

# ***SMARTEYE P-Linc***

## **Application Interface Package for the Allen Bradley Control Logix 5000**

**Application Note - SA0042**

**Revision 0**

Prepared by:



2637 Bond Street  
Rochester Hills, MI 48309

Revision 0, Dec 21, 2003

related documents:

***SMARTEYE*** P-Line User Manual

The material in this manual is for informational purposes and is subject to change without notice. SMARTEYE Corporation assumes no responsibility for any errors which may appear in this manual.

© 1996 SMARTEYE Corporation

***SMARTEYE*** is a registered trademark of SMARTEYE Corporation.

This document was produced using Microsoft® Word

Printed in U.S.A.

## Table of Contents

<b>1.0</b>	<b>INTRODUCTION.....</b>	<b>1</b>
<b>2.0</b>	<b>PARALLEL WIRING.....</b>	<b>1</b>
2.1	DATA BITS (D0-D15).....	1
2.2	READER BITS (RDR0 - RDR5) .....	1
2.3	READY BITS (RDY0, RDY1).....	2
2.4	PARITY BIT (PARITY).....	2
2.5	LABEL/ERROR BIT (L/E).....	2
2.6	DATA ACKNOWLEDGE (ACK0, ACK1).....	2
<b>3.0</b>	<b>POWER-UP.....</b>	<b>2</b>
<b>4.0</b>	<b>PARALLEL COMMUNICATION PROTOCOL .....</b>	<b>2</b>
4.1	TRANSACTION STEPS .....	3
4.2	DESIGN CONSIDERATIONS.....	3
<b>5.0</b>	<b>CONTROL LOGIX 5000 LADDER LOGIC EXPLANATION .....</b>	<b>4</b>
<b>6.0</b>	<b>APPENDIX ‘A’ (DATA TABLE LAYOUT).....</b>	<b>A-1</b>
<b>7.0</b>	<b>APPENDIX ‘B’ (EXAMPLE CONTROL LOGIX 5000 PROGRAM) .....</b>	<b>B-1</b>

## 1.0 Introduction

The Smarteye P-Linc connects up to 8 Smarteye Electronic Assemblies: (SEA-8s and/or Twins) on a serial RS485 multidrop communication line. Smarteye reader data is received on the serial communication line and converted to parallel digital 24VDC outputs. The digital outputs of the P-Linc are directly wired to digital input modules of a PLC. Two bits of the output data, the data ready bits, signal the preparation of new data on the output lines. A control device recognizes new data, receives it, and then provides two output bits which match the data ready bits. These outputs are received by the P-Linc on its two 24VDC input lines and interpreted as data acknowledge bits.

The P-Linc parallel port has 28 bits: 26 outputs and 2 inputs. The 26 outputs consist of: 16 data bits, 6 reader bits, 2 data ready bits, a parity bit and a label or error bit. The two input bits are used for data acknowledge. The 16 data bits and 6 reader bits can be configured as either binary or BCD.

## 2.0 Parallel Wiring

The P-Linc circuit card provides plug-in screw terminals for the 26 parallel output and 2 parallel input connections. Discrete 18 gauge wire such as Belden 8501 or equivalent is recommended for these connections. Parallel wiring details can be found on drawing #SP2004/01-411 in the "Smarteye P-LINC User Manual Appendix A".

Notes:

1. Data presented by a P-Linc on its parallel output lines is binary; i.e., each output represents a bit in a binary field (off = 0, on = 1).
2. The least significant bit of the data field is D0 and the most significant is D15. The least significant bit of the reader field is RDR0 and the most significant is RDR5.
3. Connecting all P-Linc parallel outputs to an input module may not be necessary. Example: binary mode with 10 bit labels, D0 - D9 must be wired, D10 - D15 will always be 0 and wiring is therefore optional.
4. The P-Linc is configured for true-high operation; i.e., a binary 1=24VDC and a binary 0=GND.

### 2.1 Data Bits (D0-D15)

The data bits (D0 - D15) define the value of the label or error number. This field may be binary or BCD dependent on the position of SW2.1. (Please refer to the P-Linc User Manual for additional information on dip switch configuration.)

### 2.2 Reader Bits (RDR0 - RDR5)

The reader bits (RDR0 - RDR5) define the value of the reader number. This field may be binary or BCD dependent on the position of SW2.1.

### 2.3 Ready Bits (RDY0, RDY1)

A change of state in the two ready bits signals the control device that the P-Linc has new data on its output lines. If the ready bits are 01 or 10, then the control device should receive the new data and perform appropriate control action. If the ready bits are 00 or 11, then the control device should ignore the data (take no control action). A full description of this protocol appears in section 4. - 'Parallel Communication Protocol'.

### 2.4 Parity Bit (PARITY)

The parity bit enables the control device to determine the validity of data received from the P-Linc. Thus, if an input/output point should fail on either the control device or the P-Linc, or if a connection between the two should be lost, then the control device can detect the fault and issue an alarm and/or stop the controlled operation until repairs are made.

The value of the parity bit is always set so that an **odd** sum results when the number of bits turned on in the data field (D0-D15), the reader field (RDR0-RDR5), and the Label/Error bit (L/E), are added with the parity bit itself.

