

**FB112 - <offline>**

"FB\_PRORUNNER\_mk5"      Verticaal conveyor prorunner mk5

**Name:**                      **Family:**

**Author:**    Qimarox                      **Version:** 2.2

**Block version:** 2

**Time stamp Code:**                      10/10/2017 08:20:02 AM

**Interface:**                      10/10/2017 08:20:02 AM

**Lengths (block/logic/data):** 01960    01464    00014

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
iInitialize	Bool	0.0	FALSE	Initialize the Prorunner mk5
iReset	Bool	0.1	FALSE	Reset command for failures
iEnableElevator	Bool	0.2	FALSE	Enable clearence emergengystop, thermal etc.
iEnableAutomatic	Bool	0.3	FALSE	Enable mode automatic
iEnableManual	Bool	0.4	FALSE	FALSE = Mode manual, TRUE = Mode automatic
iManualForward	Bool	0.5	FALSE	Pushbutton Manual forward (normal run)
iManualReverse	Bool	0.6	FALSE	Pushbutton Manual reverse
iUpstream_Ready	Bool	0.7	FALSE	Upstream >> Prorunner; Conveyor is running and ready for product
iDownstream_Photocell_S1	Bool	1.0	FALSE	Downstream >> Prorunner; Object ready for infeed (photocell S1 covered)
iProductRelease_B1	Bool	1.1	FALSE	Sensor infeed product
iBeginInfeedConveyor_B6	Bool	1.2	FALSE	
iEndInfeedConveyor_B7	Bool	1.3	FALSE	Photocell on infeedconveyor
iStartTimeFrame_B3	Bool	1.4	FALSE	Sensor Forksecurity for outfeed
iProductOnOutfeed_B8	Bool	1.5	FALSE	Dropoff position diagonal occupied
iProductLeftOutfeed_B9	Bool	1.6	FALSE	Dropoff position straight occupied
iMonitorRotation_B5	Bool	1.7	FALSE	Sensor on bottom sprocket (Not standard)
iEmptyTimeEnabled	Time	2.0	T#0MS	Empty time enabled
iTimeInfeedAllowed	Time	6.0	T#0MS	Time of window infeed allowed
iTimeProductOnInfeed	Time	10.0	T#0MS	Time for delay on B7
iTimeCheckSensors	Time	14.0	T#0MS	Delay check sensors in 0,1 sec.

Name	Data Type	Address	Initial Value	Comment
iTimeMonitorInfeed	Time	18.0	T#0MS	Time monitoring of product infeed
iTimeCheckOutfeed	Time	22.0	T#0MS	Time off delay on B3, when time is past outfeed must be empty, if not lift stop
iTimeMonitorRotation	Time	26.0	T#0MS	Time for monitoring of rotation bottom sprocket(0 = disabled)
iTimeDelayMonitorRotatio	Time	30.0	T#0MS	Time to delay monitor rotation at startup
OUT		0.0		
oProrunnerForward	Bool	34.0	FALSE	Forward vertical conveyor
oProrunnerReverse	Bool	34.1	FALSE	Reverse vertical conveyor
oDownstream_Ready	Bool	34.2	FALSE	Prorunner >> Downstream; Ready for object infeed
oInfeedConveyor	Bool	34.3	FALSE	Infeed conveyor in Prorunner
oOutfeedConveyor	Bool	34.4	FALSE	Outfeed conveyor in Prorunner
oProrunnerIsEmpty	Bool	34.5	FALSE	Vertical conveyor is empty
oFaultTimeFrameSensor	Bool	34.6	FALSE	Fault Time frame sensor (sensor not triggered)
oFaultProdReleaseSensor	Bool	34.7	FALSE	Fault sensor product release (sensor not triggered)
oFaultProductOnOutfeed	Bool	35.0	FALSE	Fault product on outfeed conveyor present
oFaultProductInfeedtime	Bool	35.1	FALSE	Fault Overtime box inserted
oFaultProductPosition	Bool	35.2	FALSE	Fault product position on infeed
oFaultMonitorRotation	Bool	35.3	FALSE	Fault motor rotation
IN_OUT		0.0		
STAT		0.0		
sDelayForwardReverse	TON	36.0		
sDelayRelease	TOF	58.0		
sTimeDelayInfeed_B7	TON	80.0		
sTimeMonitorInfeed	TON	102.0		
sDelayProrunnerEmpty	TON	124.0		
sDelayCheckOutfeed	TOF	146.0		
sTimeCheckReleaseSensor	TON	168.0		
sTimeCheckTimeFrameSenso	TON	190.0		

