

AB KTx RIO TLM

Table of contents

1 Overview.....	3
1.1 Overview.....	3
1.1.1 Introduction.....	3
2 Warranty.....	4
2.1 Terms of Use.....	4
3 Hardware Configuration.....	5
3.1 Hardware Configuration.....	5
4 Driver Configuration.....	6
4.1 Driver Configuration.....	6
4.1.1 Enabling the TLM.....	6
4.1.2 Configuration File Format.....	8
4.1.2.1 Sample Configuration File.....	8
4.1.2.2 [DRIVER].....	10
4.1.2.3 [CARDS].....	10
4.1.2.4 [INPUTS] and [OUTPUTS].....	11
5 Driver Initialization.....	14
5.1 Driver Initialization.....	14
6 Runtime Errors.....	16
6.1 Runtime Errors.....	16
6.1.1 Scanning Errors.....	16
6.1.2 Block Transfer Errors.....	18
7 TLI Ladder Instructions.....	20
7.1 TLIs (Ladder Instructions).....	20
7.1.1	20

7.1.2 KTXGETFAULTMAP..... 20
7.1.3 KTXRACKEXISTS..... 21
8 All..... 22

1. Overview

1.1. Overview

1.1.1. Introduction

This document describes the installation, usage, and functionality of a TOPDOC Loadable Module (TLM) for [SoftPLC](#) that is used with the Allen-Bradley 1784-KTX, -KTXD, -KTS, -PKTX, -PKTXD, and -PKTS Cards. The TLM works with up to any 8 such boards concurrently in the same computer. (Each -PKTXD and KTXD board counts as two boards because they have two RIO channels on them.) Henceforth let the term "KTx" refer to any of the above cards.

SoftPLC can drive one or more Allen-Bradley KTx card channels in remote I/O scanner mode connected to 1771 I/O, SLC-500 I/O (1746), Block I/O (1791), Flex I/O (1794) or any devices that communicate using Allen-Bradley Remote I/O protocol (RIO). The 1784-KTX and the 1784-KTXD ISA cards differ in that the KTX has only one channel, whereas the KTXD has two channels; a given channel on either the KTX or the KTXD can be configured to communicate on DH+ networks or to act as a remote I/O scanner. The KTS can only be configured to act as a remote I/O scanner and has only one channel. The same can be said for the PCI cards: -PKTX, -PKTXD, and -PKTS, respectively.

Note:

This TLM only concerns itself with RIO protocol. The SoftPLC support for DH+ is in a different document.

This manual discusses how to configure SoftPLC to use the KTx driver, KTX.TLM, as well as details on configuring remote I/O, and on how the driver reports errors.

You may use any of the 3 different cards listed above as a remote I/O scanner. And you can combine cards to use as many as 8 KTx I/O channels concurrently. A common configuration is to use one of the channels of a KTXD or PKTXD for remote I/O, and the other for O.N.E. communications to Data Highway Plus. (Again, this document only addresses remote I/O.)

Each KTx card/channel can support a maximum number of remote I/O adapters, and this maximum is determined by which of these two constraints is encountered first:

1. A maximum of 32 remote I/O adapters, regardless of the size of the rack that those adapters reside in.
2. A maximum of 256 input words and 256 outputs words. Using 8 words per "logical rack" limits you to 32 logical full 128 point racks.

2. Warranty

2.1. Terms of Use

Because of the variety of uses of the information described in this manual, the users of, and those responsible for applying this information must satisfy themselves as to the acceptability of each application and use of the information. In no event will SoftPLC Corporation be responsible or liable for its use, nor for any infringements of patents or other rights of third parties which may result from its use.

SOFTPLC CORPORATION MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE.

SoftPLC Corporation reserves the right to change product specifications at any time without notice. No part of this document may be reproduced by any means, nor translated, nor transmitted to any magnetic medium without the written consent of SoftPLC Corporation.

SoftPLC, and TOPDOC are registered trademarks of SoftPLC Corporation.

© Copyright 2005-2009 SoftPLC Corporation ALL RIGHTS RESERVED

First Printing: February, 1995

Latest Printing: August, 2009

SoftPLC Corporation
25603 Red Brangus Drive>
Spicewood, Texas 78669
USA Telephone: 1-800-SoftPLC
Fax: 512/264-8399
URL: <http://softplc.com>
Email: support@softplc.com

3. Hardware Configuration

3.1. Hardware Configuration

You need 3 pieces of information to setup the hardware for each KTx I/O channel:

1. How to setup the IRQ.
2. How to setup the dual port memory address.
3. How to wire and connect the remote I/O cabling.