### 2.5 Label/Error Bit (L/E)

The label or error (L/E) bit defines the data bits (D0 - D15) as an error number or a label number. **If this bit is a 1, the data is a label number. If the bit is a 0, the data is an error number.**

### 2.6 Data Acknowledge (ACK0, ACK1)

The two data acknowledge bits are the only parallel **inputs** received by the P-Linc. A control device must set the data acknowledge bits to the **same** value as the two data ready bits, signaling recognition of the most recent transition of data ready. A full description of this protocol appears in section 4. - 'Parallel Communication Protocol'.

## 3.0 Power-Up

When power is applied an LED on the power supply will illuminate. The P-Linc reads its setup switches and initializes its internal data. The control port transmit LED will begin to flash and the receive LED will flash if the control port is wired properly to one or more SEA units. One of the parallel port RDY bit LEDs will illuminate when valid data is on the parallel port.

## 4.0 Parallel Communication Protocol

The communication protocol between a P-Linc and a control device (e.g., programmable controller) is relatively simple. Nevertheless, it is important that a logic designer for the control device be thoroughly familiar with the protocol.

## 4.1 Transaction Steps

All data messages originate at an SEA unit. A set of messages in both directions is required for an SEA unit to deliver the data to the P-Linc and receive assurance of its delivery. The complete set of messages between an SEA Unit and the P-Linc, as well as the P-Linc and the PLC, is a data message transaction. The key steps of a data message transaction include the following:

1. The P-Linc selects the next SEA unit to poll from its on line list and sends a box poll command to that SEA unit.
2. The SEA unit responds with a data message.
3. The P-Linc sends a brief message back to the SEA unit, acknowledging receipt of the data message.
4. The P-Linc extracts data from the serial data message and establishes it on the parallel port.
5. Once the data has settled on the data lines the P-Linc will establish a new state for the data ready bits and write this new state to RDY0 and RDY1 of the parallel output lines.
6. Meanwhile, the PLC which has been monitoring the data ready outputs of the P-Linc, recognizes the change in the data ready bits.

If the data ready bits are 01 or 10, then the PLC receives the remainder of the data from the parallel outputs and takes appropriate action.

Whatever the new pattern of data ready bits, the control device sets data acknowledge bits to match.

7. When the P-Linc receives ACK0 and ACK1 equal to RDY0 and RDY1, then the transaction is complete.

## 4.2 Design Considerations

A P-Linc, regardless of the speed of its output buffers, cannot guarantee that all new output data will transition at the same instant. In other words, there is a short transient interval during which output data 'settles'. A PLC samples data presented by a P-Linc at scattered instances in time, including an instant during which the output lines of a P-Linc are settling.

The steps of a data message transaction are designed to prevent a control device from receiving data that is sampled during a settling period. In order to prevent this, the transaction steps allow output data to settle before the data ready bits change. Thus, a control device should only receive data when it recognizes a change in the data ready bits.

There are four possible patterns of the two data ready bits: 00, 01, 10, and 11. Only 01 and 10 are associated with usable data. Thus, a control device should receive data only when the data ready bits change and have a value of 01 or 10. This prevents a receipt of data during a settling period of the data ready bits, when 00 and 11 are possible values.

There is only one instance in which data ready is deliberately set (non-transient) to a value other than 01 or 10. At powerup, when there is no data yet established for the parallel port the initial value of data ready is 00. The associated parallel data may be ignored by the control device, as with any other instance of 00 for data ready.

A control device should reflect all values of data ready in its data acknowledge outputs, even values of 00 and 11. A P-Linc will not begin a new data message transaction until data ready of the previous transaction is matched by data acknowledge.

## 5.0 Control Logix 5000 Ladder Logic Explanation

The ladder file provided is intended to be installed in a separate task from all other logic. All user defined data types and program level tags must be pasted into the Control Logix controller program before the logic is pasted. Upon pasting the logic update rung 0 with the actual rack address of the I/O cards. Also Update rungs 17 and 18 with the actual bit addresses of the acknowledge outputs.

A data table layout is included in Appendix A, and an example program is included in Appendix B which is described in this section.

The Control Logix controller logic begins by buffering the current input words (Controller Level Tags) into into the internal words (Program Level Tags) in rung 0.

The Control Logix controller logic must recognize when the states of the ready bits have changed. This is accomplished by storing the last state of the ready bits and comparing them to the new state. See rung 1. If new data is available, then the new state of the ready bits is copied to the last state and the ACK outputs. See rungs 17 and 18. If new data is not available then the program skips down to rung 17.

The parity is calculated by adding the number of bits that are on. The 16 data bits, the 6 reader bits, the parity bit, and the L/E bit are each checked. If the bit is on, a constant value of one is added to the parity sum word. The final result of the parity sum word should be an odd number, or the LSB should be a one. See rung 10.

If the parity is ok, the program proceeds to mask off the reader number, see rung 11, test reader numbers for high and low user set limits, see rung 12, test label numbers for high and low user set limits, see rung 13, and write the valid label or error to the user data words, see rungs 14 and 15. A label ready word or error ready word (read event flags) is set to 1 when new data is moved into the user data words. The read event flags are intended to be trigger words. When one of these trigger words transitions from 0 to 1, this means that a new label or error is now in the user data word. The user program should read the label or error from the user data word and clear the trigger word so it can be set again for the next label read.

The fault bit is set when the parity test fails, or when a label is read which is larger or smaller than the defined range, see rung 16.

The integer constant words shown on the PLC memory map drawing must be set up before running the program.



