**********Draft**********

Kanade-Lucas-Tomasi feature tracker

03/19/22

Raspberry Pico

**********Draft**********

Testing with a reduced image.

cp ~/pico-lifting/testfiles/64/test1.pgm img0.pgm
cp ~/pico-lifting/testfiles/64/test1.pgm img1.pgm

```
Starting with 64 X 64 image
```

example1.c.64 in example1.c
make
./example1
sudo cp bb.bin /root/

Command (1 = Send or 0 = Wait):
need to copy the data received from host to img1
img1 = 0x2003ffd8 img2 = 0x20040fd8

```
Finding The Principal component analysis (PCA)
```

In first image:
Feature #0: (37.000000,29.000000) with value of 19844

Feature #1: (28.000000,39.000000) with value of 9270
Feature #2: (24.000000,29.000000) with value of 4393
Feature #3: (38.000000,39.000000) with value of 465

cp ~/pico-lifting/testfiles/64/lena_rgb_64.pgm img0.pgm
cp ~/pico-lifting/testfiles/64/lena_rgb_64.pgm img1.pgm
example1.c.64 in example1.c
make
sudo cp bb.bin /root/a.bin
In first image:
Feature #0: (32.000000,24.000000) with value of 4472
Feature #1: (39.000000,34.000000) with value of 3461
Feature #2: (29.000000,36.000000) with value of 3100

Goals of klt-test:

Create a library for the pico with the C from

```
*********************************************************************
NOTICE:

This code is now in the public domain.  The Stanford Office of
Technology Licensing has removed all licensing restrictions.

*********************************************************************

KLT
An implementation of the Kanade-Lucas-Tomasi feature tracker

Version 1.3.4

Authors: Stan Birchfield
         stb@clemson.edu

         Thorsten Thormaehlen
         thormae@tnt.uni-hannover.de
         (implemented affine code)

         Thanks to many others for various bug fixes.

Date: August   30, A.D. 2007
      May      10, A.D. 2007
      March    28, A.D. 2006
      November 21, A.D. 2005
      August   17, A.D. 2005
      June     16, A.D. 2004
      October   7, A.D. 1998

The code can be obtained from http://www.ces.clemson.edu/~stb/klt
(alternatively http://www.vision.stanford.edu/~birch/klt),
where the official manuals reside.  For your convenience, unofficial
manuals have been placed in the current subdirectory 'doc'.

Starting with 64 X 64 image
```

Finding The Principal component analysis (PCA)

Trying to add the following code to /rp2040-freertos-project this following the freertos libarary steps.

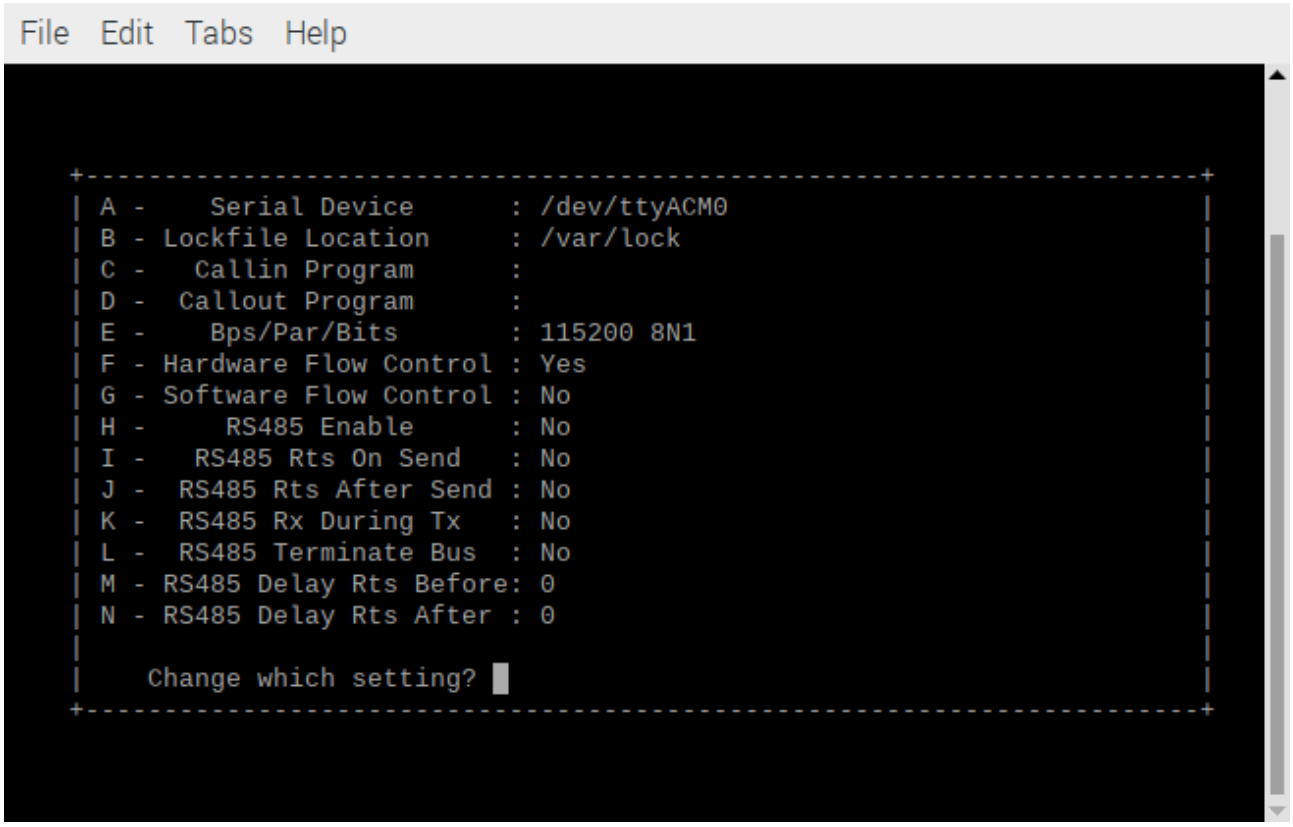
allocate.c error.c klt_util.c selectGoodFeatures.c
convolve.c klt.c   pyramid.c