Note:

For any *PCI* -KTx card, items 1 and 2 from the above list are handled automatically for you by the PCI bios or the Linux kernel. On the other hand, for *ISA* cards, see Allen-Bradley's publication **1784-6.5.22, Communication Interface Card (Cat. Nos. 1784-KTX, 1784-KTXD, and 1784-KTS) User Manual**. This publication comes with the KTx card from Allen-Bradley. Most all you need is in chapters 2 and 4 of that manual.

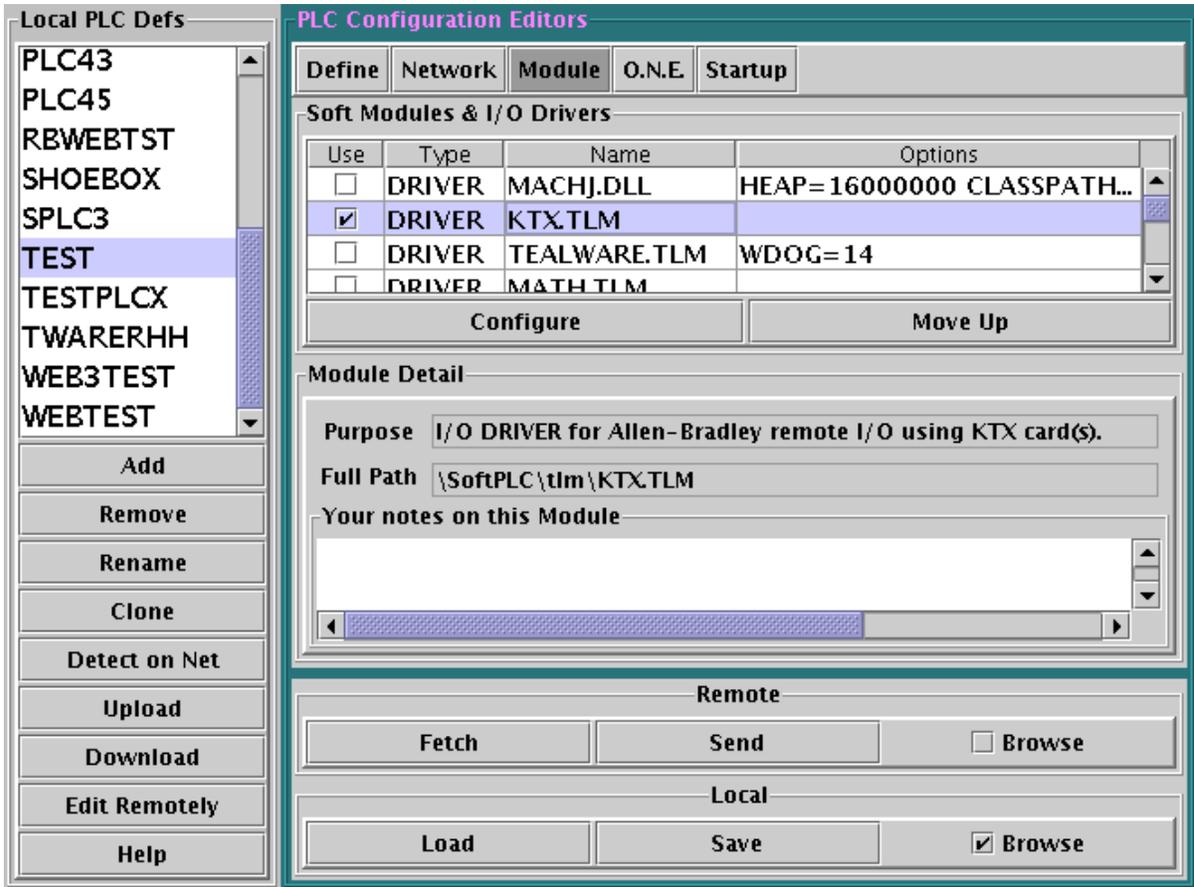
4. Driver Configuration

4.1. Driver Configuration

On version 3.x SoftPLC, the driver exists entirely within the TLM. But on version 4.x SoftPLC, the driver code is split between a Linux kernel driver and the TLM. Therefore, there is an extra step required when using version 4.x that is not required for version 3.x SoftPLC. Instructions on how to enable the Linux kernel driver follow. If you are using version 3.x SoftPLC, you can skip ahead to the next section, [Enabling the TLM](#).