## **6.0 Appendix 'A' (Logix 5000 PLinc Data Table Layout)**

Control Logix 5000 Memory Map for Parallel Interface Application – Logix 5000 PLinc  
Data Table Layout.xls

Tag Level	Tag Names	Input Word 0															
Controller	Rack_03_0:I.Data[0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
Program	Input_0	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	RDY1	RDY0	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0

Input Word 1 Layout																	
Tag Level	Tag Names	Input Word 1															
Controller	Rack_03_0:I.Data[1]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		V	V	V	V	V	V	V	V	V	V	V	V	V	V	V	V
Program	Input_1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Output Word Layout																																																	
Tag Level	Tag Names																																																
Controller	Rack_03_0:O.Data[0]																																																
	<table border="1"> <thead> <tr> <th colspan="16">Output Word 0</th> </tr> <tr> <th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th> </tr> </thead> <tbody> <tr> <td>NOT USED</td><td>NOT USED</td><td>ACK1</td><td>ACK0</td> </tr> </tbody> </table>	Output Word 0																15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	NOT USED	ACK1	ACK0													
Output Word 0																																																	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0																																		
NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	NOT USED	ACK1	ACK0																																		

Internal Application Bits		
Tag Level	Tag Names	Description
Program	Internals.New_Data_Available	New P-Linc Data Available for Calculation
Program	Internals.Parity_Is_OK	Checksum Calculated on Odd Number Parity OK
Program	Internals.Label_Range_Is_OK	Carrier Number Calculated to Be Within Valid Range
Program	Internals.Plinc_Data_Fault	P-Linc Data Received Calculated to Be Invalid
Program	Internals.Reader_Range_Is_OK	Reader Number Calculated to Be Within Valid Range
Program	Internals.PLC_Ack_0	Last State Acknowledge State 0
Program	Internals.PLC_Ack_1	Last State Acknowledge State 1
Program	Internals.First_Scan_Bit	Program Does Not Execute on First Scan

Constant Application Words		
Tag Level	Tag Names	Description
Program	Constants.Reader_Number_Low_Limit	Constant Value For Limit Statements
Program	Constants.Reader_Number_High_Limit	Constant Value For Limit Statements
Program	Constants.Carrier_Number_Low_Limit	Constant Value For Limit Statements
Program	Constants.Carrier_Number_High_Limit	Constant Value For Limit Statements
Program	Constants.Constant_63_For_Conversion	Constant Value For Calculation



