

TB164 (Rev1) - Using Custom PLC Messaging

Overview

This document provides technical information for PLC programmers desiring to use custom PLC messaging. As of software version v8.22 (DOS) and for the future Linux releases, additional support has been added that extends the custom messaging ability between XPLC programs and CNC software. The XPLCCOMP compiler and executor have been modified to support a MSG command and the CNC software has been modified to parse a text file of messages (plcmsg.txt) and respond to message requests from the PLC. A new machine parameter (0) has been added to specify the PLC bit number to which the physical E-stop is connected.

How custom PLC messaging works

There are two types of messages presently defined. The first type is called the stop message type (synchronized with the stop bit) and the second is called the asynchronous type. The stop type messages are displayed when the stop bit (OUT75) is set in the PLC program. If the PLC program sets the stop bit, then the CNC software searches for a stop type message and displays the custom message instead of the "Emergency stop detected" message, provided that the E-stop PLC bit number (as specified in machine parameter 0) does not indicate that it was caused by the physical E-stop. The asynchronous type message is displayed as soon as it is received.

A raw message is a 32-bit integer, of which the lower eight bits specify the type of message and the upper 24 bits specify the message number. This allows for over 16 million unique messages to be organized into 256 different types. Presently, the following types have been defined:

1 – Stop

2 - Asynchronous

A raw message number of 0 indicates that there is no message to be communicated. Types 3-255 are reserved for future expansion. The table below shows some examples of raw message numbers broken down into the type and number fields.

Raw	Type	Number	Notes
257	Stop (1)	1	$257 = 1 + 1*256$
514	Async (2)	2	$514 = 2 + 2*256$
2561	Stop (1)	10	$2561 = 1 + 10*256$
2818	Async (2)	11	$2818 = 2 + 11*256$

The XPLC source program sets the raw message number by using the MSG action command. The format of this command is:

MSG Wn

...where Wn represents one of the word memory locations W1-W256. The CNC software queries the PLC executor as part of its background processing. When a non-zero raw message number is read, the CNC software will look at the message number (upper 24 bits) and attempt to find a defined message for this

number. If no message with this number is found, the message "UNDEFINED PLC MESSAGE (xxxx)" appears in the message window, where xxxx is the message number. Message numbers and their accompanying message text are defined by creating a text file named plcmsg.txt consisting of lines in the following format:

message_number message_text

Each defined message is a separate line. Note that messages are truncated to approximately 40 characters for display purposes. If message_text starts with a number, then this message will also be logged and can be viewed from the F7 – Utility, F9 – Logs, F1 – Errors screen. Here is an example plcmsg.txt file:

```
1 Custom Message 1
2 9002 Custom Message 2
```

Example:

The example PLC source codes and plcmsg.txt files shown below will display the state of INP1 whenever a change in state occurs. Be sure to set machine parameter 0 = 11.

Sample source for cnc10.plc (or cnc7.plc):

```
out75 = inp11 ; E_stop is INP11
out76 = mem1 ; PLC operation bit = 0 (to prevent halting CPU)
```

Sample source for PC.PLC:

```
if l==1 then set mem49
if inp1 then (pd1)
if pd1 then w1 = 2 + 1*256, msg w1
if !inp1 then (pd2)
if pd2 then w1 = 2 + 2*256, msg w1
```

Sample plcmsg.txt file:

```
1 INP1 OPENED
2 INP1 CLOSED
```

Document History

Rev1 Created on 2004-01-21 by #000