4.1.1. Enabling the TLM

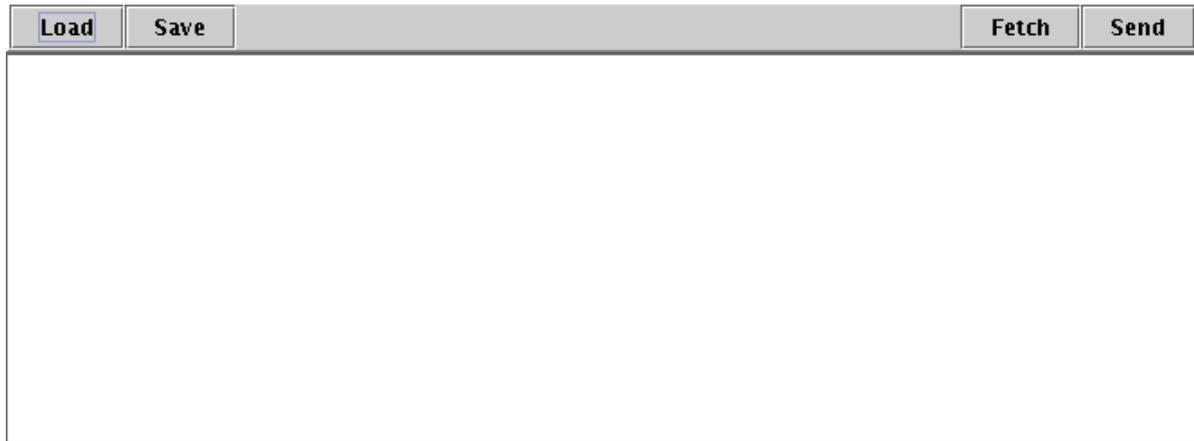
To make the driver available to SoftPLC, you have to enable it through the **Module** tab within the PLC Configuration Editor in TOPDOC NexGen:



Module Editor

Click the **Fetch** button to load the existing configuration from the attached SoftPLC. Then find the KTX.TLM driver from the list under **Name**, select the **Use** checkbox. Then click on **Send**. This will tell the SoftPLC to use the TLM after its next power cycle. But before power cycling the SoftPLC, you will edit the TLM's configuration file.

To edit the configuration file, click on **Configure**, and the following editor comes up. From here you can **Fetch**, **Send**, **Save**, or **Load** the configuration file.



Text File Editor

Click the **Fetch** button to retrieve the configuration file from the SoftPLC, or **Load** if you already have it saved from before.

Make the necessary edits to map your specific I/O, as described within this chapter.

Then **Save** to store the edited file on your local drive, and **Send** to download the new configuration to your SoftPLC.

Before power cycling SoftPLC, you should make sure your I/O racks are powered up. The TLM will not be fully function unless the racks are powered up and the cabling is intact. You can now power up SoftPLC and watch the syslog for version 4.x, or the console for version 3.x SoftPLC.

4.1.2. Configuration File Format

This section describes the format of the ASCII text configuration file used by the driver. There are a number of regions or sections within the file, each identified by a keyword surrounded by square brackets:

- [DRIVER]
- [CARDS]
- [INPUTS]
- [OUTPUTS]

