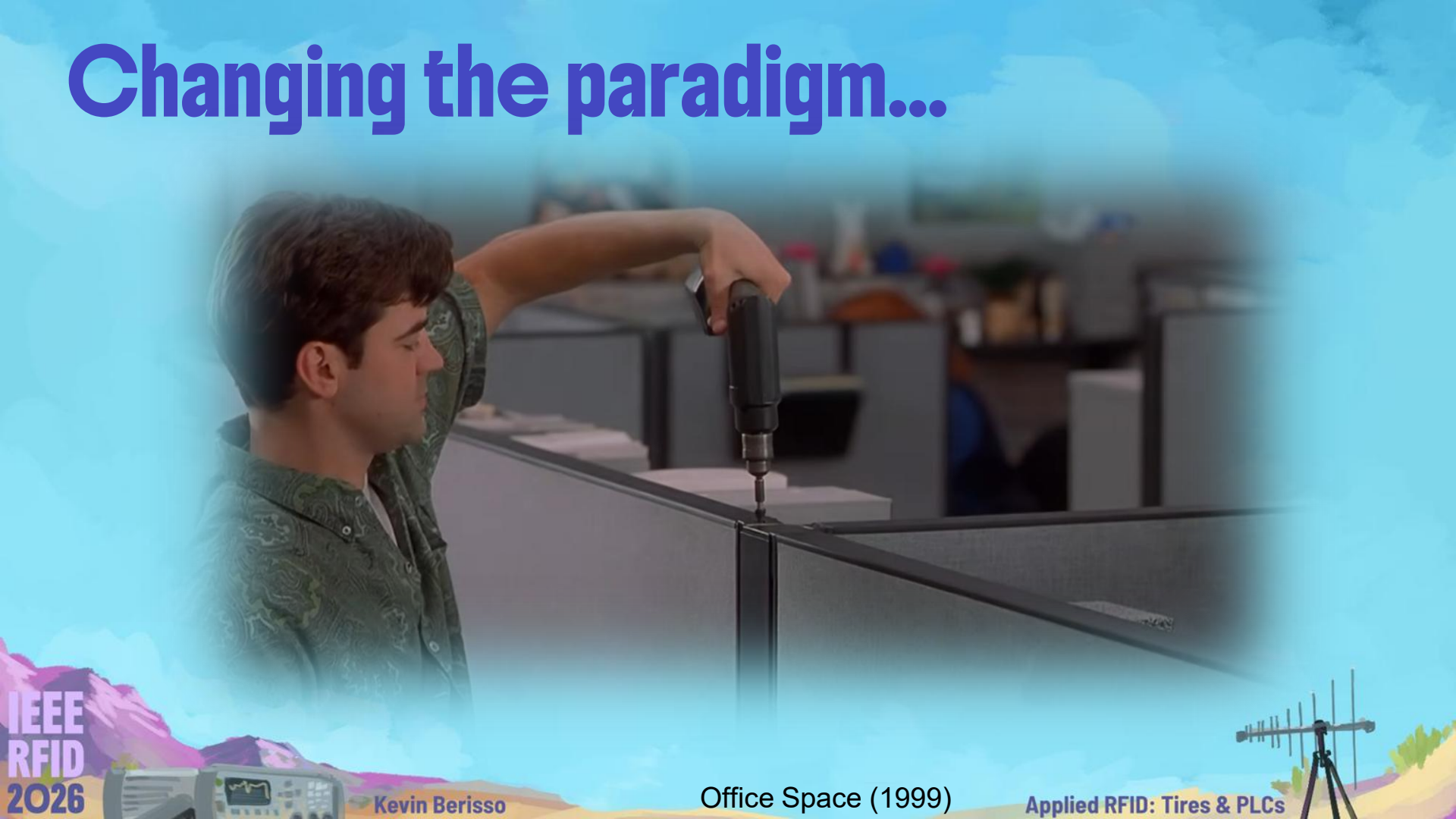
Applied RFID: PLCs

IEEE RFID 2026 Tutorial presented by **Kevin Berisso**

Changing the paradigm...



Office Space (1999)

RFID Use Cases

https://ww

https://www.radley.com/material-traceability/rfid/

id-tires/

This is a short POV video of our new RFID tunnel solution that was generously donated by Avery Dennison...

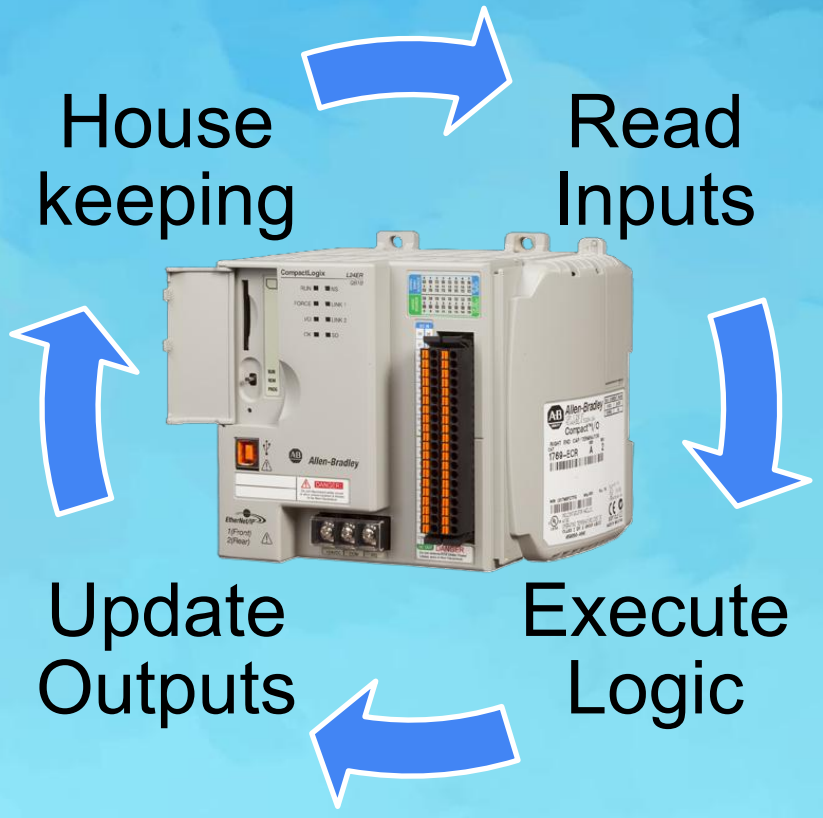
PLCs

What are PLCs

https://ladderlogicworld.com/relay-logic-vs-ladder-logic/

PLCs

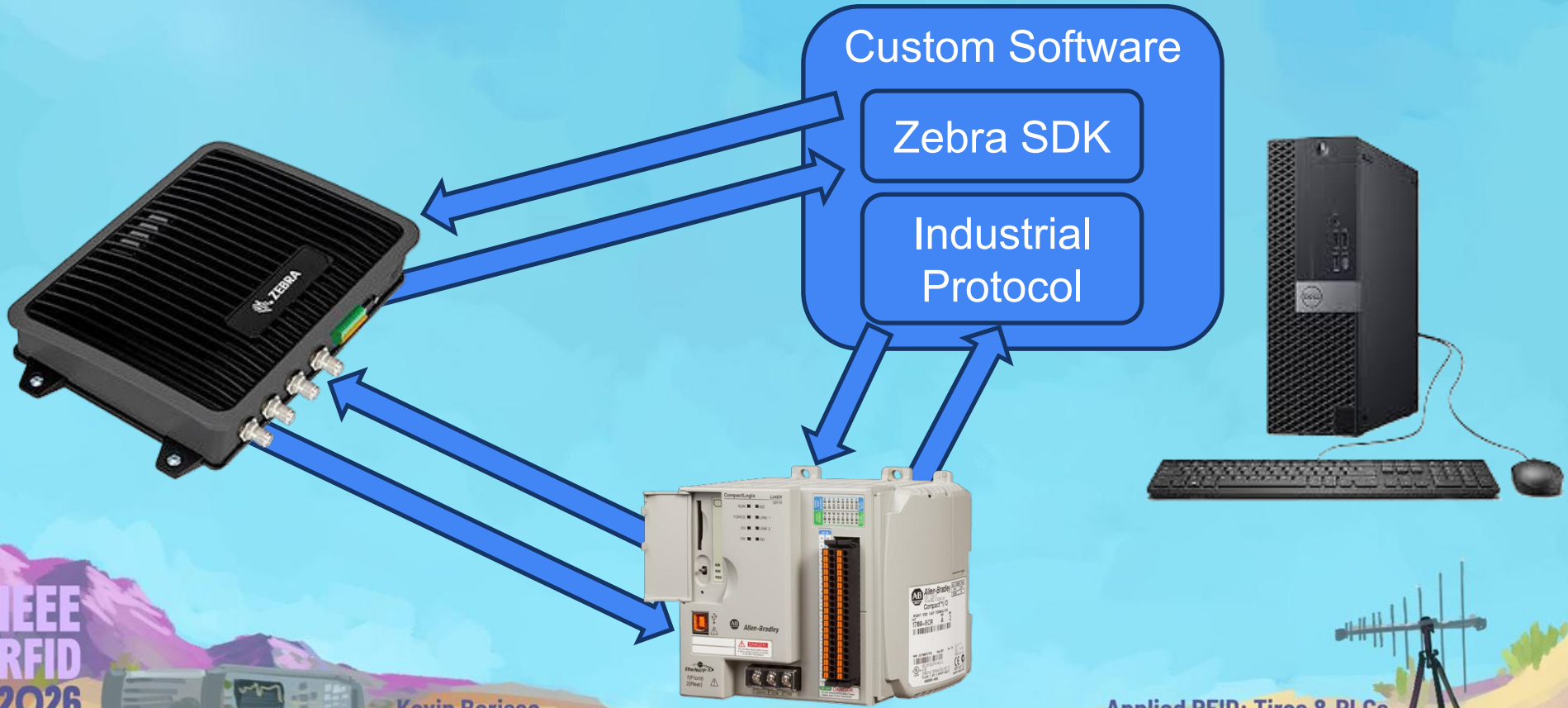
1. Read input table
2. Execute logic
3. Write physical outputs
4. Process housekeeping functions, watchdogs, etc.



PLCs – Speed Wins!

Parts per Minute	Parts per Second	ms per Part
60	1.0	1000
100	1.6	600
200	3.3	300
400	6.7	150
800	13.3	75
1000	16.7	60
6000	100.0	10

PLCs - System Complexity



PLCs

Logic	Main Program (µs)			Main Task (ms)		
	L23	L24	L306	L23	L24	L306
Empty Project	128	33	54	1.894	1.790	0.065
GSV only	290	103	64	2.304	1.750	0.074
FFL (100) + GSV	316	102	65	2.230	1.787	0.074

PLCs

https://en.wikipedia.org/wiki/Allen-Bradley

LADDER DIAGRAM(LD)

SEQUENTIAL FUNCTION CHART(SFC)

PLC Programming Languages

FUNCTION BLOCK DIAGRAM (FBD)

INSTRUCTION LIST(IL)

STRUCTURED TEXT(ST)

PLCs

PLCs

Logic	Main Program (µs)			Main Task (ms)		
	L23	L24	L306	L23	L24	L306
Empty Project	128	33	54	1.894	1.790	0.065
GSV only	290	103	64	2.304	1.750	0.074
FFL (100) + GSV	316	102	65	2.230	1.787	0.074
Extract SN	256	149	72	2.630	1.757	0.121

Industrial RFID solutions

BALLUFF

PEPPERL+FUCHS

RFID, INC.

TURCK

Rockwell Automation

ZEBRA

SICK

Sensor Intelligence.

Industrial RFID solutions

Zebra

New licensing model
Supported functions

Zebra

ZEBRA
Home
Status
Operation Statistics
Configure Reader
Read Tags
Communication
Zebra IoT Connector
LLRP
SNMP
Wireless
Services
Serial Port Configuration
802.1x EAP
Date Time
IP Sec
Change Password
GPIO
Applications
Profiles
Firmware
System Log
Diagnostics
Shutdown
Logout
Reader Communication Parameters
Configure Network Settings
Web Server: HTTPS
Shell: SSH
Disable IPV6 Stack:
Receive RA packets:
Avahi: Enable
RDMP: Enable
Netbios: Disable
Node Server Run Status:
Network Connect App: ZebraEthernetipApp
Network Connect App Autostart:
Network Connect App Status:
Set Properties

Zebra

RSLinx Classic Gateway - [RSWho - 1]

File Edit View Communications Station DDE/OPC Security Window Help

Autobrowse Refresh Browsing - node 141.225.160.26 found

AB_ETHIP-1, Ethernet

27.0.0.1], FactoryTalk Linx - Desktop, BERISSO-DELL

ryTalk Linx - Desktop, HCOE-HL8NM34

, 1769-L23E-QBFC1 Ethernet Port, 1769-L23E-QBFC1 Ethernet Port

, 1769-L24ER-QB1B LOGIX5324ER, 1769-L24ER-QB1B/A LOGIX5324ER

2198-H015-ERS, 2198-H015-ERS

Address	Private Address	Device Type	Online Name	Status
		Unrecognized Device	DataMan DM300 Reader	OK
		Unrecognized Device	In-Sight 8401	
		Unrecognized Device	HOCE-4M8NM34	OK
		Unrecognized Device	In-Sight 7000 Series	OK
		Unrecognized Device	HCOE-BM8NM34	

Address	Private Address	Device Type
.66		Unrecognized Device
.182		Unrecognized Device
.159		FX9600
.250		FX9600
.107		FX9600
.150		FX9600
.227		FX9600
.196		FX9600
.140		FactoryTalk Linx - Desktop
	127.0.0.1	

C1000micro EtherNet/IP (TM)
C1000micro EtherNet/IP (TM)
C1000micro EtherNet/IP (TM) OK
C1000micro EtherNet/IP (TM)
OE-BM8NM34 OK
C1000micro EtherNet/IP (TM) OK
taMan DM303 Reader
805MP OK
OE-BM8NM34
nelView-Plus OK
3-L16BWA B/00.00
3-L16AWA B/15.02 OK
0-LC20-20QWB
0-LC20-20QWB OK
802M
802M
802M OK
9600
9600 OK
9600
9600
9600 OK
9600
CE-6M8NM34 OK
RISSO-DELL

IEEE RFID 2026

Kevin Berisso

Applied RFID: Tires & PLCs

Zebra

Module Properties: Local (FX9600 2.001)

- General
- Connection*
- Module Info
- Internet Protocol
- Vendor

General

Type: FX9600 RFID Reader

Vendor: Zebra Technologies

Parent: Local

Name: FX9600

Description:

Module Definition

Change ...

