

---

# REDUNDANT CONTROLLER IMPLEMENTATION

V1.00.02

---

<b>Section</b>	<b>Page</b>
INTRODUCTION	2
WALLCLOCK DISCIPLINE	3
• SET SYSTEM VALUE (SSV)	3
○ RSLOGIX V16	3
○ RSLOGIX V19	3
• PRECISION TIME PROTOCOL (PTP)	4
○ RSLOGIX V19	4
SEQUENCE OF EVENTS (SOE)	7
• CST / UTC	7
○ RSLOGIX V16	7
○ RSLOGIX V19	9
• PRECISION TIME PROTOCOL (PTP)	9
○ RSLOGIX V19	9

---

## INTRODUCTION

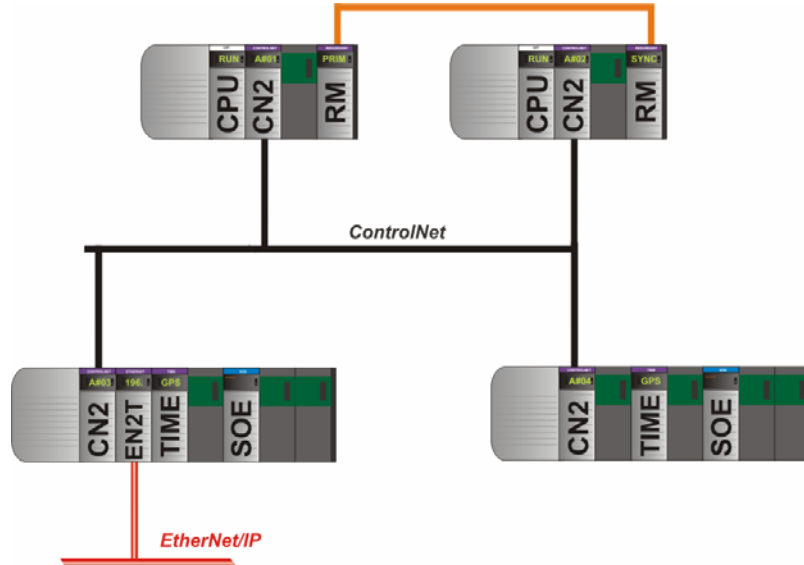
This document serves to describe the implementation of time synchronization (using the 1756HP-TIME module) in a redundant Allen-Bradley ControlLogix system. Currently the user can either use RSLogix v16 or RSLogix v19 when implementing the redundant control system (where v19 supports the Precision Time Protocol or PTP). An explanation will be given on synchronizing controllers as well as doing sequence of events (SOE) solutions.

If the user has any questions or queries please email: [tech@hiprom.com](mailto:tech@hiprom.com)

# WALLCLOCK DISCIPLINE

## Set System Value (SSV)

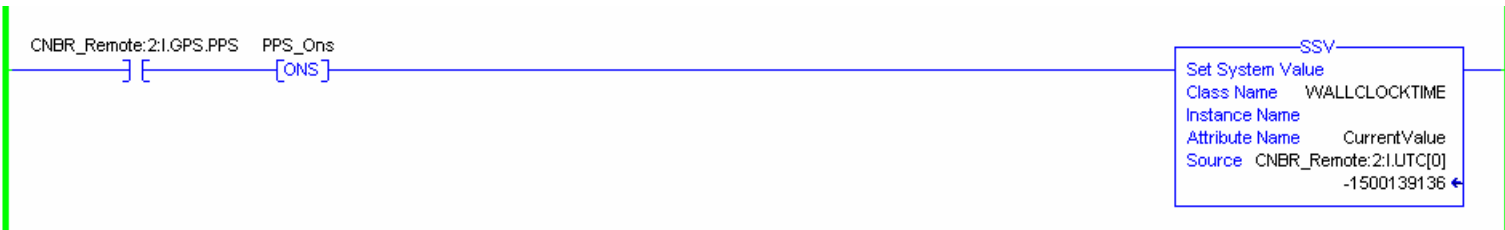
The following redundant PLC architecture will be used as an example:



### RSLogix v16

The user can set the wallclock by writing the current UTC value to it. Because the CPU is not in the same rack as the 1756HP-TIME module the UTC offset cannot be used to discipline the clock because the CST of the local and remote racks are different.

The PPS bit can be used to discipline the clock every second as shown below:



Thus at the beginning of every second the SSV instruction will discipline the clock by using the UTC value received from the 1756HP-TIME module.