4.1.2.1. Sample Configuration File

```
;Configuration file for KTX.TLM.
;Any part of any line in this text file to the right of a semicolon is a
```

```
;comment and will be ignored by the TLM.

;There are 4 sections in this file: [DRIVER], [CARDS], [INPUTS],
;and [OUTPUTS].

;[DRIVER] is a place for global parameters, and is not used yet.

;[CARDS] lists each KTx card in the system on a separate line.
; There are columns for physical ADDRESS, IRQ, BAUDRATE, and WATCHDOG.
; BAUDRATE as one of 57, 115, or 230 kbaud, and the WATCHDOG timeout
; value in msec. Each card gets an implied CARDNUMBER based on
; the order of presentation, the first being card 0.
; The ADDRESS is different depending on whether the card is an ISA
; or PCI card. If ISA, then the 20 bit physical address is given,
; e.g. "D8000". If PCI, then "PCI" is given because the pci bios
; will assign the card its physical ADDRESS. The IRQ is likewise
; different depending on whether the card is an ISA or PCI card.
; For ISA use the irq number according to the jumper, for PCI cards,
; use "PCI" because the pci bios will assign the IRQ.

;[INPUTS] lists blocks of words that are to be copied from each
;card into the SoftPLC input image table as part of the I/O scan.
;Only I:xxx type addresses may be used here. UseableAddr is the
;memory address within SoftPLC. PhysicalAddr is the address that
;is determined by the physical rack and slot used in the field.
;Each row in this section describes a block whose length is
;given by NumWords. A block may be as long as you like, and
;sometimes you will have only one block, other times you might
;have over one hundred. This scheme lets you map any PhysicalAddr
;to any real input UseableAddr.

;[OUTPUTS] lists blocks of words that are to be copied from
;SoftPLC's output image table onto a card as part of the I/O scan.
;Only O:xxx type addresses may be used here. UseableAddr is the
;memory address within SoftPLC. PhysicalAddr is the address that
;is determined by the physical rack and slot used in the field.
;Each row in this section describes a block whose length is
;given by NumWords. A block may be as long as you like, and
;sometimes you will have only one block, other times you might
;have over one hundred. This scheme lets you map any UseableAddr
;to any one or more PhysicalAddr's. The same UseableAddr may
;be mapped to more than one output PhysicalAddr.

;A "word" corresponds to a module group, and this is the granularity
;given by this configuration strategy.
;Unlike AB PLC's, with this SoftPLC KTX driver, both the I:ABC and O:ABC
;at a given ABC UseableAddr element may both be used simultaneously.

[DRIVER]
; not used yet.
```

```
[ CARDS ]
;ADDRESS      IRQ      BAUDRATE      WATCHDOG      CARDNUMBER
PCI ,         PCI ,    57 ,          500            ;0
;d6000 ,     10 ,     115 ,        500            ;1
;d5000 ,     14 ,     230 ,        500            ;2

[ INPUTS ]
;UseableAddr, NumWords, CardNumber, PhysicalAddr
I:000         256      0          I:000
;I:400        256      1          I:000

[ OUTPUTS ]
;UseableAddr, NumWords, CardNumber, PhysicalAddr
O:000         256      0          O:000
;O:400        256      1          O:000

;EOF
```

4.1.2.2. [DRIVER]

This section is for future possible global driver information and is currently not used.

4.1.2.3. [CARDS]

There may be up to 8 cards, with each card listed on its own row. A "card" is a RIO channel, and may be one of two on a KTXD card, or may be the only RIO channel on a KTS or KTX card. In this section you list for each "card":

Note:
As of this writing, only the version of this driver for 4.x SoftPLC supports PCI cards. The 3.x version supports only ISA cards, and the 4.x version supports both ISA and PCI cards. Bear this in mind when interpreting the following table.

Column Name	Description
ADDRESS	For a PCI card: simply "PCI". For an ISA card: the base physical memory address is given in 20 bit address form. This is a 5 digit hex number. For example D000:0000 would be given as D0000. D000:8000 would be given as "D8000".
IRQ	For a PCI card: simply "PCI". For an ISA card: this is the unique IRQ jumper position chosen on this RIO channel. Legal values are 3 through 15 decimal. The IRQ must match the setting on the KTx for this card. For version 4.x SoftPLC, the

	IRQ may be shared by other devices, but not if using version 3.x SoftPLC. See Allen-Bradley's KTx manual for potential conflicts. Recommended values are 5, 10, or 11.
BAUDRATE	Provides the baudrate as one of 57, 115, or 230, meaning 57.6 kbaud, 115.2 kbaud, or 230.4 kbaud respectively.
WATCHDOG	Provides the watchdog timer setting for the card in msec. If the card does not hear from SoftPLC within this interval, the card will shutdown all outputs. Use zero (0) to disable the watchdog function. A normal value is something like 3-4 times your largest anticipated program scan time. Most applications can live with 500 milliseconds. If you set it too close to the scan time, then you may get extraneous watchdog timeouts triggered by the KTX.

4.1.2.4. [INPUTS] and [OUTPUTS]

SoftPLC's Input Image table consists of a block of 16 bit words, with a maximum block size being either 8, 64, 128, or 512 16-bit words, depending on whether you have the 128, 1K, 2K, or 8K kernel respectively. There is an equal sized table for the Output Image. These I/O image tables support I/O forcing. I/O forcing is very useful for digital inputs and digital outputs.