Name	Data Type	Address	Initial Value	Comment
sTimeMonitorRotation	TON	212.0		
sTimeDelayMonitotRotatio	TON	234.0		
sCounters	Struct	256.0		
CountedProducts	DInt	256.0	L#0	
sHM_FP	Struct	260.0		
Reset	Bool	260.0	FALSE	
ProductRelease	Bool	260.1	FALSE	
CheckRotation	Bool	260.2	FALSE	
CheckDropOffPosition	Bool	260.3	FALSE	
BusyWithInfeed	Bool	260.4	FALSE	
sHM_FN	Struct	262.0		
ProductReleased	Bool	262.0	FALSE	
ProductLeftOutfeed_B9	Bool	262.1	FALSE	
CheckRotation	Bool	262.2	FALSE	
TimeFrameOutfeed	Bool	262.3	FALSE	
sManualActive	Bool	264.0	FALSE	
sProductOnOutfeed	Bool	264.1	FALSE	
sStopLiftProdOnOutfeed	Bool	264.2	FALSE	
sEmptyVerticaalConveyor	Bool	264.3	FALSE	
sProductInserted	Bool	264.4	FALSE	
sExtendedRelease	Bool	264.5	FALSE	
sInfeedBusy	Bool	264.6	FALSE	
TEMP		0.0		
tAutomaticTerms	Bool	0.0		
tAutomatic	Bool	0.1		
tManual	Bool	0.2		
tSignalToDelay	Bool	0.3		
tEnableMotorOn	Bool	0.4		
tEnableMotorForward	Bool	0.5		
tEnableMotorReverse	Bool	0.6		
tProrunnerRunning	Bool	0.7		
tProrunnerRotating	Bool	1.0		
tMonitorRotatioEnabled	Bool	1.1		
tFP	Struct	2.0		
Reset	Bool	2.0		
Productrelease	Bool	2.1		
CheckDropOffPosition	Bool	2.2		
CheckRotation	Bool	2.3		
BusyWithInfeed	Bool	2.4		
tFN	Struct	4.0		
ProductReleased	Bool	4.0		

Name	Data Type	Address	Initial Value	Comment
ProductLeftOutfeed_B9	Bool	4.1		
CheckRotation	Bool	4.2		
TimeFrameOutfeed	Bool	4.3		

<b>Block: FB112    Verticaal conveyor</b>
PRORUNNER mk5 - Function block ***** DISCLAIMER ***** We (Qimarox) are not liable legally or in any other way for erroneous program functions and their consequences! Please carefully read the technical manual of the PRORUNNER mk5 before implenting the function block ***** VERSION ***** V2.0 (2011-09-05) - First working add-on instruction  V2.1 (2014-11-1) - Updated outfeed procedure (B8 & B9)  V2.2 (2017-10-10) - Updated function with sensor B6 for product position check

<b>Network: 1            Init</b>
When Init is activated, the prorunner must be empty. all memory bits will be reset and the prorunner will start all over.
<div> <div>A</div> <div>#iInitialize</div> <div>#iInitialize</div> <div>-- Initialize the Prorunner mk5</div> </div> <div> <div>JCN</div> <div>NoIn</div> <div></div> <div></div> </div> <div> <div>SET</div> <div>R</div> <div>#sProductOnOutfeed</div> <div>#sProductOnOutfeed</div> </div> <div> <div>R</div> <div>#sEmptyVerticaalConveyor</div> <div>#sEmptyVerticaalConveyor</div> </div> <div> <div>R</div> <div>#sProductInserted</div> <div>#sProductInserted</div> </div> <div> <div>R</div> <div>#sStopLiftProdOnOutfeed</div> <div>#sStopLiftProdOnOutfeed</div> </div> <div> <div>NoIn: NOP</div> <div>0</div> <div></div> <div></div> </div>