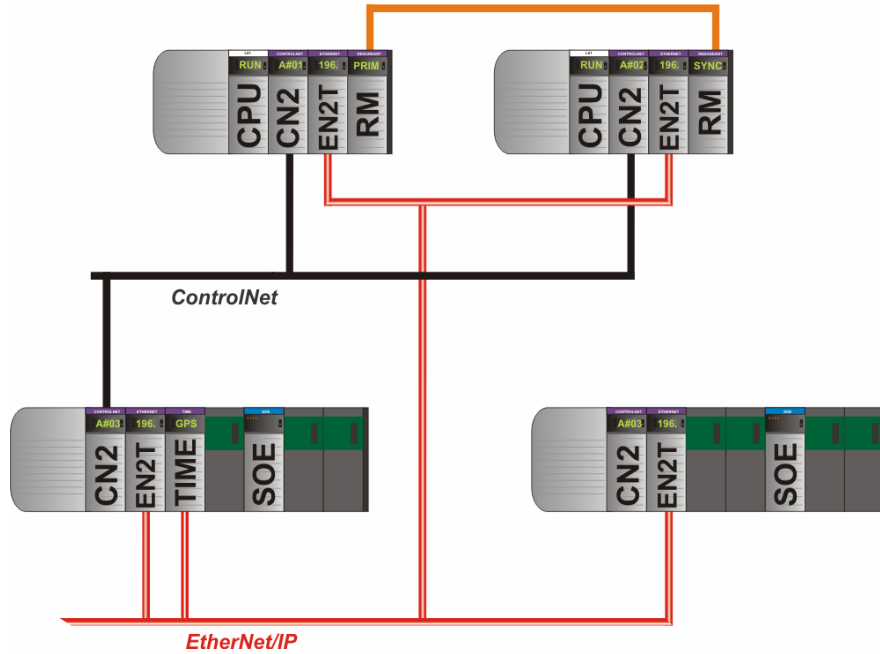
**NOTE:** The wallclock time will only be as accurate as the RPI of the remote rack + the RPI of the 1756HP-TIME module. For example if the remote rack RPI is 20ms and the 1756HP-TIME module’s RPI is 10ms the max wallclock time error will be 30ms.

### RSLogix v19

If the user is not using CIPSync (Precision Time Protocol – PTP) then the RSLogix v16 method will also apply to RSLogix v19.

## Precision Time Protocol (PTP)

The following redundant PLC architecture will be used as an example:



### RSLogix v19

When using CIPSync the user must enable 3 options. This will allow the 1756HP-TIME module to synchronize the Controllers without any ladder needed (note that the time accuracy is much better when PTP is being used):

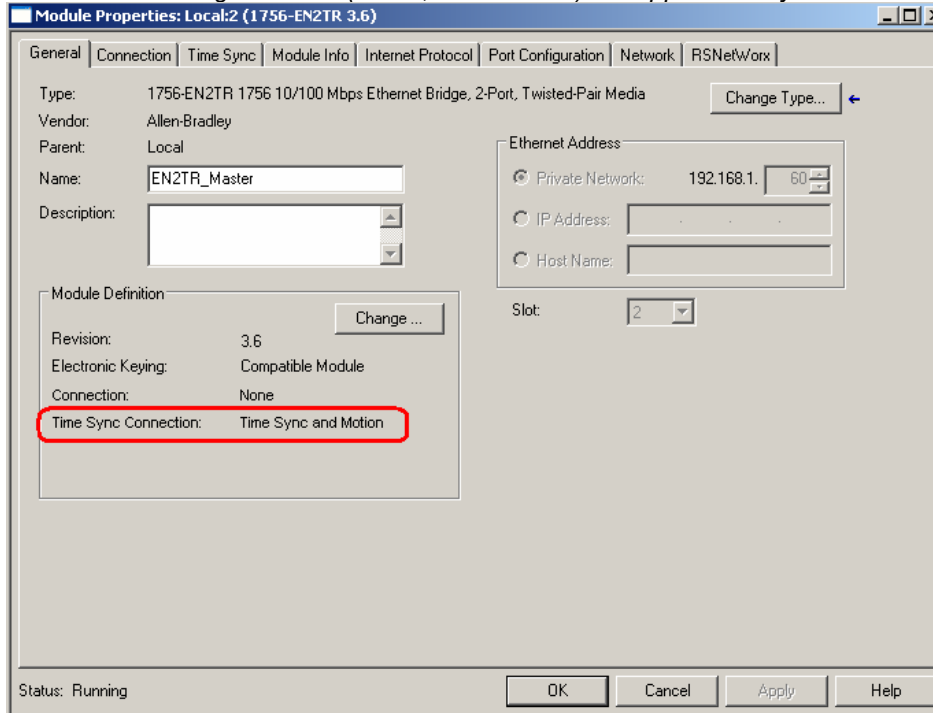
1. Set the 1756HP-TIME module to output PTP in either the config image UDT or AOP:

[-] CNBR_Remote:2:C	{...}	{..	HT:1756HI
+ CNBR_Remote:2:C.Source	1	Decimal	INT
+ CNBR_Remote:2:C.Control	2#0000 0000 0010 0000	Binary	INT
+ CNBR_Remote:2:C.PTPOutputEnable	1	Decimal	BOOL
- CNBR_Remote:2:C.NTPOutputEnable	0	Decimal	BOOL
- CNBR_Remote:2:C.IRIGBOutputEnable	0	Decimal	BOOL
- CNBR_Remote:2:C.IRIGBLockLostTx	0	Decimal	BOOL
- CNBR_Remote:2:C.PreV16Support	0	Decimal	BOOL
- CNBR_Remote:2:C.CSTMastershipEnable	0	Decimal	BOOL
+ CNBR_Remote:2:C.NTPUpdateInterval	3600	Decimal	DINT
+ CNBR_Remote:2:C.ExternalSourceAddress	{...}	{..	Decimal INT[4]
+ CNBR_Remote:2:C.Priority1	128	Decimal	INT
+ CNBR_Remote:2:C.Priority2	128	Decimal	INT
+ CNBR_Remote:2:C.PTPTimeToLive	64	Decimal	INT
+ CNBR_Remote:2:C.PTPSyncInterval	4	Decimal	SINT
+ CNBR_Remote:2:C.AdvancedConfig	0	Decimal	DINT
+ CNBR_Remote:2:C.DemoWeekNumber	0	Decimal	DINT
+ CNBR_Remote:2:C.DemoWeekSeconds	0	Decimal	DINT
+ CNBR_Remote:2:C.SpareDINT	0	Decimal	DINT