Revision: 2.001

Electronic Keying: Compatible Module

Ethernet Address

- Private Network: 192.168.1.
- IP Address: 141 . 225 . 163 . 250
- Host Name:

Zebra

Module Properties: Local (FX9600 2.001)

- General
- Connection*
- Module Info
- Internet Protocol
- Vendor

Connection

Requested Packet Interval (RPI): 20.0 ms (5.0 - 1000.0)

- [ ] Inhibit Module
- [ ] Major Fault On Controller If Connection Fails While in Run Mode
- [x] Use Unicast Connection over EtherNet/IP

Name	Value	Style	Data Type
AccessPostFilter	{...} {		ACCESS_AND_POST_FILTER
AccessPostFilter.TagPatternA	{...} {		TAG_PATTERN
AccessPostFilter.TagPatternA.BitOffset	0	Decimal	INT
AccessPostFilter.TagPatternA.TagPatternBitCount	0	Decimal	INT
AccessPostFilter.TagPatternA.TagPattern	'' {		STRING36
AccessPostFilter.TagPatternA.TagMask	'' {		STRING36
AccessPostFilter.TagPatternA.TagMaskBitCount	0	Decimal	INT
AccessPostFilter.TagPatternA.MemoryBank	0	Decimal	SINT
AccessPostFilter.TagPatternA.Pad	0	Decimal	SINT
AccessPostFilter.TagPatternB	{...} {		TAG_PATTERN
AccessPostFilter.MatchPattern	0	Decimal	SINT
AccessPostFilter.PeakRSSILowerLimit	0	Decimal	SINT
AccessPostFilter.PeakRSSIUpperLimit	0	Decimal	SINT
AccessPostFilter.PeakRSSIMatchRange	0	Decimal	SINT

Name	Value	Style	Data Type
AccessPostFilter	{...} {		ACCESS_AND_POST_FILTER
Ant1	0	Decimal	BOOL
Ant2	0	Decimal	BOOL
AntennaConfig	{...} {		ANTENNA_CONFIGURATION
AntennaConfig.AntennaID	0	Decimal	SINT
AntennaConfig.Sel	0	Decimal	SINT
AntennaConfig.Session	0	Decimal	SINT
AntennaConfig.Target	0	Decimal	SINT
AntennaConfig.RFModeIndex	0	Decimal	SINT
AntennaConfig.Pad	0	Decimal	SINT
AntennaConfig.Tari	0	Decimal	INT
AntennaConfig.TagPopulation	0	Decimal	INT
AntennaConfig.PowerLevel	0	Decimal	INT

Name	Value	Style	Data Type
FX9600:I1	{...} {		ZT:FX9600_INV_RESP:I1:0
FX9600:I1.ConnectionFaulted	1	Decimal	BOOL
FX9600:I1.StatusMask	{...} {		ZT:RFID_STATUS_MASK:I:0
FX9600:I1.PacketSequenceNumber	23	Decimal	SINT
FX9600:I1.NumberOfTagReports	0	Decimal	SINT
FX9600:I1.TagReports	{...} {		ZT:Zebra_SubStruct_Tag_Report_64:I:0[4]
FX9600:I2	{...} {		ZT:FX9600_ACCESS_RESP:I2:0
FX9600:I2.ConnectionFaulted	1	Decimal	BOOL
FX9600:I2.StatusMask	{...} {		ZT:RFID_STATUS_MASK:I:0
FX9600:I2.PacketSequenceNumber	27	Decimal	SINT
FX9600:I2.ResponseHeader	0	Decimal	SINT
FX9600:I2.TagReport	{...} {		ZT:Zebra_SubStruct_Tag_Report_64:I:0
FX9600:I2.AccessData	{...} {		ZT:Zebra_SubStruct_Access_Data:I:0

Name	Value	Style	Data Type
FX9600:O1	{...} {		ZT:FX9600_CMD_INV:O1:0
FX9600:O1.CommandType	1	Decimal	SINT
FX9600:O1.Handshake	88	Decimal	SINT
FX9600:O1.EnablePreFilter	1	Decimal	SINT
FX9600:O1.EnablePostFilter	0	Decimal	SINT
FX9600:O1.AntennaMask	0	Decimal	DINT
FX9600:O2	{...} {		ZT:FX9600_CMD_ACCESS:O2:0
FX9600:O2.CommandType	0	Decimal	SINT
FX9600:O2.Handshake	0	Decimal	SINT
FX9600:O2.MemoryBank	0	Decimal	SINT
FX9600:O2.EnableAccessFilter	0	Decimal	SINT
FX9600:O2.AntennaMask	0	Decimal	DINT
FX9600:O2.TagEPC	{...} {		ZT:String64_Struct:O:0
FX9600:O2.Password	0	Decimal	DINT
FX9600:O2.ByteOffset	0	Decimal	INT
FX9600:O2.ByteCount	0	Decimal	INT
FX9600:O2.AccessData	{...} {		ZT:Zebra_SubStruct_Access_Data:O:0

▷ Get_AccessPostFilter	{...}	{		MESSAGE
▷ Get_AntennaConf	{...}	{		MESSAGE
▷ Get_EventReport	{...}	{		MESSAGE
▷ Get_GPIOConfig	{...}	{		MESSAGE
Get_Inspection_ID	0		Decimal	BOOL
▷ Get_Prefilter	{...}	{		MESSAGE
▷ Get_ProfileClass	{...}	{		MESSAGE
▷ Get_ProfileList	{...}	{		MESSAGE
▷ Get_ReaderCaps	{...}	{		MESSAGE
▷ GPIOConfig	{...}	{		GPIO_CONFIG
▷ InputAccessResponse	{...}	{		ACCESS_RESPONSE
▷ InputInventoryResponse	{...}	{		INVENTORY_RESPONSE_12B

Read Logic

MOVE	
Source	1
Dest	OutputInventoryCommand.CommandType 1

LT	
Source A	OutputInventoryCommand.Handshake 89
Source B	127

ADD	
Source A	OutputInventoryCommand.Handshake 89
Source B	1
Dest	OutputInventoryCommand.Handshake 89

GT	
Source A	OutputInventoryCommand.Handshake 89
Source B	126

MOVE	
Source	0
Dest	OutputInventoryCommand.Handshake 89

Zebra

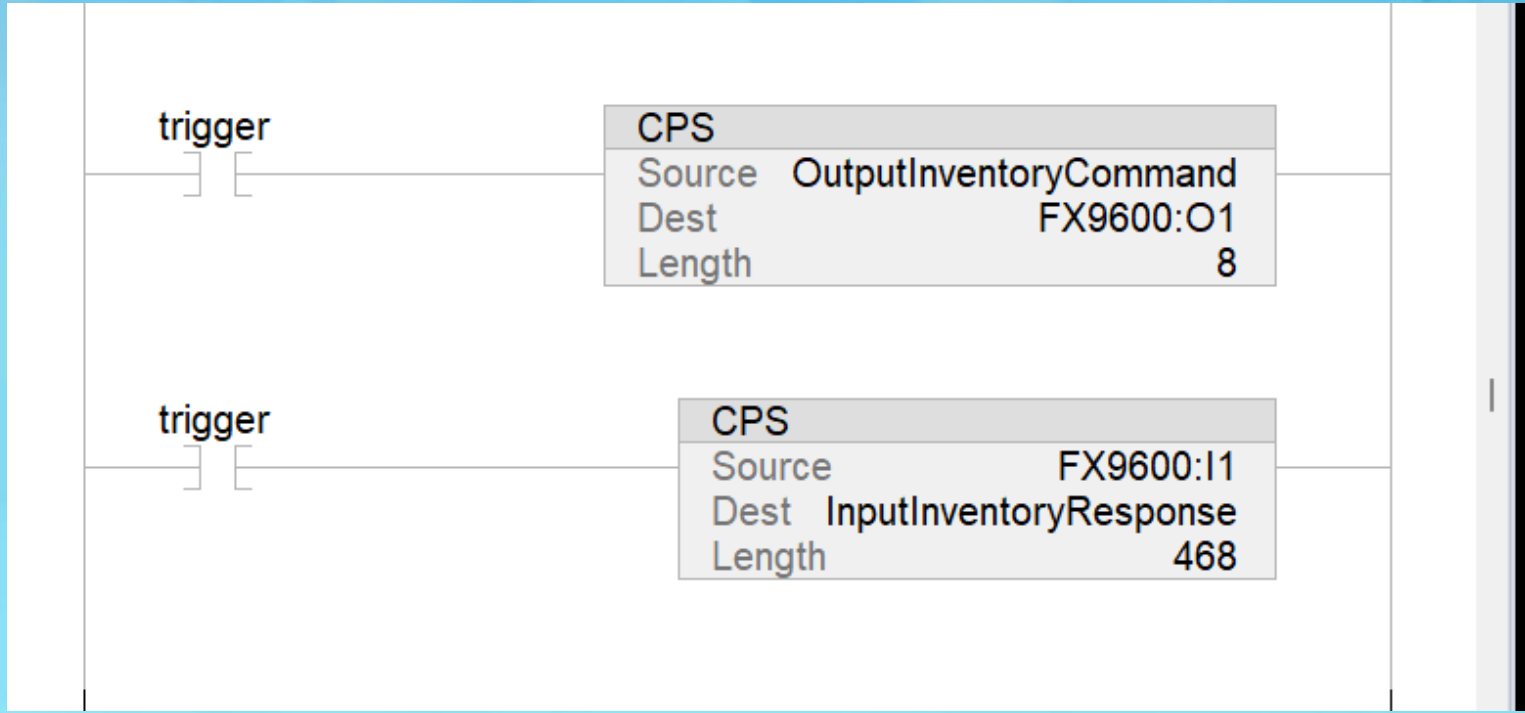
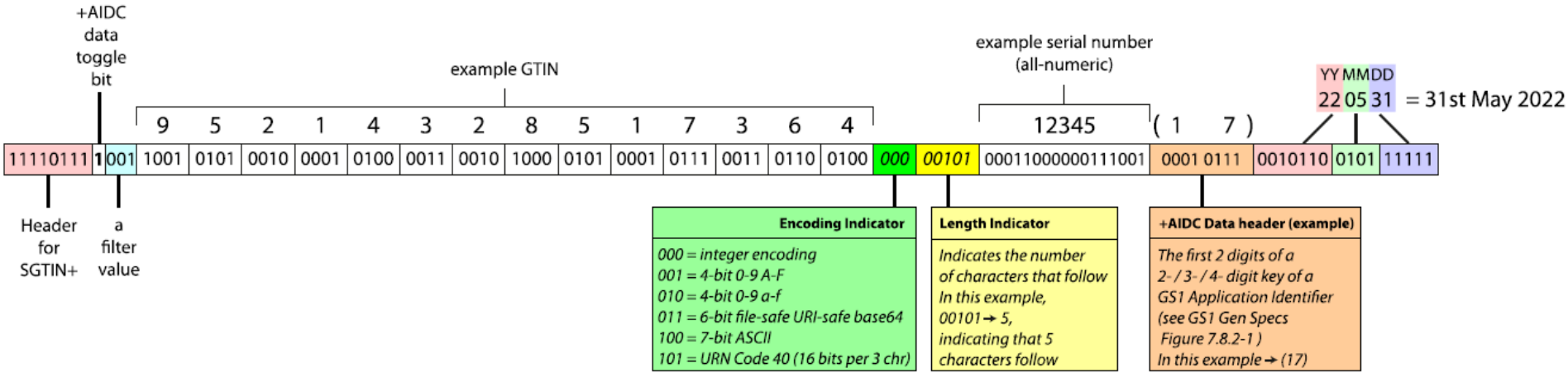

trigger
CPS
Source OutputInventoryCommand
Dest FX9600:O1
Length 8
trigger
CPS
Source FX9600:I1
Dest InputInventoryResponse
Length 468

Table 14-5 SGTIN+ coding table

Scheme	SGTIN+				
GS1 Digital Link URI syntax	https://id.gs1.org/01/{gtin}/21/{serial}				
Total Bits	Up to 216 bits				
Logical Segment	EPC Header	+Data Toggle	Filter	GTIN	Serial Number
Corresponding GS1 AI				(01)	(21)
Logical Segment Bit Count	8	1	3	56	3 bit encoding indicator + 5 bit length indicator + up to 140 bits
Logical Segment Character Count		1 digit (0 or 1)	1 digit (0-7)	14 digits	up to 20 characters
Bit Position (counting up)*	$b_0b_1...b_7$	b_8	$b_9b_{10}b_{11}$	$b_{12}b_{13}...b_{67}$	$b_{68}b_{69}b_{70}...$
Coding	11110111	+AIDC	Fixed-Bit-	Fixed-Length	Variable-length alphanumeric

Zebra

TDS 2.x SGTIN+ discussion

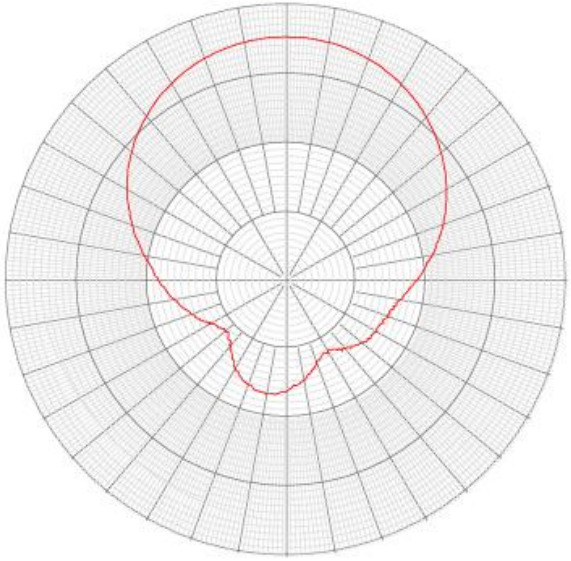


Zebra – Antenna Selection

- Considerations for use in automation
  - Read zone
  - Spacing/gapping between tags
  - Tag velocity
  - Environment