~/rp2040-freertos-project/build $ cmake ../

~/rp2040-freertos-project/build $ make

openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program klt-test/klt-test.elf verify reset exit"

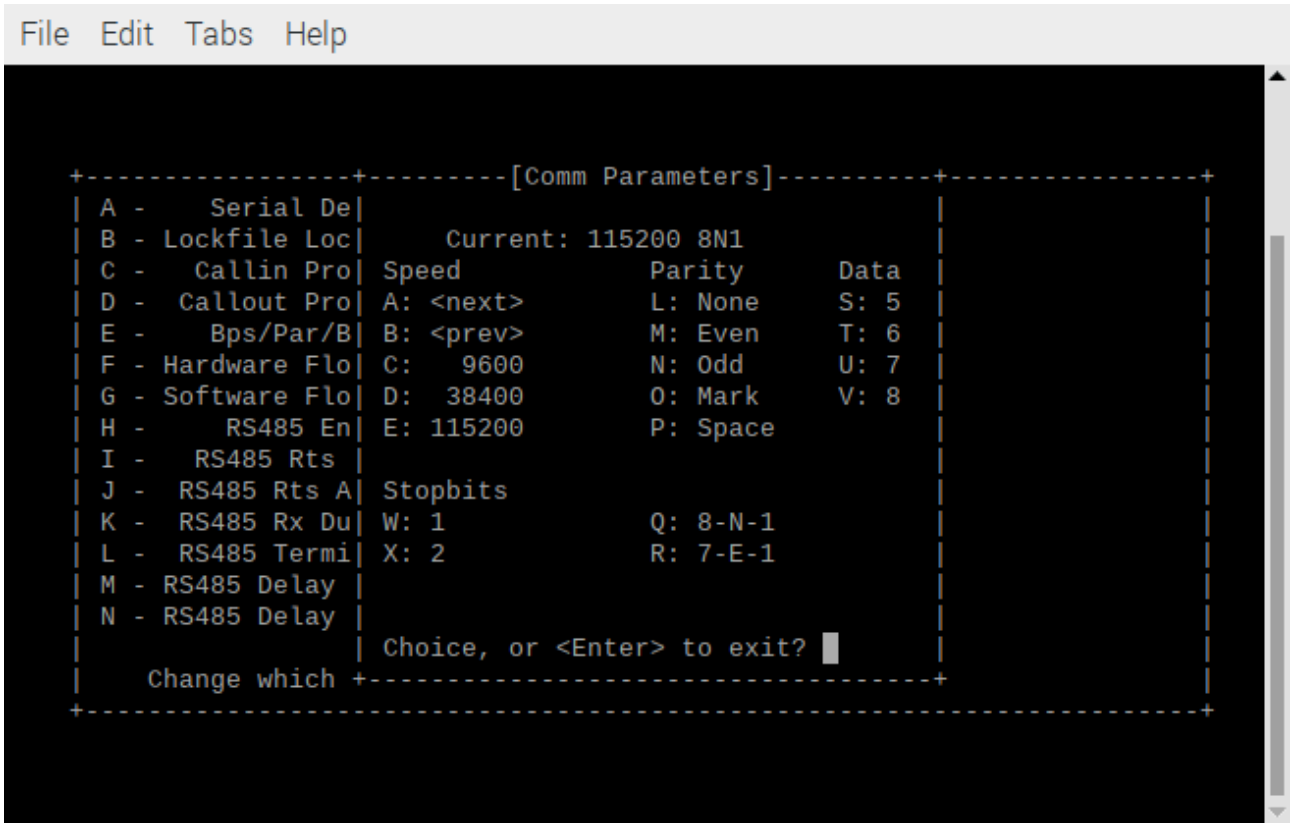
sudo minicom -s
scroll down Serial port setup
Enter