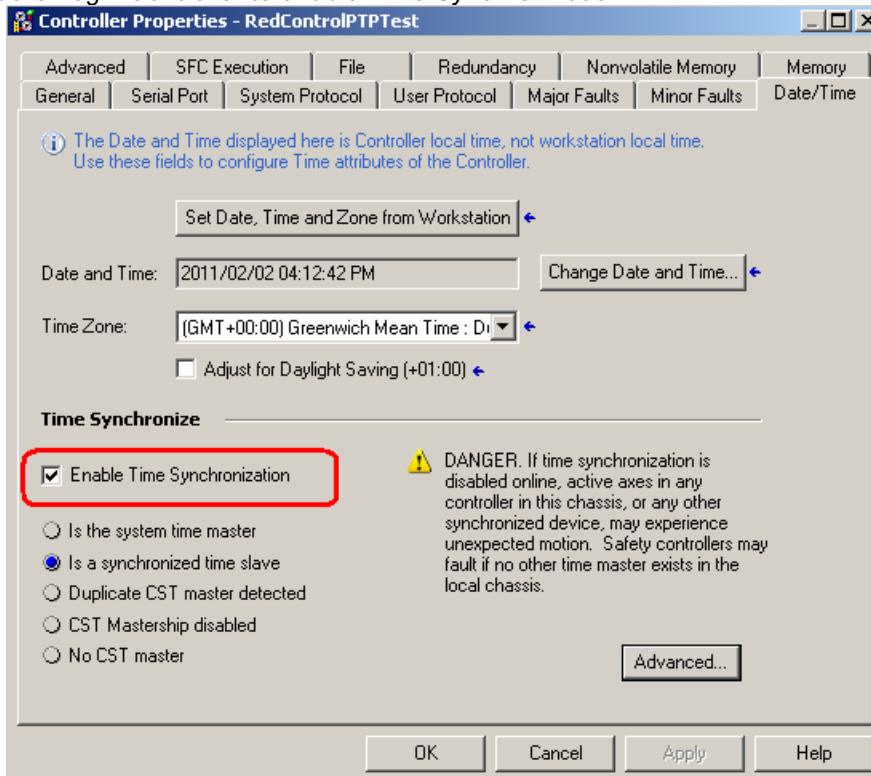


**NOTE:** If the user is using major firmware revision 2 of the 1756HP-TIME module all parameters can be set from the Add-on Profile (AOP) and no UDTs are required as the images are structured.