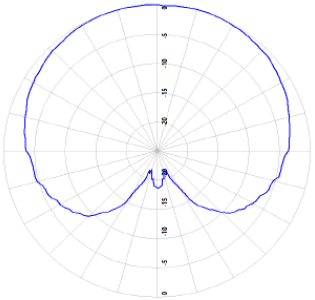
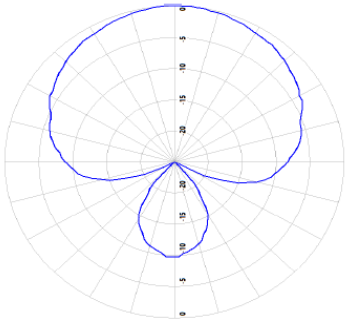
Zebra - Antennas

- Selection for use in automation



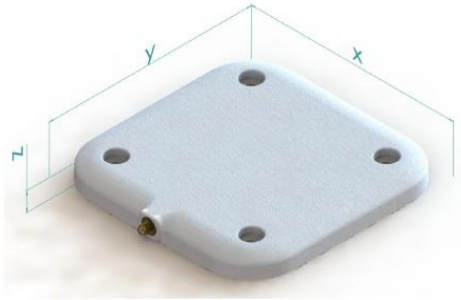
8dBiC, 70˚ beamwidth

Radiating Patterns

A5020CP XZ Plane	A5020CP YZ Plane
	

5dBiC, 105˚ beamwidth

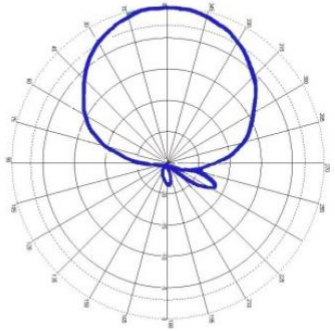
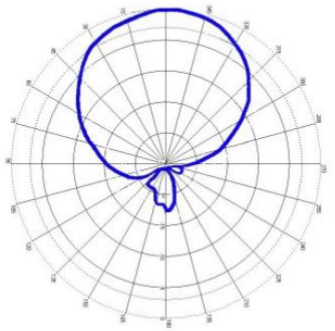
*Azimuth Planes



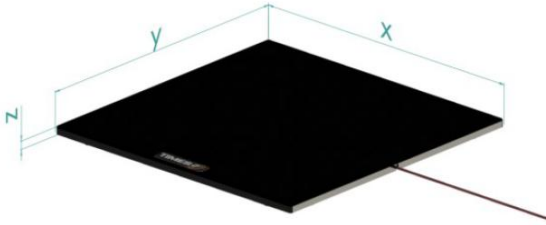
Zebra - Antennas

- Selection for use in automation

Radiation Pattern

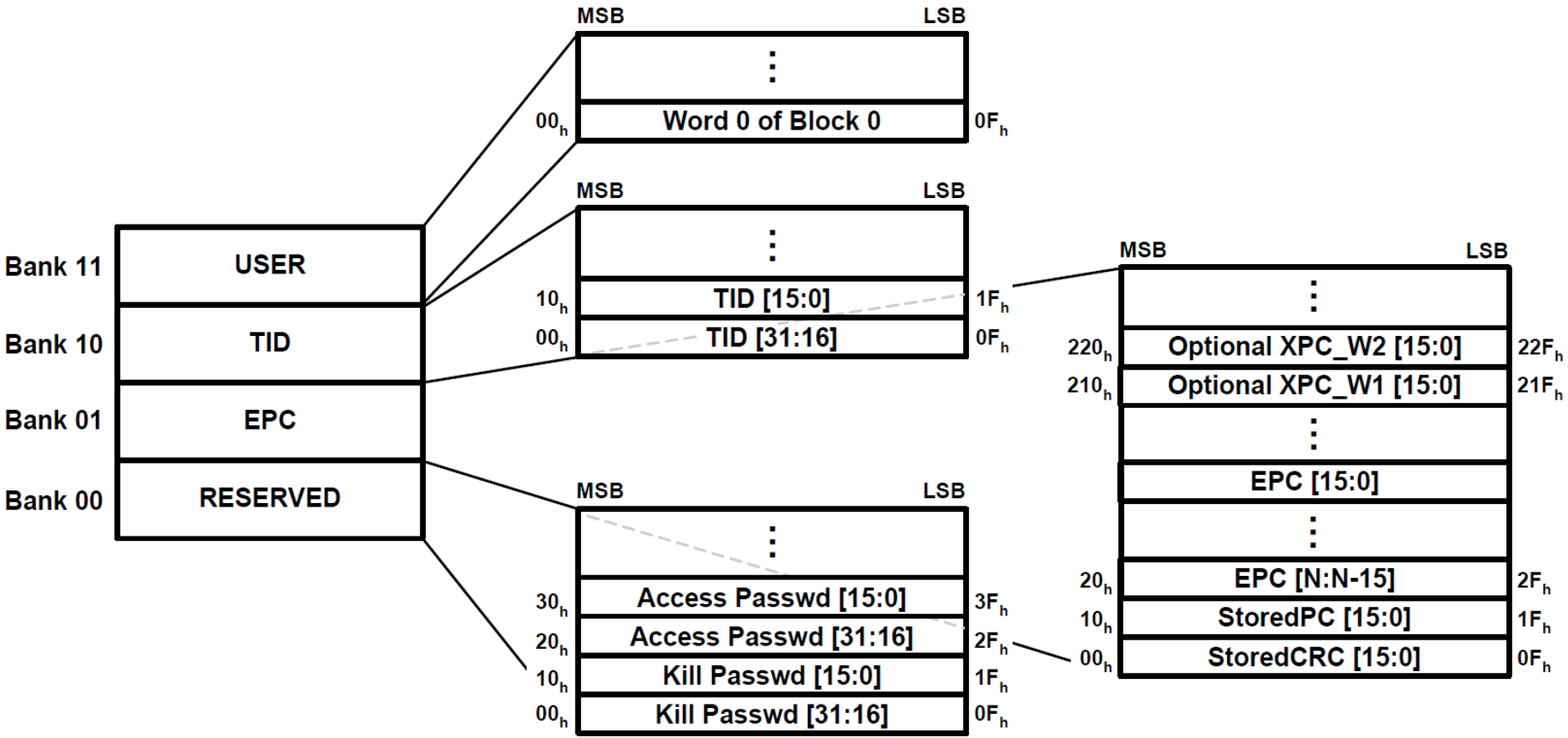
A7030C Y-Z plane	A7030C X-Z plane
	
6dBiC, 65˚ beamwidth	

Azimuth Planes



Zebra - Demo

- **Read Process**
  - Set Power
  - Set Antenna
  - Initiate Read
  - Stop Read


MSB
LSB
Word 0 of Block 0
00h
0Fh
Bank 11
USER
Bank 10
TID
Bank 01
EPC
Bank 00
RESERVED
TID [15:0]
TID [31:16]
10h
1Fh
Access Passwd [15:0]
Access Passwd [31:16]
Kill Passwd [15:0]
Kill Passwd [31:16]
30h
3Fh
20h
2Fh
Optional XPC_W2 [15:0]
Optional XPC_W1 [15:0]
220h
22Fh
210h
21Fh
EPC [15:0]
EPC [N:N-15]
StoredPC [15:0]
StoredCRC [15:0]

Zebra Write

OutputAccessCommand	{...}
OutputAccessCommand.CommandType	2
OutputAccessCommand.Handshake	6
OutputAccessCommand.MemoryBank	1
OutputAccessCommand.EnableAccessFilter	0
OutputAccessCommand.AntennaMask	1
OutputAccessCommand.TagEPC	$827$8573787r7v7$887$8C7'
OutputAccessCommand.Password	0
OutputAccessCommand.ByteOffset	18
OutputAccessCommand.ByteCount	2
OutputAccessCommand.AccessData	{...}
OutputAccessCommand.AccessData.Page...	0
OutputAccessCommand.AccessData.Page...	'$8D7'

+4 bytes for CRC/PCWord

TagList[14].DATA[0]	16#82
TagList[14].DATA[1]	16#37
TagList[14].DATA[2]	16#85
TagList[14].DATA[3]	16#37
TagList[14].DATA[4]	16#33
TagList[14].DATA[5]	16#37
TagList[14].DATA[6]	16#38
TagList[14].DATA[7]	16#37
TagList[14].DATA[8]	16#72
TagList[14].DATA[9]	16#37
TagList[14].DATA[10]	16#76
TagList[14].DATA[11]	16#37
TagList[14].DATA[12]	16#88
TagList[14].DATA[13]	16#37
TagList[14].DATA[14]	16#8a
TagList[14].DATA[15]	16#37
TagList[14].DATA[16]	16#00
TagList[14].DATA[17]	16#00

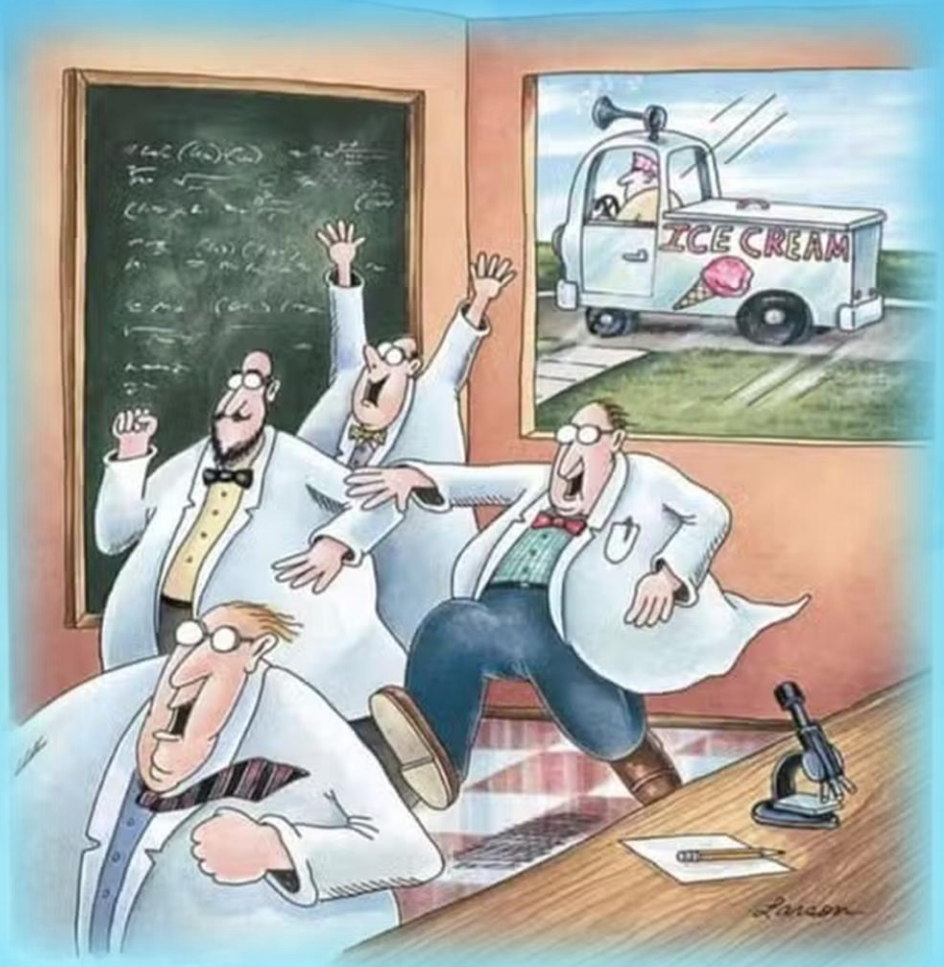
Zebra - Demo

- Read Process
- **Write Process**

- Set Power
- Set Antenna
- Set Filter
- Initiate Write
- Valiate Write

Zebra - PLC File

Questions



Applied RFID: Tires

IEEE RFID 2026 Tutorial presented by **Kevin Berisso**

Tires

TPMS Tech

TPMS Legislation

IEEE RFID 2026

Kevin Berisso

Tires

- **Tire Condition**
  No flat, leaking, exposed cord/belt, tread separation, or sidewall damage. Tread depth ≥ 4/32" (front axle), ≥ 2/32" (all others).
- **Driver Duty**
  Drivers must conduct pre-trip inspections (§392.7), review prior DVIR (§396.13), and personally verify tire pressure before driving.

CMV Tire Pressure Requirements

49 CFR § 393.75
Core Tire Conditions & Inflation

49 CFR § 392.7
Pre-Trip Equipment Inspection

49 CFR § 396.13
Driver Inspection Responsibility

49 CFR Part 396
General Carrier Maintenance

Commerce Clause
Constitutional Authority (Art. I §8)

CVSA OOS Criteria
Enforcement Standard

Tires

- **Penalties**
  Carriers: up to $19,246/violation.

  Drivers: up to $4,812/violation.

  CVSA Out-of-Service order until corrected.

CMV Tire Pressure Requirements

49 CFR § 393.75
Core Tire Conditions & Inflation

49 CFR § 392.7
Pre-Trip Equipment Inspection

49 CFR § 396.13
Driver Inspection Responsibility

49 CFR Part 396
General Carrier Maintenance

Commerce Clause
Constitutional Authority (Art. I §8)

CVSA OOS Criteria
Enforcement Standard

So, ummm… the TPMS tire report???