Normally analog data is handled differently than digital data in this driver, it uses Block transfer and analog or smart module data is not confined to the I/O image tables. (The exception is SLC-500 I/O which gives you the option of avoiding block transfer altogether.) Although block transfer is supported by this driver, it is an unfortunate burden that other drivers do not bear, and is best understood by consulting the Allen-Bradley documentation. It is not discussed here because SoftPLC is compatible with a PLC-5 in this regard.

The rows in this section of KTX.LST each define a different kind of block, and are not related to **block transfer ladder instructions** used to transfer smart module data.

Note:

When programming block transfer instructions, you should use the RackGroupSlot values as they relate to the UseableAddr fields, not PhysicalAddr fields.

Here is how to set up blocks for non-smart module data, i.e. discrete data. You may have rows in the file that are commented out with a semi-colon.

Column Name	Description
UseableAddr	This is the starting memory address within SoftPLC that will be used for a block of contiguous words. It must be I:nnn or O:nnn for INPUTS and OUTPUTS respectively, where <nnn> is the word element with in the SoftPLC datatable. Note that INPUTS must go into the I: section and OUTPUTS must go into the O: section of the datatable with this driver.
NumWords	This gives the number of words that are to be transferred as part of this block. The range can be from 1 to 256 inclusive, but must not cause the block to extend beyond the end of the image table, given the starting address in UseableAddr.
CardIndex	This is simply the 0 based index into the list of [CARDS]. The first card given in the [CARDS] section has a CardIndex of 0. The next has 1, etc.
PhysicalAddr	This is the address that is determined by the physical rack and slot used in the field as if you had a PLC-5 hooked up to the remote I/O network.

Here is an example [INPUTS] section:

```
[ INPUTS ]
;UseableAddr, NumWords, CardNumber, PhysicalAddr
I:000      256      0      I:000
I:400      256      1      I:000
```

When deciphering the I:nnn or O:nnn notation, you can think of breaking the <nnn> down into 2 fields: RRG, where RR is 2 digits for a Rack number and G is one digit for a module Group number. So I:123 means physical rack 12 and group 3. These digits are always in octal (according to Allen-Bradley PLC-5). A module group maps to a single word within the PLC.

On any given RIO blue hose segment attached to a KTx RIO channel, the highest allowed rack number may be octal 37, so this means a maximum PhysicalAddr of O:377 and I:377. Not so in SoftPLC, where the highest allowed rack number is octal 77 in the 8K SoftPLC, so UseableAddr?s can go up to O:777 and I:777. Therefore, to support the higher numbered racks in the 8K SoftPLC, we must be able to conceptually map a physical rack to a higher UseableAddr. Notice the 2nd row in the above example, which moves what is a rack with physical adapter switch settings at 00 to a UseableAddr designating rack 40 octal. All the racks on the 2nd KTx RIO channel/card have been offset to a higher UseableAddr.

In reality, with SoftPLC you have the flexibility of letting go of the strict hardware to logical mapping that an AB PLC-5 restricts you to. You can map any module group to an input image table word, so this means you can write a program without regard for the actual final addresses. It also means you can use both I:xyz and O:xyz at the same time for any given xyz. This is what AB calls complementary I/O, but SoftPLC's scheme is much more flexible because each UseableAddr can come from a different KTx card and rack.

Others who value the relationship between physical rack topology and UseableAddr mapping that AB PLC's limit you to, will want to stick with that scheme. This is merely a matter of defining fewer larger blocks in the [INPUTS] and [OUTPUTS] section.

Either technique works.

All the Physical Racks on any particular KTx I/O channel must be uniquely numbered. However, the same Physical Rack (number!) may be present on more than one I/O channel.

Any UseableAddr's not referenced as part of a block in KTX.LST, are not touched by the KTx driver and may be used by other I/O drivers within SoftPLC.

The SoftPLC KTx driver, KTX.TLM, loads several files at startup. All these files are part of the normal SoftPLC installation and must be in the same directory as the KTX.TLM file:

- KTX.LST
- ONEKTX0.BIN
- ONEKTX1.BIN
- ONEKTX2.BIN

5. Driver Initialization

5.1. Driver Initialization

Driver Initialization is a special phase of operation that happens only under either of two special circumstances, described below.

During driver initialization, the SoftPLC KTx driver uses the AUTOCONFIGURE feature of the KTx card to temporarily set the KTx's scan list. In other words, the KTx will attempt to find all powered up and communicating adapters on the I/O link. Partial rack configurations are also automatically recognized in this process.

