
Industrial Automation Crimson®: Mapping to RSLogix 5/ Tech Note 12 500-Based Processors



Abstract:

This document describes how to map Crimson tags to RSLogix 5/500-based processors and Programmable Logic Controllers (PLCs).

Products:

Red Lion Controls CR1000 Human Machine Interface (HMI), CR3000 HMI, Data Station Plus (DSP), G3 HMI, G3 Kadet HMI, Graphite® Controller, Graphite HMI, Modular Controller, and ProductVity Station™

Problem Solved: Mapping Tags to RSLogix 5/500-Based Processors

One difference between the majority of data types available for programming processors is the way that bit addressing is displayed. RSLogix 5/500-based processors/PLCs often use a “/” to signify a bit within a word or long; Crimson uses a “.” to signify a bit within a word or long. Additionally, all RSLogix 5/500-based processors transfer data in increments of 16-bit words. To map a tag to a bit within a word or long, first map the tag to the word or long, set the Treat As parameter to Bit Array Little-Endian, and then choose the desired bit from the Bit Number drop down selection. Refer to section 2.B in the [Crimson 3.0 Quick Start Guide](#) for further instructions on accessing bits within words. Refer to the [RSLogix 5](#) section for RSLogix 5-specific information.

Required Software:

Crimson® 2.0, 3.0, or 3.1

Required Operating System:

Microsoft Windows 2000, or above

Introduction

This document describes how to map Crimson tags to RSLogix 5/500-based processors for the types of data files listed in Table 1.

TABLE 1.

Data Type
Binary
Counter
Floating Point
Input
Long
Integer
Output
String
Timer

NOTE: Crimson 2 or higher is required and is only available to Windows 2000 or above users. Please update your version of Crimson to the latest; available online at www.redlion.net.

RSLogix 500

All of the information provided in this section applies to RSLogix 500-based processors/devices. All information, except for the descriptions of Input and Output Data types, also applies to RSLogix 5-based processors/devices.

Binary

RSLogix 500

Binary Data File addresses may be shown in one of two formats:

1. B(File):Element/Bit
2. B(File)/Bit.

Crimson

Crimson's mapping is similar to option 1 listed above. The Binary Data Files are transferred a word at a time, in order to access individual bits, a flag tag is required.

Counter

RSLogix 500

Counter Data File addresses are shown in the following format:

- C(File):Element
- Counters are broken down into 8 pieces:
 - CU
 - CD
 - DN
 - OV
 - UN
 - UA
 - PRE
 - ACC.
- The item selected in Figure 1 is displayed as C5:0/CU in RSLogix 500.

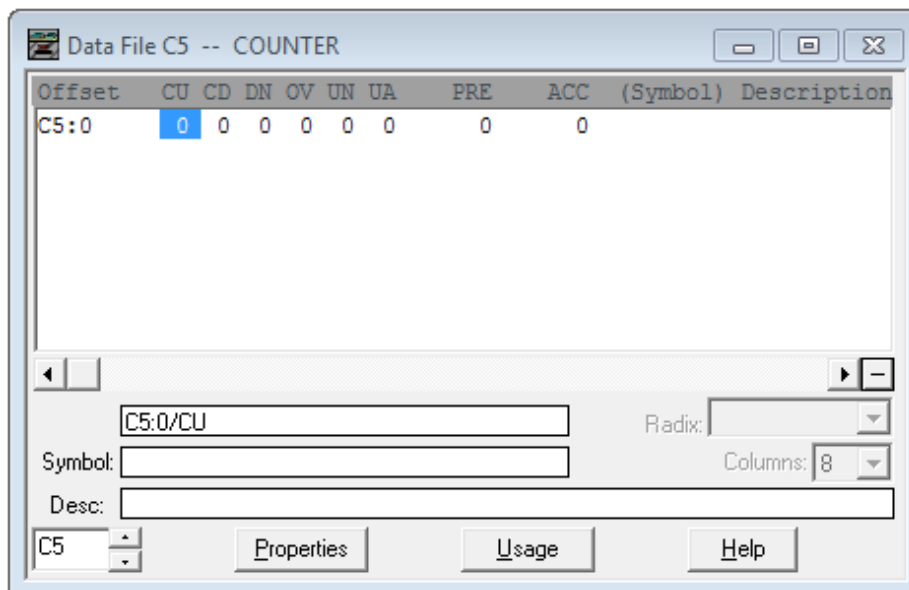


Figure 1.

