TRIKS-C CONTROLLER

Powered by Vixen

By David Rawsthorne, Tim W and John Wilson 22 June 2009

Background

While there are a myriad of computer based controller hardware [i] available that will run under the Vixen software[ii] for controlling Christmas lighting, there were no LED display panels that could be controlled thru Vixen. Robert Jordan decided that the Vixen community needed a LED sign and took it upon himself to design and prototype an LED scrolling board. While Robert had seen LED boards everywhere, he wanted on that allowed you to do text and animation. He envisaged a LED display board that could be your show sign and a place for animations during the show.

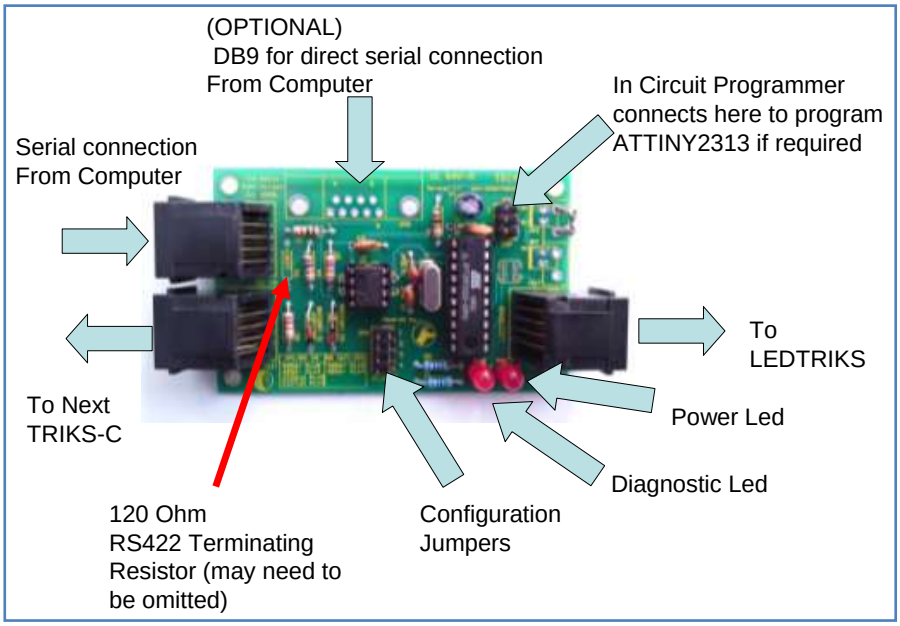
The LEDTRIKS board was popular, and the software control was quickly incorporated into Vixen by K.C. Yet it needed something extra....

Introduction to the Triks-C

The TRIKS-C was designed by TimW, and first introduced to the masses at the Melbourne Mini in 2008 http://au.youtube.com/watch?v=Kx9GqKnV9us

The boards design was expanded to allow in-situ programming of the chip, various connection and capacitor types, as well as diagnostic and power LED's as a collaboration between Wjohn and TimW.

After an initial round of Beta Testing both in Australia and the USA, it was released in time for Christmas 2008.



Why the Triks-C?

Many people felt that the Ledtriks was too much of a CPU hog for what it was doing. Much of the calculation for what was to be displayed was being done by the control computer, and this often caused significant load. In my case one full core of a Quad Core Q6600 was being run at 100% usage for just a single Ledtriks Panel to display a relatively static image.

The Ledtriks keeps control of the 768 LED's on its panel by cycling through them very quickly, faster than you can see. The actual Ledtriks board can only remember and display one row (48 Leds) at a time. So whatever sends information to the Ledtriks has to continuously keep updating the panel, one row at a time, many times a second, regardless of whether the LED's in each row are changing or not. Data, row addressing, clock signals, strobe signals all need to be generated out the parallel port to keep the display 'alive'. (Refer to the Ledtriks Manual for a more technical discussion of how all this refreshing works).

This means a direct connected Ledtriks creates a fair bit of load on the computer driving the board.

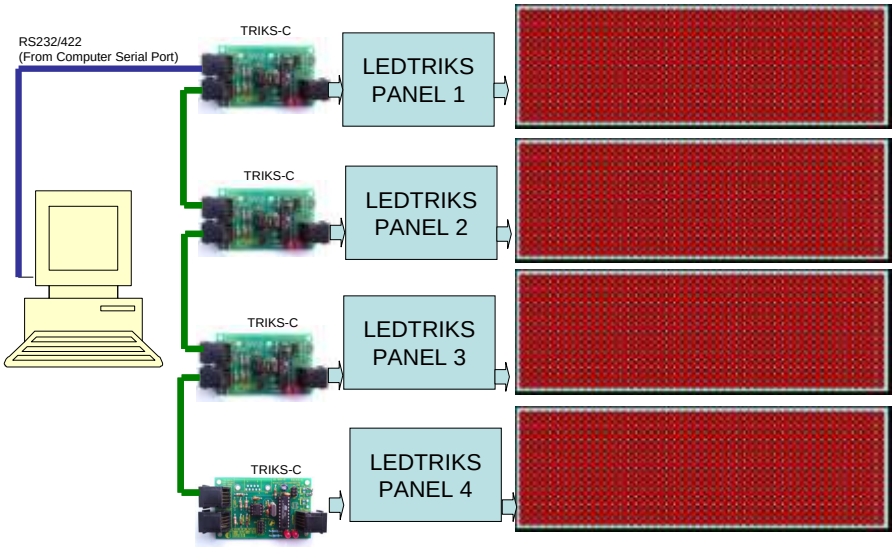
The Triks-C uses a small ATTINY2313 microcontroller to take care of this continuous refreshing. It still receives commands from the computer to update the display but the microcontroller can remember all of the rows at once – so the job of continuously refreshing the rows does not have to be done by the computer.