Possible TPMS solutions

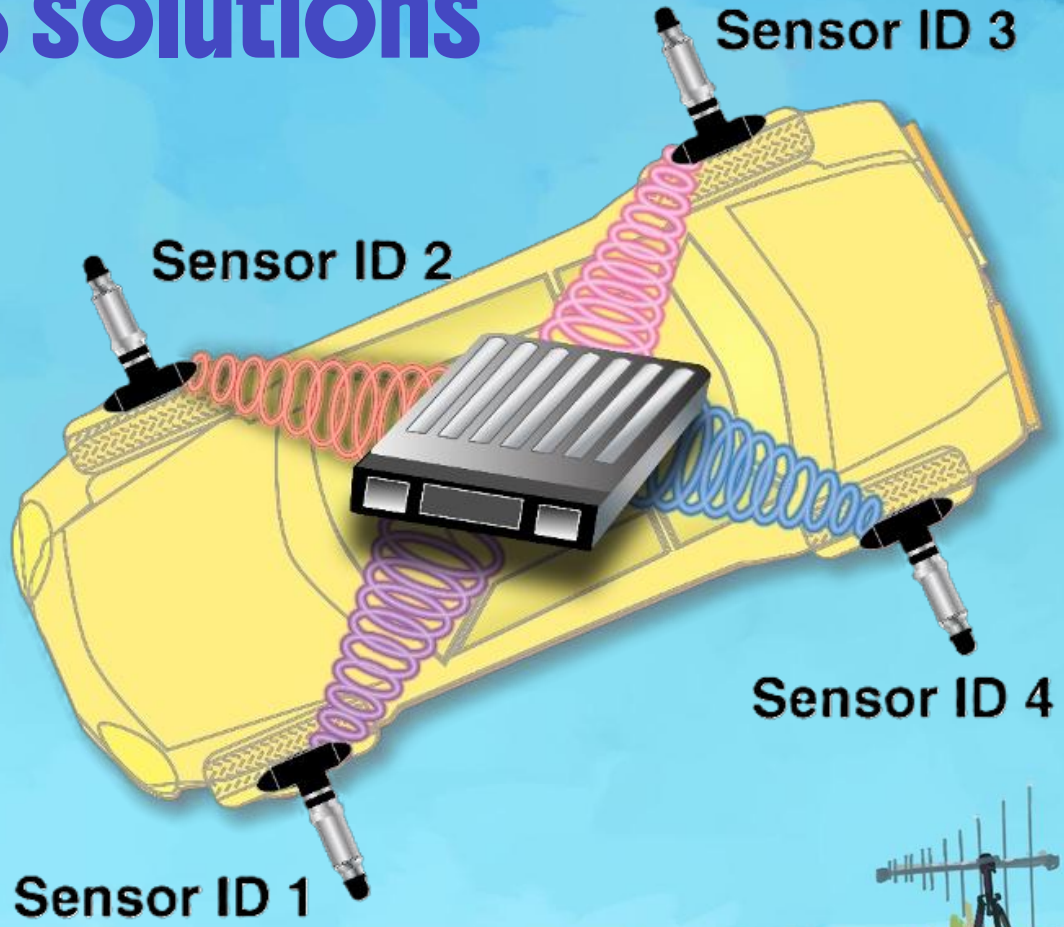
- Current Tech options
  - Stem
  - Anti-lock brake sensor

https://tires.bridgestone.com/en-us/learn/automotive/tire-maintenance/tire-pressure-monitoring-system-how-tpms-works

Possible TPMS solutions

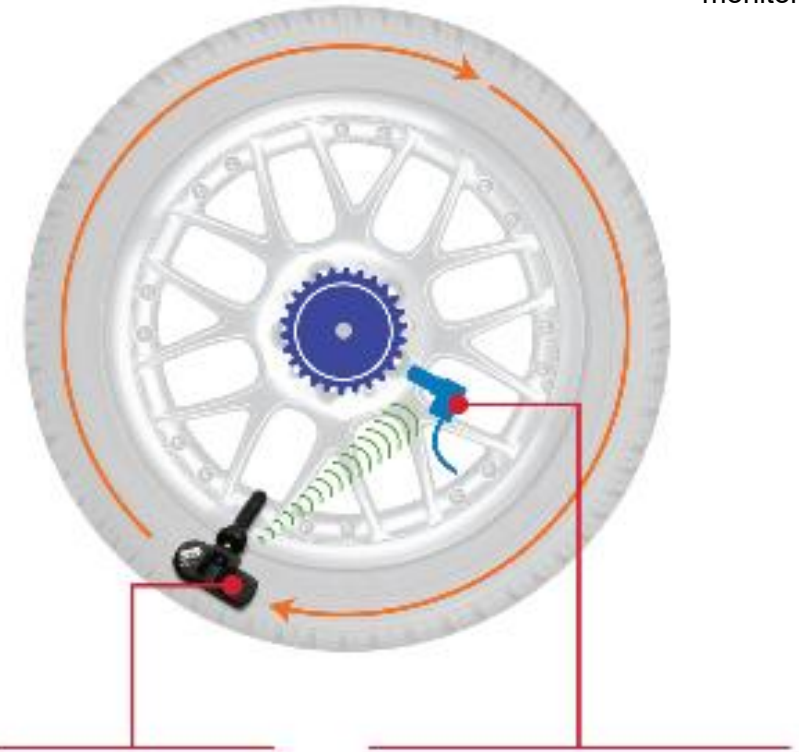
- Current Tech options
  - Stem
  - Anti-lock brake sensor

https://www.standardbrand.com/en/training/standard-insights/a-closer-look-tire-pressure-monitoring-systems-tpms-operation



https://www.standardbrand.com/en/training/standard-insights/a-closer-look-tire-pressure-monitoring-systems-tpms-operation

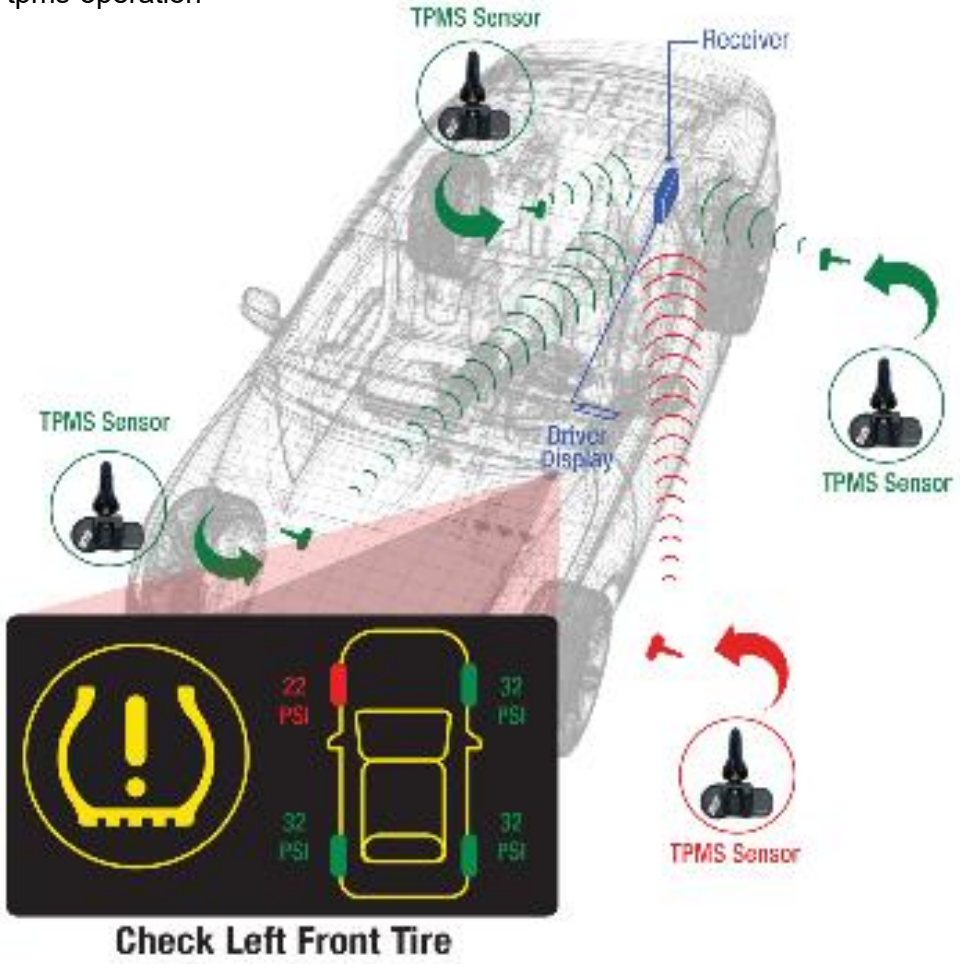
Phase Angle Location (PAL) Technology



TPMS Sensor transmits tire data during rotation

ABS Wheel Speed Sensor signal is used in conjunction with TPMS data

Wireless Auto-Locate (WAL) Technology



Possible TPMS solutions

- Current Tech options
  - Stem caps
  - Embedded sensor
  - Anti-lock brake sensor

- Issue
  - No traceability to a tire

Tires - Grant requirements

- FY2025 High Priority Program-Commercial Motor Vehicle (HPCMV), FM-MHP-0941-26-01-00
- Item 5.8 - Tire Pressure Monitoring System (TPMS)
- AGREEMENT NUMBER: 69A3602641737MHPOTN

Tires – Grant requirements

https://www.seikorfid.com/news/Michelin-RFID-Tire-Tag.html

IEEE RFID 2026

Tires - Grant requirements

- This technology should be capable of performing tire pressure monitoring functions while the CMV [commercial motor vehicle] is both in motion and stationary.

- Ultimately, FMCSA [federal motor carrier safety administrator] seeks data regarding the viability of using this technology to assess the status of CMV tires prior to a subject vehicle's dispatch (i.e., pre-trip inspections), and at any time before returning to the reporting terminal (i.e., when in motion or during NAS inspections).

Tires - Grant requirements

- Research proposals must involve monitoring CMV tire conditions (air pressure, temperature, and/or strain/weight) using sensor-enabled EPC-GEN2 RFID tags embedded in the tire. This includes analyzing the RFID tags in the tire to understand the RF performance, link budget, and sensor accuracy.

Tires - Grant requirements

- Additionally, proposed research must include monitoring RFID tags while equipped CMVs are undergoing stationary inspections (e.g., pre-trip, or NAS) or operating at highway and ramp speeds to evaluate adverse tire conditions.

Tires - Grant requirements

- Finally, the research must include an analysis of the methods of tire anomaly notification (i.e., to drivers, enforcement inspectors, and fleet maintenance personnel) and their efficacy in preventing tire anomalies from graduating to catastrophic failure.

Tires - Grant requirements

- Why is this being done – reading between the lines

  how does this fit into the overall ecosystem and what/why it will ultimately accomplish (imo)

Our Partners

asygn

midmo

SensThys

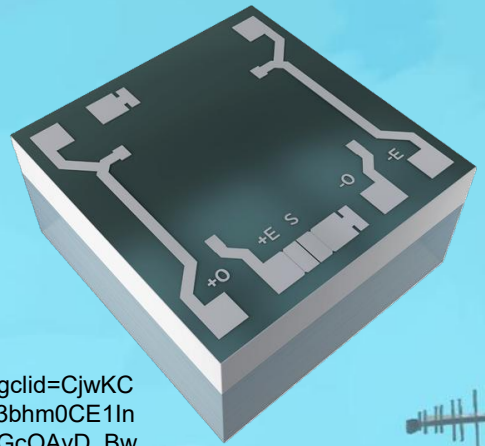
Voyantic

??
??

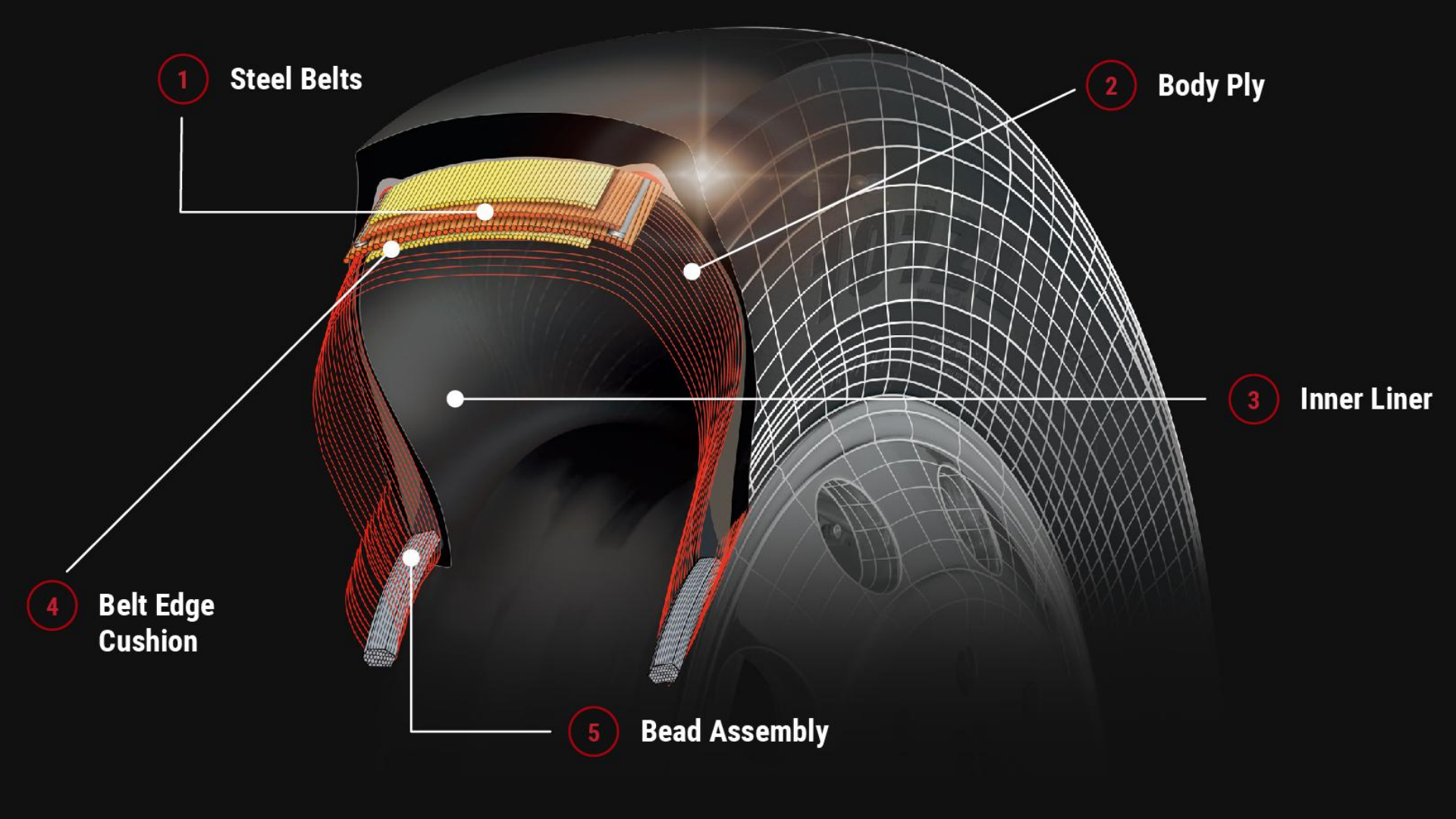
Our path

- Sensors
  - Asgyn chips - Asygn AS3212 chip
  - Pressure – Merit Sensor S-Series MEMS silicon pressure sensor
    5 to 300psi / 0.34 to 21 bar / 34.5 to 2,068 kPa
    -40°C-150°C
    1.5 mm x 1.5 mm x 0.9 mm