Crimson

Crimson mappings are slightly different than RSLogix 500 and use the following format:

- C(File):Element, this is the same as RSLogix 500
- Counters are broken down into 3 pieces:
 - 0 – STAT Status word
 - 1 – PRS Counter Preset
 - 2 – ACC Counter Accumulated Value.
- To map a tag to the count up enable bit above (C5:0/CU in RSLogix 500) in Crimson, you would map a Flag tag to C005:0000 0 - STAT. Set the Treat As: parameter to *Bit Array Little-Endian* and choose *Bit 15* for the Bit Number parameter, as shown in Figure 2.

NOTE: Bit numbers explanations are available via the RSLogix 500 Instruction Set Help function.

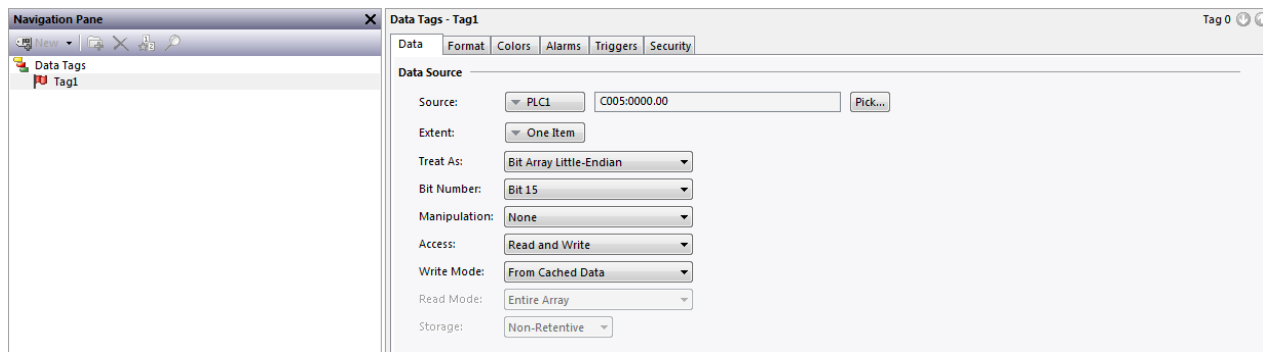


Figure 2.

Floating Point

RSLogix 500

Floating Point Data File addresses are in the following format:

- F(File):Element, ex. F8:3

Crimson

Crimson mappings are similar, but with leading 0s; F008:0003, for example.

Input

RSLogix 500

Input Point Data File addresses are in the following format:

- I:(Slot).Word/Bit
 - I is for Input
 - Slot is the backplane location of the input module
 - Word is which word of the card contains the data.

NOTE: RSLogix 500 uses a “/” to signify a bit within a word or long.

- The item highlighted in Figure 3 is displayed as I:1.0/0 in RSLogix 500.

