

Guider interface using Raspberry Pico

Warning: Please be careful when interfacing to your telescope. I cannot be responsible for any damage caused due to errors either due to your wiring etc or this software. I have tested this on my LX10.

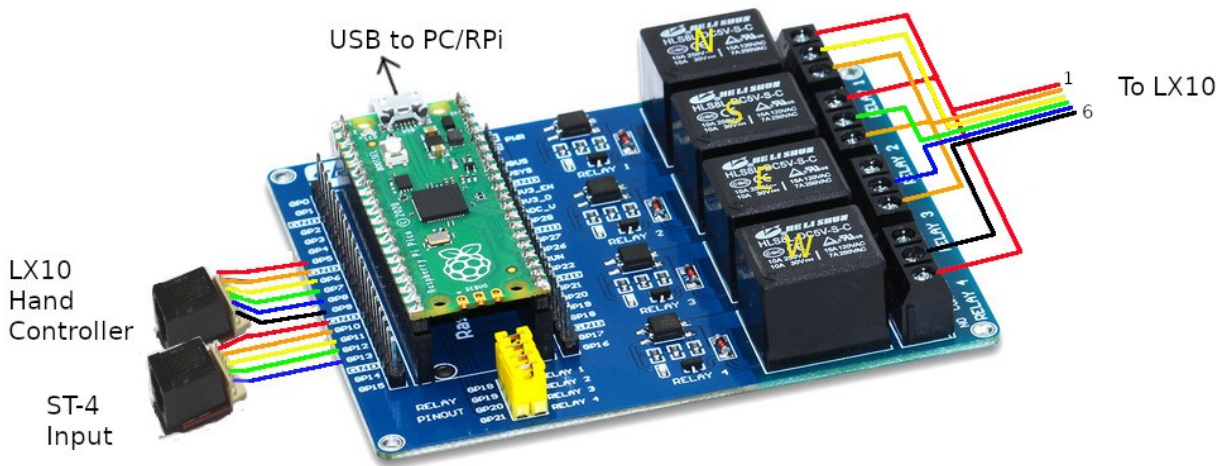
This interface enables guiding using guiding software such as PHD2 on a Pc, or my software for a Raspberry Pi. Communications between the PC / RPi and Pico is via USB. The interface also gets its 5v from the USB. Install main.py into the RP PICO

It can be interfaced to a ST-4 or other types of mount. Details are provided for connection to a ST-4 or Meade LX10. The ST-4 interface uses a 4-channel Phototransistor Output Optocoupler, the ILQ74. The Meade LX10 uses either mechanical or solid state relays.

There is also an option to use a PWM control for the LX10 but this is entirely experimental. This replaces the existing drive circuitry (not physically , it is simply switched off and the interface connected through the front panel port).

You will need to provide the interface from the Pico to the telescope/mount port. The Pico provides 4 outputs for N,S,E & W (Dec +,Dec -,RA- & RA+). These are +3.3v when active (although this can be inverted in the Micropython code for active low relay boards), note many boards available are for Arduinos requiring 5v so may not work with the Pico.

Here is one example of an available board that should work as it uses the same GP outputs, connect as shown in the generic relay interface.



Parts:

<https://thepihut.com/products/raspberry-pi-pico>

<https://thepihut.com/products/male-headers-for-raspberry-pi-pico>

<https://thepihut.com/products/raspberry-pi-pico-with-pre-soldered-headers>

<https://thepihut.com/collections/pico/products/raspberry-pi-pico-relay-board>

<https://proto-pic.co.uk/product/sparkfun-bob-14021-rj11-breakout/>

<https://proto-pic.co.uk/product/sparkfun-prt-00132-rj11-6-pin-connector/>

<https://proto-pic.co.uk/product/sparkfun-prt-10366-jumper-wire-0-1-6-pin-4/>

<https://proto-pic.co.uk/product/sparkfun-prt-10371-jumper-wire-0-1-6-pin-6/>

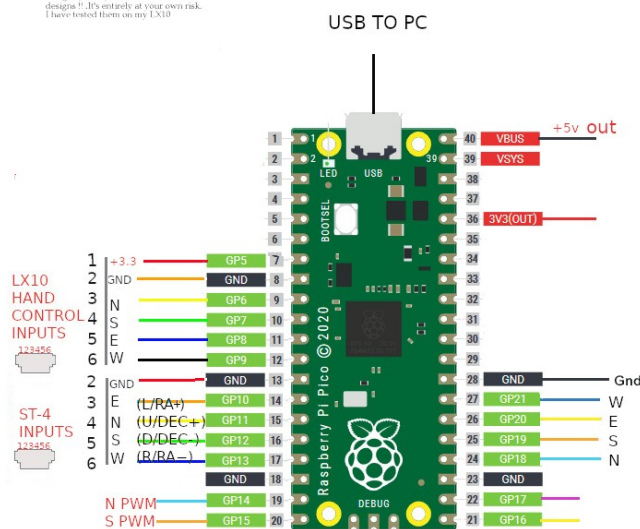
<https://proto-pic.co.uk/product/sparkfun-prt-10365-jumper-wire-0-1-5-pin-4/>