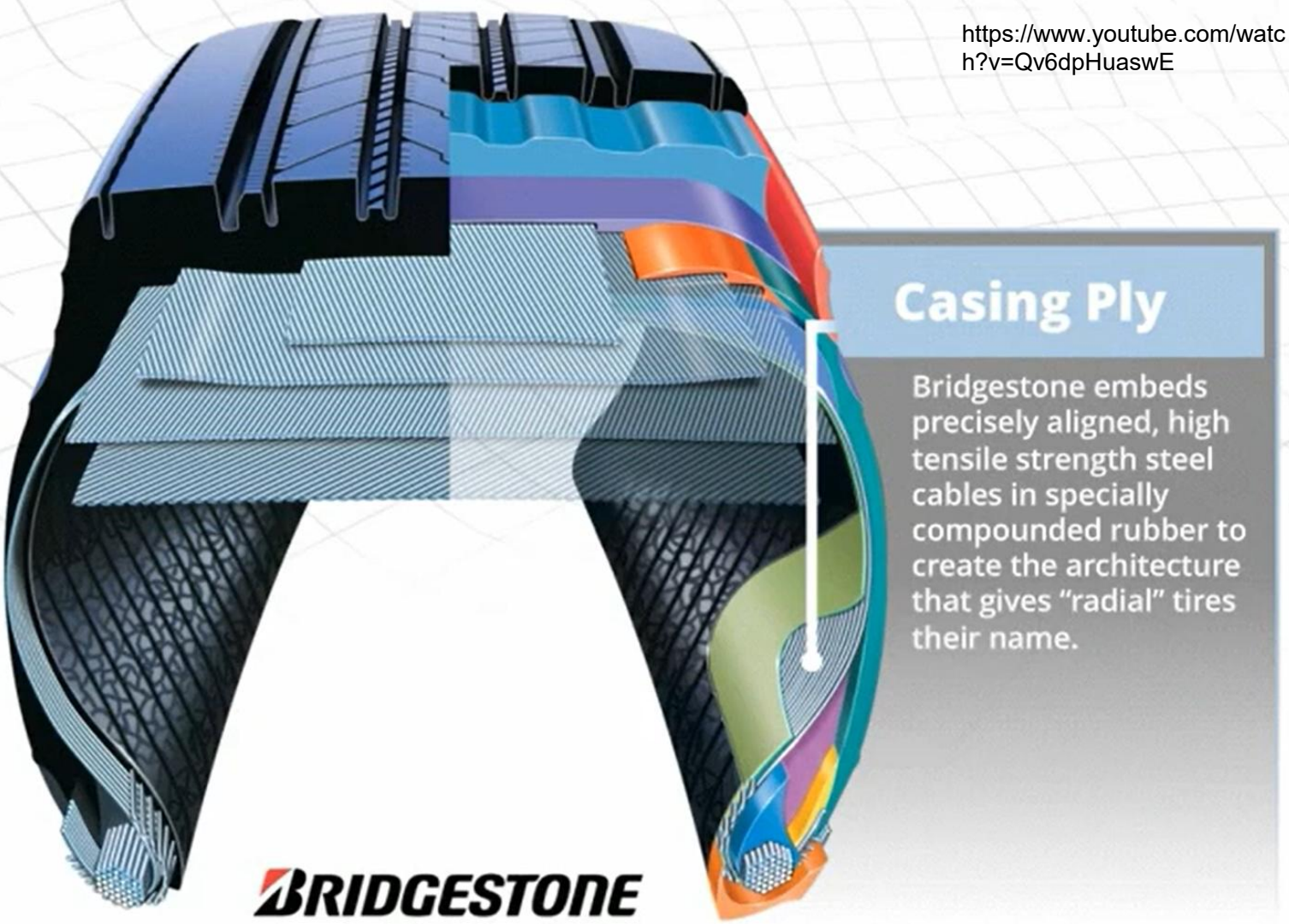
https://asygn.com/as3212-external-resistive/



https://meritsensor.com/products/s-series/?_gl=1*19copa9*_up*MQ..*_gs*MQ..&gclid=CjwKCAjwidXQBhAZEiwA4egw6AFDWQU4YWUk53bhm0CE1Inhb2xNhYfMtSv5cZRum_SfYeLwa8qKOBoC_GcQAvD_BwE&gbraid=0AAAADxc_3eXBt7b1VVc1jObju7qk42xg


1 Steel Belts
2 Body Ply
3 Inner Liner
4 Belt Edge Cushion
5 Bead Assembly

https://www.youtube.com/watch?v=Qv6dpHuaswE

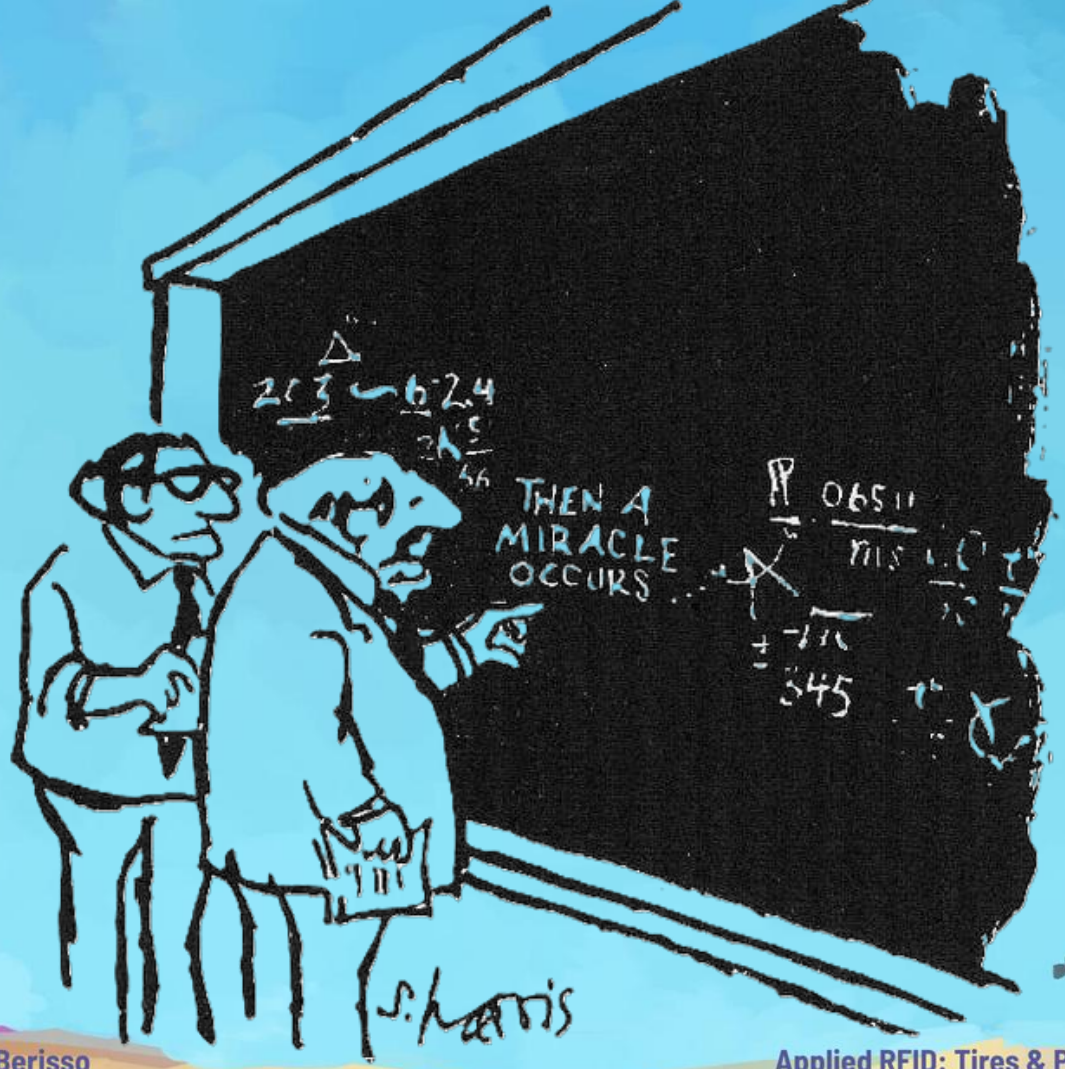


Our path

- Sensors
  - Strain - TBD

https://www.shutterstock.com/search/low-tire-pressure?dd_referrer=https%3A%2F%2Fwww.google.com%2F

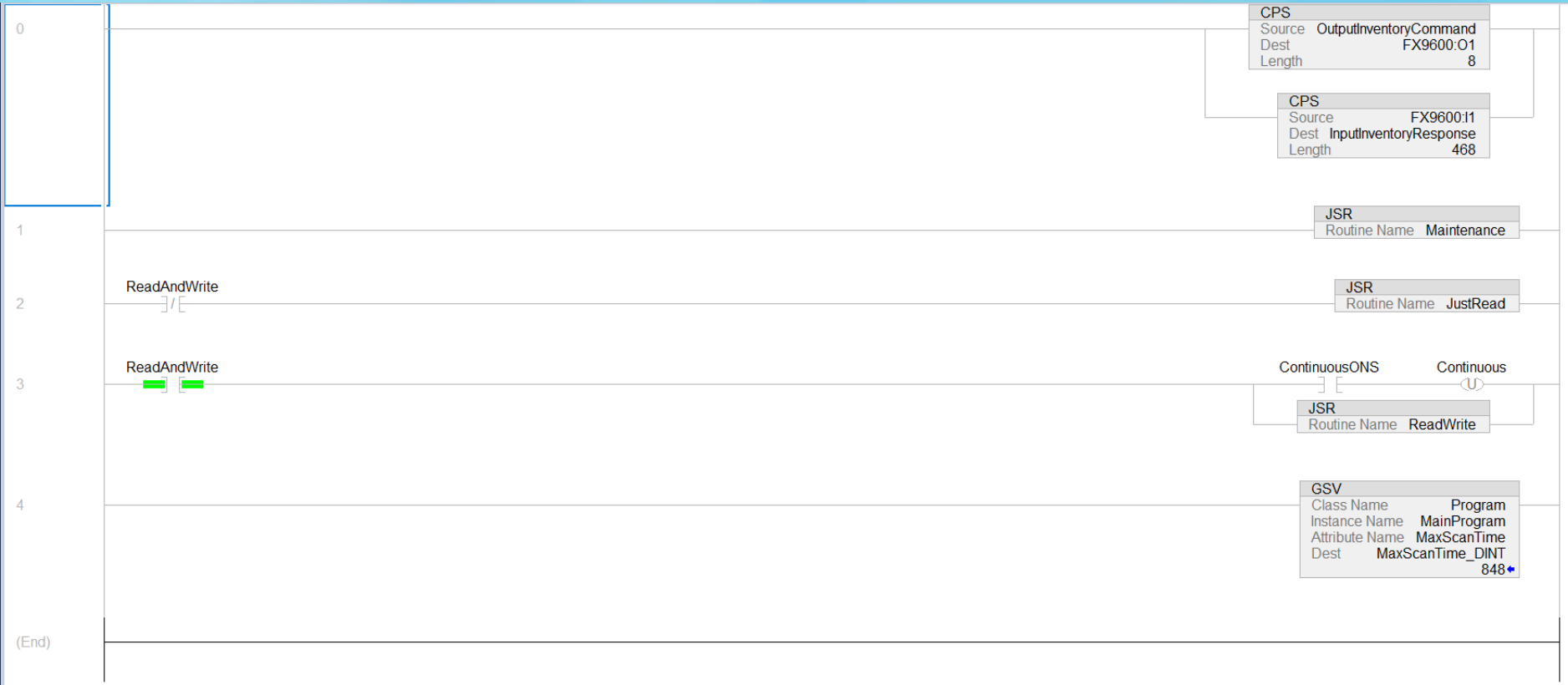
Our path



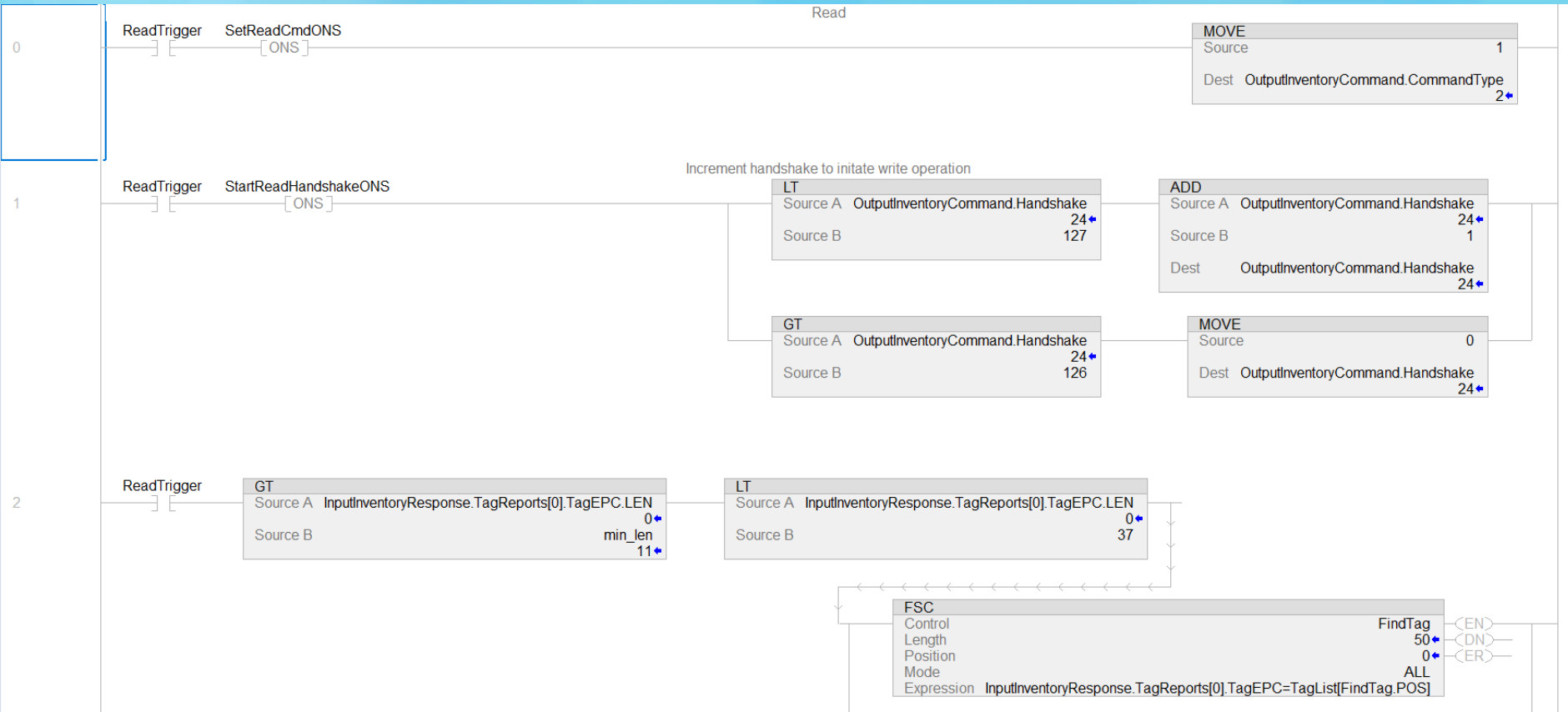
Our path

What could go wrong?