Because Parallel ports are becoming less common, and USB to Serial adapters or Multi-port serial cards are becoming a common item on our show PC's, moving the Ledtriks from Parallel to Serial seemed like a logical next step. Using a small ATTINY2313 chip and a handful of other parts, the Triks-C takes the majority of the CPU load off the controlling PC as well as giving us an RS232 or RS485 interface to allow easier connections and longer runs while still maintaining a high speed data transfer to the Ledtriks screen.

While it was possible to run the Parallel Ledtriks and Display off one pc, it usually required a separate PC just to control the Ledtriks. With the Triks-C and it's companion program LTC, the whole display can be run of almost any modestly powered PC with no issues at all.

As of the time of writing, you can run a maximum of 4 Triks-C's from the one serial port at once. This gives the user the choice of a single panel, 1 panel high and 2 wide, 1 panel wide and 2 high, or finally 2 panels high and 2 wide.

Each Triks-C interfaces just one Ledtriks board. So for a single Ledtriks you need a single Triks-C, and for 2 Ledtriks boards, you need two Triks-C's. Do not confuse the Ledtriks board with the Ledtriks LED holder board, of which 3 LED holder boards are used to hold the 768 LED's for a single Ledtriks board.



The Triks-C is quite simple in hardware terms. Most of the work is done by the ATTINY2313 microcontroller (Made by ATMEL). The front end provides the means to receive the rs422/serial signals and send them on.

The Triks-C has several features that assist its construction and operation

- Power and Diagnostic Leds
- ICSP programming header for in-circuit programming/updating
- 4 jumpers to select board address and diagnostic/standalone mode
- Power in and out (allowing cascade to Ledtriks)
- RS232/422 in /422 Out (allowing cascade of controllers)

Connecting the Triks-C

The original Ledtriks board was designed to run from a Parallel Port off a PC. The introduction of the Triks C allowed the Ledtriks boards to be run from a Serial Port.

The problem is that there is no direct Triks-c support in Vixen at this time, so a small DOS program was developed by Tim to allow Vixen to drive the Ledtriks. It is expected that once Vixen 2.5 is released, there will be work done on better Ledtriks support by KC.

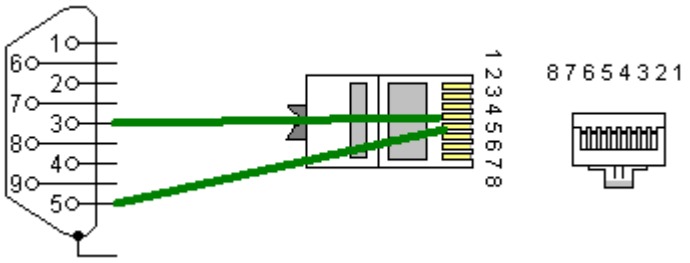
Vixen opens **LTC.exe** (LedTriks-C = L(ed)T(riks)C = LTC.exe) in a DOS window and passes on the required parameters and filenames in order to open the required .led file and send it to the specific Com port where the Triks C is connected.

The **Triks-C** connects to the computer by a Com port such as **COM4:** at 56K, No parity, 8 bits, 1 Stop Bit. This is the same setup as a standard Renard, as the Triks-C uses a modified Renard protocol.

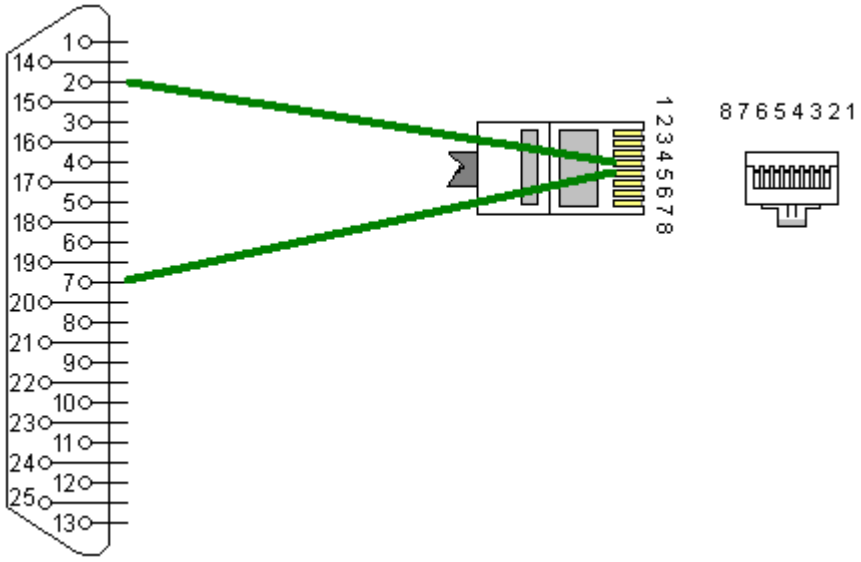
Built-in support for RS232 and RS485 allows the user to select the serial port type depending on the distance from the driving PC. Distances of 15 meters (50 feet) or less are recommended for RS232, but to avoid issues, it is recommended that runs longer than 7.5 meters (25 feet), or those in close proximity to electrical interference use the more noise tolerant RS485. Cheap RS232 to RS485 converters can be built, or sourced from places such as eBay.