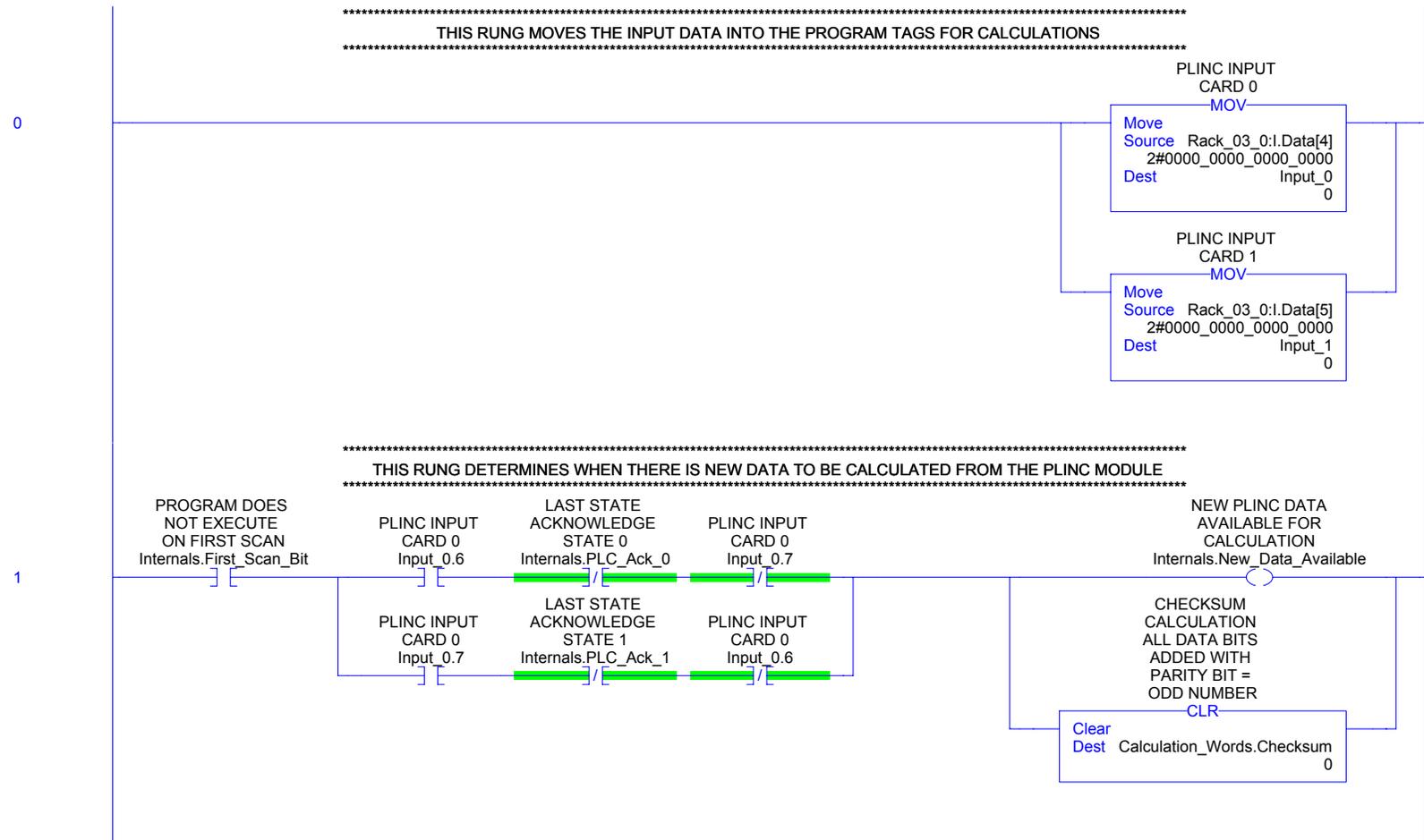




New Error Index													
Tag Level	Tag Names	Word											
		7	6	5	4	3	2	1	0	Sea or Twin 0			
		15	14	13	12	11	10	9	8	Sea or Twin 1			
		23	22	21	20	19	18	17	16	Sea or Twin 2			
		31	30	29	28	27	26	25	24	Sea or Twin 3			
		39	38	37	36	35	34	33	32	Sea or Twin 4			
		47	46	45	44	43	42	41	40	Sea or Twin 5			
		55	54	53	52	51	50	49	48	Sea or Twin 6			
		63	62	61	60	59	58	57	56	Sea or Twin 7			
		Reader 7	Reader 6	Reader 5	Reader 4	Reader 3	Reader 2	Reader 1	Reader 0				
Program	New_Error_Index.Word												

## **7.0 Appendix 'B' (Example Control Logix 5000 Program)**

Control Logix 5000 Ladder Listing for Parallel Interface Application – PLinc.acd



\*\*\*\*\*  
THIS SECTION COUNTS THE NUMBER OF BITS TURNED ON BY THE PLINC FOR A CHECKSUM. WITH THE PARITY BIT, ALL VALID DATA WILL HAVE AN ODD NUMBER  
FOR THE CHECKSUM  
\*\*\*\*\*

NEW PLINC DATA  
AVAILABLE FOR  
CALCULATION  
Internals.New\_Data\_Available

PLINC INPUT  
CARD 1  
Input\_1.0

CHECKSUM  
CALCULATION  
ALL DATA BITS  
ADDED WITH  
PARITY BIT =  
ODD NUMBER

ADD

Add	Source A	1
Source B	Calculation_Words.Checksum	0
Dest	Calculation_Words.Checksum	0

PLINC INPUT  
CARD 1  
Input\_1.1

CHECKSUM  
CALCULATION  
ALL DATA BITS  
ADDED WITH  
PARITY BIT =  
ODD NUMBER

ADD

Add	Source A	1
Source B	Calculation_Words.Checksum	0
Dest	Calculation_Words.Checksum	0