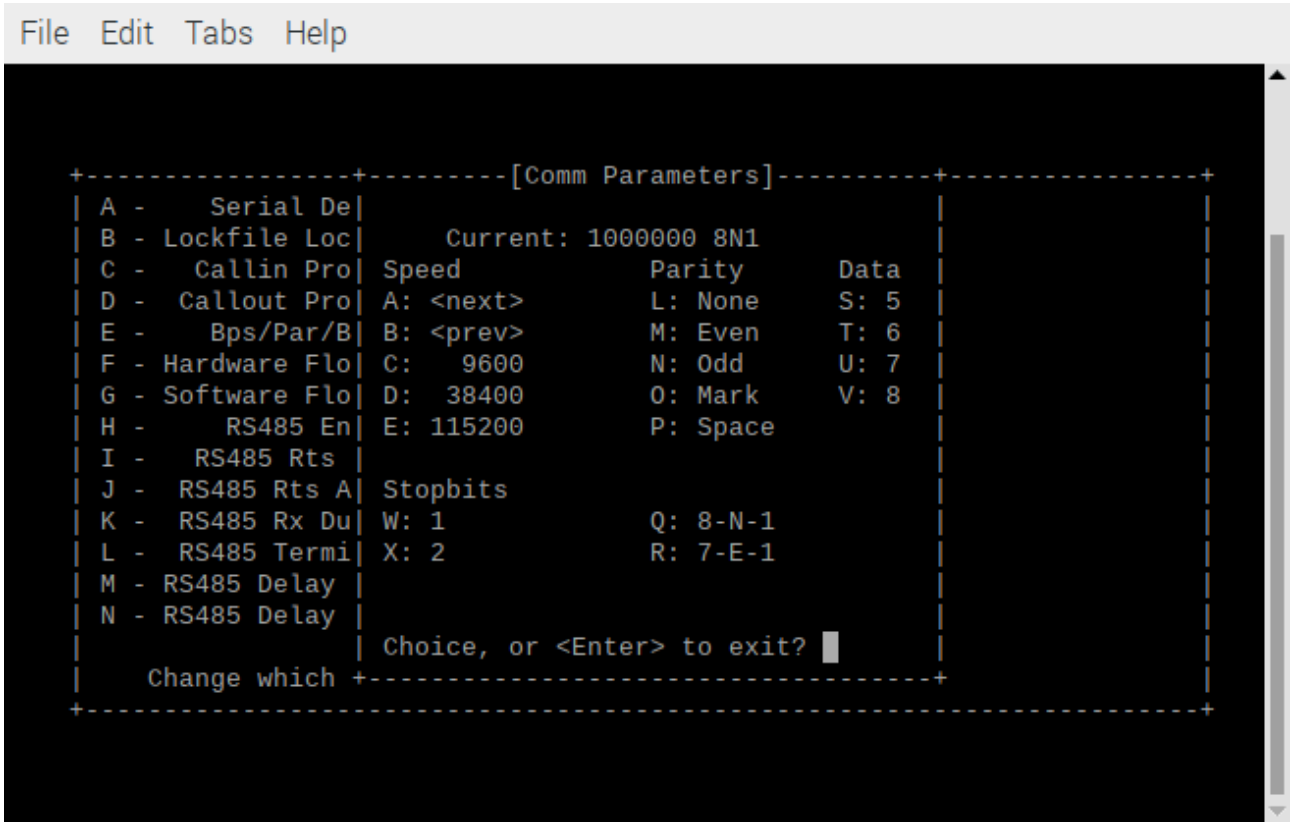
q

Need to change the baud rate to 1M