The **Triks-C** then connects to the **Ledtriks** via a short Cat 5 cable. In this case, like the connections between the Ren-C's and Grinch/595, the cable should be kept as short as possible. 10 to 15 cm (4 to 6 inches) or less is ideal.

Building the PC-Triks-C cables



RS232 (DB9)		**Triks-C In (RJ-45)**
Pin 3 (TXD)	----------	Pin 4
Pin 5 (GND)	----------	Pin 5



RS232 (DB25)		**Triks-C In (RJ-45)**
Pin 2 (TXD)	----------	Pin 4
Pin 7 (GND)	----------	Pin 5

RS485/422		**Triks-C In (RJ-45)**
(485B-)	----------	Pin 4
(485A+)	----------	Pin 5

Obtaining the board and parts

The Triks-C and Ledtriks boards can be sourced from **Wjohn** in the DoItYourselfChristmas.Com forum. Prices vary, depending on postage and volume, so please contact John for the exact price of your order. The Triks-C contains only a few components, and a Mouser BOM is shown below. The parts can also be ordered from places like Futurlec.com.au, and may be cheaper for Australians, depending on the exchange rate. Note that there are some parts not available from Futurlec, such as the RS-485 Interface IC.

Mouser BOM

Mouser Part Number	Qty per Triks-C	Description
815-AB-11.0592-B2	1	11.0592 MHz Crystal, HC49 case
581-SA102A220JAR	2	Axial Ceramic Capacitors 200volts 22pF 5%
581-SA105E104MAR	2	Axial Ceramic Capacitors 50volts 0.1uF 20%
140-XRL16V10-RC	1	Radial Electrolytic Capacitors 16V 10uF 20% **(See note 1)**
604-WP7104GT	2	3mm Diameter LED Standard GREEN TRANSPARENT **(See note 2)**
595-SN65LBC179P	1	RS-485 Interface ICs LP Diff Pairs
556-ATTINY2313-20PU	1	Microcontrollers - RISC 2kB Flash 0.128kB EEPROM 18 I/O Pins
271-1K-RC	2	1/4W 1% Metal Film Resistors 1Kohms 1%
271-470-RC	2	1/4W 1% Metal Film Resistors 470ohms 1%
271-27K-RC	2	1/4W 1% Metal Film Resistors 27Kohms 1%
271-10K-RC	1	1/4W 1% Metal Film Resistors 10Kohms 1%
271-120-RC	1	1/4W 1% Metal Film Resistors 120ohms 1%
78-1N5229B	1	4.3V, 0.5W Zener Diode
78-1N5239B	1	9.1V, 0.5W Zener Diode
571-5520251-4	3	Modular Jacks 8/8 SIDE ENTRY
538-10-89-7081	1	C-Grid Connectors VERT BRKAWAY HDR 8P dual row tin
538-10-89-7061	1	C-Grid Connectors VERT BRKAWAY HDR 6P dual row tin **(See note 3)**
571-1-390261-6	1	IC Sockets 20P ECONOMY TIN **(See note 4)**
571-1-390261-2	1	IC Sockets 8P ECONOMY TIN **(See note 4)**
636-182-009-213R531	1	9 Pin Serial connection

Notes:

1: 10V capacitor is also acceptable – Mouser p/n 140-XRL10V10-RC

2: 5mm LED's are acceptable, and any solid color can be used. Red, Yellow and Green have all been tested OK by the author of the manual.

3: Used for on-board programming. While this is recommended, this may be left off if not required.

4: IC Sockets are recommended, but may be left off is user wishes to solder IC's direct to board.

Programming the ATTINY2313

The ATMEL ATTINY2313 is programmed in the same way as PICS are programmed (but with a different programmer). There are many circuits available to make programmers capable of programming the chip.

Several DIYC members have built and used a kit programmer from www.ADAFRUIT.com (USBtinyISP AVR Programmer Kit). This sort of programmer connects to the PC via USB and directly onto the TRIKS-C ISP header – the chip can be programmed without moving it from the board!

To talk to the programmer from the PC you will need some software. For the programmer mentioned above (and many others) a good programmer is WinAvr (AVRDUDE) – Get it from :

http://sourceforge.net/projects/winavr/

Note there are other How-To pages on the ADAFRUIT site to setup and use the programmers. Also the program is invoked from the command line (Windows START/Run/type CMD). A Gui front end can be found here:

There are 2 key programming steps to setting up a blank ATTINY2313 chip to work in the Triks-C

Firstly the 'fuses' must be programmed. The fuses tell the chip (among other things) that it has an external crystal that it should use as a clock. By default the intenal crystal will not work (the chip will run slow and serial communication speed will be wrong)

The NORMAL FUSE bits need to be set to 0EFh

The HIGH Fuse bits need to be set to 0DFh

The Extended Fuse bits need to be set to 0FFh

The AVRDUDE command for this (using the above programmer) is:

avrdude -c usbtiny -p attiny2313 -U lfuse:w:0xEF:m -U hfuse:w:0xDF:m -U efuse:w:0xFF:m