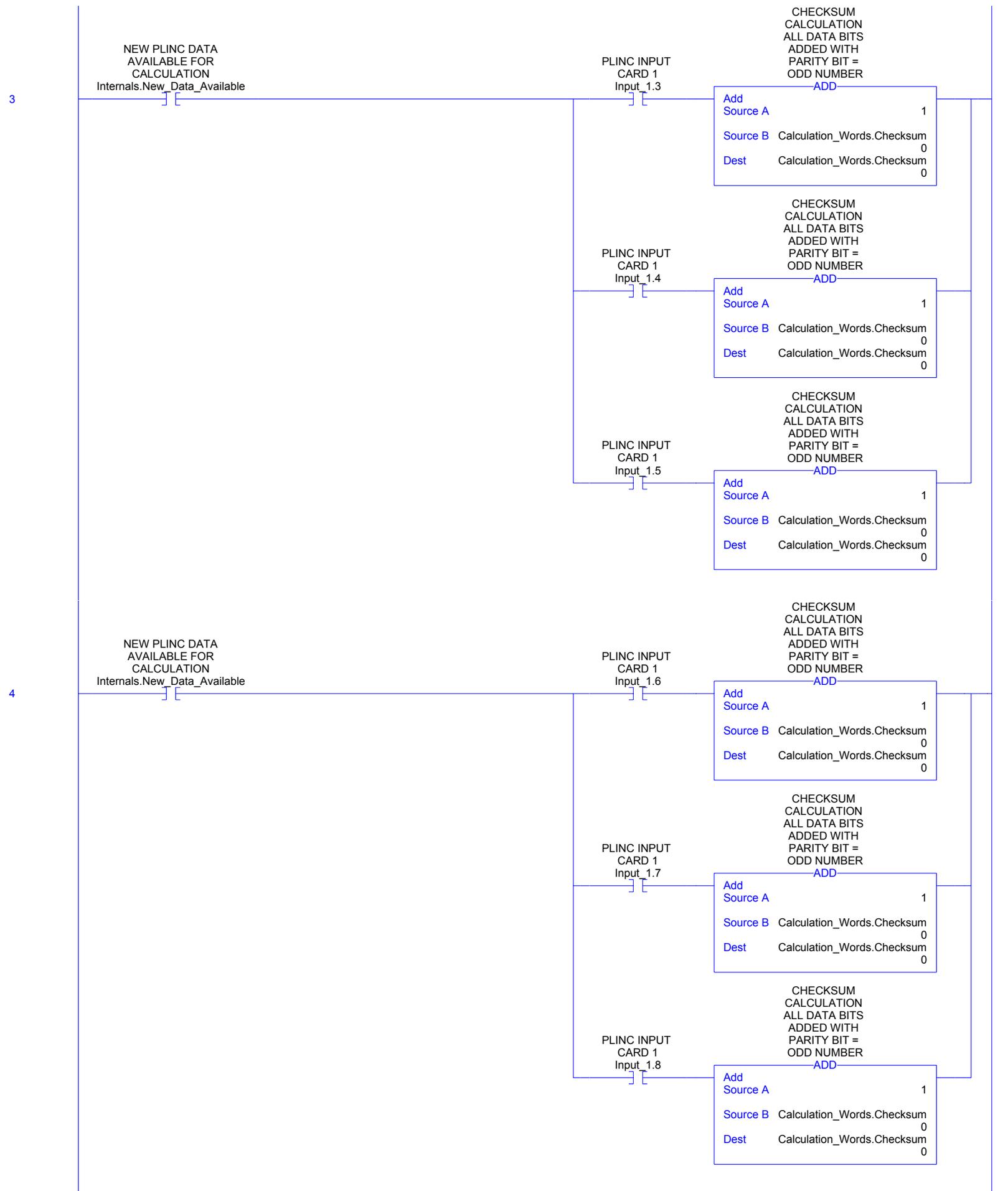
PLINC INPUT  
CARD 1  
Input\_1.2

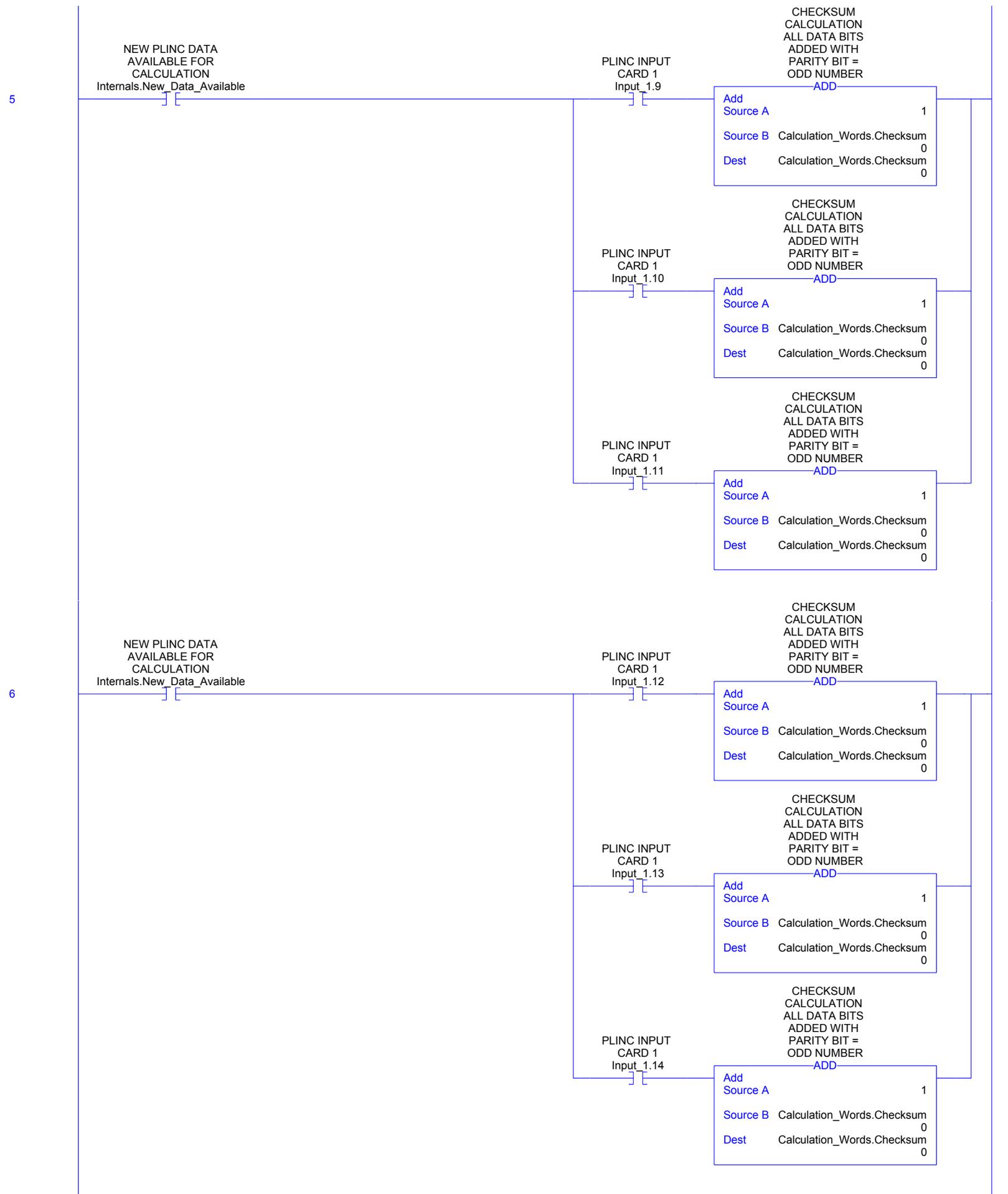
CHECKSUM  
CALCULATION  
ALL DATA BITS  
ADDED WITH  