<https://magpi.raspberrypi.org/articles/programming-raspberry-pi-pico-with-python-and-micropython>

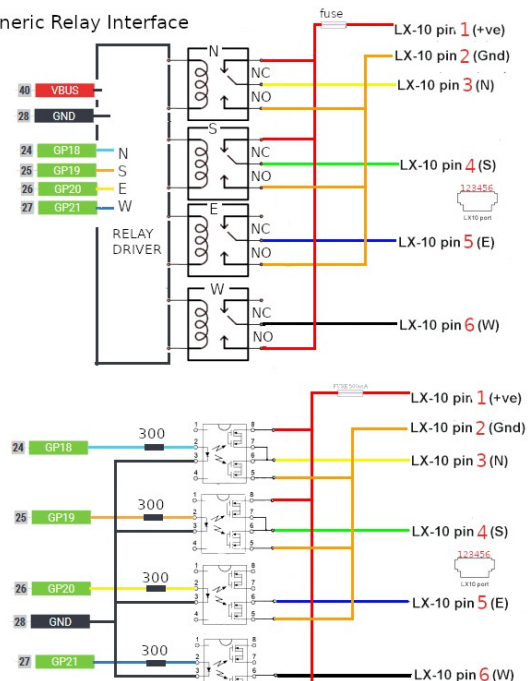
<https://datasheets.raspberrypi.org/pico/raspberry-pi-pico-python-sdk.pdf>

Raspberry Pi Pico

I can accept no responsibility if anyone does any damage to their equipment etc using any of my software or hardware designs !! It's entirely at your own risk. I have tested them on my LX10



Generic Relay Interface



You can connect a LX10 hand controller into the Pico to also provide manual control. There are also ST-4 control inputs, or you could make your own hand controller using 4 push buttons from the ST-4 inputs to gnd.

I have tested it with PHD2 v2.6.9, ASCOM Platform 6.5.SP1 and drivers for Meade LX200 Classic and Autostar #494, #495, and #497.

There is also the option to try using PWM to drive the LX10. This is for experimenting and is not proven to be sufficiently stable for astrophotography yet. At your own risk !!.

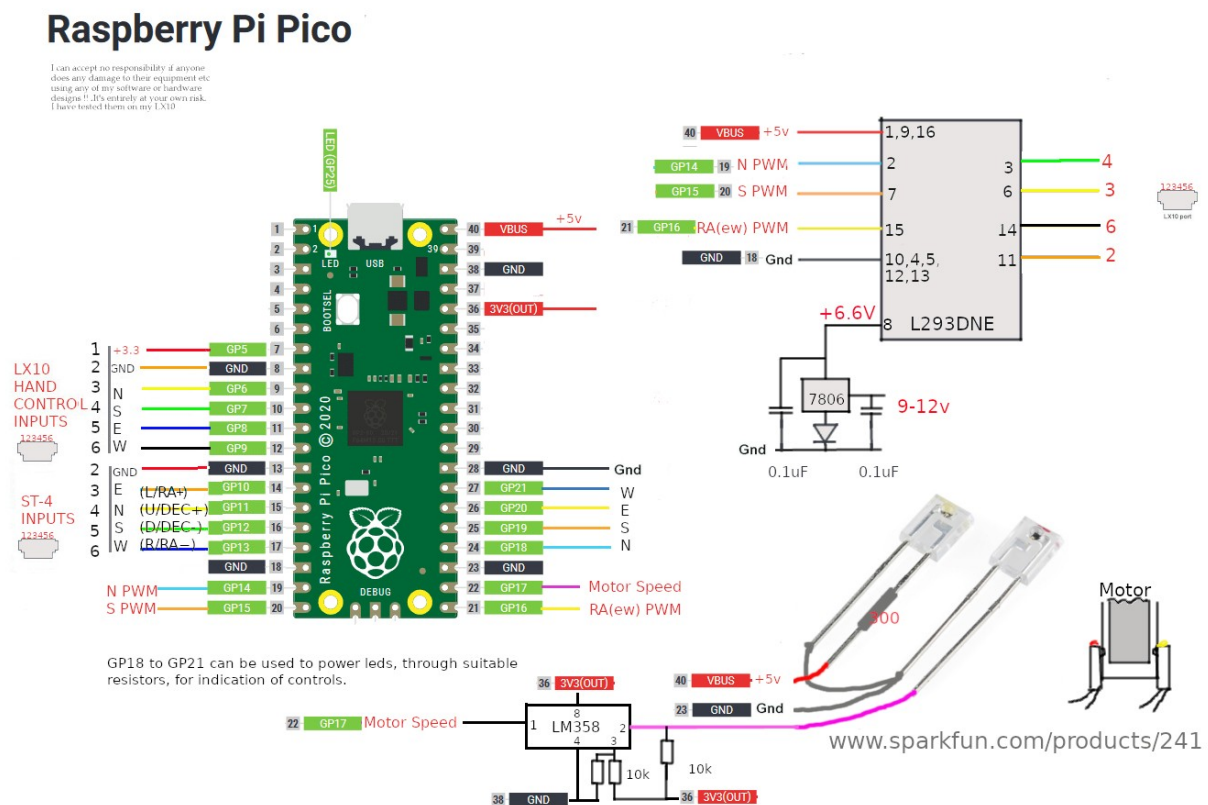
To provide motor speed detection you need to fit a led transmitter and detector to the motor in the LX10, which involves drilling a hole in the motor. This was a modification proposed by Mark on the LX10 forum.