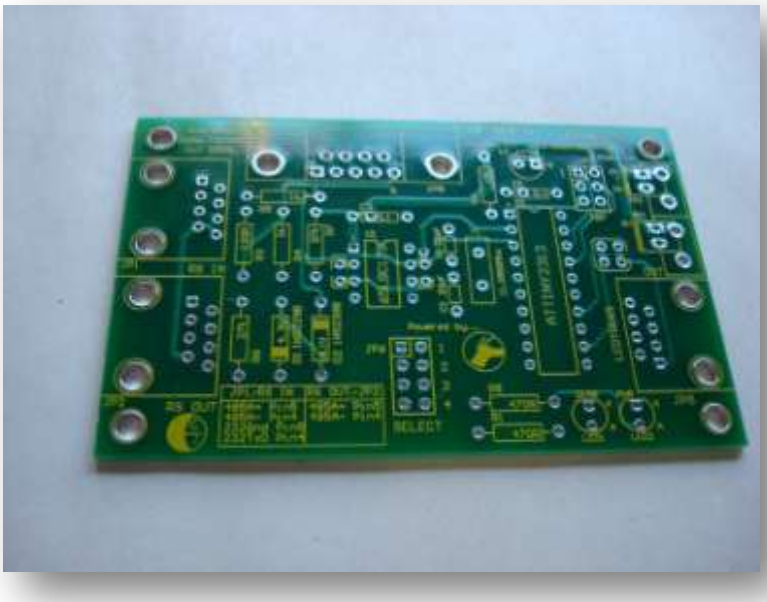
The second programming step is to load the code (the .hex file)

The AVRDUDE command for this (using the above programmer and the triksc02.hex file) is:

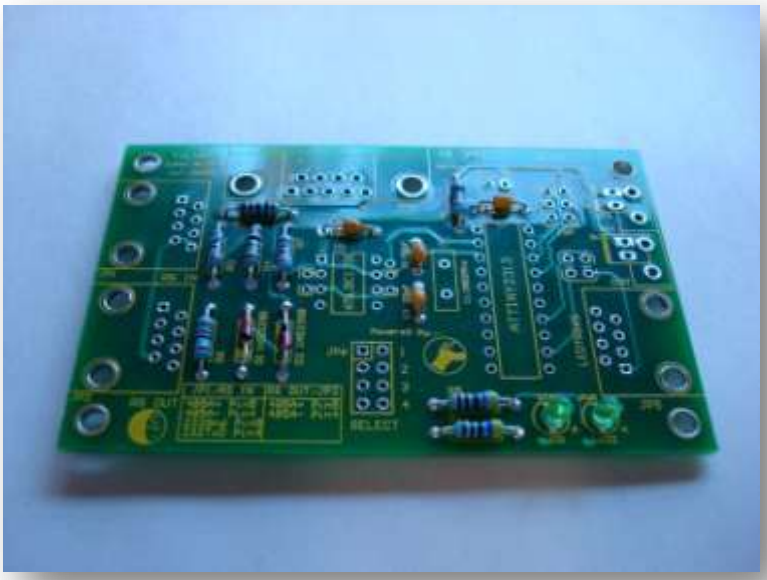
avrdude -c usbtiny -p attiny2313 -U flash:w:triksc02.hex

Building the Board

The Triks-C board is a relatively simple board to build, and those with moderate experience should be able to complete the board in less than 1 hour. Even those relatively new to soldering should be able to complete in less than 2hrs. IC sockets are recommended for all users, but are not compulsory.



The bare board, shown above, is the latest Triks-C available at the time of writing. (Photo by Mike, www.landolights.com)



This image shows the capacitors, resistors, LED's and Zener Diodes in place. Note the polarity of the LED's and Zener Diodes. (Photo by Mike, www.landolights.com)

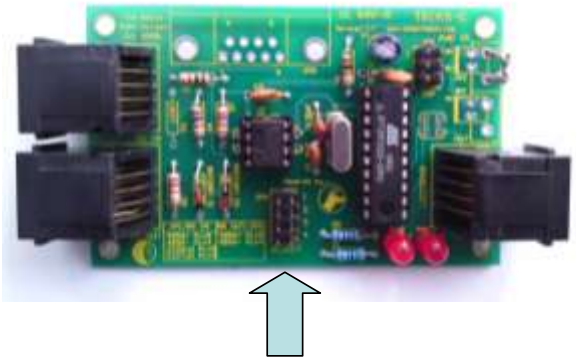
This shows the IC Sockets, RJ 45's and 6 & 8 pin connectors. The optional 9 pin serial connector is not installed. The input from the PC is via the RJ45 socket on top left, or by the serial port on the top, just left of centre. Connection to the next Triks-C, in the case of a multo panel display is via the RJ45 bottom left. Connection to the Ledtriks is via the RJ45 socket on the right side. (Photo by Mike, www.landolights.com)

Here the optional 9 Pin serial port has been added, as well as a connector for the power (Optional, as many people prefer to solder wire direct to the board). The IC's have been installed with the correct orientation as shown on the board's component screen print. (Photo by Mike, www.landolights.com)

Operating modes and Configuration

Firstly we should mention that the address for every Ledtriks panel you are running with a Triks-C should be set to address 1 on the Ledtriks board. The Ledtriks no longer handles the addressing, as this is now done by the Triks-C

There are 4 addressing Jumpers on the TRIKS-C labelled JP6 1, 2, 3 & 4. The address jumper should be placed in the chosen position BEFORE powering up the Triks-C. If no jumpers are present, the Ledtriks defaults to panel 1.



