How to Convert the CCD Camera Cookbook 245 and 211 Acquisition Interface Board to USB

By Simple-Circuit

Usually I don't start my project text with a caution. However, this project is not a step by step process with hold your hand instructions. Nor has it been through rigorous testing on different systems. In fact, it is a marginal hardware design that just happens to work. The software is functional for reading the camera and storing images. Some features are still a work in progress. Source code for the interface board is provided so you can hack away. The application is under development and source will be released on completion.

The project is designed to use the camera body, CCD chip and preamp board. The old interface box gets tossed. The new card has the voltage regulation, analog to digital conversion and the USB interface. Power from the USB generates the plus and minus 15V but the DS0026 drivers must be replaced with TC1426 CMOS drivers. This is required to drop the current draw low enough for DC to DC converters not to overload the USB power.

The new card uses a Teensy4.0 processor to drive the CCD clocks and read the ADC. It also takes care of the USB communication. The new analog to digital converter is a MCP3302 12-bit SPI interface device. Although the Teensy comes with a built in ADC, the external converter has better camera performance.

The TC245 readout speed is comparable to that of the parallel port interface. Here are some typical values:

Fast	0.30sec
Focus	0.43sec
Find	0.75sec
256 internal bin	2.0sec
256 external bin	4.6 sec
378	7.25sec
756	9.9sec

The digital I/O for ARM processor in the Teensy requires mapping of the internal register to the external pins. The following table shows the J1 connector mapping to the data pins, the internal register bit and the function for each camera type.

J1 Pin	Tensy4.0 Data I/O	Register Bit 2^N	245 Function	211 Function
11	14	18	SRG1	SRG
13	15	19	SRG2	IAG
9	16	23	SRG3	--
15	17	22	TRG	SRG CDS Mod
4	18	17	IAG	Low Dark Current
2	19	16	SAG	--
J2 Center Pin	20	26	Low Dark Current	Amp Electroluminescense

To generate clocks in the Arduino sketch, the clock bits are defined then added to generate the required transitions.

```
//TC245 Camera Constants
#define SRG1 0x00040000   //1<<18       //pin-14 = srg1 clock
#define SRG2 0x00080000  //1<<19       //pin-15 = srg2 clock
#define SRG3 0x00800000  //1<<23       //pin-16 2 = srg3 clock
#define TRG  0x00400000  //1<<22       //pin-17 = trg clock
#define IAG  0x00020000  //1<<17      //pin-18 = iag clock
#define SAG  0x00010000  //1<<16      //pin-19 = sag clock
#define LOWNOISE 0x04000000 //1<<26  //pin-20 = low dark current
#define CCDINIT  LOWNOISE     //normal static state of clock lines

//Image area clear clock sequence
#define CL1 CCDINIT+TRG
#define CL2 CCDINIT+TRG+SAG
#define CL3 CCDINIT+SAG+IAG+SRG1+SRG2+SRG3
#define CL4 CCDINIT+IAG+SRG1+SRG2+SRG3
```

The data register bits are set to the clock levels by ANDing the register with a mask to zero the I/O bits then the new data is ORed with the result.

```
void outlow(uint32_t k){
 GPIO6_DR  = (GPIO6_DR & 0xF030FFFF) | k;
 delayNanoseconds(500);
}
```

Commands are processed by receiving a start character = 255. The command interpreter then reads input data as follows:

Byte-1: Mode of operation
Byte-2: Options bit-0 low dark current mode and bit-1 no amplifier electroluminescence
Byte-3 and Byte-4: Image X-shift (two 7-bit values to make 14-bit data)
Byte-5 and Byte-6: Image Y-shift (two 7-bit values to make 14-bit data)
Byte-7, Byte-8 and Byte-9: Integration Time in ms (three 7-bit values to make 21-bit data)

The program will set the mode and options. The CCD will be cleared of charge and integration runs for the time in milliseconds or until any character is received.

After integration, the CCD is read and data is sent as two 8-bit characters (16-bit data) for each pixel. Data is sent starting at the upper left corner. Row by row are sent until all row by height (e.g. 252 x 242) values are sent. The measured integration time in milliseconds is sent as three 7-bit characters (21-bit value). An application written using the Lazarus IDE communicates with the Teensy and displays the images.