<b>Network: 2            Reset</b>
<div> <div>A</div> <div>#iReset</div> <div>#iReset</div> <div>-- Reset command for failures</div> </div> <div> <div>FP</div> <div>#sHM_FP.Reset</div> <div>#sHM_FP.Reset</div> <div></div> </div> <div> <div>=</div> <div>#tFP.Reset</div> <div>#tFP.Reset</div> <div></div> </div>

<b>Network: 3            Release manual/automatic</b>
<div> <div>A</div> <div>#iEnableAutomatic</div> <div>#iEnableAutomatic</div> <div>-- Enable mode automatic</div> </div> <div> <div>A</div> <div>#iEnableElevator</div> <div>#iEnableElevator</div> <div>-- Enable clearence emergengystop, ther</div> </div> <div> <div>AN</div> <div>#iEnableManual</div> <div>#iEnableManual</div> <div>-- FALSE = Mode manual, TRUE = Mode aut</div> </div> <div> <div>=</div> <div>#tAutomaticTerms</div> <div>#tAutomaticTerms</div> <div></div> </div> <div> <div>A</div> <div>#tAutomaticTerms</div> <div>#tAutomaticTerms</div> <div></div> </div> <div> <div>AN</div> <div>#oFaultMonitorRotation</div> <div>#oFaultMonitorRotation</div> <div>-- Fault motor rotation</div> </div>

```

AN      #oFaultProdReleaseSensor  #oFaultProdReleaseSensor -- Fault sensor product release (
AN      #oFaultTimeFrameSensor    #oFaultTimeFrameSensor -- Fault Time frame sensor (sensor
AN      #sEmptyVerticaalConveyor  #sEmptyVerticaalConveyor
=      #tAutomatic                #tAutomatic

A      #iEnableElevator           #iEnableElevator    -- Enable clearence emergengystop, ther
A      #iEnableManual             #iEnableManual      -- FALSE = Mode manual, TRUE = Mode aut
AN      #iManualForward            #iManualForward    -- Pushbutton Manual forward (normal ru
AN      #iManualReverse            #iManualReverse      -- Pushbutton Manual reverse
S      #sManualActive             #sManualActive

ON      #iEnableElevator           #iEnableElevator    -- Enable clearence emergengystop, ther
ON      #iEnableManual             #iEnableManual      -- FALSE = Mode manual, TRUE = Mode aut
R      #sManualActive             #sManualActive

A      #sManualActive             #sManualActive
AN      #oFaultMonitorRotation     #oFaultMonitorRotation -- Fault motor rotation
=      #tManual                   #tManual