2. Set the Ethernet bridge module (EN2T, EN2TR etc.) to support CIPSync:

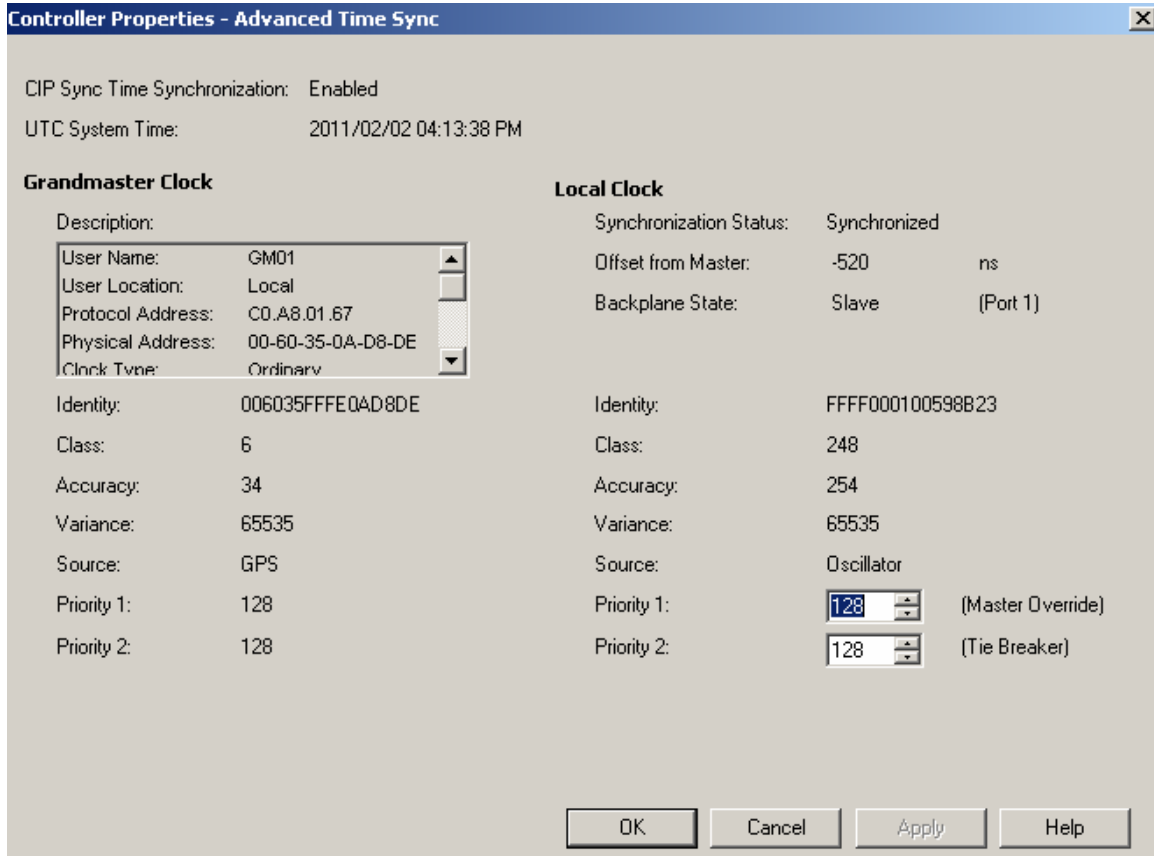


3. Set the Logix Controller to enable Time Synchronization:



The Ethernet Bridge will serve as a boundary clock (PTP slave on Ethernet and PTP master on ControlBus). Thus the Logix Controller will synchronize to the Ethernet Bridge which in turn will synchronize to the 1756HP-TIME module.

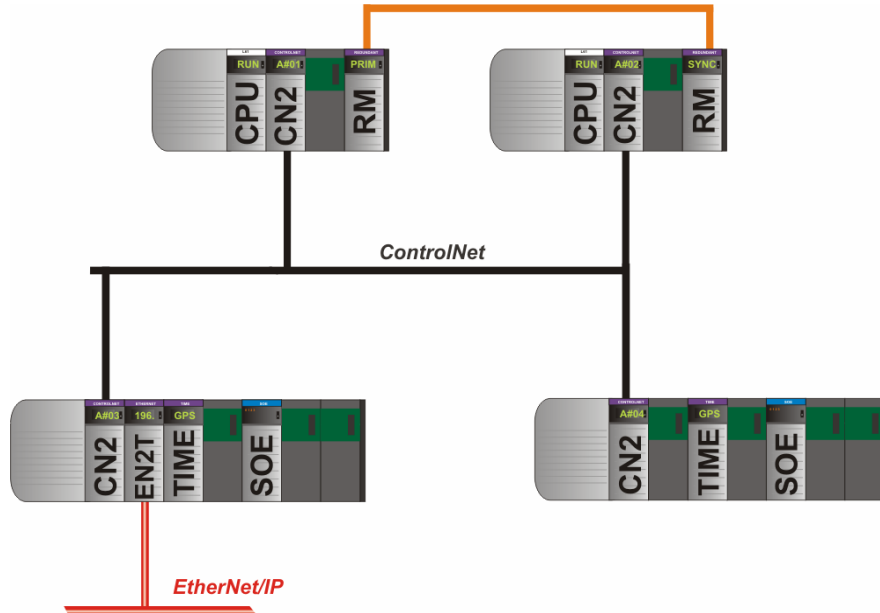
Once the Logix Controller has synchronized to the TIME module the user can confirm the time accuracy by selecting the *Advanced* button in the Date/Time tab of the Controller AOP:



# SEQUENCE OF EVENTS (SOE)

## CST / UTC

The following redundant PLC architecture will be used as an example:



### RSLogix v16

When doing sequence of events (SOE) in a redundant system the user will need to add a 1756HP-TIME module per rack as to provide the CST and UTC pairs to each SOE module. This will allow the SOE module to timestamp events in UTC time. Thus the user will need to enable the CST Mastership on the 1756HP-TIME module and copy the UTC and CST pairs to each SOE module output assembly. The CST Mastership can be enabled either via the Config image UDT or AOP:

[-] CNBR_Remote:2:C	{...}	{..	HT:1756HI
[-] CNBR_Remote:2:C.Source	1	Decimal	INT
[-] CNBR_Remote:2:C.Control	2#0000_0000_0010_0000	Binary	INT
[-] CNBR_Remote:2:C.PTPOutputEnable	0	Decimal	BOOL
[-] CNBR_Remote:2:C.NTPOutputEnable	0	Decimal	BOOL
[-] CNBR_Remote:2:C.IRIGBOutputEnable	0	Decimal	BOOL
[-] CNBR_Remote:2:C.IRIGBLockLostTx	0	Decimal	BOOL
[-] CNBR_Remote:2:C.PreV16Support	0	Decimal	BOOL
[-] CNBR_Remote:2:C.CSTMastershipEnable	1	Decimal	BOOL
[-] CNBR_Remote:2:C.NTPUpdateInterval	3600	Decimal	DINT
[-] CNBR_Remote:2:C.ExternalSourceAddress	{...}	{..	Decimal INT[4]
[-] CNBR_Remote:2:C.Priority1	128	Decimal	INT
[-] CNBR_Remote:2:C.Priority2	128	Decimal	INT
[-] CNBR_Remote:2:C.PTPTimeToLive	64	Decimal	INT
[-] CNBR_Remote:2:C.PTPSyncInterval	4	Decimal	SINT
[-] CNBR_Remote:2:C.AdvancedConfig	0	Decimal	DINT
[-] CNBR_Remote:2:C.DemoWeekNumber	0	Decimal	DINT
[-] CNBR_Remote:2:C.DemoWeekSeconds	0	Decimal	DINT
[-] CNBR_Remote:2:C.SpareDINT	0	Decimal	DINT