Here is a list of the modes that control the readout:

TC245:

1 – Clear Image Area
2 – Move Image to Storage Area
3 – Clear Storage Area
4 – Clear Serial Register
5 – Read Reset and Reference Values (returns two 16- bit values as four characters)
7 – Get 252x242 image internal binning
8 – Get 252x242 image internal binning low noise readout
9 – Get 126x121 image internal binning
10 – Get 126x121 center image (shifted X and Y)
11 – Get 96x82 center image (shifted X and Y)
12 – Get 252x242 image externally binned
13 – Get 126x121 image internal binning
14 – Get 378x242 image externally binned
15 – Get 378x242 image externally binned reference subtracted
16 – Get 756x242 image externally binned
17 – Get 756x242 image externally binned reference subtracted
99 – Send camera type as two bytes of 245

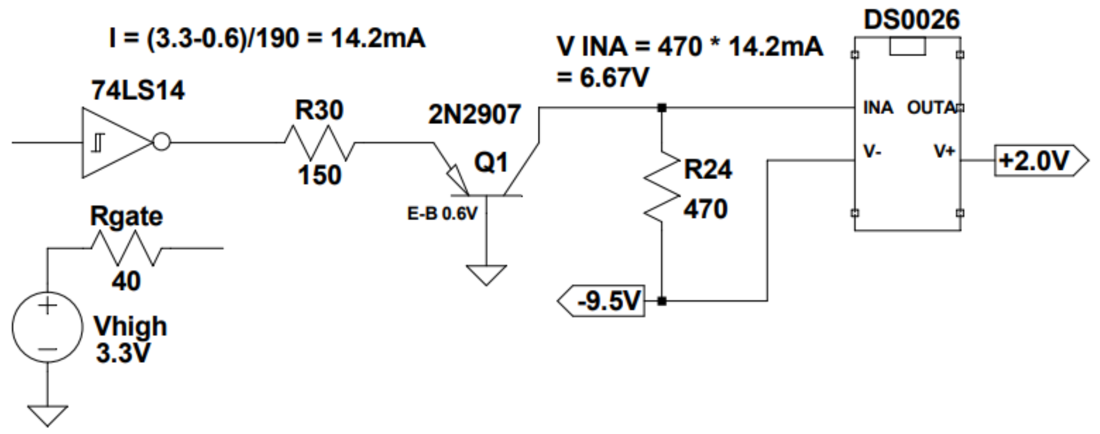
TC211:

19 – Get 192x156 image
20 – Get 96x82 internally binned image
21 – Get 96x82 center image (shifted X and Y)
22 – Get 192x165 image with reference subtracted
23 – Get 192x165 image with reference subtracted (sampled twice for lower noise)
24 – Read Reset and Reference Values (returns two 16- bit values as four characters)
25 – Get 192x165 image read with double correlated sampling (requires mod to preamp board)
99 – Send camera type as two bytes of 211

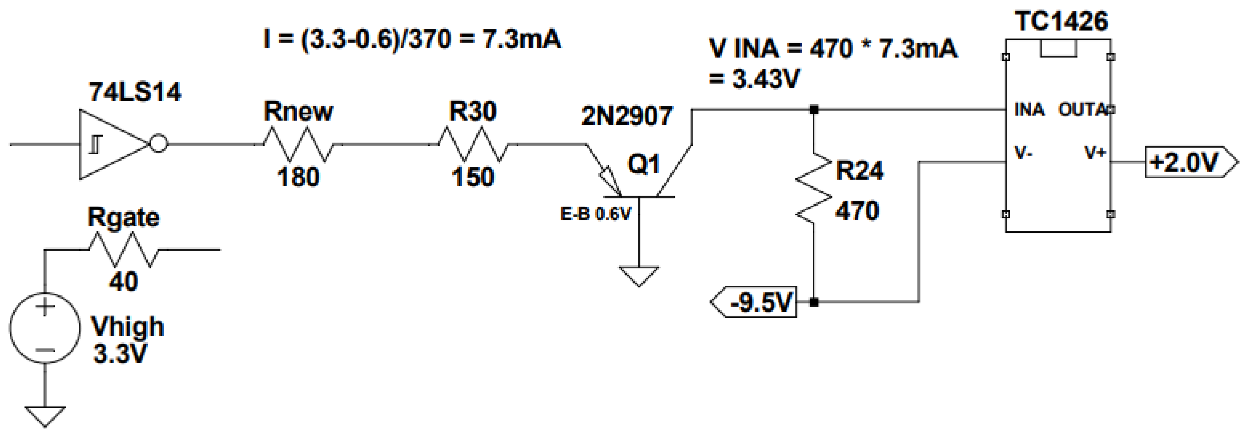
Code for the Teensy is found in sketches: cook_book211.ino and cook_book245.ino

As stated earlier, the DS0026 chips will need to be replaced with lower power CMOS TC1426 types. Also, the current in the level shifter must be reduced. The new card will have some series resistors that drop the current. Nominally, each level shifter requires about 14.2mA. The TC245 camera has six level shifters and the total current is 85.2mA. By adding a series 180 ohm resistor, the current drops to 7.3mA and this lowers the draw to about 44mA. The logic high at the driver will be around 3.4V which is just above the 3V logic high minimum for the TC1426.