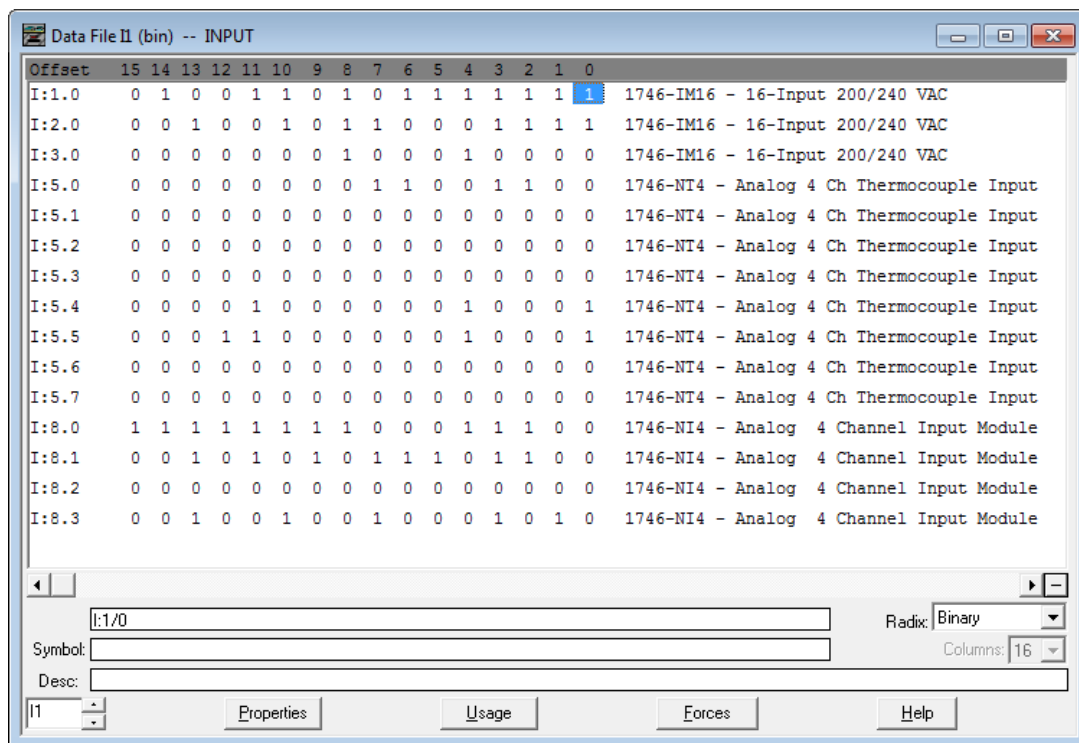


Figure 3.

Crimson

Crimson mappings are slightly different than RSLogix 500 and use the following format:

- I(Slot):Word.Bit
 - I is for Input
 - Slot is the backplane location of the input module
 - Word is which word of the card contains the data.

NOTE: Crimson uses a “.” to signify a bit within a word or long.

- To map a tag to the input above (I:1.0/0 in RSLogix 500) in Crimson, you would map a tag to I001:0000. Then, set the Treat As parameter to *Bit Array Little-Endian* and choose *Bit 0* for the Bit Number parameter as shown in Figure 4.

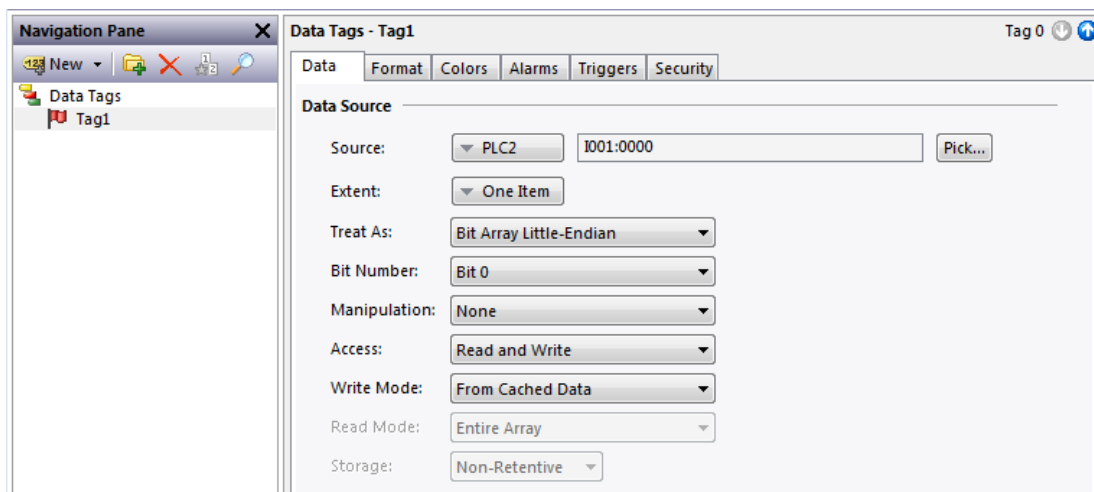


Figure 4.

Long

RSLogix 500

Long Data File addresses are in the following format:

- L(File):Element, ex. L9:7

Crimson

Crimson mappings are similar, but with leading 0s; L009:0007, for example.

Integer

RSLogix 500

Integer Data File addresses are in the following format:

- N(File):Element, ex. N7:0

Crimson

Crimson mappings are similar, but with leading 0s; N007:0000, for example.

Output

RSLogix 500

Output Point Data File addresses are in the following format:

- O:(Slot):Word/Bit
 - O is for Output
 - Slot is the backplane location of the input module
 - Word is which word of the card contains the data.