**NOTE:** If the user is using major firmware revision 2 of the 1756HP-TIME module all setting can be done from the Add-on Profile (AOP) and no UDTs are required as the images are structured.

The user can copy the CST and UTC pairs as shown below:

1

COP

Copy File

Source CNBR\_Remote:2:I.UTC[0]

Dest CNBR\_Remote:4:O.UCTTime[0]

Length 2

2

COP

Copy File

Source CNBR\_Remote:2:I.CST[0]

Dest CNBR\_Remote:4:O.CSTTime[0]

Length 2

[-] CNBR_Remote:4:O	{...}	{..}		AB:1756
[+] CNBR_Remote:4:O.EventAck	0		Decimal	DINT
[+] CNBR_Remote:4:O.NewDataAck	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
[+] CNBR_Remote:4:O.PointToRetrieve	0		Decimal	INT
[-] CNBR_Remote:4:O.ResetEvents	0		Decimal	BOOL
[-] CNBR_Remote:4:O.RetrieveByPoint	0		Decimal	BOOL
[-] CNBR_Remote:4:O.UCTTime	{...}	{..}	Decimal	DINT[2]
[+] CNBR_Remote:4:O.UCTTime[0]	656828160		Decimal	DINT
[+] CNBR_Remote:4:O.UCTTime[1]	301915		Decimal	DINT
[-] CNBR_Remote:4:O.CSTTime	{...}	{..}	Decimal	DINT[2]
[+] CNBR_Remote:4:O.CSTTime[0]	279410533		Decimal	DINT
[+] CNBR_Remote:4:O.CSTTime[1]	544		Decimal	DINT

Once this is done the SOE module will indicate that it is reporting back UTC timestamps:

[-] CNBR_Remote:4:I	{...}	{..}		AB:1756
[+] CNBR_Remote:4:I.Fault	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
[+] CNBR_Remote:4:I.Data	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
[+] CNBR_Remote:4:I.NewData	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
[+] CNBR_Remote:4:I.EventOverflow	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
[+] CNBR_Remote:4:I.ReturningUCTTime	1		Decimal	BOOL
[+] CNBR_Remote:4:I.EventNumber	0		Decimal	DINT

When an event occurs on the SOE module will timestamp the UTC value.



[-] CNBR_Remote:4:I	{...}	{..		AB:1756
[+] CNBR_Remote:4:I.Fault	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
[+] CNBR_Remote:4:I.Data	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
[+] CNBR_Remote:4:I.NewData	2#0000_0000_0000_0000_0000_0000_0000_0001		Binary	DINT
[+] CNBR_Remote:4:I.EventOverflow	2#0000_0000_0000_0000_0000_0000_0000_0000		Binary	DINT
- CNBR_Remote:4:I.ReturningUCTTime		1	Decimal	BOOL
[+] CNBR_Remote:4:I.EventNumber		2	Decimal	DINT
[-] CNBR_Remote:4:I.Timestamp	{...}	{..		AB:1756
[-] CNBR_Remote:4:I.Timestamp[0]	{...}	{..		AB:1756
[-] CNBR_Remote:4:I.Timestamp[0].OffOn	{...}	{..	Decimal	DINT[2]
[+] CNBR_Remote:4:I.Timestamp[0].D...	1655458586		Decimal	DINT
[+] CNBR_Remote:4:I.Timestamp[0].D...	301915		Decimal	DINT
[-] CNBR_Remote:4:I.Timestamp[0].UnUdt	{...}	{..	Decimal	DINT[2]
[+] CNBR_Remote:4:I.Timestamp[0].D...	1657324453		Decimal	DINT
[+] CNBR_Remote:4:I.Timestamp[0].D...	301915		Decimal	DINT

This can be copied into a variable of type *LINT* and viewed in *Date/Time* format to see the Gregorian Date.