The original level shifter:



The lower current level shifter:

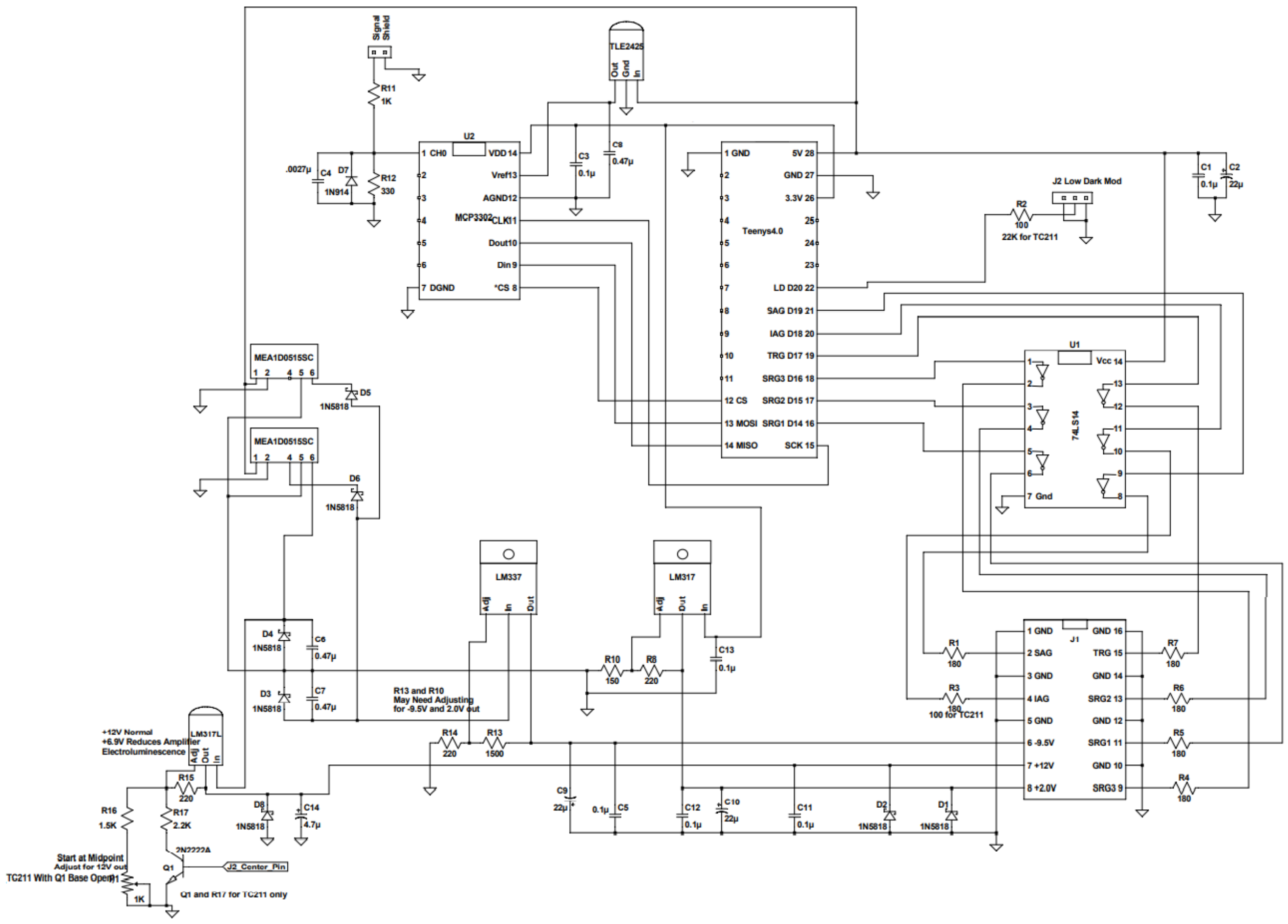


I am using two MEA1D0515SC DC to DC converters to generate the ±15V for the circuit. A single higher current converter will probably work. The parts used in this design are ones I had on hand. You are welcome to try other parts.