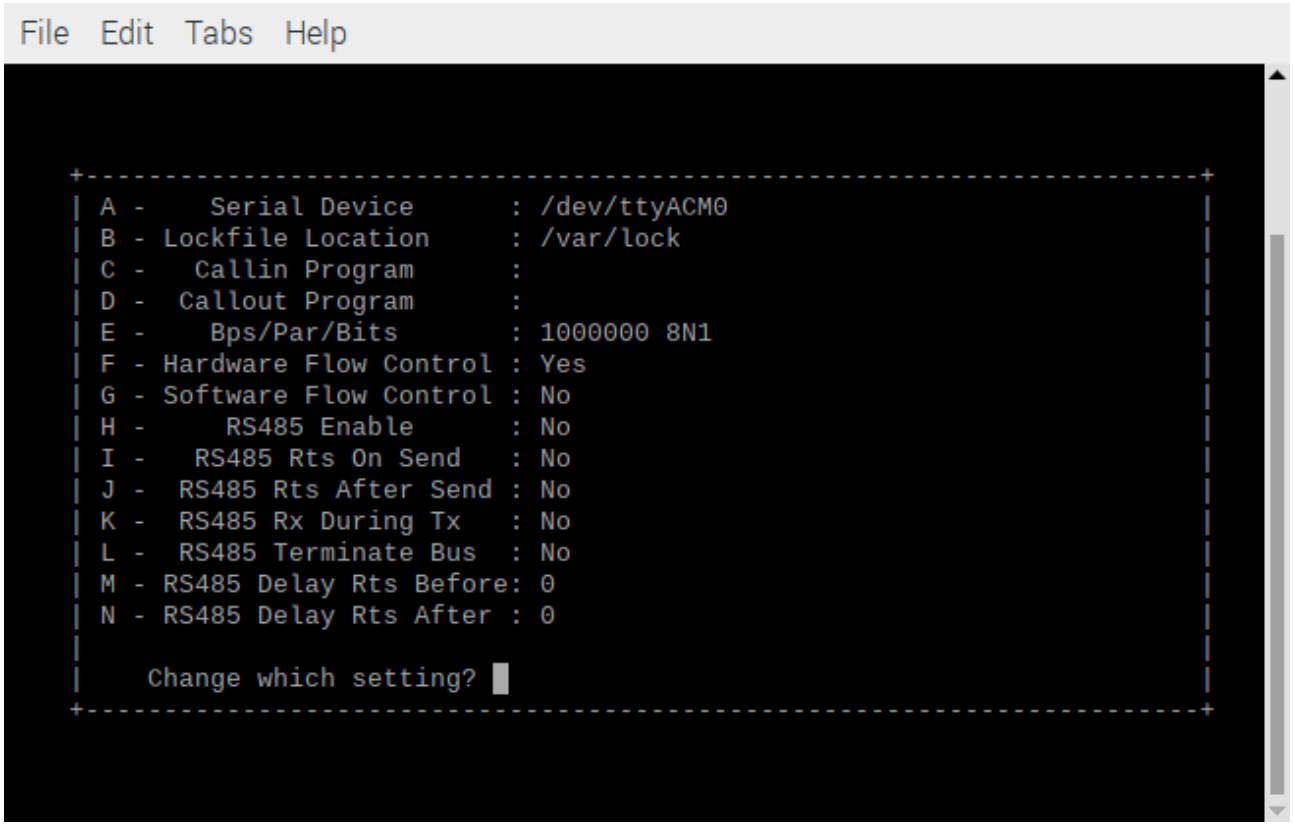
Depress E



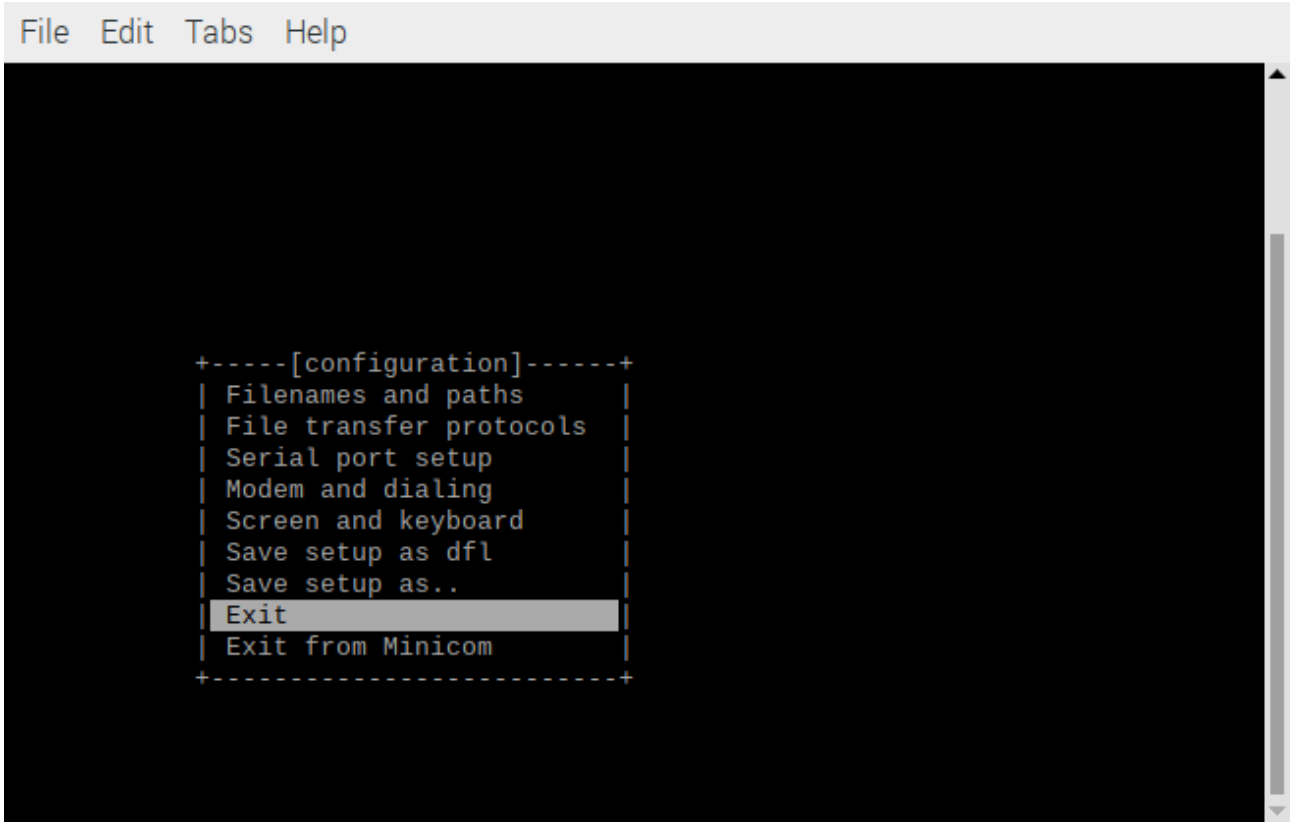
Depress A 6 times.