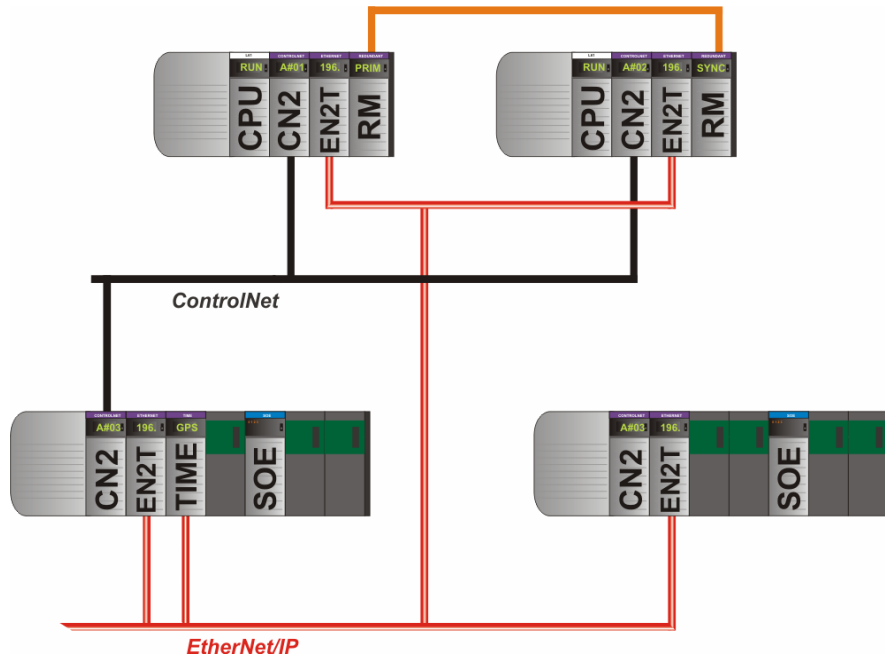
EventTime	DT#2011-02-03-09:05:06.630426(GMT+02:00)	Date/Time	LINT
-----------	--	-----------	------

### RSLogix v19

If the user is not using CIPSync (Precision Time Protocol – PTP) then the RSLogix v16 method will also apply to RSLogix v19.

### Precision Time Protocol (PTP)

The following redundant PLC architecture will be used as an example:



**RSLogix v19**

When using CIPSync (PTP) the user must enable 3 options. This will allow the 1756HP-TIME module to synchronize the SOE modules without any ladder needed:



**NOTE:** Only major revision 2 of the 1756-IB16ISOE module will support CIPSync (PTP).

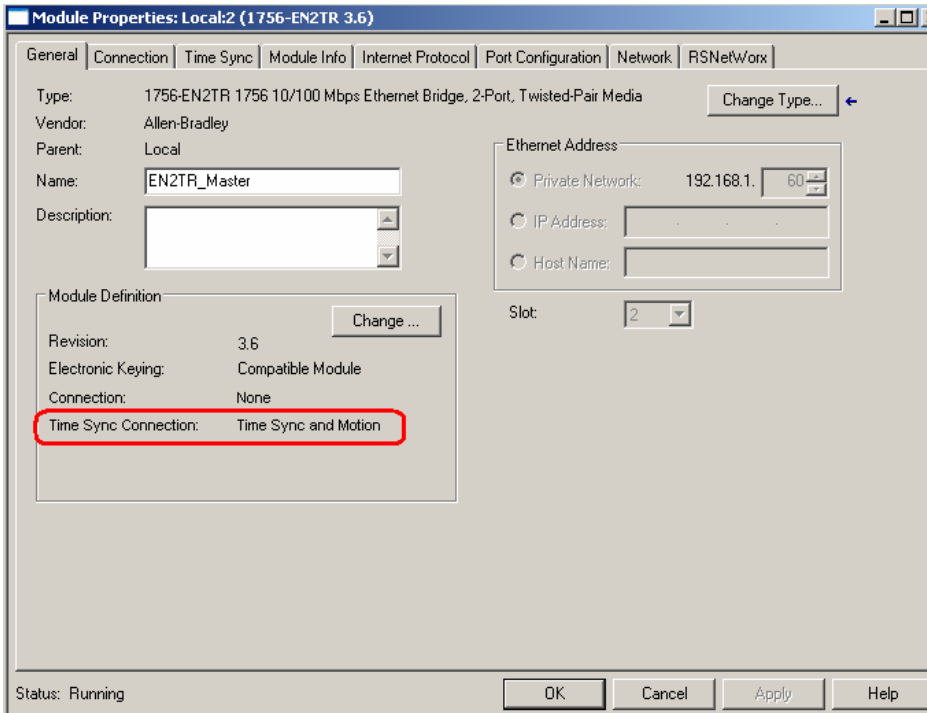
1. Set the 1756HP-TIME module to output PTP in either the config image UDT or AOP:

[-] CNBR_Remote:2:C	{...}	{..}		HT:1756HI
[+] CNBR_Remote:2:C.Source	1		Decimal	INT
[+] CNBR_Remote:2:C.Control	2#0000 0000 0010 0000		Binary	INT
[+] CNBR_Remote:2:C.PTPOutputEnable	1		Decimal	BOOL
[+] CNBR_Remote:2:C.NTPOutputEnable	0		Decimal	BOOL
[+] CNBR_Remote:2:C.IRIGBOutputEnable	0		Decimal	BOOL
[+] CNBR_Remote:2:C.IRIGBLockLostTx	0		Decimal	BOOL
[+] CNBR_Remote:2:C.PreV16Support	0		Decimal	BOOL
[+] CNBR_Remote:2:C.CSTMastershipEnable	0		Decimal	BOOL
[+] CNBR_Remote:2:C.NTPUpdateInterval	3600		Decimal	DINT
[+] CNBR_Remote:2:C.ExternalSourceAddress	{...}	{..}	Decimal	INT[4]
[+] CNBR_Remote:2:C.Priority1	128		Decimal	INT
[+] CNBR_Remote:2:C.Priority2	128		Decimal	INT
[+] CNBR_Remote:2:C.PTPTimeToLive	64		Decimal	INT
[+] CNBR_Remote:2:C.PTPSyncInterval	4		Decimal	SINT
[+] CNBR_Remote:2:C.AdvancedConfig	0		Decimal	DINT
[+] CNBR_Remote:2:C.DemoWeekNumber	0		Decimal	DINT
[+] CNBR_Remote:2:C.DemoWeekSeconds	0		Decimal	DINT
[+] CNBR_Remote:2:C.SpareDINT	0		Decimal	DINT

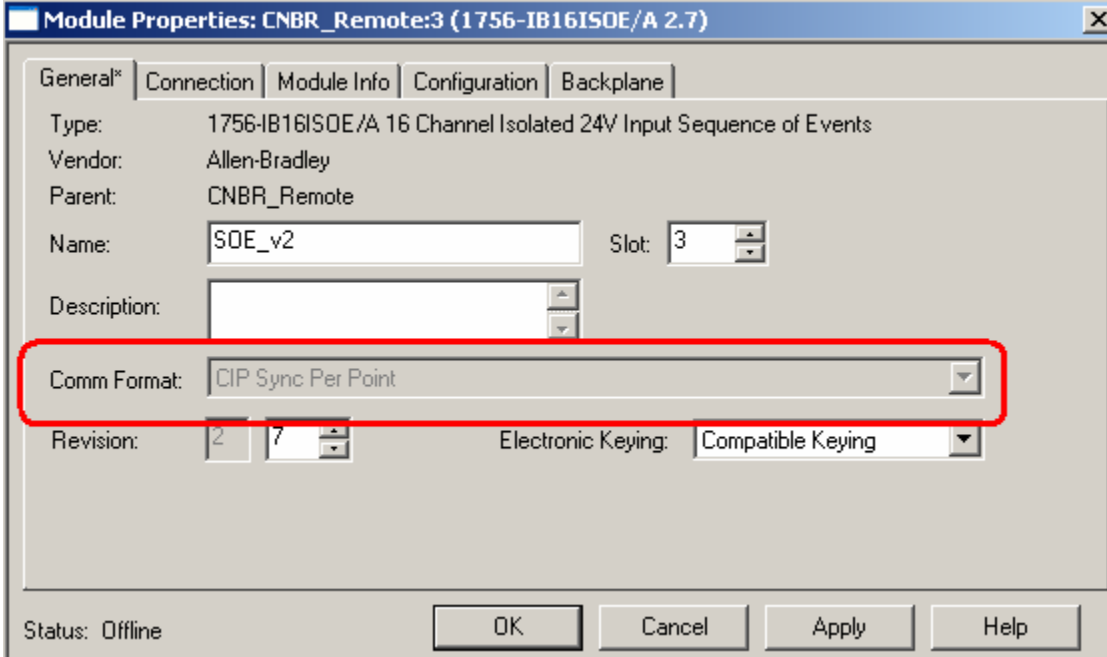


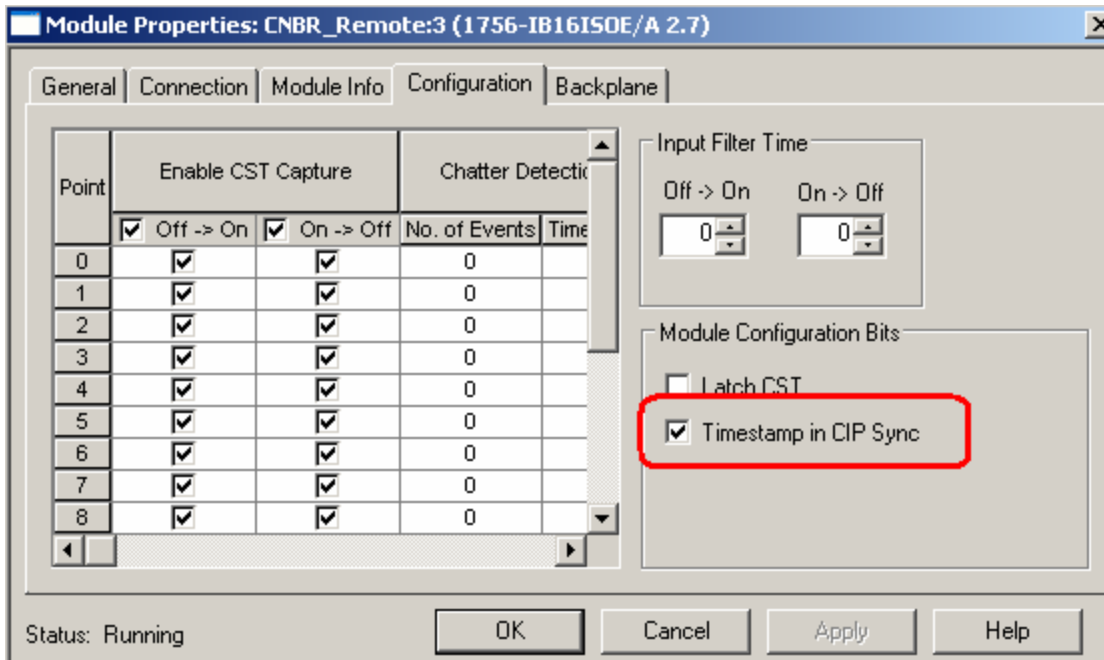
**NOTE:** If the user is using major firmware revision 2 of the 1756HP-TIME module all setting can be done from the Add-on Profile (AOP) and no UDTs are required as the images are structured.