JP 6 Configuration		
	1	(Reserved)
	2	(Sets Stand Alone Mode)
	3	Panel Address
	4	

Panel 1

	1
	2
	3
	4

Panel 2

	1
	2
	3
▬	4

Panel 3

	1
	2
▬	3
	4

Panel 4

	1
	2
▬	3
▬	4

Diagnostic/Stand Alone

	1
▬	2
	3
	4

The positions are 1 to 4, from top to bottom.

Position 1: A test position. Not yet implemented

Position 2: This is used as a Stand-alone test of the board (i.e. no computer attached). It places the Triks-C into a scrolling test where the Triks-C name and firmware version is shown on 2 lines, and is scrolled off to the left by an alternating solid block and angled line test. The Diag light on the Triks-C also flashes at about 2 Hz in this mode.

Positions 3 & 4 are used to select the panel number in a multi panel setup. The Panel layout, when viewed from the front, is:

1 2

3 4

Panel 1: No jumpers on 3 or 4

Panel 2: Jumper on position 4 only

Panel 3: Jumper on position 3 only

Panel 4: Jumper on both positions 3 AND 4

Now the board is identified as the correct panel the first thing to test is that it is operational. Connect the Ledtriks to the Triks-C but leave the computer unconnected.

Suggested Testing approach:

- Set jumper J2 to run the diagnostic sequence
- Powerup the Ledtriks and the Triks C
- You should see a power LED on the Triks C
- You should see a splash screen followed by a scrolling diagonal and block on the Panel
- This will test the function of the board and your panel
- The DIAG Led will flash
- If the board doesn't scroll – make sure J2 is in place The jumper is checked only at power on so it needs to be in place before you powerup
- If the board scrolls slowly its likely that the internal clock is still selected (reprogram the fuses)

Now you are ready to try a connection to the PC

- Remove J2
- Fire up LTC and send #CLS
- If the screen goes blank you are in business!

Using the LTC software

The LTC.exe program should be placed in the same directory as your .led files, usually \sequences under your Vixen directory.

To run the program just open a cmd window (start/run 'cmd') navigate to the directory and run LTC by typing LTC at the command prompt.

The first time LTC is run it will detect that there is no TRIKSC.DAT file and ask you some questions regarding the com port you want to use (i.e. COM1: COM5: etc) and then set frame delays. Most people will not have to change any of the delays from their default value.

Later versions of the LTC program also ask for the number of panels connected, and the user can select 1, 2 3 or 4. Select 1 for single panel operation, 2 for two panel wide. If the user wants a 2 high, 1 wide selection, select 3, and for a 2*2 select 4

After TRIKSC.DAT has been created, any subsequent runs of LTC (without command line parameters) will show you the list of available command line parameters.

Running a .led sequence from the command line

LTC works with command line parameters. Each parameter is separated by a single space, and you can string multiple sequences into the one long sequence. For example:

LTC example.led #FDL100 #FRM #DLY5000 #CLS

This tells LTC to open the file “example.led”, override the default frame delay and use a delay of 100ms per frame (#fdl100), display it frame by frame (#FRM), wait 5 seconds after the end of the file (#DLY5000), and clear the screen at the end of the file (#CLS). Note that the Frame Delay has to go before the #frm as LTC looks at the commands in order.

Another way to run multiple commands is to place each command 1 per line, in a plain text file and call the file instead of stringing the commands on the command line. For example

LTC example2.txt

And the text file example2.txt could look something like this:

E:\Vixen\Sequences\Eyes.led

#FDL50

#FRM

#RPT5

#CLS

That would open eyes.led, use a frame delay of 50ms, display it in frame by frame mode, repeat the entire .led file 5 times, and then clear the screen.

The path to the .led file is not always needed, but I found it was more reliable, especially when testing, running and tweaking across multiple PC’s, so it is strongly recommended.

As of V0.9 of LTC, the available options are:

#CLS	= clear screen
#BLK	= blank screen
#SHW	= Show screen
#RL	= Roll to Left (EDGE TRANSMISSION)
#RR	= Roll to Right
#RD	= Roll Down
#RU	= Roll UP
#FRM	= Whole Frame

#RPTxxxx = Repeat all xx times

#DLYnnnn = Wait nnnn ms

#LOOP = Repeat until keypress

#FDLnnnn = Delay between Frames nnnn (ms)

#HDLnnnn = Delay for Horizontal scroll (ms)

#VDLnnnn = Delay for Vertical scroll (ms)

#RDLnnnn = Delay for Rotate (ms)

#NPx = Number of active panels (1-4)

#TIME = Display time to execute this file (Must be the last option on the command line or .txt file)

#ROTR = Rotate Clockwise (right)

#ROTL = Rotate AntiClockwise (left)

#RINCnnn = Rotation Steps (nnn degrees)

#RTOnnn = Rotation Angle (nnn degrees)