NOTE: RSLogix 500 uses a “/” to signify a bit within a word or long.

- The item highlighted in Figure 5 is displayed as O:9.2 in RSLogix 500.

Offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
O:4.0	1	1	0	1	1	0	1	0	0	0	0	0	1	0	1	1	1746-OW16 - 16-Output (RLY) 240 VAC
O:5.0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:5.1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:5.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:5.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:5.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:5.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:5.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:5.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NT4 - Analog 4 Ch Thermocouple Input
O:6.0	1	0	0	0	1	1	0	1	1	0	0	0	0	0	1	0	1746-OW16 - 16-Output (RLY) 240 VAC
O:7.0									1	0	1	1	1	0	0	0	1746-OX8 - 8-Output Isolated Relay
O:9.0	0	1	0	0	1	1	0	0	1	1	0	0	0	1	1	0	1746-NO4I - Analog 4 Ch. Current Output
O:9.1	0	0	0	1	1	0	0	0	0	1	1	0	0	0	1	0	1746-NO4I - Analog 4 Ch. Current Output
O:9.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NO4I - Analog 4 Ch. Current Output
O:9.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1746-NO4I - Analog 4 Ch. Current Output

Symbol: O:9.2 Radix: Binary Columns: 16

Buttons: Properties, Usage, Forces, Help

Figure 5.

Crimson

Crimson's mapping is slightly different than RSLogix 500 and uses the following format:

- O(Slot):Word.Bit
 - O is for Output
 - Slot is the backplane location of the input module
 - Word is which word of the card contains the data.

NOTE: Crimson uses a "." to signify a bit within a word or long.

- To map a tag to the output above (O:9.2 in RSLogix 500) in Crimson, you would map a tag to O009:0002 as shown in Figure 6.

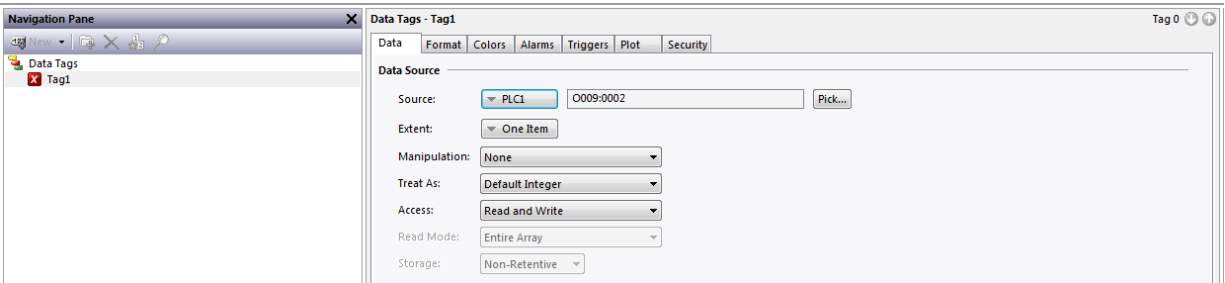


Figure 6.

String

RSLogix 500

String Point Data File addresses are in the following format:

- ST(File):Element ex. ST10:0
- Strings are broken down into 2 pieces
 - LEN
 - String Text.

Crimson

Crimson only has a single character space available, so strings are referred to as R registers in Crimson; the File and Element values also have leading zeros; ex. R010:0000, for example.

- After mapping the Crimson tag to the String register, the Length parameter MUST be set to 80 characters, as shown in Figure 7.

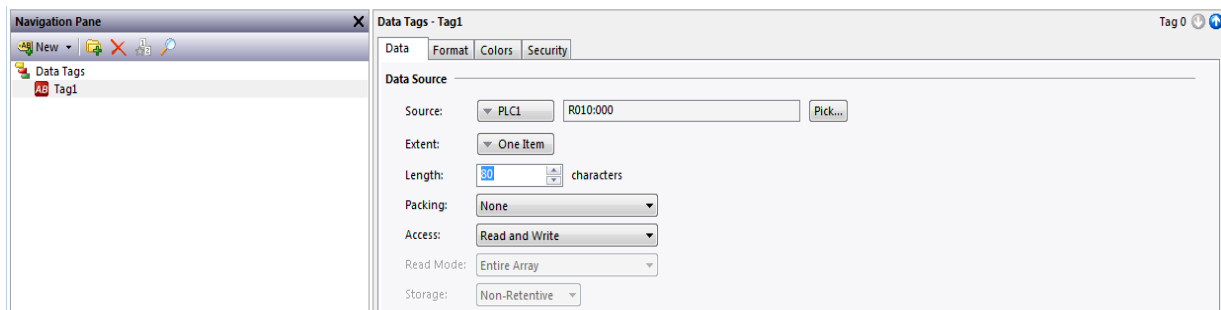


Figure 7.