After the temporary list of all adapters is obtained via the KTx AUTOCONFIGURE feature, then two additional checks are done by KTX.TLM:

1. Check that all **PhysicalAddrs** in KTX.LST may be serviced by an RIO adapter which is known to be present. If some needed adapter is missing, then the entire SoftPLC will not be able to enter RUN mode. (The [INPUTS] and [OUTPUTS] sections of KTX.LST determine which adapters are "needed". If you wish to not use an adapter, then you should comment out or delete those blocks in KTX.LST that reference the PhysicalAddrs that fall under the domain of that adapter. Or you can reduce the NumWords in a block.)
2. Check that all adapters found are actually fulfilling some function with respect to the **PhysicalAddrs** in KTX.LST. Any unused but available adapters are removed from the scan list and will be ignored by the KTx card and SoftPLC.

The KTX.LST file is read, parsed, and the above driver initialization procedure happens under either of these two conditions:

1. During a SoftPLC REM/PROG to REM/PROG mode change, with no intervening mode. You can cause this to happen with TOPDOC, and our support group refers to this as **double clutching** the system.
2. When the KTx driver is initialized during SoftPLC start up, i.e. when SoftPLC is first loaded into memory. This means that **the I/O racks need to be powered up no later than the SoftPLC computer is**, otherwise you will have to invoke condition 1) just above.

Therefore, to add a new rack to the I/O link, the user must change KTX.LST, then change SoftPLC's mode from REM/PROG to REM/PROG, or shut down and then restart SoftPLC.

Note:

Inhibiting racks is not supported by setting "rack inhibit" bits in the datatable. It was felt that this cost too much CPU time for a feature that can be accomplished in a better way. Instead, you must remove all the blocks from the KTX.LST file that have PhysicalAddrs that fall under the domain of the undesired adapter(s).

6. Runtime Errors

6.1. Runtime Errors

6.1.1. Scanning Errors

The KTx card reports processing problems by putting a fault code into status word 15. Your ladder logic should monitor this word, copy it when not zero, and reset S15 to zero, since the driver never resets the word. The meaning of fault and error codes stored in S15 is documented in the table below and this information supersedes the similar information in the TOPDOC for SoftPLC help system.

Some of these errors occur only during driver initialization discussed above, and this subset of errors will prevent you from placing the system into REM/RUN mode. In order to enter RUN mode, ALL needed adapters must have been powered up and communicating during driver initialization, and S15 should reflect a value of 0 at that time. Otherwise the next attempted transition to RUN mode will fail, even if that is months later. The only way to catch these errors is to either have a monitor hooked up to SoftPLC so you see them on the console, or you can use TOPDOC or your MMI to read from SoftPLC while it is in FAULT mode. SoftPLC will not enter RUN mode under these circumstances.

Other errors can occur while in RUN mode, and these fall into two subsets: those that will fault the CPU and those that will not. When the CPU faults, it turns off all OUTPUTS to the best of its ability and enters FAULT mode.

The one error that you should pay most attention to is number 5324. This can happen in RUN mode and indicates that at least one I/O rack adapter is not communicating properly. (Remember that in order to enter RUN mode, ALL needed adapters had to have been communicating during driver initialization. So this means that an adapter that once was working is now not working.) Error 5324 will not FAULT the CPU. If you want it to, then you will need to add logic to EQU word S15 for 5324 and then set bit S11/6, setting bit S11/6 should fault the CPU. Test this logic to verify it actually faults the CPU as you desire. To determine which I/O rack adapters are faulted, use the TOPDOC Loadable Instruction (TLI) named KTXGETFAULTMAP described in the TLI section.

S15 Error Code	Description
5301	Illegal baudrate in the KTX.LST file
5302	The MemAddr field in KTX.LST for a CARD is not correct, or the card is conflicting with another device at the same memory address.