It provides a PWM signal to power the RA motor at the required rate, and by monitoring the motor speed can maintain a constant rate (set for 20.89Hz). It also provides PWM for the DEC motor.

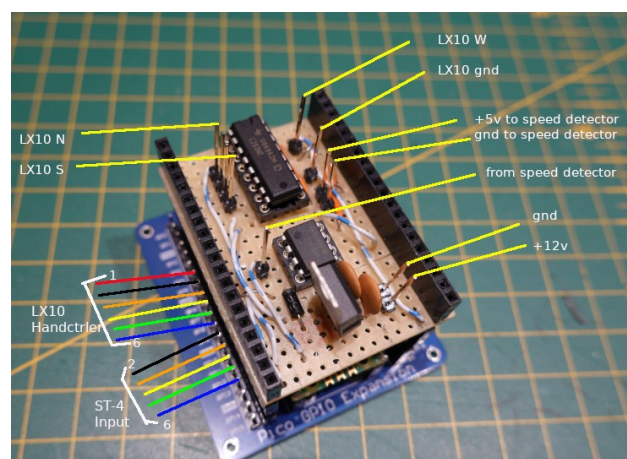
PWM is disabled in the software as supplied, simply change it to `pwm_on = 1` to enable.

The outputs from the L293DNE are connected to the LX10 port. ENSURE the LX10 is switched OFF.

This can also be controlled by PHD2 etc.

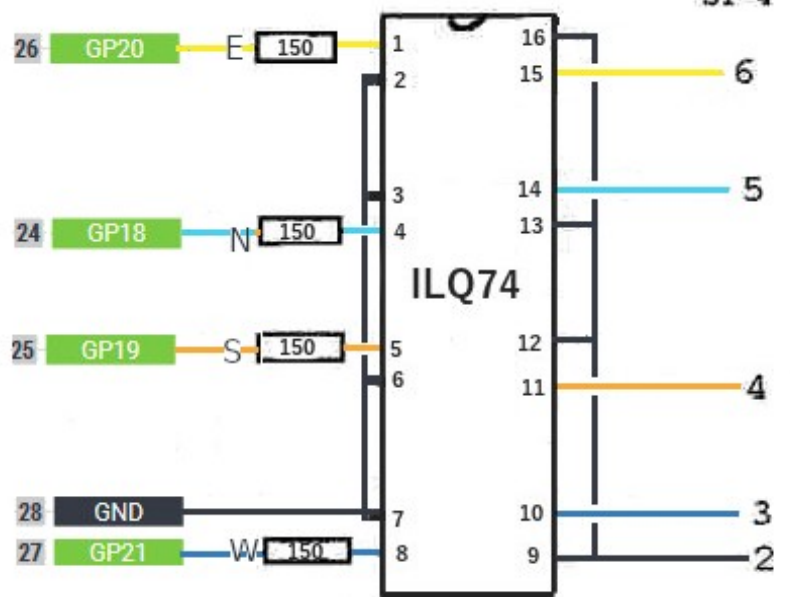


PWM Prototype

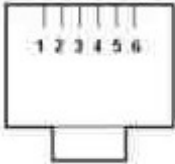


ST-4 output interface.

Pico Connections



AUTO GUIDE



Auto Guide Port Pins	Function
1	Not connected
2	Ground
3	Move RA Positive
4	Move Dec Positive
5	Move Dec Negative
6	Move RA Negative

LX200 ASCII Commands between PC and Pico.