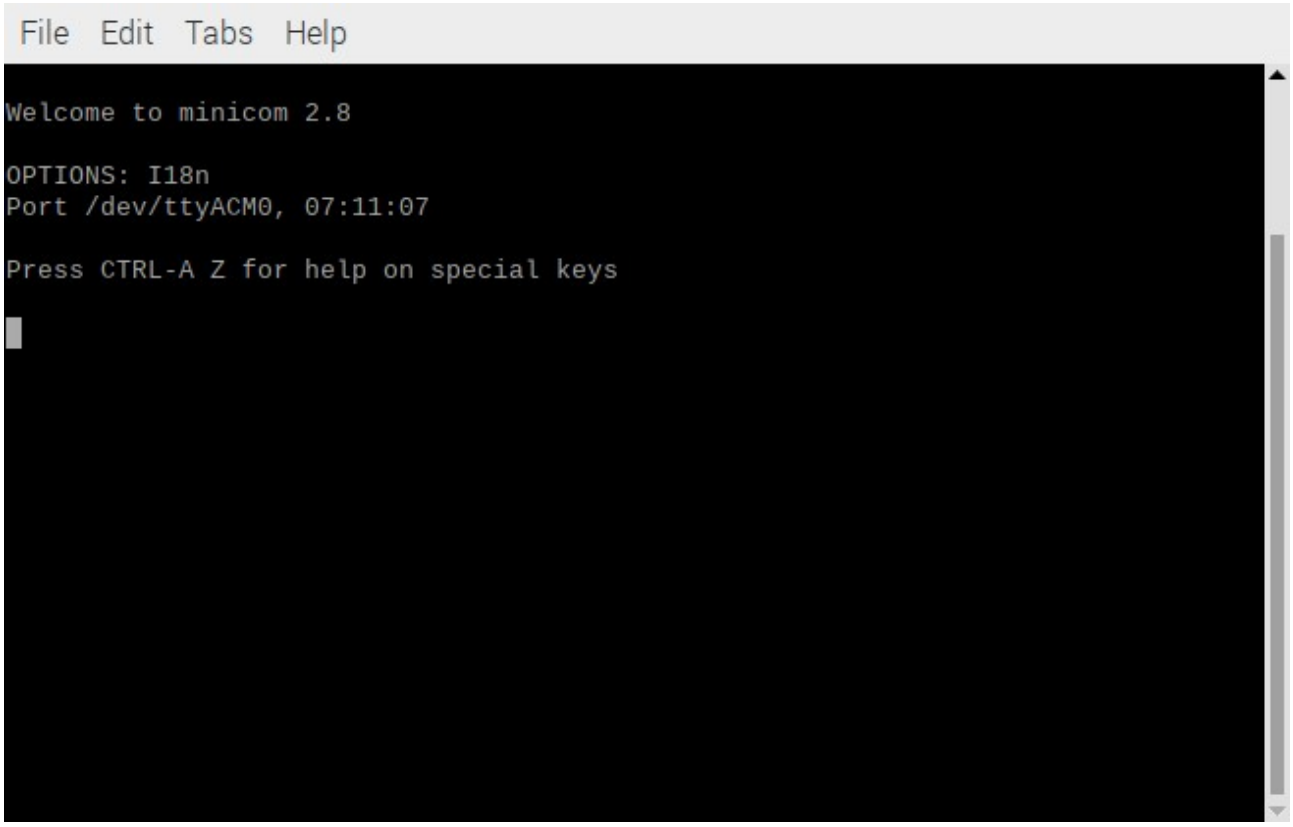
Depress Enter



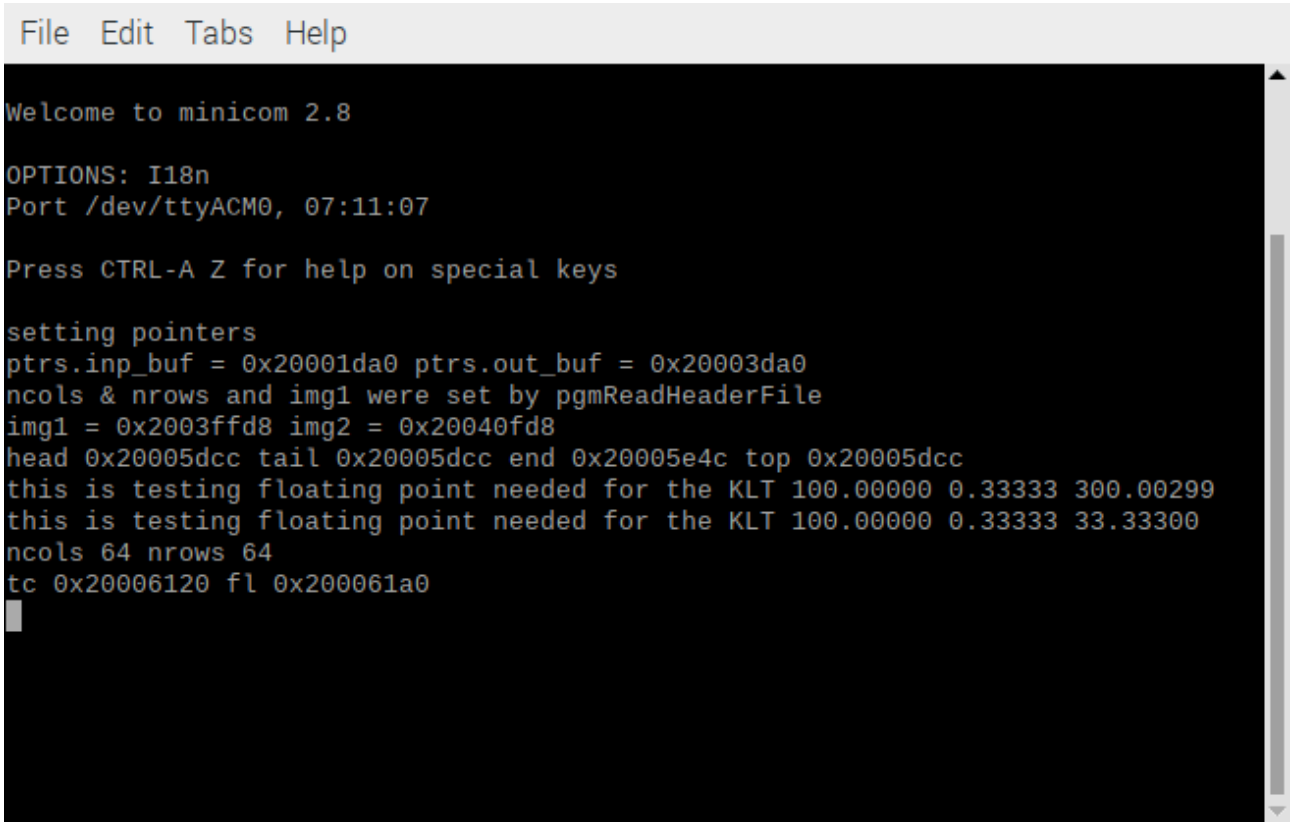
Depress Enter



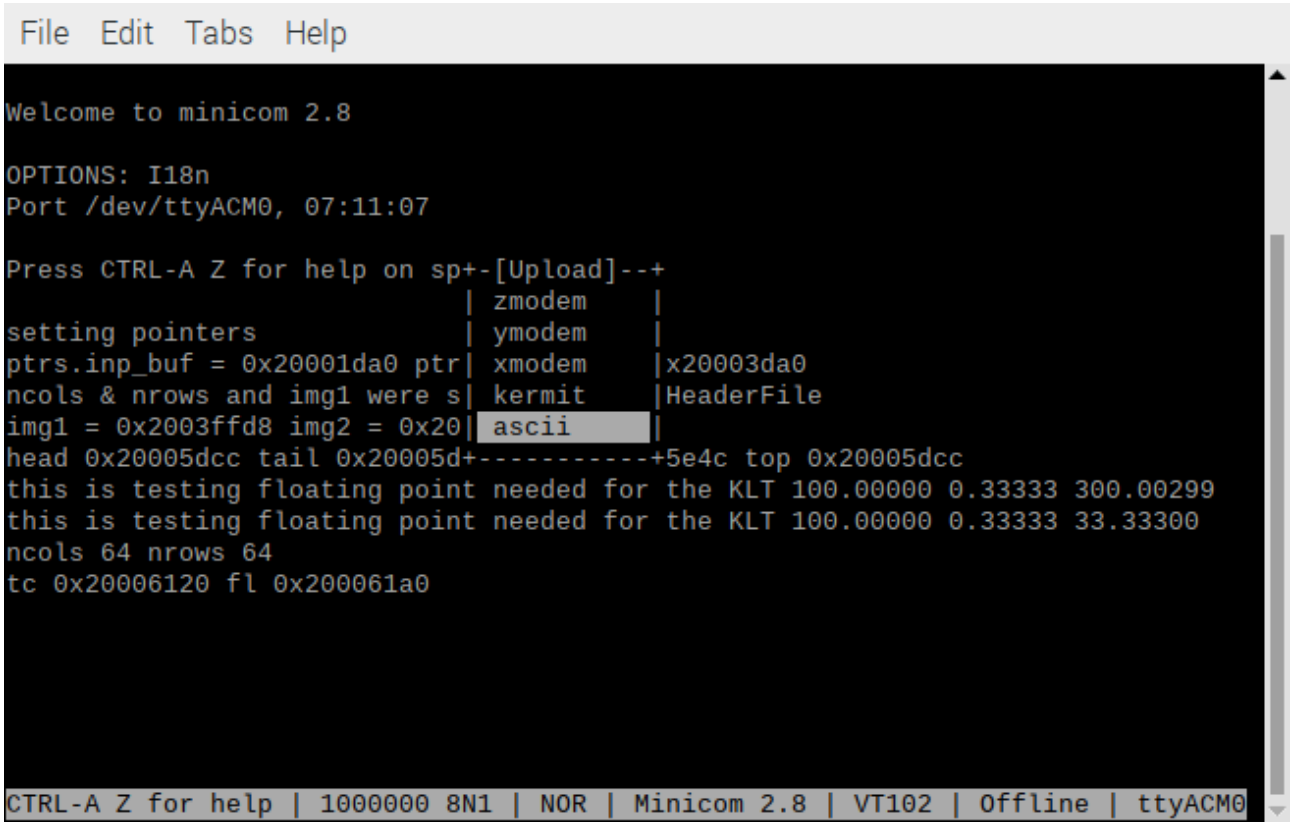