5303	One of the files ONEKTX*.BIN is missing.
5304	KT diagnostics timeout at CARD load time.
5305	One of the KT startup diagnostics tests failed.
5306	A mode change was attempted while one of the CARDS was faulted.
5307	A file read error happened while reading one of the ONEKTX*.BIN files.
5308	A CARD is not processing load requests properly.
5309	A CARD did not respond to the IRQ test. The IRQ setting for a card is wrong or it is conflicting with another device. KTX.TLM dis not see the interrupt as it should have.
5310	A CARD concluded that it was not ready to run.
5312	A mode change was denied by a CARD.
5313	KTX.TLM timed out a CARD waiting for it to change mode.
5314	The KTX.TLM timed out a CARD waiting for it to AUTOCONFIGURE the adapters.
5315	The KTX.TLM received an error response from a CARD while doing an AUTOCONFIGURE.
5316	A CARD was flagged as dead by KTX.TLM because it was not responsive to the WATCHDOG checks done by the KTX.TLM.
5317	The service thread could not be created. There may not be enough RAM in the host.
5318	KTX.TLM timed out a card waiting for it to complete the GET_SCANLIST command.
5319	The KTX.TLM detected that a CARD has been marked as dead.
5320	You have an [INPUTS] or [OUTPUTS] block in your KTX.LST for which there is no powered up, communicating adapter.
5324	At least one of your I/O rack adapters is faulted.

5325	There was insufficient RAM in the host for the BLOCK TRANSFER buffers.
5326	A CARD did not prepare its input mailbox in time.
5331	KTX.LST is not found in the same directory as KTX.TLM.
5332	KTX.LST has a format error in it.
5333	The event flag could not be created.

6.1.2. Block Transfer Errors

All block transfer errors are stored in the 3rd word of the respective block transfer "control block". When using BTW or BTR instructions, it is recommended that you use the BTxxx:yyy datatable elements to store the control block information rather than the N (integer) blocks. The advantage of BT elements for control blocks, is that TOPDOC can present a more discernable view of the control block, and will name the 3rd word DLEN. In either case, N or BT control block, upon a successful block transfer the 3rd control block word contains the number of 16 bit words actually transferred and the DN bit comes on. Upon a failed block transfer, this 3rd word instead gets a negative error code, and the ER bit is also set in the control block.

The .NR and .TO bits in the block transfer control block are not used in this driver and therefore have no meaning. If an adapter does not respond to a block transfer command within 4 seconds, the corresponding control block will timeout with an error code of -36 in the 3rd word.

Note:

SoftPLC versions 3.x and 4.x do not support "N" type blocks for Block Transfer Instructions.

Control Block Error Codes in 3rd control block word	Description
-28	The Rack Group Slot in the block transfer control block points to an adapter that is not in the scan list.
-30	If even for an instant, you had 2 block transfer requests queued up for the same module group slot. There can only be one active at one time. For bi-directional modules, you must stagger the read and write in time. The PLC-5 Instruction

	Set Reference Guide, section 15, has an example under "Bidirectional Alternating Block Transfer" that will work fine with SoftPLC.
-36	The block transfer timed out. The module did not respond to the request, or you asked for a number of words unexpected by the module.
-38	Host requested a write, module responded with a read request, or vice versa.
-39	Length mismatch between what the host requested and what the adapter requested.
-40	Module requested more data than host supplied.
-50	A block transfer was requested for a module attached to a faulted adapter.

7. TLI Ladder Instructions

7.1. TLIs (Ladder Instructions)

7.1.1.

This driver includes a pair of Ladder Logic instructions, called **TLIs**. For TOPDOC online, these instructions are automatically added to the instruction menu the moment you go online to a SoftPLC that has the KTX.TLM installed.

Note:

The rung logic preceding these instructions must be true for the instructions to operate.

Both instructions work with a block of 32 integer words. Each of the 32 words corresponds to a logical I/O rack, whose number is given by the rack adapter switch settings. Remember, field I/O racks may be placed in any I/O image table location due to the flexibility of KTX.LST. **The TLIs use the rack numbers as they are determined by adapter switch settings, not the mapped logical addresses in the KTX.LST file.**

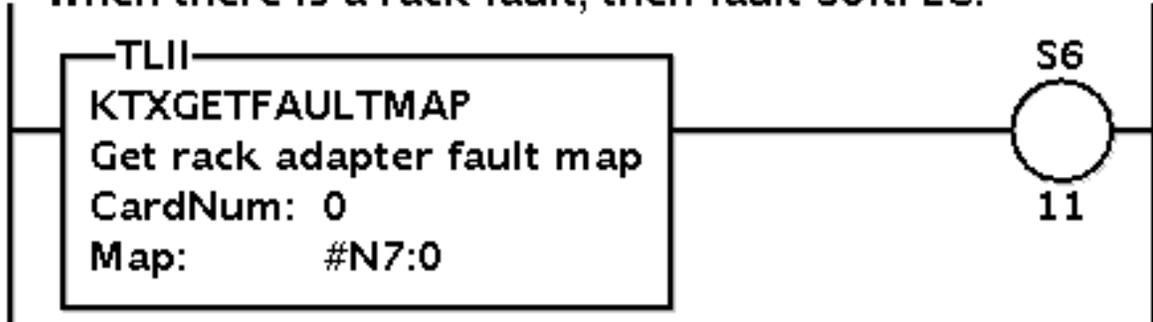
For example, rack octal 017 (decimal 15), would be mapped to the sixteenth word in the block.