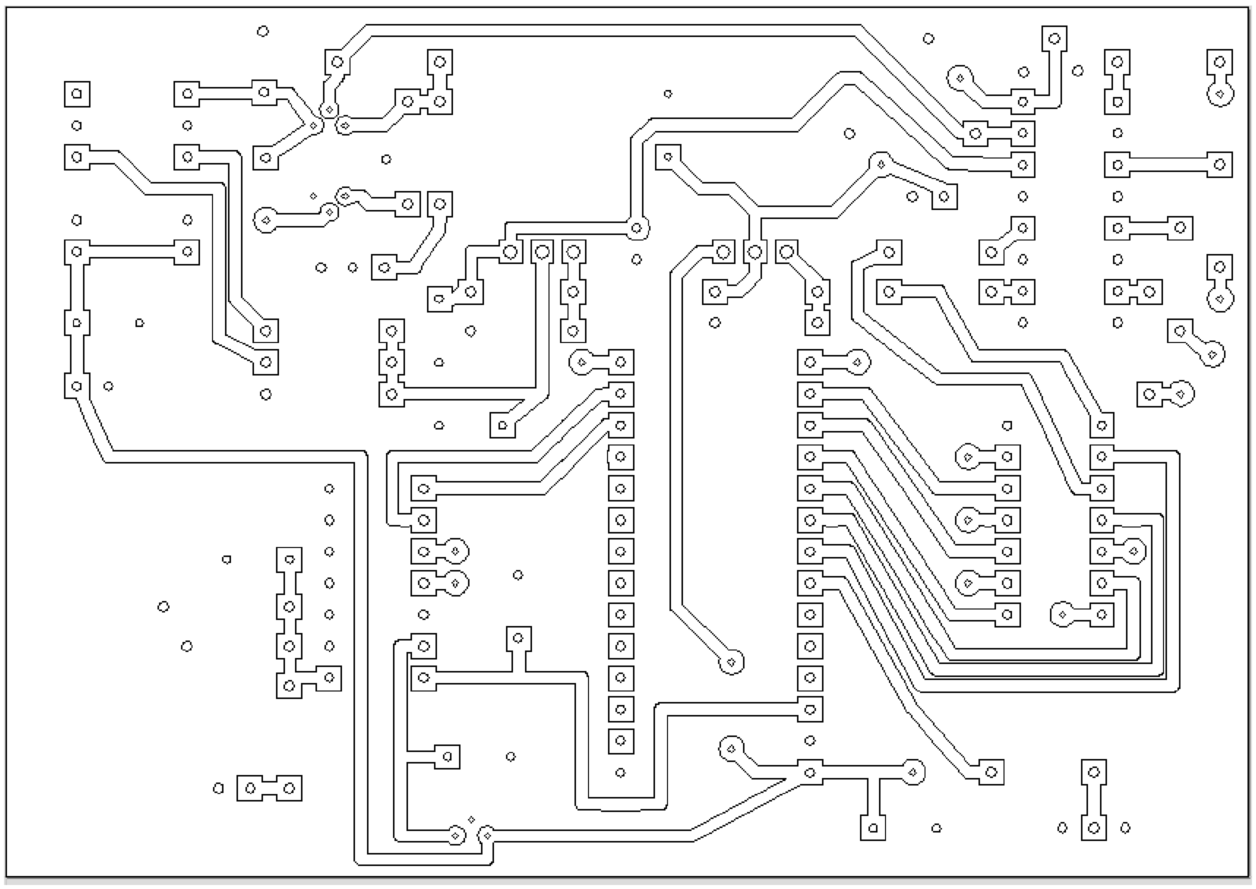
The TC211 camera requires a shifting of the 12V supply to reduce the amplifier glow during integration. Resistor R2 is replaced with a 22K resistor and the center pin of J2 connects to the base of Q1. The control comes from the D20 of the Teensy4.0 processor.

For the TC245 camera, J2 is used to level shift the 2.5V bias on the CCD. This is for the Reduced dark current modification. Resistor R2 remains at 100 ohms. The TC211 cameras with the Reduced dark current modification uses the D18 or IAG line. Resistor R3 is changed to 100 ohms to ensure the level shifter has enough current.