```

Network: 4	Empty verticaal conveyor
------------	--------------------------

```

AN      #iDownstream_Photocell_S1  #iDownstream_Photocell_S1 -- Downstream >> Prorunner; Obj
AN      #iBeginInfeedConveyor_B6   #iBeginInfeedConveyor_B6
AN      #iEndInfeedConveyor_B7     #iEndInfeedConveyor_B7 -- Photocell on infeedconveyor
AN      #iProductOnOutfeed_B8      #iProductOnOutfeed_B8 -- Dropoff position diagonal occup
AN      #iProductLeftOutfeed_B9    #iProductLeftOutfeed_B9 -- Dropoff position straight occu
A      #oProrunnerForward          #oProrunnerForward -- Forward  vertical conveyor
=      #tProrunnerRunning          #tProrunnerRunning

CALL    #sDelayProrunnerEmpty      #sDelayProrunnerEmpty
IN:=#tProrunnerRunning            #tProrunnerRunning
PT:=#iEmptyTimeEnabled            #iEmptyTimeEnabled -- Empty time enabled
Q :=#sEmptyVerticaalConveyor      #sEmptyVerticaalConveyor
ET:=

A      #sEmptyVerticaalConveyor    #sEmptyVerticaalConveyor
S      #oProrunnerIsEmpty          #oProrunnerIsEmpty -- Vertical conveyor is empty

O      #iDownstream_Photocell_S1  #iDownstream_Photocell_S1 -- Downstream >> Prorunner; Obj
O      #iBeginInfeedConveyor_B6   #iBeginInfeedConveyor_B6
O      #iEndInfeedConveyor_B7     #iEndInfeedConveyor_B7 -- Photocell on infeedconveyor
O      #iProductOnOutfeed_B8      #iProductOnOutfeed_B8 -- Dropoff position diagonal occup
O      #iProductLeftOutfeed_B9    #iProductLeftOutfeed_B9 -- Dropoff position straight occu
R      #oProrunnerIsEmpty          #oProrunnerIsEmpty -- Vertical conveyor is empty

```

Network: 5	Monitoring outfeed conveyor for next carrier
------------	--

<p>When a product is put on the outfeedconveyor, it must be put in photocel -B8. this photocell sets a memory bit, so the software knows there is a product</p>
---

present on the outfeed conveyor even if this product is not in the photocell -B8 anymore.

The product will travel in photocell -B9, and on a negative edge of this photocell the memory bit is reset.

When the next carrier of the Prorunner activates sensor -B3 a time frame will start. when this time is past and the memory bit ProductOnOutfeed is still present, the Prorunner must stop until this memory bit is reset.

A	#iStartTimeFrame_B3	#iStartTimeFrame_B3 -- Sensor Forksecurity for outfeed
FP	#sHM_FP.CheckDropOffPosition	#sHM_FP.CheckDropOffPosition
=	#tFP.CheckDropOffPosition	#tFP.CheckDropOffPosition
CALL	#sDelayCheckOutfeed	#sDelayCheckOutfeed
IN:=	#tFP.CheckDropOffPosition	#tFP.CheckDropOffPosition
PT:=	#iTimeCheckOutfeed	#iTimeCheckOutfeed -- Time off delay on B3, when time is past outfeed must be empty, if not lift stop
Q	:=#sDelayCheckOutfeed.Q	#sDelayCheckOutfeed.Q
ET:=		
A	#sDelayCheckOutfeed.Q	#sDelayCheckOutfeed.Q
FN	#sHM_FN.TimeFrameOutfeed	#sHM_FN.TimeFrameOutfeed
=	#tFN.TimeFrameOutfeed	#tFN.TimeFrameOutfeed
A	#iProductLeftOutfeed_B9	#iProductLeftOutfeed_B9 -- Dropoff position straight occupied
FN	#sHM_FN.ProductLeftOutfeed_B9	#sHM_FN.ProductLeftOutfeed_B9
=	#tFN.ProductLeftOutfeed_B9	#tFN.ProductLeftOutfeed_B9
A	#iProductOnOutfeed_B8	#iProductOnOutfeed_B8 -- Dropoff position diagonal occupied
O	#iProductLeftOutfeed_B9	#iProductLeftOutfeed_B9 -- Dropoff position straight occupied
S	#sProductOnOutfeed	#sProductOnOutfeed
AN	#iProductOnOutfeed_B8	#iProductOnOutfeed_B8 -- Dropoff position diagonal occupied
A	#tFN.ProductLeftOutfeed_B9	#tFN.ProductLeftOutfeed_B9
R	#sProductOnOutfeed	#sProductOnOutfeed
A	#sProductOnOutfeed	#sProductOnOutfeed
A	#tFN.TimeFrameOutfeed	#tFN.TimeFrameOutfeed
S	#sStopLiftProdOnOutfeed	#sStopLiftProdOnOutfeed
AN	#sProductOnOutfeed	#sProductOnOutfeed
R	#sStopLiftProdOnOutfeed	#sStopLiftProdOnOutfeed
AN	#sEmptyVerticalConveyor	#sEmptyVerticalConveyor
A	#oOutfeedConveyor	#oOutfeedConveyor -- Outfeed conveyor in Prorunner
A	#sProductOnOutfeed	#sProductOnOutfeed
A	#tFN.TimeFrameOutfeed	#tFN.TimeFrameOutfeed
S	#oFaultProductOnOutfeed	#oFaultProductOnOutfeed -- Fault product on outfeed conveyor present
A	#iReset	#iReset -- Reset command for failures
O(		
AN	#sProductOnOutfeed	#sProductOnOutfeed
AN	#iProductOnOutfeed_B8	#iProductOnOutfeed_B8 -- Dropoff position diagonal occupied
AN	#iProductLeftOutfeed_B9	#iProductLeftOutfeed_B9 -- Dropoff position straight occupied
)		
R	#oFaultProductOnOutfeed	#oFaultProductOnOutfeed -- Fault product on outfeed conveyor present