#RXCnnnn = Rotate Center X co-ord

#RYCnnnn = Rotate Center Y co-ord

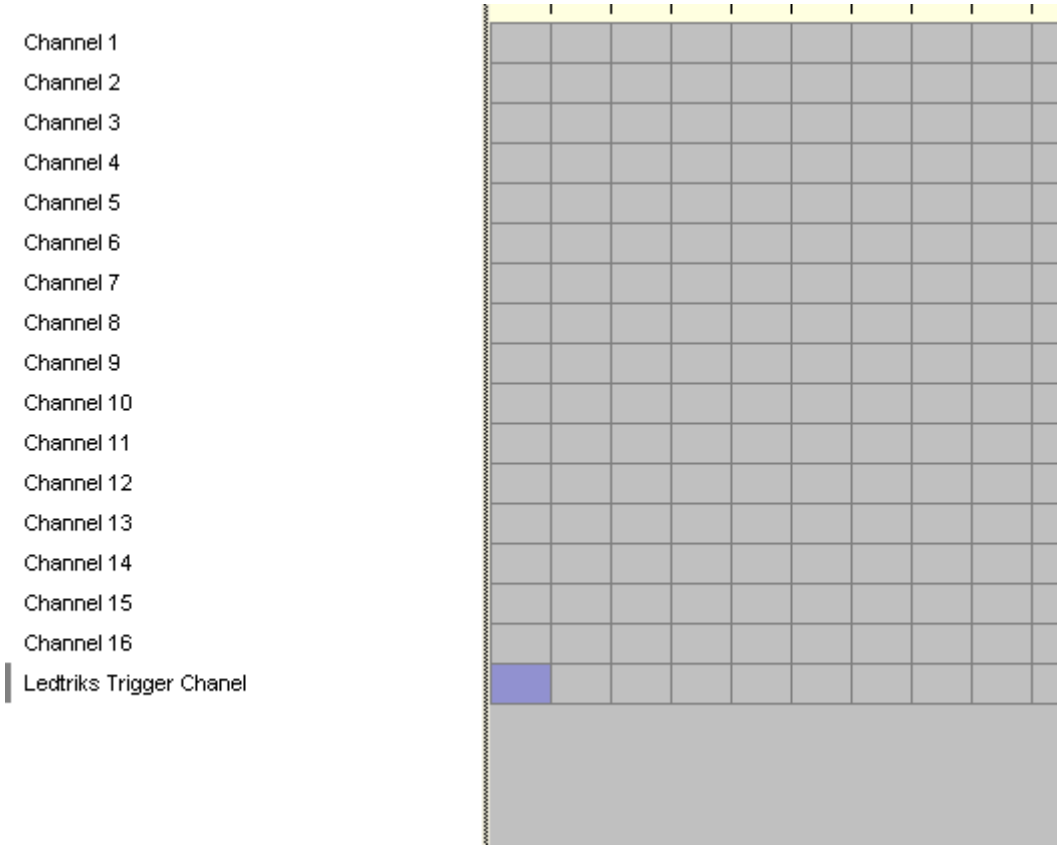
#STA = Send TRIKS-C to stand alone mode(until next command received)v0.2 of the TRIKSC firmware is required for #STA

#CALxxx = Coundown timer (2panels) xxx secs

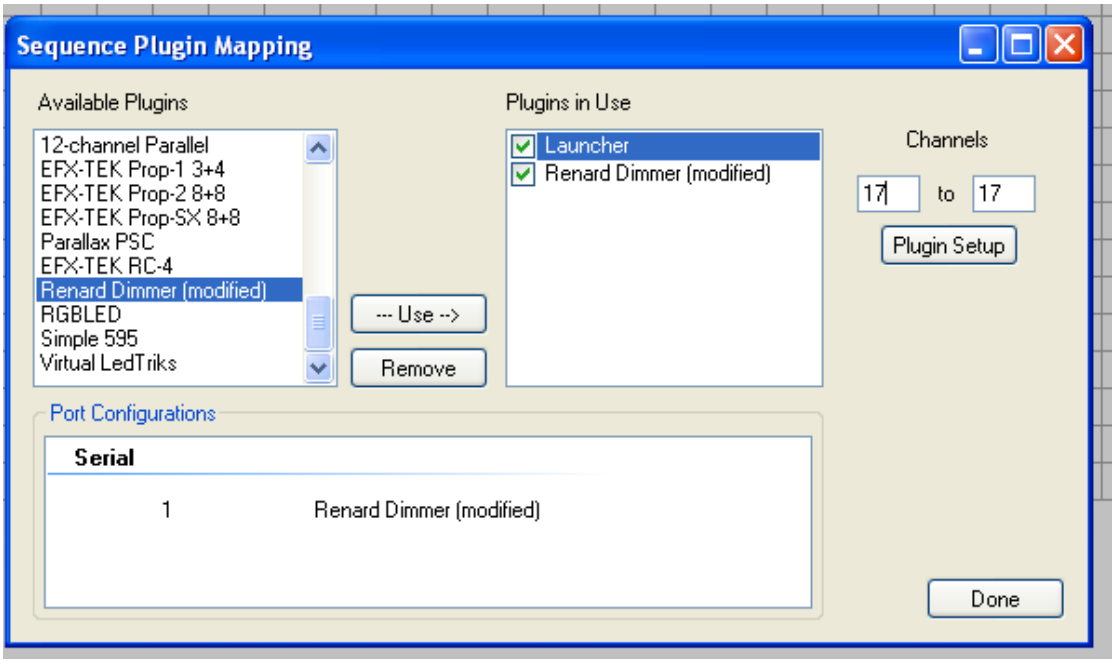
Interfacing LTC with Vixen

LTC is designed as an interface between Vixen and the Triks-C. The following steps show the interface using Vixen 2.1, but earlier versions of Vixen are very similar.

Firstly you need to create a “Trigger Channel” for Vixen to use. I personally create this as the last channel.



The above image shows the Ledtriks Trigger Channel after a 16 channel sequence, and named to easily identify it.



The chosen channel should then be set to use the Launcher Plugin. The above sequence has channels 1-16 using the Renard plugin, and channel 17 using the Launcher.