Scroll down to Exit Depress Enter



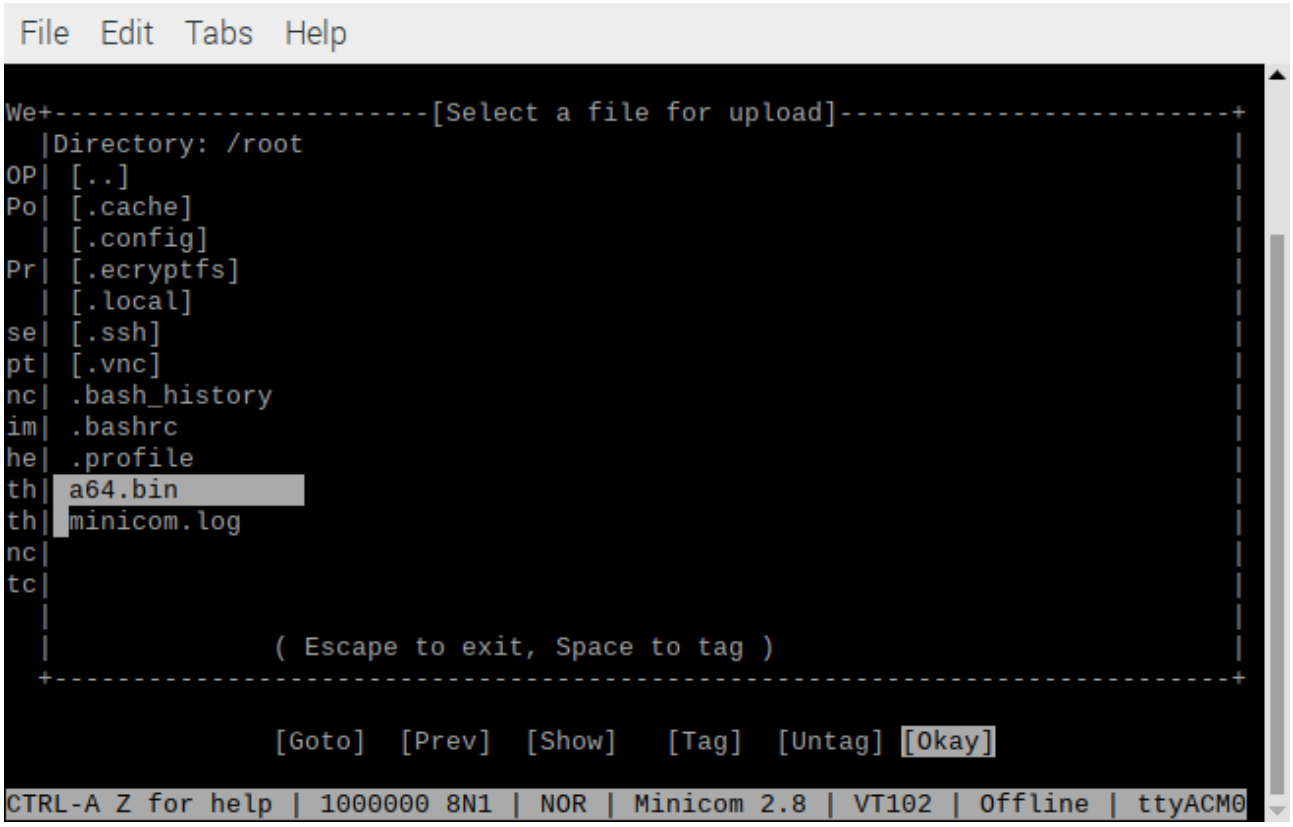
openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program klt-test/klt-test.elf verify reset exit"