Network: 6	Delay rotation when changing rotation CW or CCW
------------	---

```
AN      #oProrunnerForward      #oProrunnerForward -- Forward  vertical conveyor
AN      #oProrunnerReverse      #oProrunnerReverse -- Reverse vertical conveyor
=      #tSignalToDelay          #tSignalToDelay

CALL    #sDelayForwardReverse    #sDelayForwardReverse
IN:=#tSignalToDelay              #tSignalToDelay
PT:=T#1S
Q      :=#tEnableMotorOn        #tEnableMotorOn
ET:=

O      #tEnableMotorOn          #tEnableMotorOn
O      #oProrunnerForward      #oProrunnerForward -- Forward  vertical conveyor
=      #tEnableMotorForward      #tEnableMotorForward

O      #tEnableMotorOn          #tEnableMotorOn
O      #oProrunnerReverse      #oProrunnerReverse -- Reverse vertical conveyor
=      #tEnableMotorReverse      #tEnableMotorReverse
```

Network: 7	Motor verticaal conveyor
------------	--------------------------

```
A      #tAutomatic              #tAutomatic
AN      #oFaultProductInfeedtime #oFaultProductInfeedtime -- Fault Overtime box inserted
AN      #oFaultProductPosition   #oFaultProductPosition -- Fault product position on infeed
AN      #sStopLiftProdOnOutfeed  #sStopLiftProdOnOutfeed
A      #tEnableMotorForward      #tEnableMotorForward
O

A      #tManual                  #tManual
A      #tEnableMotorForward      #tEnableMotorForward
A      #iManualForward           #iManualForward -- Pushbutton Manual forward (normal run)
=      #oProrunnerForward        #oProrunnerForward -- Forward  vertical conveyor

A      #tManual                  #tManual
A      #tEnableMotorReverse      #tEnableMotorReverse
A      #iManualReverse           #iManualReverse -- Pushbutton Manual reverse
=      #oProrunnerReverse        #oProrunnerReverse -- Reverse vertical conveyor
```

Network: 8	Release new product in Prorunner
------------	----------------------------------

```
A      #iProductRelease_B1       #iProductRelease_B1 -- Sensor infeed product
FP      #sHM_FP.ProductRelease   #sHM_FP.ProductRelease
=      #tFP.Productrelease       #tFP.Productrelease

A      #iBeginInfeedConveyor_B6  #iBeginInfeedConveyor_B6
FP      #sHM_FP.BusyWithInfeed   #sHM_FP.BusyWithInfeed
=      #tFP.BusyWithInfeed       #tFP.BusyWithInfeed

A      #iDownstream_Photocell_S1 #iDownstream_Photocell_S1 -- Downstream >> Prorunner; Object ready for infeed (photocell S1 covered)
FN      #sHM_FN.ProductReleased  #sHM_FN.ProductReleased
=      #tFN.ProductReleased      #tFN.ProductReleased

CALL    #sDelayRelease           #sDelayRelease
IN:=#tFP.Productrelease          #tFP.Productrelease
PT:=#iTimeInfeedAllowed         #iTimeInfeedAllowed -- Time of window infeed allowed
Q      :=#sExtendedRelease       #sExtendedRelease
```