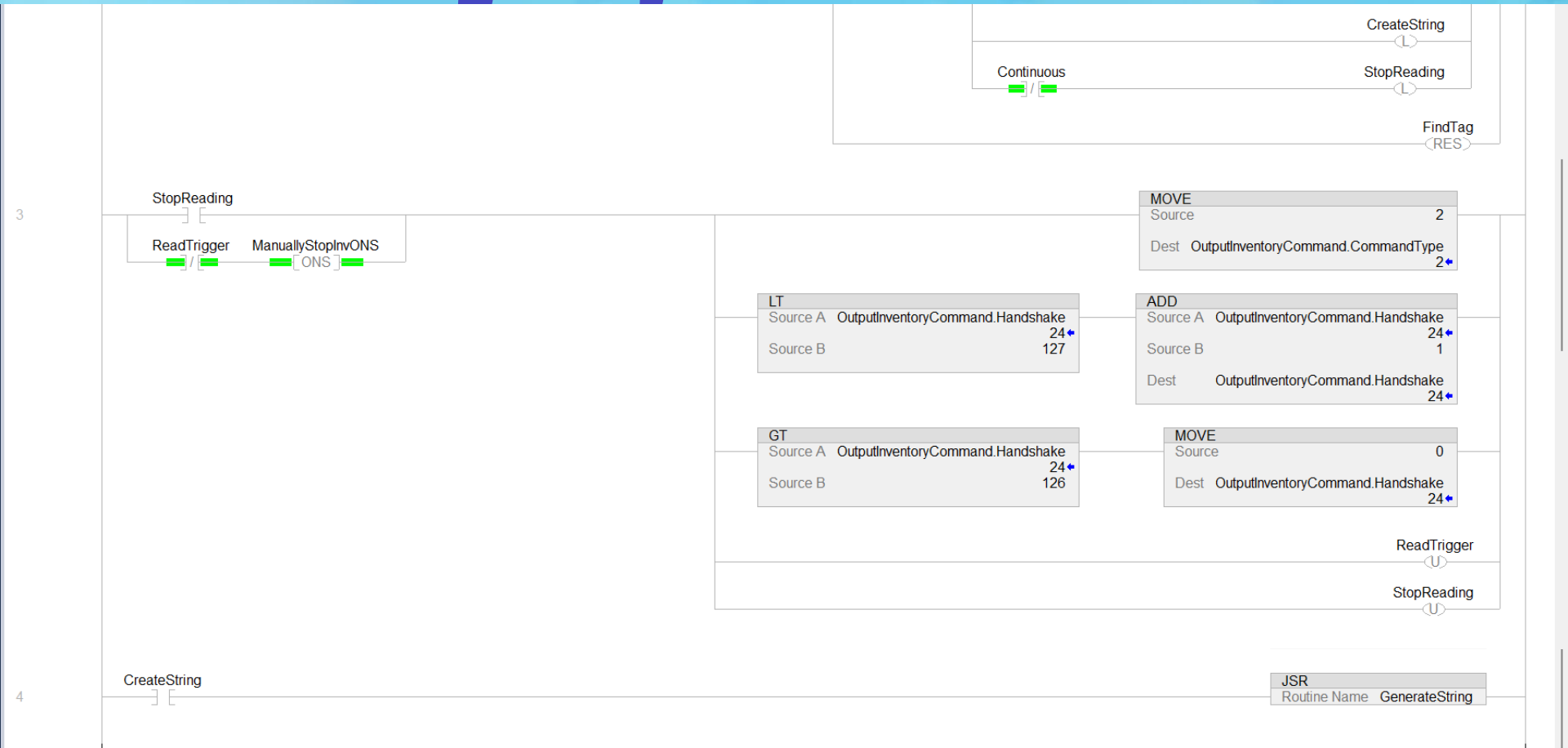
Ladder Logic - mainroutine



Ladder Logic - justread



Ladder Logic - justread



Ladder Logic - maintenance

Ladder Logic - maintenance

2 Copy96

COP
Source TagList[TagListFFL.POS-1]
Dest Store_96bit
Length 1

3 Copy128

COP
Source TagList[TagListFFL.POS-1]
Dest Store_128bit
Length 1

4 Place96

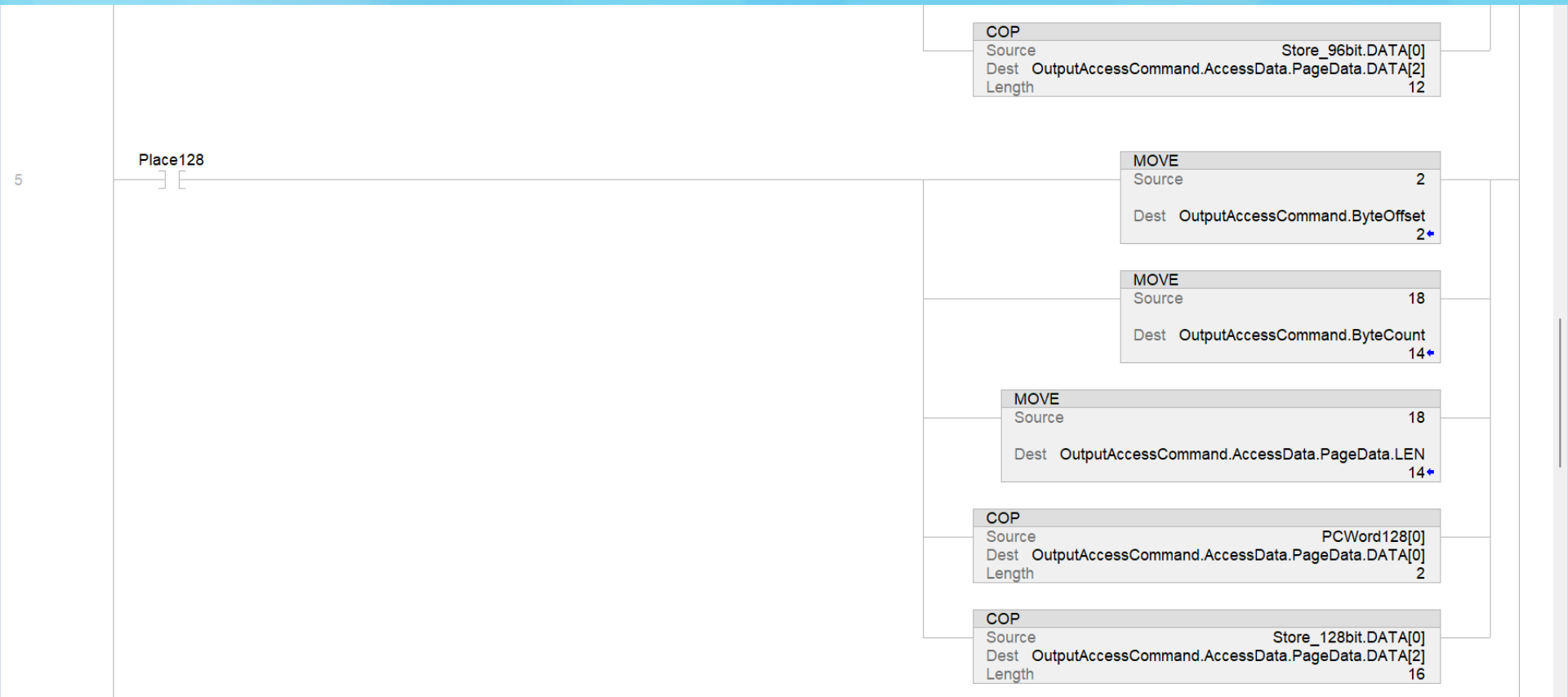
MOVE
Source 2
Dest OutputAccessCommand.ByteOffset
2

MOVE
Source 14
Dest OutputAccessCommand.AccessData.PageData.LEN
14

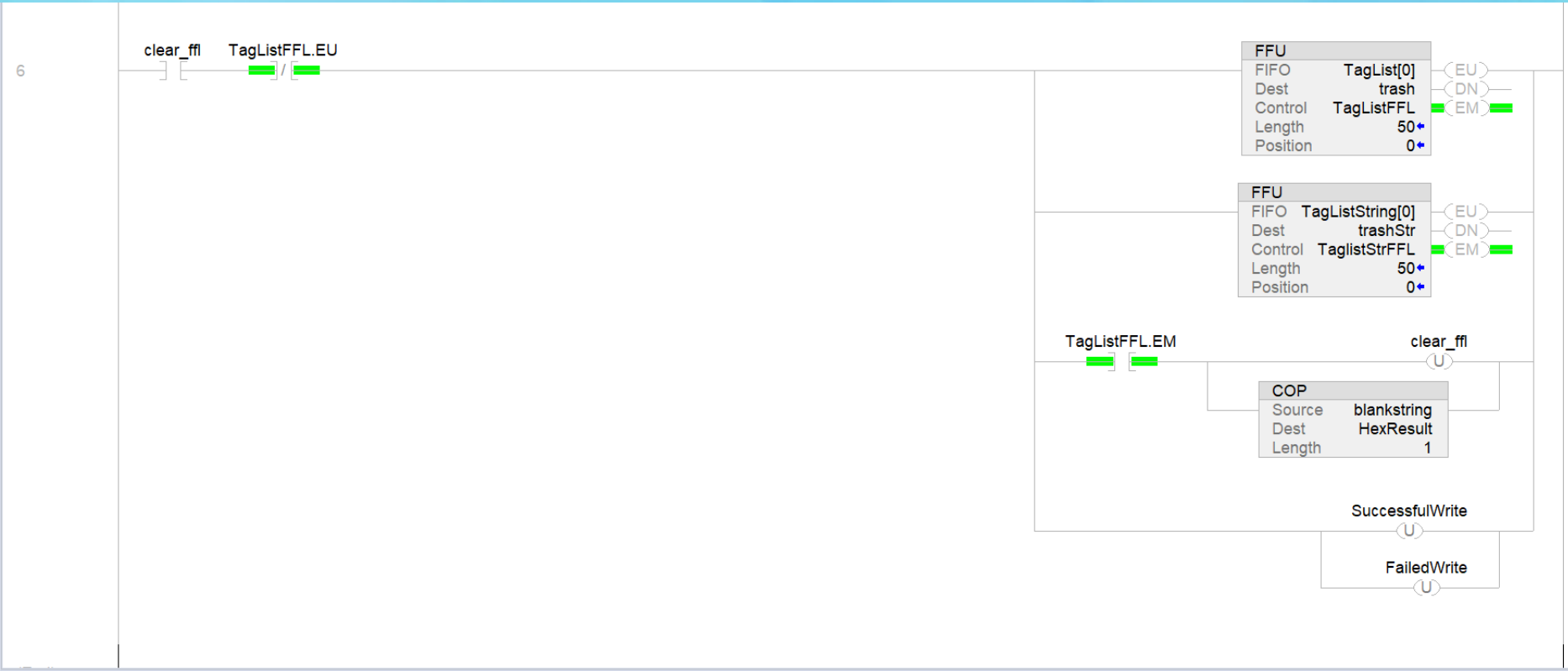
MOVE
Source 14
Dest OutputAccessCommand.ByteCount
14

COP
Source PCWord96[0]
Dest OutputAccessCommand.AccessData.PageData.DATA[0]
Length 2