PARITY BIT =  
ODD NUMBER

ADD

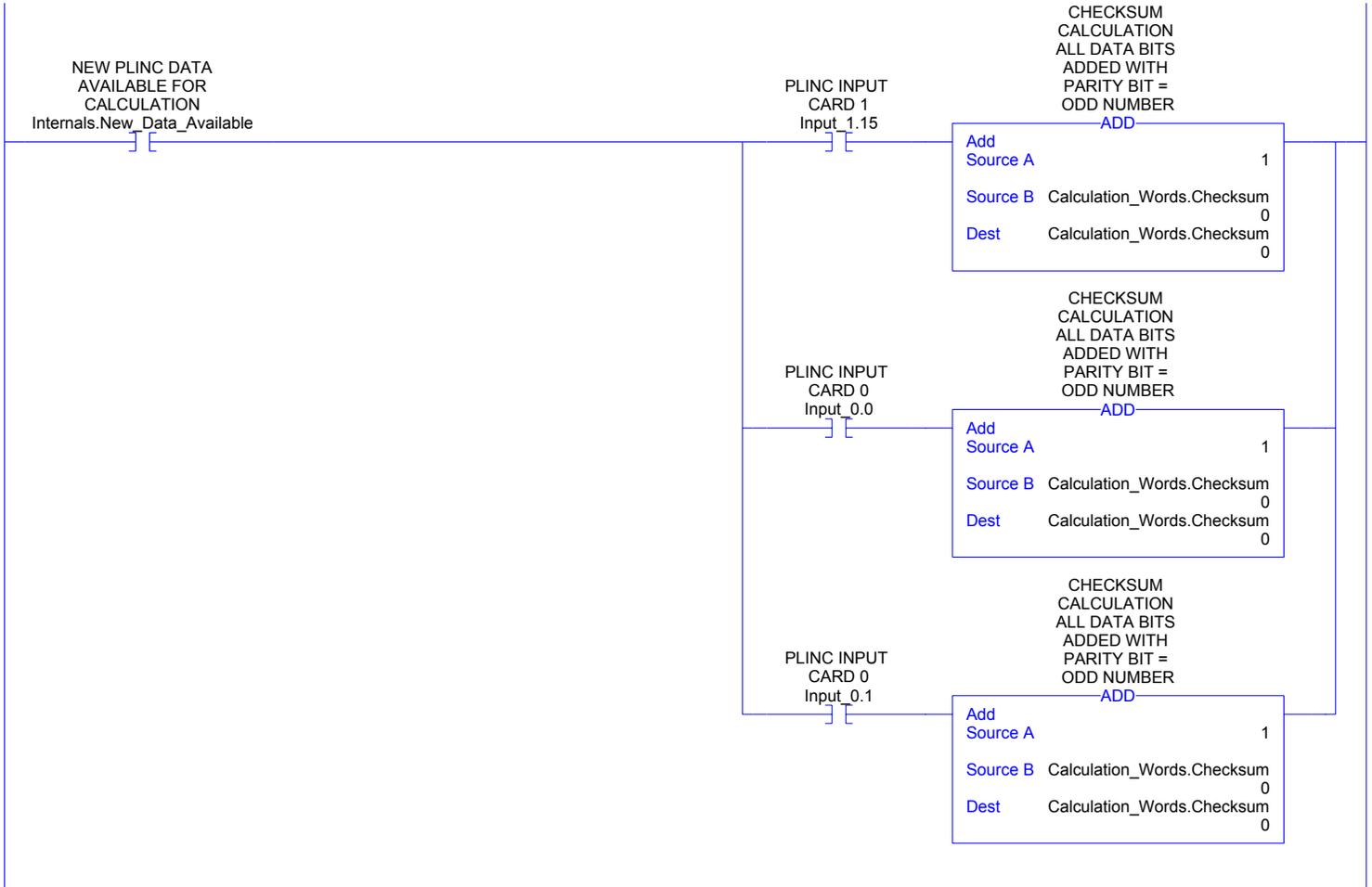
Add	Source A	1
Source B	Calculation_Words.Checksum	0
Dest	Calculation_Words.Checksum	0