ET:=

```
A      #tFN.ProductReleased      #tFN.ProductReleased
JCN    pkvp
L      #sCounters.CountedProducts #sCounters.CountedProducts
+      L#1
T      #sCounters.CountedProducts #sCounters.CountedProducts
```

pkvp: NOP 0

Network: 9	Signal to downstream infeed conveyor
------------	--------------------------------------

```
A      #tAutomatic      #tAutomatic
A      #oInfeedConveyor  #oInfeedConveyor -- Infeed conveyor in Prorunner
A      #sExtendedRelease #sExtendedRelease
AN     #sProductInserted #sProductInserted
AN     #iEndInfeedConveyor_B7 #iEndInfeedConveyor_B7 -- Photocell on infeedconveyor
O
A      #tAutomatic      #tAutomatic
AN     #iDownstream_Photocell_S1 #iDownstream_Photocell_S1 -- Downstream >> Prorunner; Object ready for infeed (photocell S1 covered)
=      #oDownstream_Ready  #oDownstream_Ready -- Prorunner >> Downstream; Ready for object infeed
```

Network: 10	Motor infeed conveyor
-------------	-----------------------

```
CALL  #sTimeDelayInfeed_B7      #sTimeDelayInfeed_B7
      IN:=#iEndInfeedConveyor_B7 #iEndInfeedConveyor_B7 -- Photocell on infeedconveyor
      PT:=#iTimeProductOnInfeed  #iTimeProductOnInfeed -- Time for delay on B7
      Q :=#sTimeDelayInfeed_B7.Q #sTimeDelayInfeed_B7.Q
      ET:=
A      #tAutomatic      #tAutomatic
AN     #oProrunnerIsEmpty  #oProrunnerIsEmpty -- Vertical conveyor is empty
A(
ON     #sTimeDelayInfeed_B7.Q #sTimeDelayInfeed_B7.Q
O      #iBeginInfeedConveyor_B6 #iBeginInfeedConveyor_B6
)
AN     #oFaultProductInfeedtime #oFaultProductInfeedtime -- Fault Overtime box inserted
AN     #oFaultProductPosition  #oFaultProductPosition -- Fault product position on infeed
=      #oInfeedConveyor        #oInfeedConveyor -- Infeed conveyor in Prorunner
```

Network: 11	Monitoring product infeed time
-------------	--------------------------------

```
A      #tAutomatic      #tAutomatic
A      #tFN.ProductReleased #tFN.ProductReleased
S      #sProductInserted  #sProductInserted

A      #sProductInserted  #sProductInserted
A(
A      #iEndInfeedConveyor_B7 #iEndInfeedConveyor_B7 -- Photocell on infeedconveyor
AN     #iBeginInfeedConveyor_B6 #iBeginInfeedConveyor_B6
O
A      #tFP.Reset      #tFP.Reset
A      #oFaultProductInfeedtime #oFaultProductInfeedtime -- Fault Overtime box inserted
)
R      #sProductInserted  #sProductInserted
R      #sInfeedBusy      #sInfeedBusy
```



```
CALL #sTimeMonitorInfeed      #sTimeMonitorInfeed
IN:=#sProductInserted        #sProductInserted
PT:=#iTimeMonitorInfeed      #iTimeMonitorInfeed -- Time monitoring of product infeed
Q :=#sTimeMonitorInfeed.Q    #sTimeMonitorInfeed.Q
ET:=

A      #sTimeMonitorInfeed.Q   #sTimeMonitorInfeed.Q
S      #oFaultProductInfeedtime #oFaultProductInfeedtime -- Fault Overtime box inserted
```