Status

RSLogix 500

Status Data File addresses are in the following format:

- S:Element, ex. S:42 (Clock Calendar Seconds)

Crimson

Crimson mappings are similar, but with leading 0s; S:0042, for example.

Timer

RSLogix 500

Timer Data File addresses are in the following format:

- T(File):Element ex. T4:0
- Timers are broken down into 6 pieces:
 - EN
 - TT
 - DN
 - BASE
 - PRE
 - ACC.
- The item highlighted in Figure 8 displayed as T4:0/TT in RSLogix 500.

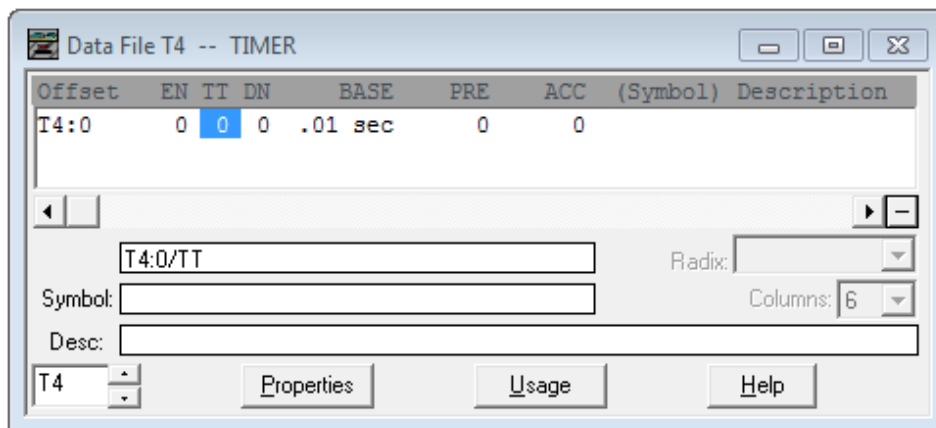


Figure 8.

Crimson

Crimson's mapping is slightly different than RSLogix 500, using the following format:

- T(File):Element, this is the same as RSLogix 500
- Timers are broken down into 3 pieces:
 - 0 – STAT Status word
 - 1 – PRS Timer Preset
- 2 – ACC Timer Accumulated Value. To map a tag to the count up enable bit above (T4:0/TT in RSLogix 500) in Crimson you would map a Flag tag to T004:0000 0 - STAT. Set the Treat As: parameter to *Bit Array Little-Endian* and choose Bit 14 for the Bit Number, as shown in Figure 9.

NOTE: Bit numbers explanations are available via the RSLogix 500 Instruction Set Help function.

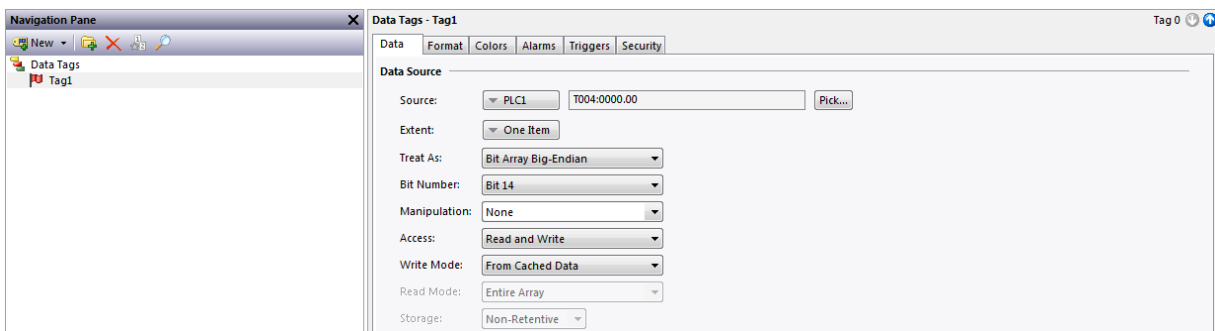


Figure 9.