Ok now we have the plugin added, we need to set it up for the triggering of the sequences. Before we go on, we should decide what Ledtriks animations you wish to run and make a list of them, assigning each a number from 100 down to 1. Should you need more than 100, you will need to add a second trigger channel, giving you an additional 100 triggers.

Here is my list:

100 - bomb(4p).led

99 - bowling(4p).led

98 - hohoho(4p).led

Because I have chosen to use .txt files to make things easier, I have 3 text files named bomb(4p).txt, bowling(4p).txt, and hohoho(4p).txt.

The bomb(4p).txt contains the following 4 lines:

E:\Vixen\Sequences\bomb(4p).led

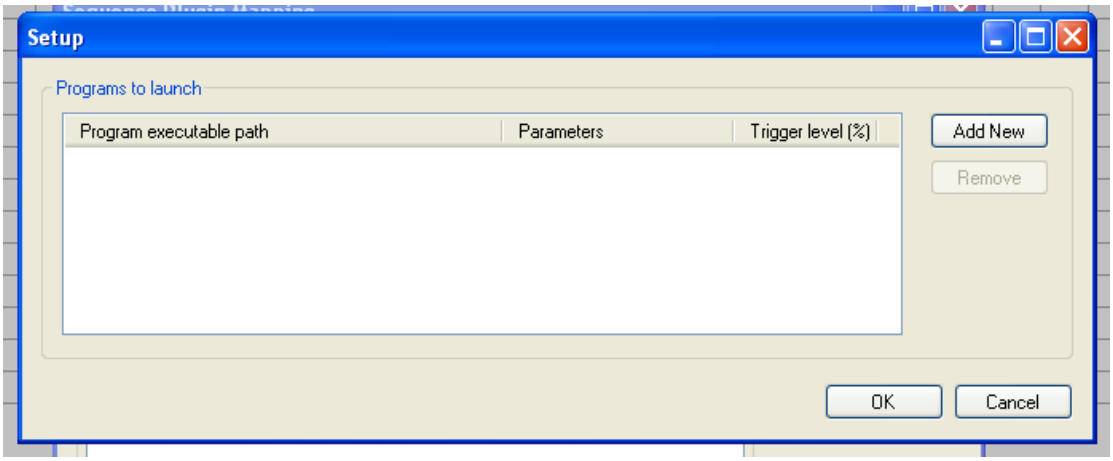
#FDL50

#FRM

#CLS

The path to the .led file is not always needed, but I found it was more reliable, especially when testing, running and tweaking across multiple PC's, so it is strongly recommended.

OK to add these to Vixen, highlight the Launcher plugin as shown in the image above, then select "Plugin Setup" button. You will then be shown a screen as shown below.



From here click ***Add New***.

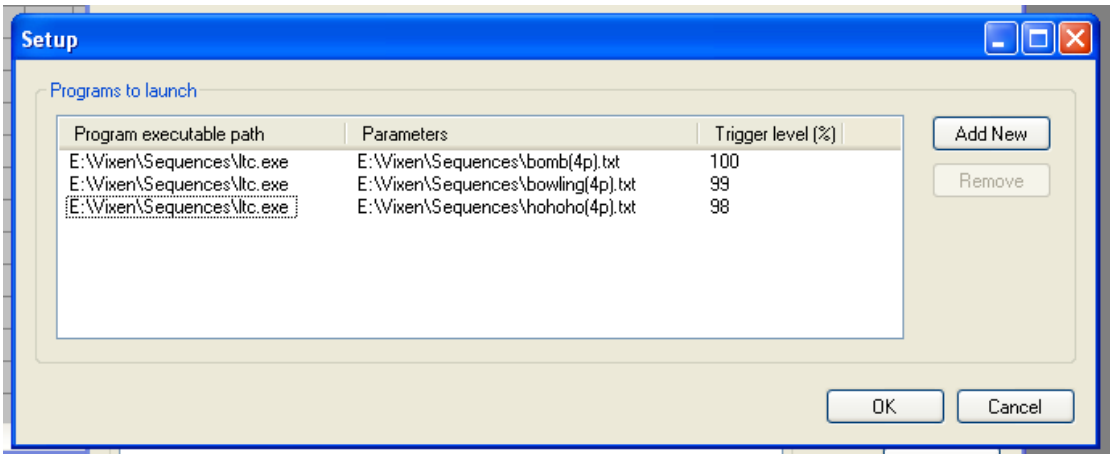
The executable path is the FULL path to LTC.EXE, such as

E:\Vixen\Sequences\ltc.exe

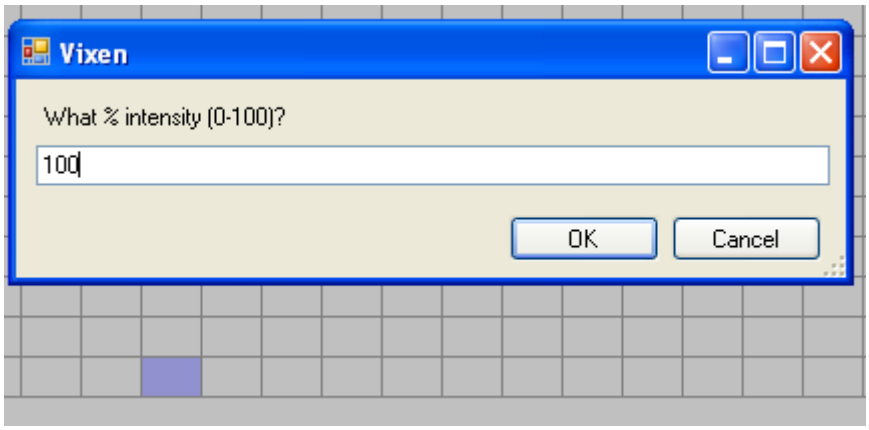
The parameters is easy when you use a text file – it's the full path to the text file - E:\Vixen\Sequences\bomb(4p).txt

Trigger % is the lever you choose to trigger the particular Ledtriks animation, and to make it easy, I use the numbers as above.