Network: 12	Monitoring product position
-------------	-----------------------------

```
A      #tAutomatic            #tAutomatic
A      #oDownstream_Ready     #oDownstream_Ready -- Prorunner >> Downstream; Ready for o
                                bject infeed
A      #tFP.BusyWithInfeed    #tFP.BusyWithInfeed
S      #sInfeedBusy           #sInfeedBusy

A      #sInfeedBusy           #sInfeedBusy
A(
A      #iEndInfeedConveyor_B7 #iEndInfeedConveyor_B7 -- Photocell on infeedconveyor
AN     #iBeginInfeedConveyor_B6 #iBeginInfeedConveyor_B6
O
A      #tFP.Reset             #tFP.Reset
A      #oFaultProductPosition #oFaultProductPosition -- Fault product position on infeed
)
R      #sInfeedBusy           #sInfeedBusy

A      #tAutomatic            #tAutomatic
AN     #sInfeedBusy           #sInfeedBusy
A      #iBeginInfeedConveyor_B6 #iBeginInfeedConveyor_B6
S      #oFaultProductPosition #oFaultProductPosition -- Fault product position on infeed
```

Network: 13	Motor outfeed conveyor
-------------	------------------------

```
A      #tAutomatic            #tAutomatic
AN     #oProrunnerIsEmpty     #oProrunnerIsEmpty -- Vertical conveyor is empty
A      #iUpstream_Ready       #iUpstream_Ready -- Upstream >> Prorunner; Conveyor is running
                                and ready for product
=      #oOutfeedConveyor      #oOutfeedConveyor -- Outfeed conveyor in Prorunner
```

Network: 14	Monitoring product release and time frame sensor
-------------	--

```
A      #oProrunnerForward     #oProrunnerForward -- Forward vertical conveyor
A      #tAutomatic            #tAutomatic
AN     #iProductRelease_B1    #iProductRelease_B1 -- Sensor infeed product
AN     #oFaultProdReleaseSensor #oFaultProdReleaseSensor -- Fault sensor product relea
                                se (sensor not triggered)
=      #sTimeCheckReleaseSensor.IN #sTimeCheckReleaseSensor.IN

CALL #sTimeCheckReleaseSensor #sTimeCheckReleaseSensor
IN:=#sTimeCheckReleaseSensor.IN #sTimeCheckReleaseSensor.IN
PT:=#iTimeCheckSensors        #iTimeCheckSensors -- Delay check sensors in 0,1 sec.
Q :=#sTimeCheckReleaseSensor.Q #sTimeCheckReleaseSensor.Q
ET:=

A      #sTimeCheckReleaseSensor.Q #sTimeCheckReleaseSensor.Q
S      #oFaultProdReleaseSensor #oFaultProdReleaseSensor -- Fault sensor product relea
                                se (sensor not triggered)
```

```

A      #oProrunnerForward      #oProrunnerForward -- Forward vertical conveyor
A      #tAutomatic              #tAutomatic
AN     #iStartTimeFrame_B3      #iStartTimeFrame_B3 -- Sensor Forksecurity for outfeed
AN     #oFaultTimeFrameSensor  #oFaultTimeFrameSensor -- Fault Time frame sensor (sen
=      #sTimeCheckTimeFrameSenso.IN  #sTimeCheckTimeFrameSenso.IN
                                           sor not triggered)

CALL   #sTimeCheckTimeFrameSenso  #sTimeCheckTimeFrameSenso
IN:=#sTimeCheckTimeFrameSenso.IN  #sTimeCheckTimeFrameSenso.IN
PT:=#iTimeCheckSensors            #iTimeCheckSensors -- Delay check sensors in 0,1 sec.
Q :=#sTimeCheckTimeFrameSenso.Q  #sTimeCheckTimeFrameSenso.Q
ET:=

A      #sTimeCheckTimeFrameSenso.Q  #sTimeCheckTimeFrameSenso.Q
S      #oFaultTimeFrameSensor      #oFaultTimeFrameSensor -- Fault Time frame sensor (sen
                                           sor not triggered)