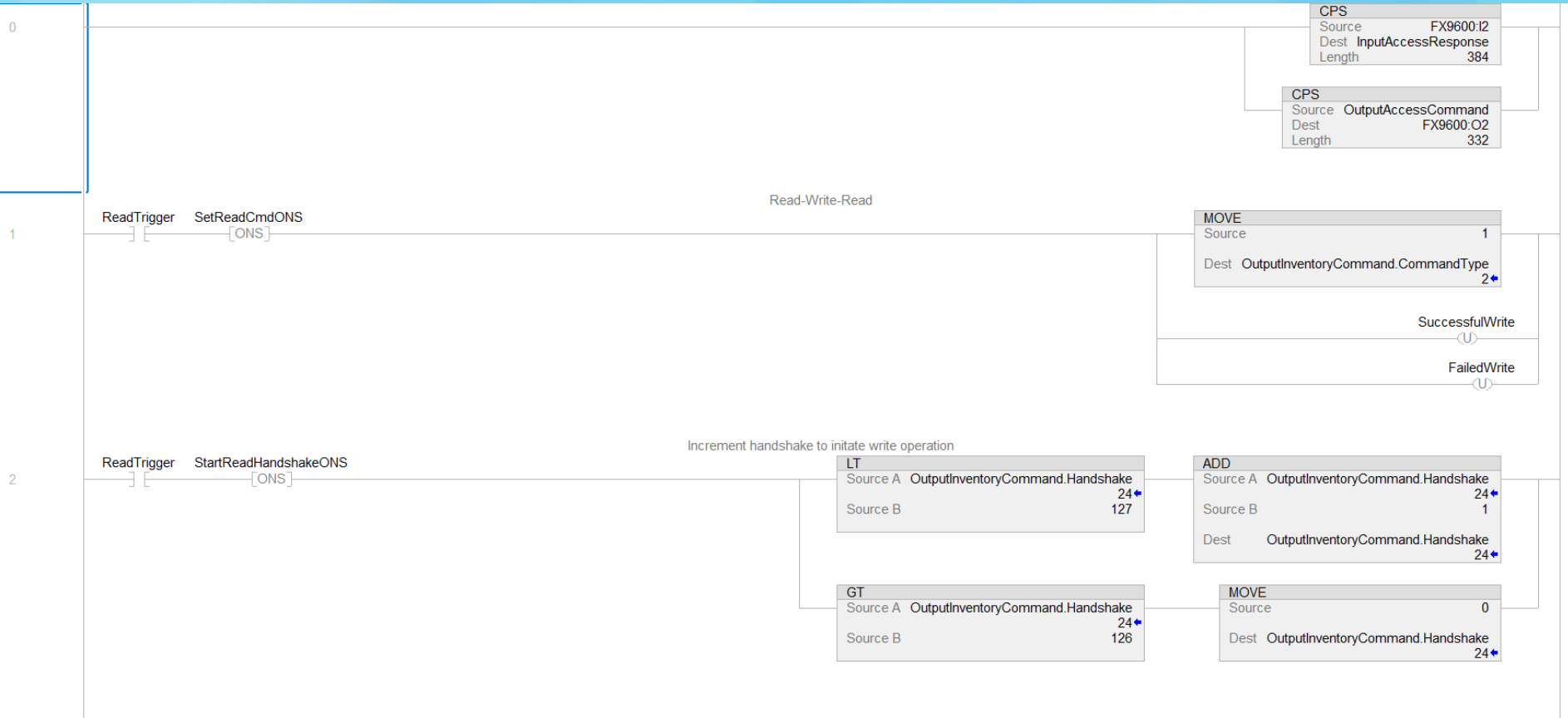
Ladder Logic - maintenance



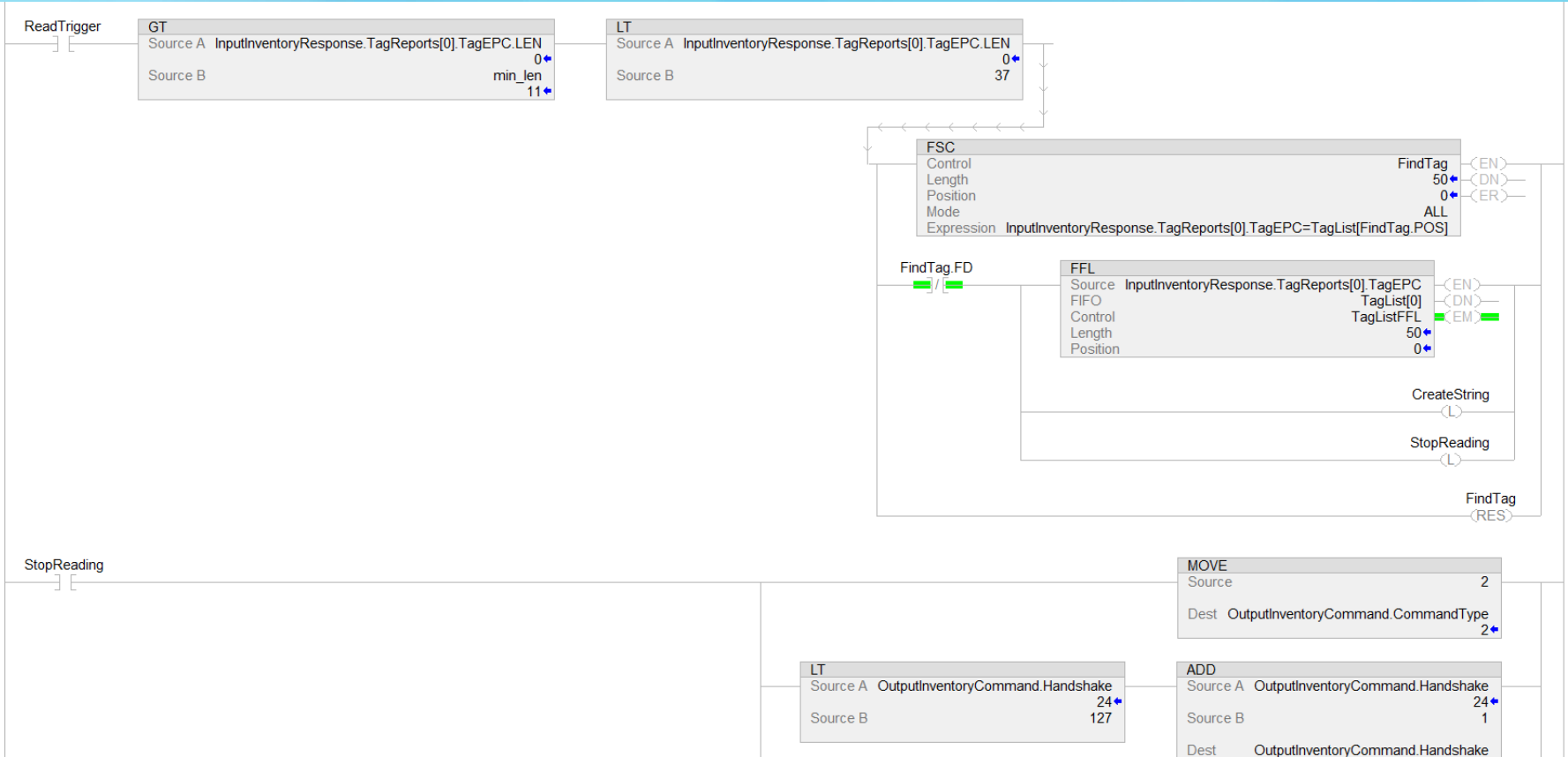
Ladder Logic - maintenance



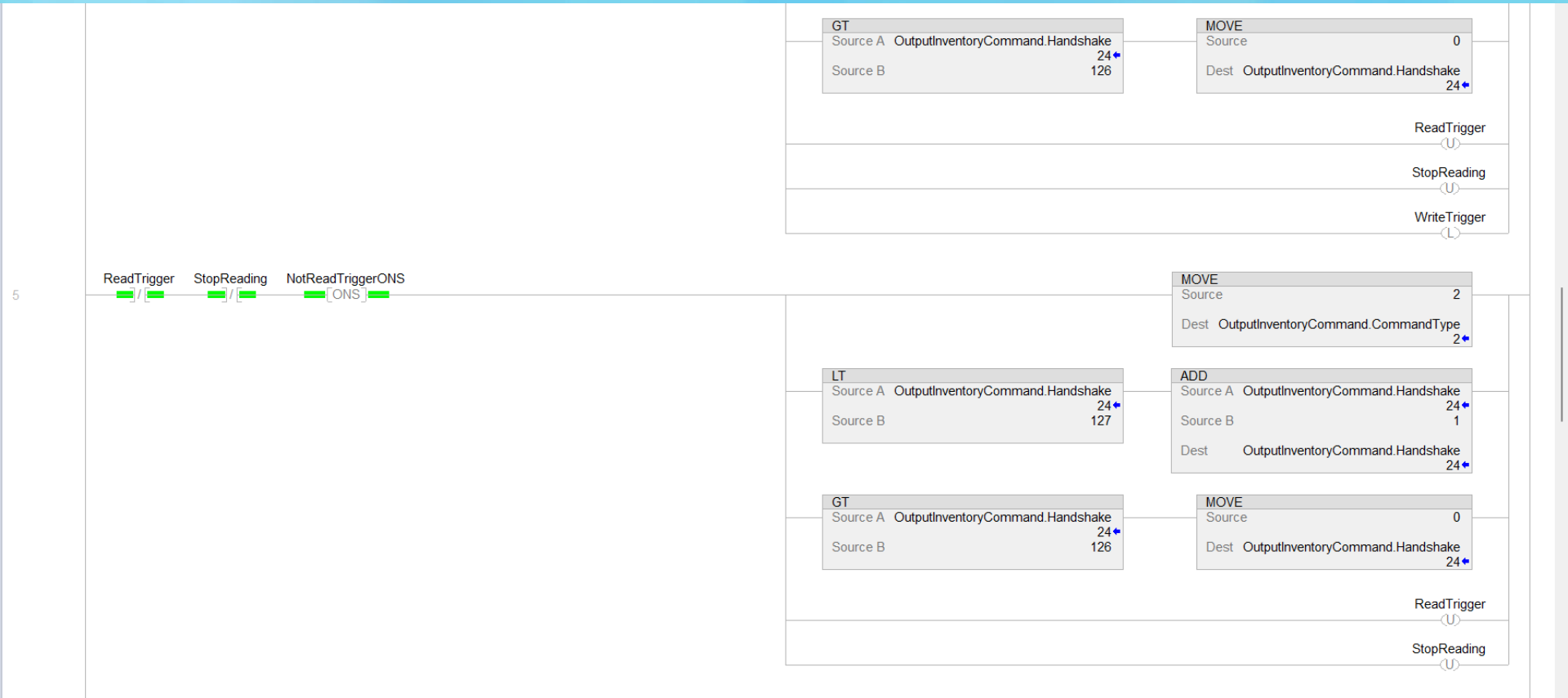
Ladder Logic - readwrite



Ladder Logic - readwrite

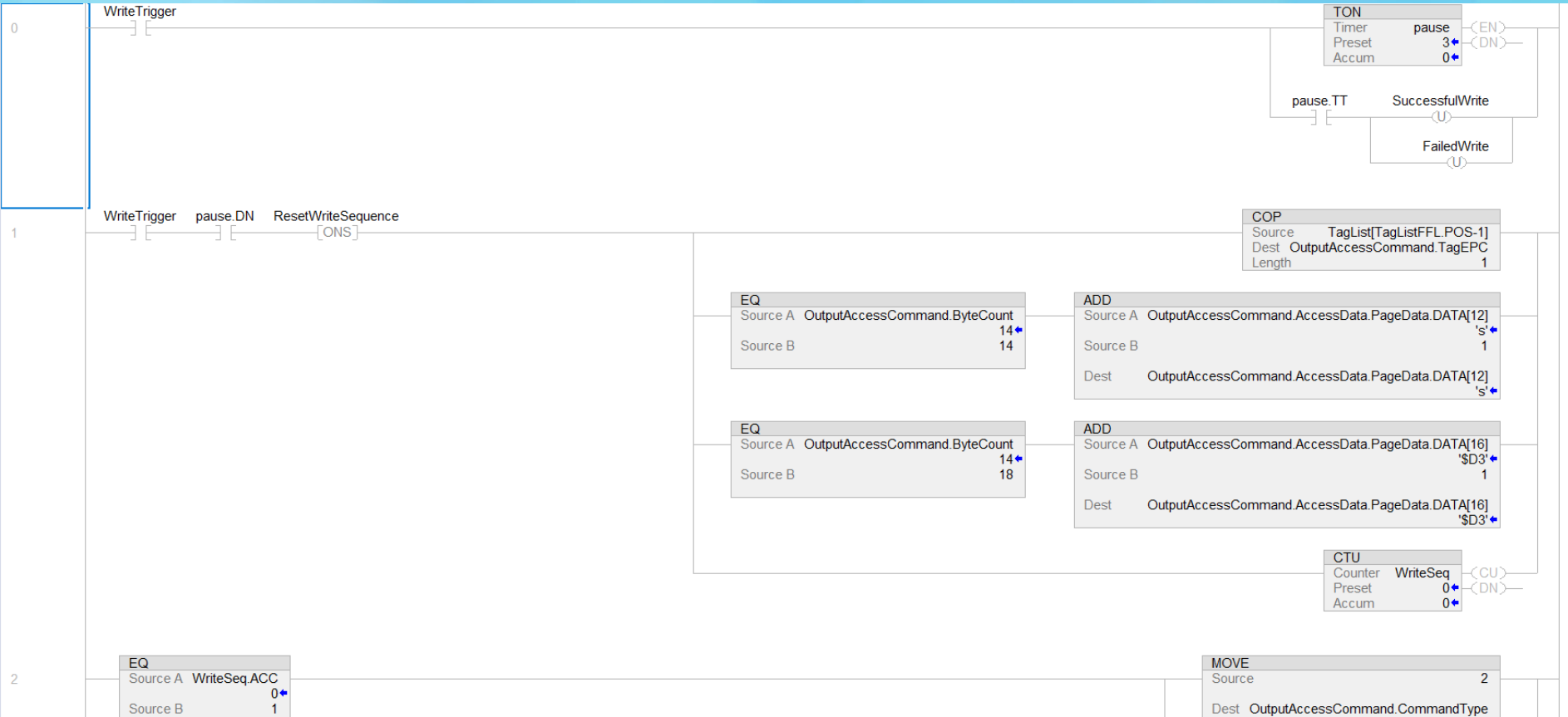


Ladder Logic - readwrite

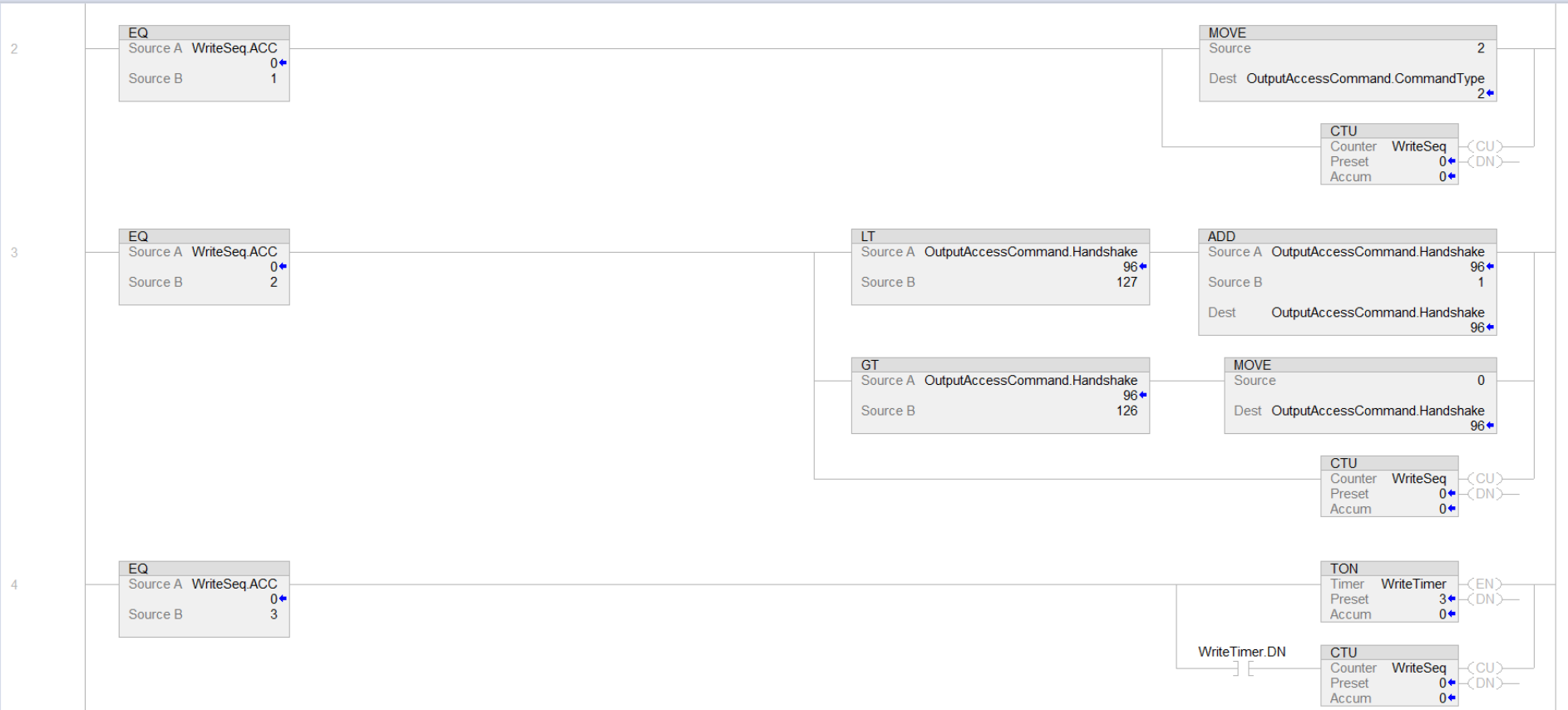


Ladder Logic - readwrite

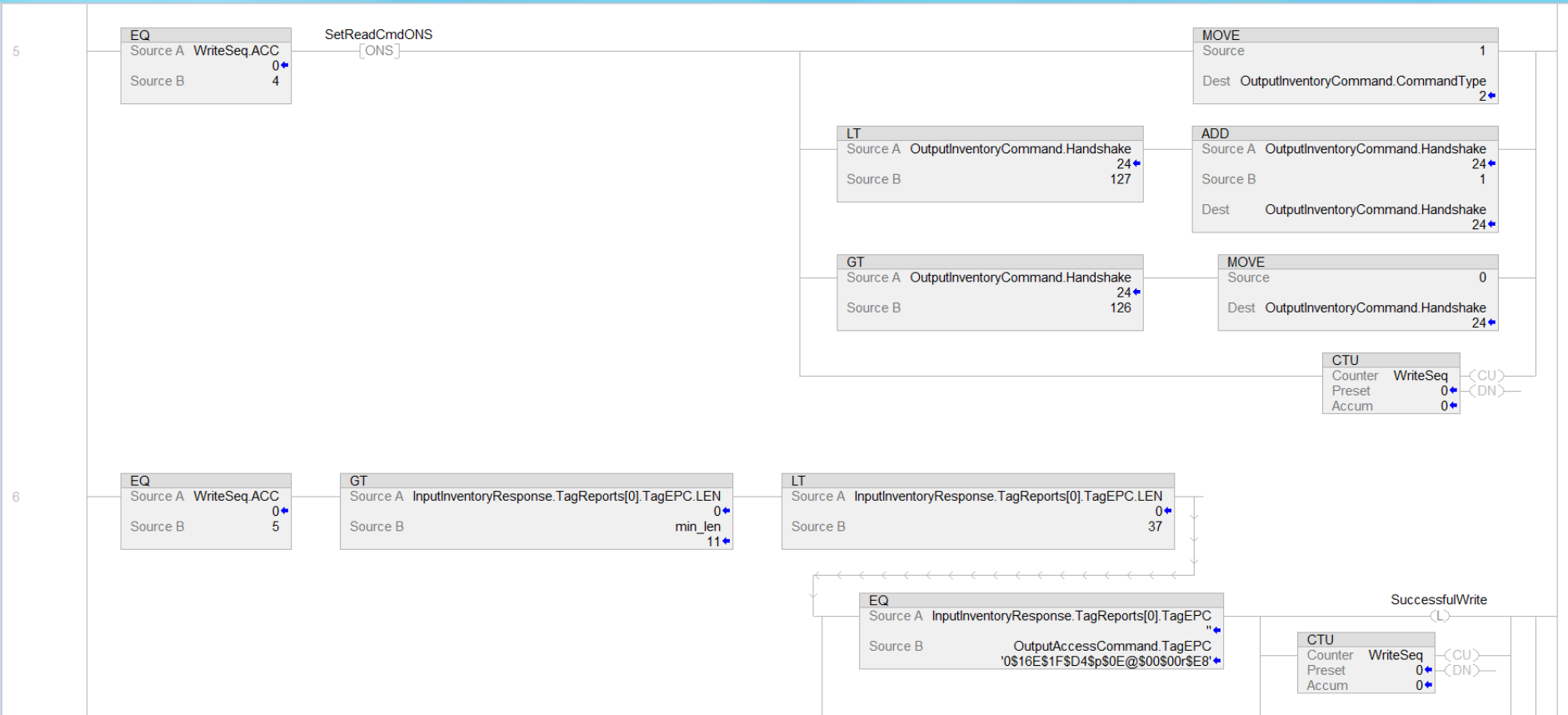
Ladder Logic - writeroutine



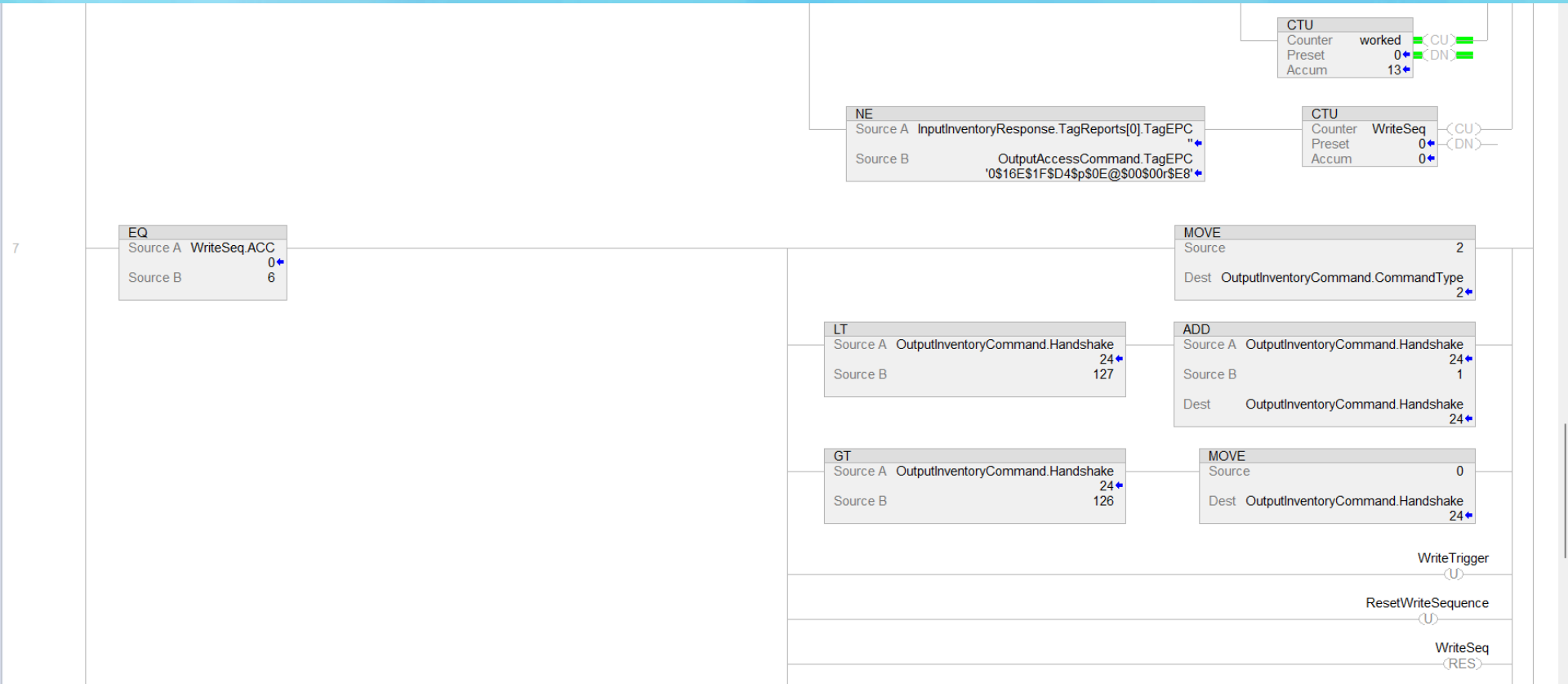
Ladder Logic - writeroutine



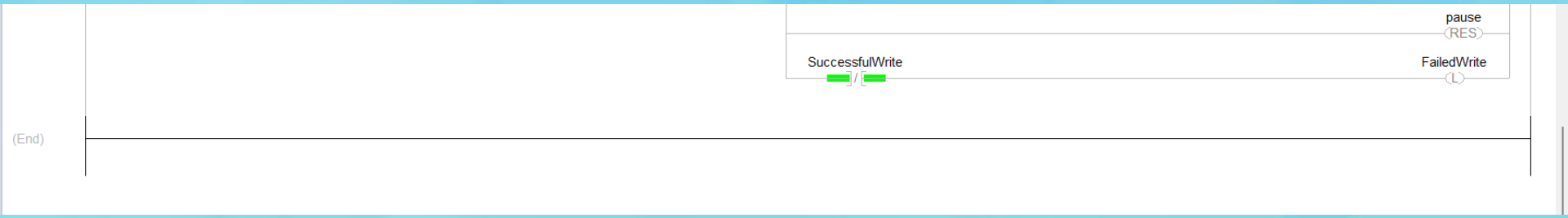
Ladder Logic - writeroutine



Ladder Logic - writeroutine



Ladder Logic - writeroutine



Ladder Logic - generatestring

```
// ============================================================
// Convert TagList SINT array to human-readable hex string
// ============================================================
HexResult.LEN := 0;

FOR i := 0 TO (TagList[TagListFFL.POS-1].LEN - 1) DO

    // Extract high nibble
    NibbleHigh := (TagList[TagListFFL.POS-1].DATA[i] AND 16#F0) / 16;
    IF NibbleHigh < 0 THEN NibbleHigh := NibbleHigh + 16; END_IF;

    // Extract low nibble
    NibbleLow := TagList[TagListFFL.POS-1].DATA[i] AND 16#0F;
    IF NibbleLow < 0 THEN NibbleLow := NibbleLow + 16; END_IF;

    // High nibble to ASCII
    CASE NibbleHigh OF
        0:  TempChar := 16#30; // '0'
        1:  TempChar := 16#31; // '1'
        2:  TempChar := 16#32; // '2'
        3:  TempChar := 16#33; // '3'
        4:  TempChar := 16#34; // '4'
        5:  TempChar := 16#35; // '5'