Here is the setup for the 3 files I used



Now off to the normal Vixen page to trigger some Ledtriks animations.

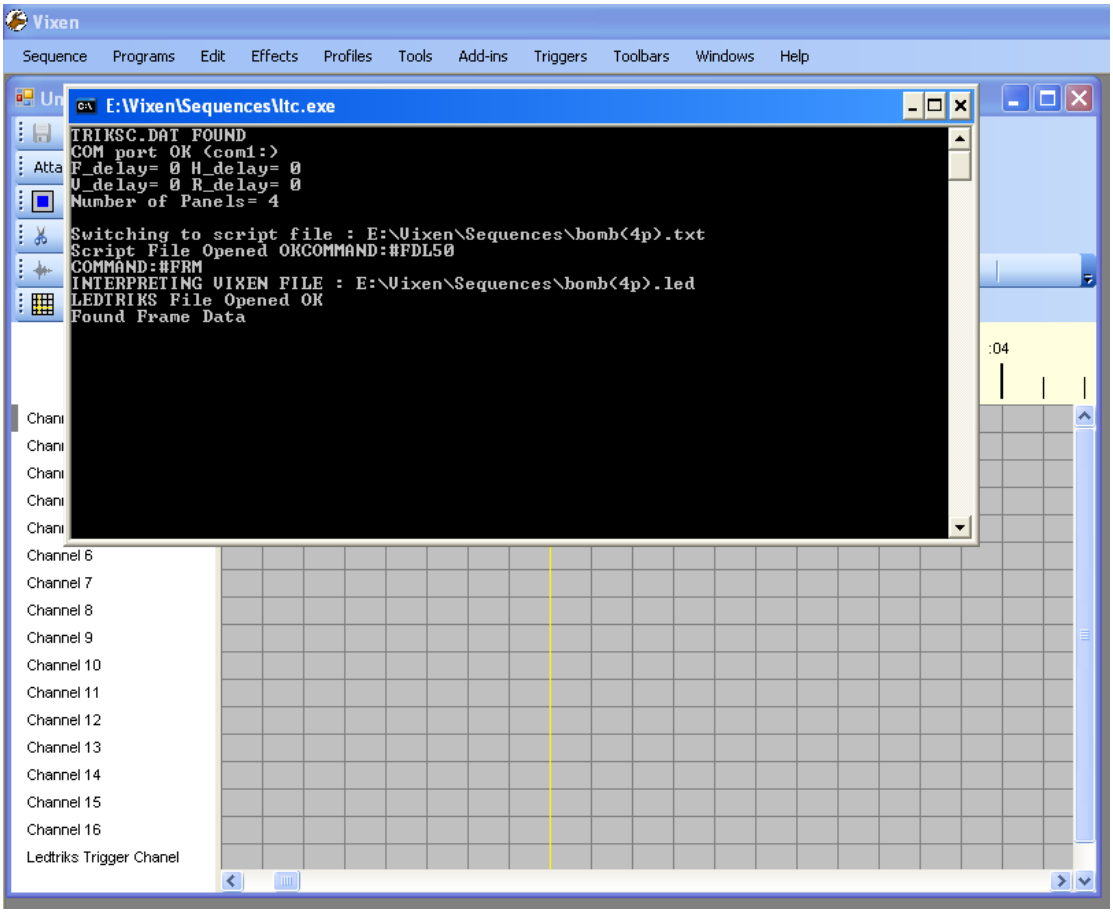


This sets the cell selected – JUST one cell!!! To 100%, and this will trigger bomb(4p).txt. An issue that I find annoying is that it's very hard to see if you have entered a low % value, so I personally start at 100% and go down to about 20%, but that's up to the user.

NOTE: Depending on your operating system, Vixen and LTC may not like spaces in the path name. This happens on some PC's and not others, so I recommend moving your Vixen directory to c:\vixen

My main PC running Windows XP had no issues last year, but the show PC also running XP would not work with spaces in the path. I never got to the end of it, as moving vixen was MUCH easier.

Also note you may need the full path to the .led file inside the .txt file as shown in the examples earlier in the document.



This shows the DOS box opening and running LTC.exe with the requested .txt file.

Credits

Ledtriks Initial Design: RJ

Ledtriks Board Design and modifications: Wjohn

Triks-C: TimW

Triks-C Board Design: TimW and WJohn

Beta Testers: TimW, WJohn, daviddth, Frank, KiwiChristmas, Marcus, WildWillie, nromanel, santacarl, rca, shocker, scorpia, Mike Sullivan and Keith R

Other Help: I dunno, but I bet John and Tim got heaps of help from other people.

This manual was written by David Rawsthorne (daviddth), with LOTS of help from TimW and WJohn. Some sections and images are copied in total from TimW's beta manual, which has never seen the light of day. Photo credits for the building of the Triks-C board go to Mike (oldcqr) from www.landolights.com

[i] The most well know site is www.computerchristmas.com
[ii] Vixen is a software program developed by KC Oates. The website is www.vixenlights.com