7.1.2. KTXGETFAULTMAP

This instruction copies a bitmap of faulted I/O adapters for a particular KTX card into the datatable. This instruction is a permissive instruction, and evaluates as TRUE only if an adapter is faulted. It can be used as a condition for invoking a JSR or setting a summary alarm bit in the same rung. It only writes to the indicated bits and no others. It takes two parameters.

The following example uses a TRUE evaluation result, indicating at least one faulted rack, to turn on bit S6/11 which will then fault SoftPLC.

When there is a rack fault, then fault SoftPLC.



Parameter Name	Description
CardNum	is the zero based index into your [CARDS] section of the KTX.LST file. That is, it refers to a particular card. Zero is the first card, 1 is the second card, etc.
Map	is the start of a 32 word block that will receive the faulted rack adapter bitmap. The format of the bitmap is given below. Each of the 32 words corresponds to a logical rack.

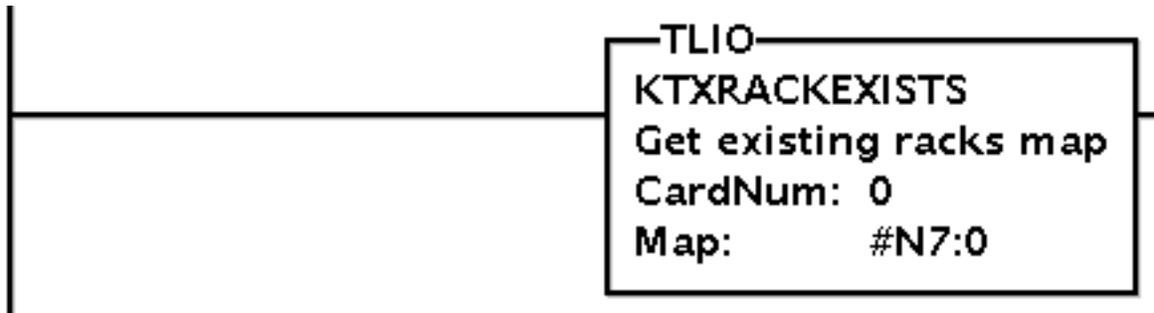
The Map will always be set to zeros unless an I/O rack adapter is faulted. Also, remember from earlier that when an I/O rack adapter is faulted S15 is set to 5324 by the KTX.TLM independent of this TLI. S15 is not ever cleared by the KTX.TLM, you must do that in logic. Only the least significant 4 bits of each word make up the map:

This bit when set	Corresponds to a fault in
0	first 1/4 rack starting I/O group 0
1	second 1/4 rack starting I/O group 2
2	third 1/4 rack starting I/O group 4
3	fourth 1/4 rack starting I/O group 6

7.1.3. KTXRACKEXISTS

This instruction copies a bitmap of existing I/O adapters for a particular KTX card into the datatable. Immediately after loading each KTx card with its firmware, the firmware goes out and detects the existence of racks on the RIO cabling. This instruction is an output instruction. It can be used to fetch the findings of a KTX card/channel. It only writes to the

indicated bits and no others. It takes two parameters:



Parameter Name	Description
CardNum	is the zero based index into your [CARDS] section of the KTX.LST file. That is, it refers to a particular card. Zero is the first card, 1 is the second card, etc.
Map	is the start of a 32 word block that will receive the existence status bitmap. The format of the bitmap is given below. Each of the 32 words corresponds to a full rack.

Below is the layout of the 32 Map words. Only bits 8, 9, 10, and 11 are part of this map:

This bit when set	Corresponds to the existence of
8	first 1/4 rack starting I/O group 0
9	second 1/4 rack starting I/O group 2
10	third 1/4 rack starting I/O group 4
11	fourth 1/4 rack starting I/O group 6

Note:
Both TLIs can point to the same Map block since each TLI only affects the bits indicated, but you would only do this for the same CardNum. Each card should get its own block of Map words.

8. All