2

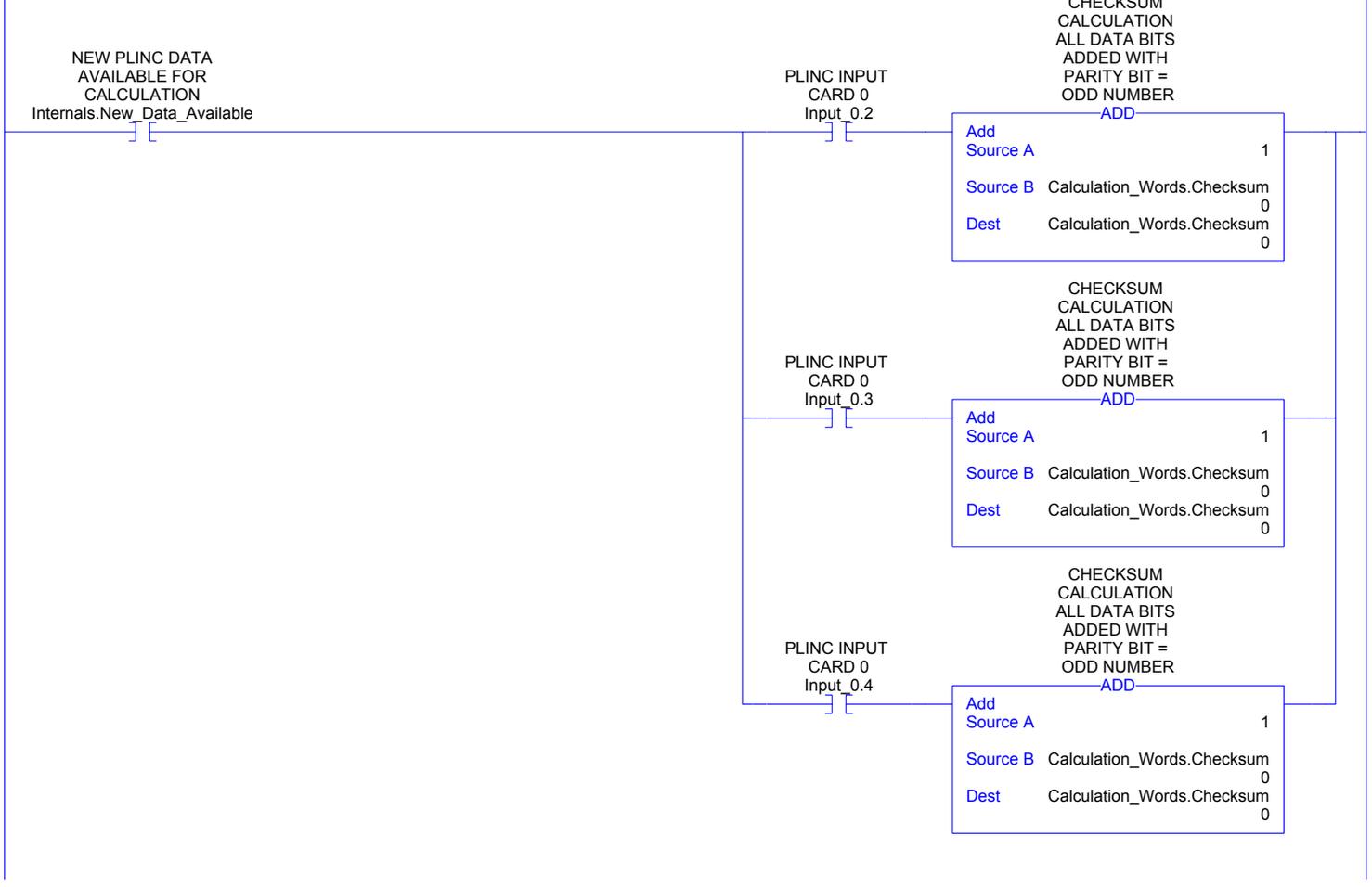


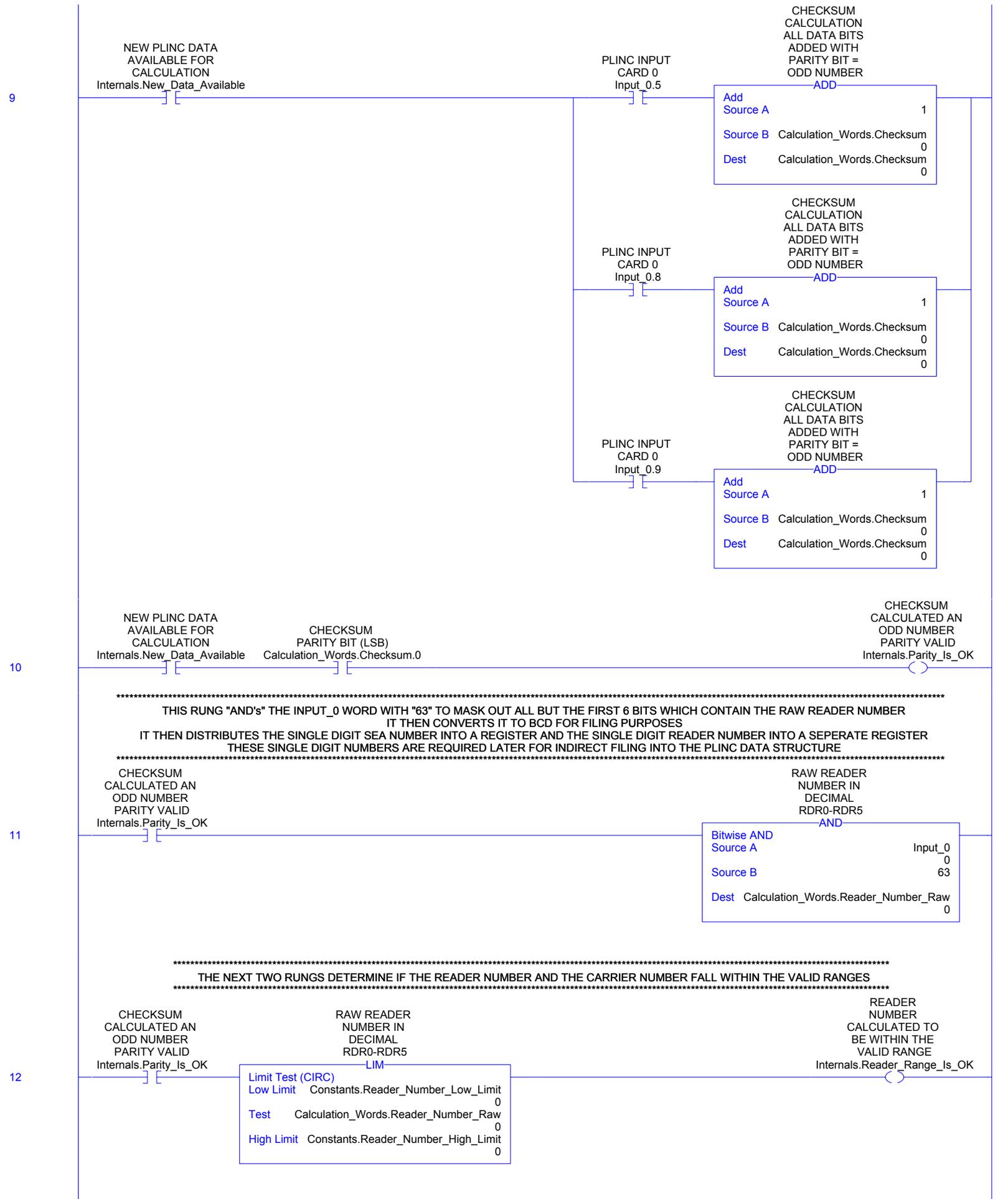


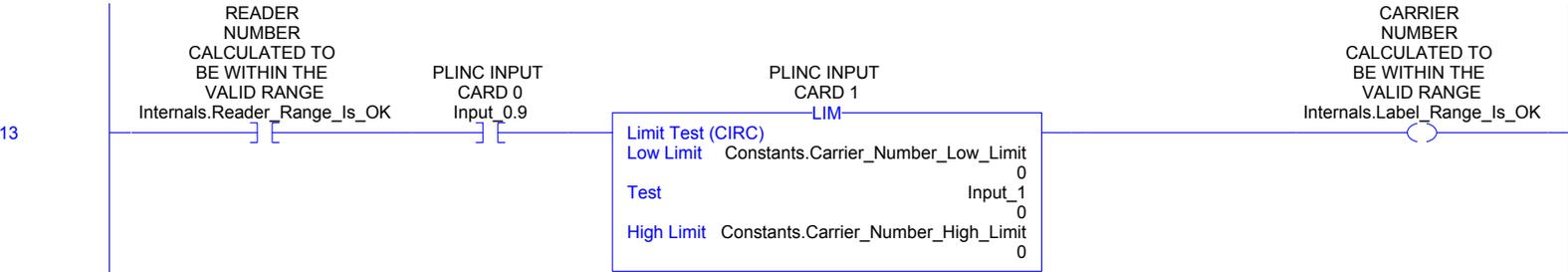
7



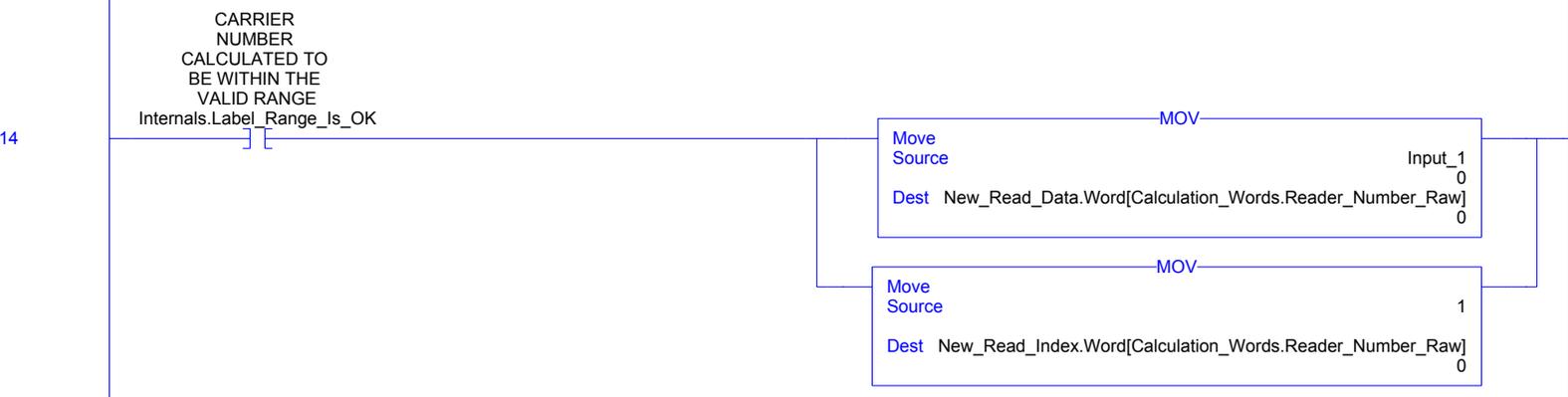
8







\*\*\*\*\*  
THIS RUNG FILES THE NEW CARRIER LABEL DATA INTO THE PLINC DATA TABLE. IT USES THE PRECALCULATED READER NUMBER TO INDERECTLY FILE INTO THE CORRECT LOCATION WITHIN THE TABLE.  
IT ALSO SETS MOVES A "1" INTO THE NEW READ INDEX USING THE SAME ADDRESSING. THE ONE INDICATES THAT THE READ IS NEW, AND IS RESET BY THE SYSTEM USING THE DATA.  
\*\*\*\*\*



\*\*\*\*\*  
THIS RUNG FILES THE NEW ERROR DATA INTO THE PLINC DATA TABLE. IT USES THE PRECALCULATED READER NUMBER TO INDERECTLY FILE INTO THE CORRECT LOCATION WITHIN THE TABLE.  
IT ALSO SETS MOVES A "1" INTO THE NEW ERROR INDEX USING THE SAME ADDRESSING. THE ONE INDICATES THAT THE ERROR IS NEW, AND IS RESET BY THE SYSTEM USING THE DATA.  
\*\*\*\*\*