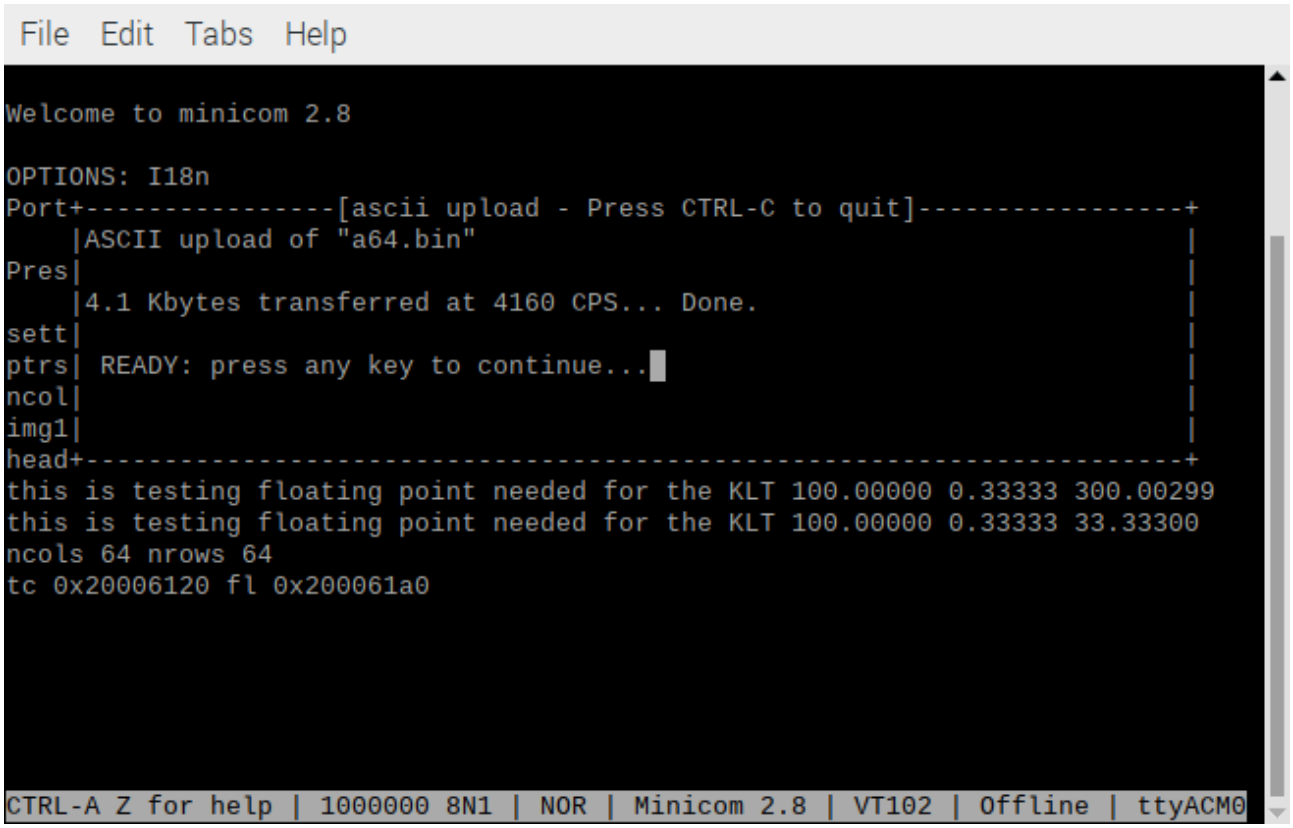
ctrl A S Scroll down to ascii



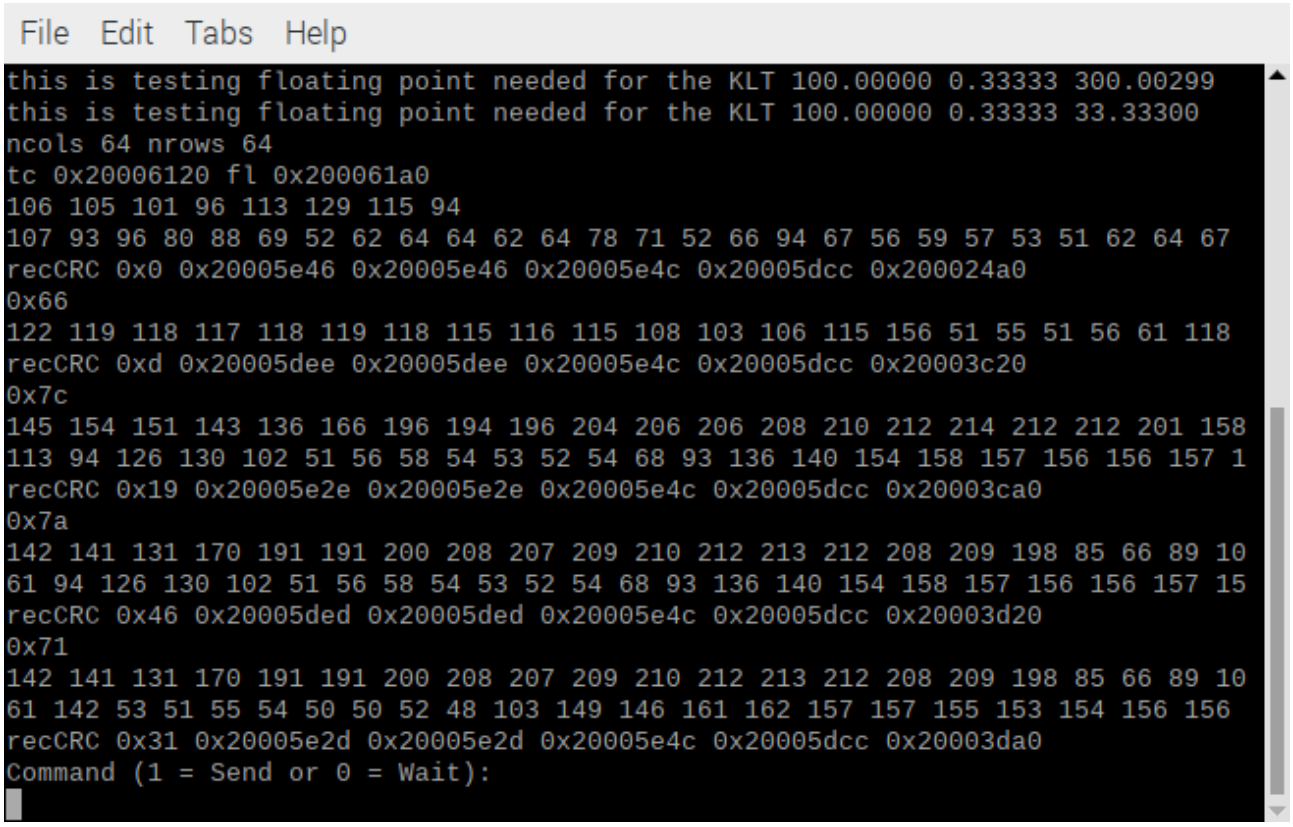
Depress Enter



Scroll down to a64.bin Depress Enter



Depress Enter


File Edit Tabs Help
this is testing floating point needed for the KLT 100.00000 0.33333 300.00299
this is testing floating point needed for the KLT 100.00000 0.33333 33.33300
ncols 64 nrows 64
tc 0x20006120 fl 0x200061a0
106 105 101 96 113 129 115 94
107 93 96 80 88 69 52 62 64 64 62 64 78 71 52 66 94 67 56 59 57 53 51 62 64 67
recCRC 0x0 0x20005e46 0x20005e46 0x20005e4c 0x20005dcc 0x200024a0
0x66
122 119 118 117 118 119 118 115 116 115 108 103 106 115 156 51 55 51 56 61 118
recCRC 0xd 0x20005dee 0x20005dee 0x20005e4c 0x20005dcc 0x20003c20
0x7c
145 154 151 143 136 166 196 194 196 204 206 206 208 210 212 214 212 212 201 158
113 94 126 130 102 51 56 58 54 53 52 54 68 93 136 140 154 158 157 156 156 157 1
recCRC 0x19 0x20005e2e 0x20005e2e 0x20005e4c 0x20005dcc 0x20003ca0
0x7a
142 141 131 170 191 191 200 208 207 209 210 212 213 212 208 209 198 85 66 89 10
61 94 126 130 102 51 56 58 54 53 52 54 68 93 136 140 154 158 157 156 156 157 15
recCRC 0x46 0x20005ded 0x20005ded 0x20005e4c 0x20005dcc 0x20003d20
0x71
142 141 131 170 191 191 200 208 207 209 210 212 213 212 208 209 198 85 66 89 10
61 142 53 51 55 54 50 50 52 48 103 149 146 161 162 157 157 155 153 154 156 156
recCRC 0x31 0x20005e2d 0x20005e2d 0x20005e4c 0x20005dcc 0x20003da0
Command (1 = Send or 0 = Wait):

Depress 1

```
File  Edit  Tabs  Help
142 141 131 170 191 191 200 208 207 209 210 212 213 212 208 209 198 85 66 89 10