RSLogix 5

Previously described mapping, with the exception of the Input and Output data types, holds true for PLC 5 devices. The information in this section applies to RSLogix 5-based processors/devices, only.

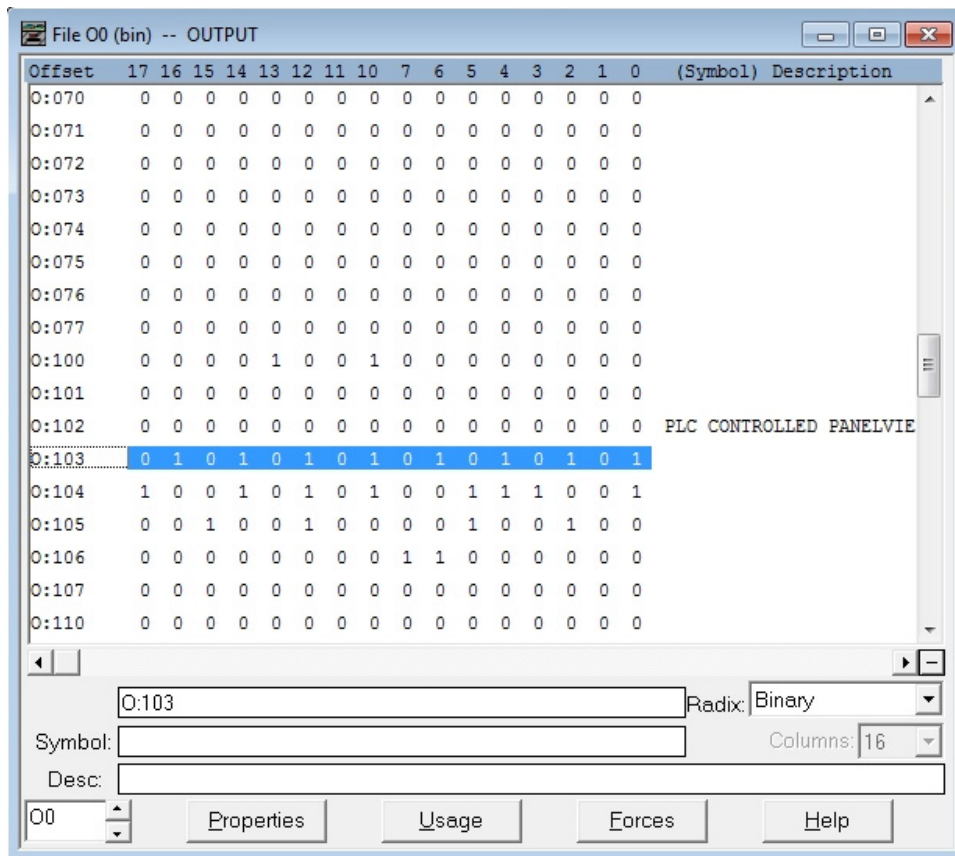
RSLogix 5

Input and Output Point Data File addresses are in the following format:

- O:Word/Bit
 - O is for Output
 - Word is in OCTAL

NOTE: RSLogix 5 uses a “/” to signify a bit within a word or long.

- The item highlighted in the following screen capture is displayed as O:103 in RSLogix 5:



Crimson

Crimson's mapping is slightly different than RSLogix 5, it uses the following format:

- O(Slot):Word.Bit
 - O is for Output
 - Word is which word of the card contains the data, in DECIMAL

NOTE: Crimson uses a "." to signify a bit within a word or long.

To map a tag to the previous output (O:103 in RSLogix 5) in Crimson, you would map a tag to O00:0067.

Disclaimer

It is the customer's responsibility to review the advice provided herein and its applicability to the system. Red Lion makes no representation about specific knowledge of the customer's system or the specific performance of the system. Red Lion is not responsible for any damage to equipment or connected systems. The use of this document is at your own risk. Red Lion standard product warranty applies.

Red Lion Technical Support

If you have any questions or trouble contact Red Lion Technical Support by emailing support@redlion.net or calling 1-877-432-9908.

For more information: <http://www.redlion.net/support/policies-statements/warranty-statement>