2. Set the Ethernet bridge module (EN2T, EN2TR etc.) to support CIPSync:



3. Set the SOE module to timestamp using CIPSync (PTP):





The Ethernet Bridge will serve as a boundary clock (PTP slave on Ethernet and PTP master on ControlBus) for each rack. Thus the SOE module will synchronize to the Ethernet Bridge which in turn will synchronize to the 1756HP-TIME module.

Once the SOE module has synchronized to the TIME module it will indicate that the timestamps will provide the UTC from the TIME module via CIPSync:

+	CNBR_Remote:3:I.EventOverflow	2#0000_0000_0000_0000_0000_0000_0000_0001	Binary	DINT
-	CNBR_Remote:3:I.ReturningUCTTime	0	Decimal	BOOL
-	CNBR_Remote:3:I.ReturningCIPSyncTime	1	Decimal	BOOL
-	CNBR_Remote:3:I.CIPSyncValid	1	Decimal	BOOL
-	CNBR_Remote:3:I.CIPSyncTimeout	0	Decimal	BOOL
+	CNBR_Remote:3:I.EventNumber	10	Decimal	DINT
+	CNBR_Remote:3:I.LocalClockOffset	{...}	{. Decimal	DINT[2]
+	CNBR_Remote:3:I.Offsettimestamp	{...}	{. Decimal	DINT[2]
-	CNBR_Remote:3:I.Timestamp	{...}	{.	AB:1756
-	CNBR_Remote:3:I.Timestamp[0]	{...}	{.	AB:1756
-	CNBR_Remote:3:I.Timestamp[0].OffOn	{...}	{. Decimal	DINT[2]
+	CNBR_Remote:3:I.Timestamp[0].OffOn[0]	921144119	Decimal	DINT
+	CNBR_Remote:3:I.Timestamp[0].OffOn[1]	301903	Decimal	DINT
-	CNBR_Remote:3:I.Timestamp[0].OnOff	{...}	{. Decimal	DINT[2]
+	CNBR_Remote:3:I.Timestamp[0].OnOff[0]	921144144	Decimal	DINT
+	CNBR_Remote:3:I.Timestamp[0].OnOff[1]	301903	Decimal	DINT

When an event occurs the SOE module will timestamp the UTC value.

+	CNBR_Remote:3:I.EventOverflow	2#0000_0000_0000_0000_0000_0000_0000_0001	Binary	DINT
-	CNBR_Remote:3:I.ReturningUCTTime	0	Decimal	BOOL
-	CNBR_Remote:3:I.ReturningCIPSyncTime	1	Decimal	BOOL
-	CNBR_Remote:3:I.CIPSyncValid	1	Decimal	BOOL
-	CNBR_Remote:3:I.CIPSyncTimeout	0	Decimal	BOOL
+	CNBR_Remote:3:I.EventNumber	10	Decimal	DINT
+	CNBR_Remote:3:I.LocalClockOffset	{...}	{. Decimal	DINT[2]
+	CNBR_Remote:3:I.Offsettimestamp	{...}	{. Decimal	DINT[2]
-	CNBR_Remote:3:I.Timestamp	{...}	{.	AB:1756
-	CNBR_Remote:3:I.Timestamp[0]	{...}	{.	AB:1756
-	CNBR_Remote:3:I.Timestamp[0].OnOff	{...}	{. Decimal	DINT[2]
+	CNBR_Remote:3:I.Timestamp[0].OnOff[0]	921144119	Decimal	DINT
+	CNBR_Remote:3:I.Timestamp[0].OnOff[1]	301903	Decimal	DINT
-	CNBR_Remote:3:I.Timestamp[0].OnOff	{...}	{. Decimal	DINT[2]
+	CNBR_Remote:3:I.Timestamp[0].OnOff[0]	921144144	Decimal	DINT
+	CNBR_Remote:3:I.Timestamp[0].OnOff[1]	301903	Decimal	DINT

This can be copied into a variable of type *LINT* and view in *Date/Time* format to see the Gregorian Date.

EventTime	DT#2011-02-02-18:33:52.708407 (GMT+02:00)	Date/Time	LINT
-----------	---	-----------	------



**HIPROM**

**TEL:** +27 11 787 4458

**FAX:** +27 11 787 7937

**POSTAL**

P.O. Box 732

Pinegowrie

South Africa

2123

**PHYSICAL**

369 Pretoria ave

Ferndale, Randburg

South Africa