61 142 53 51 55 54 50 50 52 48 103 149 146 161 162 157 157 155 153 154 156 156
recCRC 0x31 0x20005e2d 0x20005e2d 0x20005e4c 0x20005dcc 0x20003da0
Command (1 = Send or 0 = Wait):
need to copy the data received from host to img1
img1 = 0x2003ffd8 img2 = 0x20040fd8
0 img1 161 ptrs.buf 161
1 img1 157 ptrs.buf 157
2 img1 156 ptrs.buf 156
3 img1 157 ptrs.buf 157
4 img1 159 ptrs.buf 159
4091 img1 62 ptrs.buf 62
4092 img1 67 ptrs.buf 67
4093 img1 63 ptrs.buf 63
4094 img1 59 ptrs.buf 59
4095 img1 91 ptrs.buf 91
need to copy the data from img1 to img2
0 img2 161 img1 161
1 img2 157 img1 157
2 img2 156 img1 156
3 img2 157 img1 157
4 img2 159 img1 159
4091 img2 62 img1 62
4092 img2 67 img1 67
4093 img2 63 img1 63
4094 img2 59 img1 59
4095 img2 91 img1 91
img1 = 0x2003ffd8 img2 = 0x20040fd8

In first image:
Feature #0:  (24.000000,32.000000) with value of 4472
Feature #1:  (34.000000,39.000000) with value of 3461
Feature #2:  (36.000000,29.000000) with value of 3100
Feature #3:  (-1.000000,-1.000000) with value of -1
Feature #4:  (-1.000000,-1.000000) with value of -1
Feature #5:  (-1.000000,-1.000000) with value of -1
Feature #6:  (-1.000000,-1.000000) with value of -1
```

When img0.pgm and img1.pgm both have the 64 x 64 image lena_rgb_64.pgm from /pico-lifting/testfiles/ and the program
./example1 is executed in klt-feature-detect/klt. The image feat2.ppm is created.
The Principal component analysis (PCA) are in feat1.txt
In first image:
Feature #0: (32.000000,24.000000) with value of 4472
Feature #1: (39.000000,34.000000) with value of 3461
Feature #2: (29.000000,36.000000) with value of 3100