```

Network: 15      check bottom sprocket for rotation

Fist mak a positive and negative edge of the sensor on the bottom sprocket.  
The monitoring can be delayed at the start of the motor for a variable time.  
The monitoring can be disabled if time is 0 ms.  
When the monitoring is enabled the sensor resets the timer at every edge.  
When the time is past there is a fault warning and the Prorunner will stop rotating.

```

A      #iMonitorRotation_B5      #iMonitorRotation_B5 -- Sensor on bottom sprocket (Not s
FP     #sHM_FP.CheckRotation      #sHM_FP.CheckRotation
=      #tFP.CheckRotation        #tFP.CheckRotation

A      #iMonitorRotation_B5      #iMonitorRotation_B5 -- Sensor on bottom sprocket (Not s
FN     #sHM_FN.CheckRotation      #sHM_FN.CheckRotation
=      #tFN.CheckRotation        #tFN.CheckRotation

O      #oProrunnerForward        #oProrunnerForward -- Forward vertical conveyor
O      #oProrunnerReverse        #oProrunnerReverse -- Reverse vertical conveyor
=      #tProrunnerRotating      #tProrunnerRotating

CALL   #sTimeDelayMonitotRotatio  #sTimeDelayMonitotRotatio
IN:=#tProrunnerRotating          #tProrunnerRotating
PT:=#iTimeDelayMonitorRotatio    #iTimeDelayMonitorRotatio -- Time to delay monitor rota
Q :=#sTimeDelayMonitotRotatio.Q  #sTimeDelayMonitotRotatio.Q
ET:=

A (
L      #iTimeMonitorRotation      #iTimeMonitorRotation -- Time for monitoring of rotatio
L      T#0MS                      n bottom sprocket (0 = disabled)
<>I
)
A      #sTimeDelayMonitotRotatio.Q  #sTimeDelayMonitotRotatio.Q
=      #tMonitorRotatioEnabled    #tMonitorRotatioEnabled

A      #tProrunnerRotating        #tProrunnerRotating
A      #tMonitorRotatioEnabled    #tMonitorRotatioEnabled
A      #iEnableElevator          #iEnableElevator -- Enable clearence emergengystop, t
                                           hermical etc.

AN     #oFaultMonitorRotation    #oFaultMonitorRotation -- Fault motor rotation
AN     #iReset                  #iReset -- Reset command for failures
AN     #tFP.CheckRotation        #tFP.CheckRotation

```

AN	#tFN.CheckRotation	#tFN.CheckRotation
=	#sTimeMonitorRotation.IN	#sTimeMonitorRotation.IN
CALL	#sTimeMonitorRotation	#sTimeMonitorRotation
	IN:=#sTimeMonitorRotation.IN	#sTimeMonitorRotation.IN
	PT:=#iTimeMonitorRotation	#iTimeMonitorRotation -- Time for monitoring of rotation bottom sprocket(0 = disabled)
	Q :=#sTimeMonitorRotation.Q	#sTimeMonitorRotation.Q
	ET:=	
A	#sTimeMonitorRotation.Q	#sTimeMonitorRotation.Q
S	#oFaultMonitorRotation	#oFaultMonitorRotation -- Fault motor rotation

Network: 16

Reset failures

A	#tFP.Reset	#tFP.Reset
R	#oFaultProdReleaseSensor	#oFaultProdReleaseSensor -- Fault sensor product release (sensor not triggered)
R	#oFaultTimeFrameSensor	#oFaultTimeFrameSensor -- Fault Time frame sensor (sensor not triggered)
R	#oFaultProductOnOutfeed	#oFaultProductOnOutfeed -- Fault product on outfeed conveyor present
R	#oFaultProductInfeedtime	#oFaultProductInfeedtime -- Fault Overtime box inserted
R	#oFaultProductPosition	#oFaultProductPosition -- Fault product position on infeed
R	#oFaultMonitorRotation	#oFaultMonitorRotation -- Fault motor rotation