        6:  TempChar := 16#36; // '6'
        7:  TempChar := 16#37; // '7'
        8:  TempChar := 16#38; // '8'
        9:  TempChar := 16#39; // '9'
        10: TempChar := 16#41; // 'A'
        11: TempChar := 16#42; // 'B'
        12: TempChar := 16#43; // 'C'
        13: TempChar := 16#44; // 'D'
        14: TempChar := 16#45; // 'E'
        15: TempChar := 16#46; // 'F'
    END_CASE;
    HexResult.DATA[HexResult.LEN] := TempChar;
    HexResult.LEN := HexResult.LEN + 1;

    // Low nibble to ASCII
    CASE NibbleLow OF
        0:  TempChar := 16#30; // '0'
        1:  TempChar := 16#31; // '1'
        2:  TempChar := 16#32; // '2'
        3:  TempChar := 16#33; // '3'
        4:  TempChar := 16#34; // '4'
        5:  TempChar := 16#35; // '5'
        6:  TempChar := 16#36; // '6'
        7:  TempChar := 16#37; // '7'
        8:  TempChar := 16#38; // '8'
        9:  TempChar := 16#39; // '9'
        10: TempChar := 16#41; // 'A'
        11: TempChar := 16#42; // 'B'
        12: TempChar := 16#43; // 'C'
        13: TempChar := 16#44; // 'D'
        14: TempChar := 16#45; // 'E'
        15: TempChar := 16#46; // 'F'
    END_CASE;
    HexResult.DATA[HexResult.LEN] := TempChar;
    HexResult.LEN := HexResult.LEN + 1;
```

Ladder Logic - generatestring

```
56      END_CASE;
57      HexResult.DATA[HexResult.LEN] := TempChar;
58      HexResult.LEN := HexResult.LEN + 1;
59
60  END_FOR;
61  COP(HexResult, TagListString[TagListFFL.POS-1], 1);
62  TaglistStrFFL.POS :=TagListFFL.POS;
63
64  CreateString := 0;
```