="0" Address="10" />
```

```
<Tag Name="SYS.Total3" Type="SYS" Init="0" Address="11" />
```

```
<Tag Name="SYS.Total4" Type="SYS" Init="0" Address="12" />
```

```
<Tag Name="SYS.Total1RST" Type="SYS" Init="0" Address="13" />
```

```
<Tag Name="SYS.Total2RST" Type="SYS" Init="0" Address="14" />
```

```
<Tag Name="SYS.Total3RST" Type="SYS" Init="0" Address="15" />
```

```
<Tag Name="SYS.Total4RST" Type="SYS" Init="0" Address="16" />
```

```
<Tag Name="AITag0" Type="AI" Init="0" Address="0" />
```

```
<Tag Name="AITag1" Type="AI" Init="0" Address="1" />
```

```
<Tag Name="AITag2" Type="AI" Init="0" Address="2" />
```

```
<Tag Name="AITag3" Type="AI" Init="0" Address="3" />
```

```
<Tag Name="DITag0" Type="DI" Init="0" Address="0" />
```

```
<Tag Name="DITag1" Type="DI" Init="0" Address="1" />
```

```
<Tag Name="DITag2" Type="DI" Init="0" Address="2" />
```

```
<Tag Name="DITag3" Type="DI" Init="0" Address="3" />
```

```
<Tag Name="DITag4" Type="DI" Init="0" Address="4" />
```

```
<Tag Name="DITag5" Type="DI" Init="0" Address="5" />
```

```
<Tag Name="DITag6" Type="DI" Init="0" Address="6" />
```

```
<Tag Name="DITag7" Type="DI" Init="0" Address="7" />
```

```
<Tag Name="DOTag0" Type="DO" Init="0" Address="0" />
```

```
<Tag Name="DOTag1" Type="DO" Init="0" Address="1" />
```

```
<Tag Name="DOTag2" Type="DO" Init="0" Address="2" />
```

```
<Tag Name="DOTag3" Type="DO" Init="0" Address="3" />
```

```
<Tag Name="CNTTag0" Type="CNT" Init="0" Address="0" />
```

```
<Tag Name="CNTTag1" Type="CNT" Init="0" Address="1" />
```

```
<Tag Name="CNTTag2" Type="CNT" Init="0" Address="2" />
```

```
<Tag Name="CNTTag3" Type="CNT" Init="0" Address="3" />
```

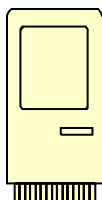
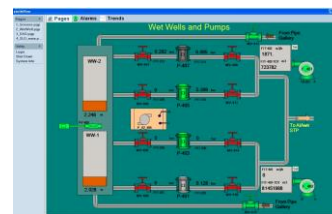
```
<Tag Name="AOTag0" Type="AO" Init="0" Address="0" />
```



pbsHMI  
clients



pbsHMI  
clients



RDBMS  
Server

TCP/IP



pbsHMI Server

DNP3 / IEC104 / GSP / OPCUA



pbs2008RIO

Modbus Remote I/O



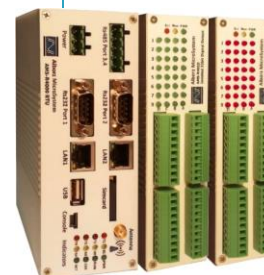
MAPCS-R



AMS-4000-GW



AMS-4000-RTU



Modbus



Power Meter - Flow Computer

# IEC870-5-104 Master SCADA Tested Connection

- ▶ TestHarness [www.trianglemicroworks.com](http://www.trianglemicroworks.com)
- ▶ WinPP104 <http://www.ipcomm.de/>
- ▶ High-leit SCADA [www.ids.de](http://www.ids.de)
- ▶ Cegelec ViewStar 750 Master SCADA
- ▶ pbsControl IEC870-5-104 Mater OPC Server  
[www.pbscontrol.com](http://www.pbscontrol.com)
- ▶

# DNP3 Master SCADA Tested Connection

- ▶ TestHarness [www.trianglemicroworks.com](http://www.trianglemicroworks.com)
  - ▶ Kepware OPC Server
  - ▶ ACS Master SCADA Software
  - ▶ Telvent Master SCADA
- 