:Mx move in required direction, where x = n,s,e or w. eg. **:Mn**

:MgxYYYY move in required direction for a required period where x = n,s,e or w and Y = 0000 to 9999. eg **:Mgn1000** move North for 1 second.

:Qx quit move, where x = n,s,e,w or # (all). eg. **:Qn**

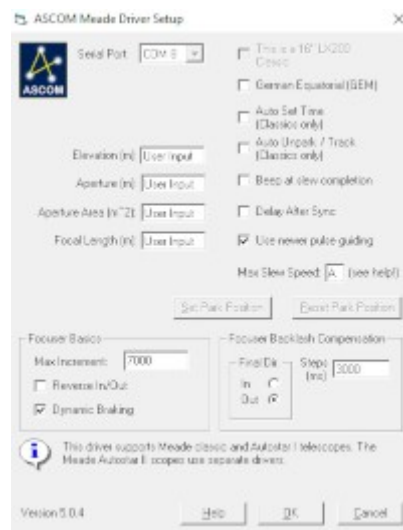
The following commands are also acknowledged to enable connection with PHD2:

:GR,:GS,:GW,:AT0,:GD,:GA,:Gt,:GVT,:GVP,:GVE,:GVN,:GVD,:?+

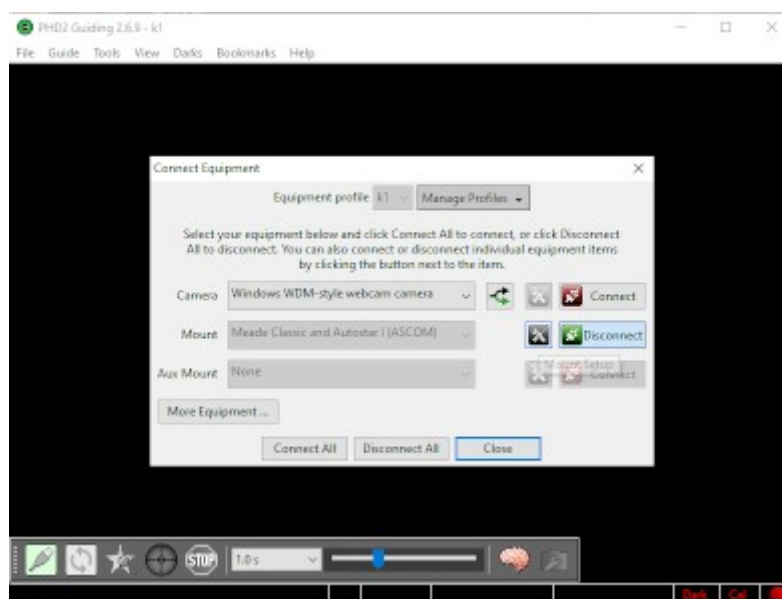
Connecting to PHD2 on a Windows 10 PC.

Run PHD Guiding 2

Click on **Guide** and **Connect Equipment** and choose the required **Serial Port**. (Use Device Manager to find it if unknown). Click on **OK**



Choose the **Meade Classic and Autostar 1 (ASCOM)** Mount (assuming you have installed the driver) and click **Connect**. Click **Close** if successfully connected.



To test click on **Tools** and then **Manual Guide**, then click on **North, South, East or West** and listen for the scope moving.

Using Pi AG Lite (PiAGL.py) for guiding

Copy PiAGL.py into '/home/pi' on the Raspberry Pi.
Install opencv with 'sudo apt install opencv-python3

Python3 PiAGL.py to run.

You can use a Pi Camera or a Philips 740/900 USB webcam * attached to your guidescope.
Connect the pico to a USB port on the Pi. It should appear as '/dev/ttIAMC0'.

(* Note there is an issue with RaspbIOS and Philips webcams, you need to carry out a sudo rpi-update for them to work, hopefully this will be fixed in future RaspbIOS updates)

To test the connection click on North/South/West/East and look for the telescope moving.



Screenshot using a Pi Camera.

There are various parameters that can be adjusted.

For the Pi camera: **AEB, FPS, Mode, Shutter mS, ISO, Brightness & Contrast.**

For the Philips webcams: **Gain, Gamma, Exposure, Auto Gain, Brightness & Contrast.**

South, North, West & East, send short (250mS) corrections.

AUTO GUIDE, switches guiding ON/OFF

Threshold, sets detection threshold for pixels. 0 for Auto

Zoom, Zoom image in / out

RA & DEC offsets, moves image to allow centering on screen

Minimum Correction, no correction will be made below this level

Interval, determines when corrections sent , timed in video frames

EXIT, exit script

(* Note there is an issue with Raspbian and Philips webcams, you need to carry out a `sudo rpi-update` for them to work, hopefully this will be fixed in future Raspbian updates)