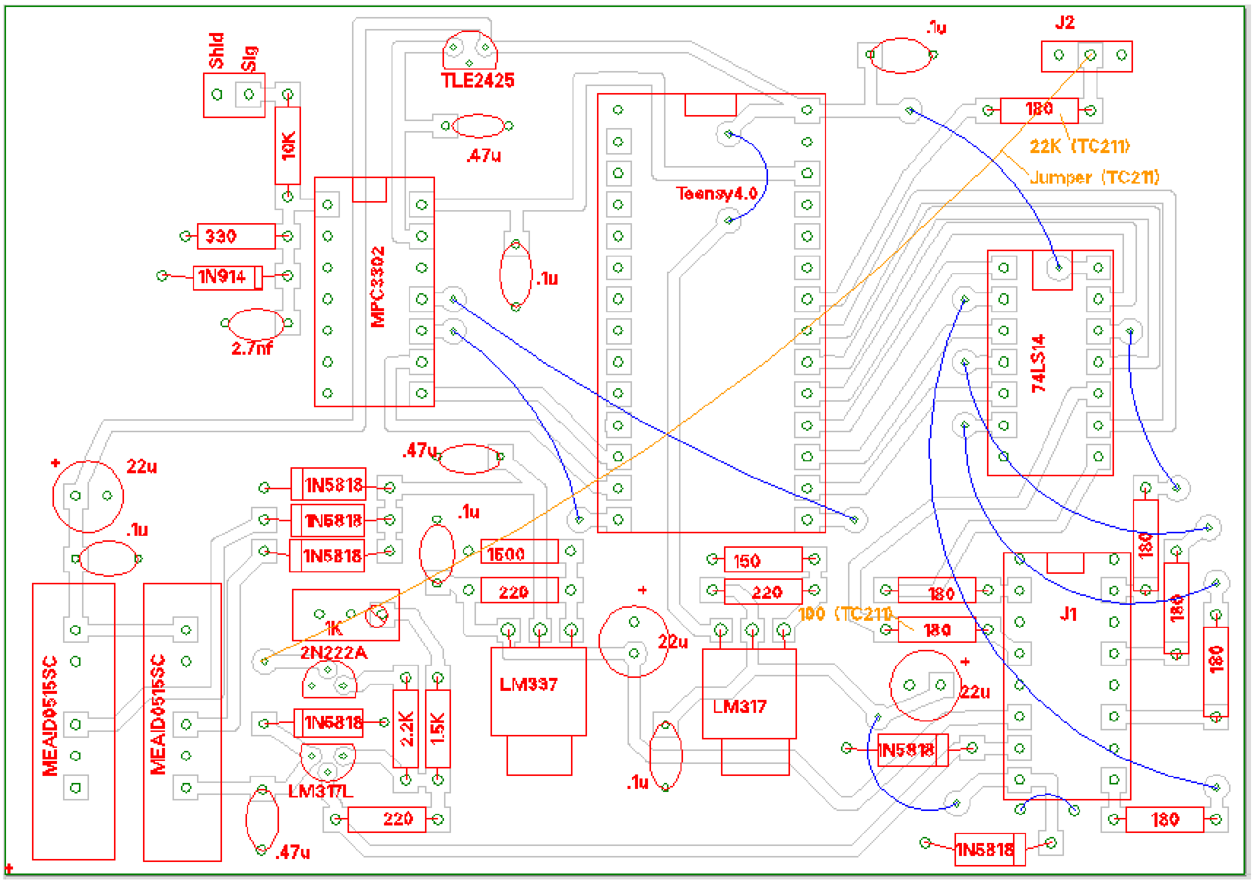
The new interface board schematic diagram:



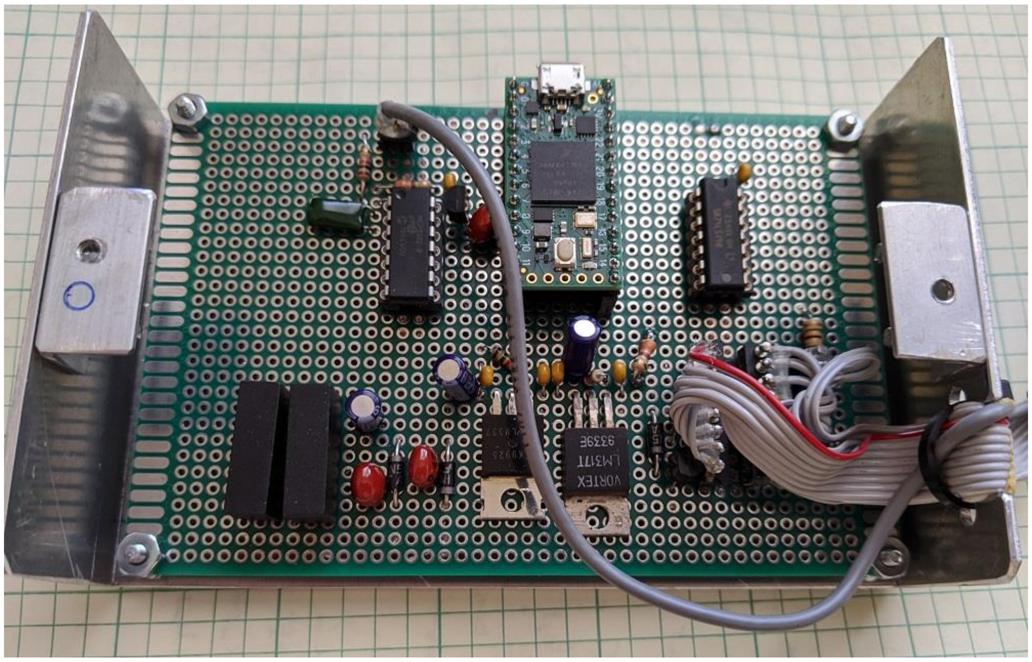
The circuit can be built on a 100mm X 70mm X 1.5mm single sided circuit card (bottom side). It is tight but fits. Several jumper wires are required. CNC drill and profile files for GRBL routers is provided in files cbdrill.gcode and cbprofile.gcode



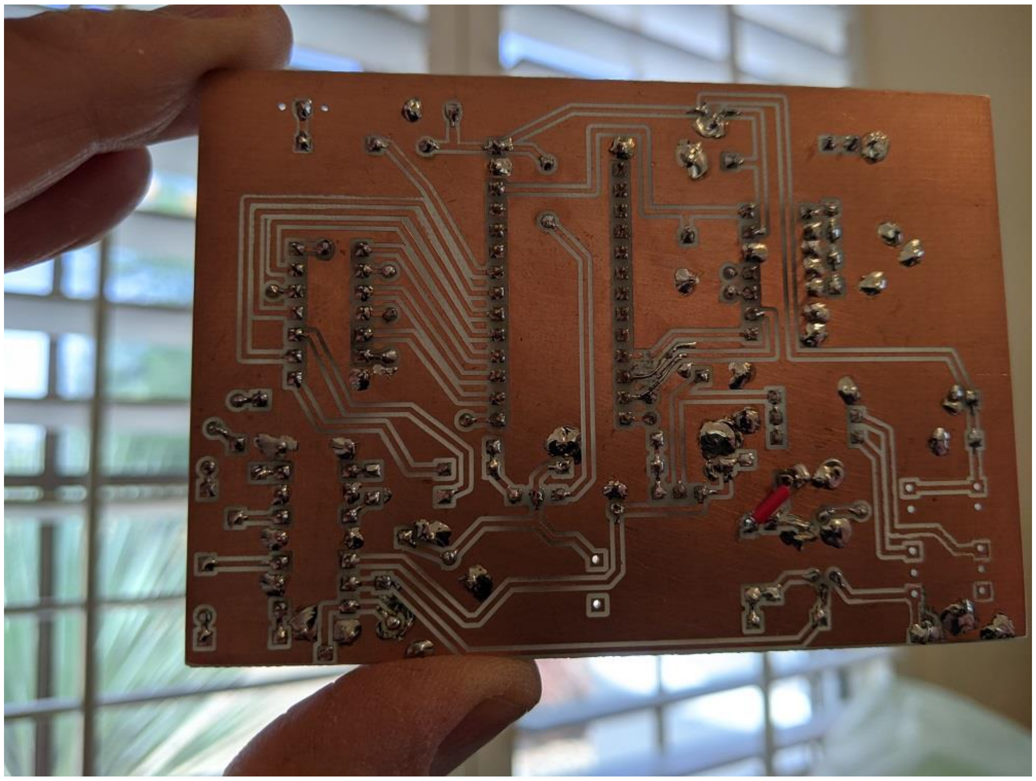
Component top side placement. The blue wires are required jumpers. The extra TC211 jumper is shown in brown.



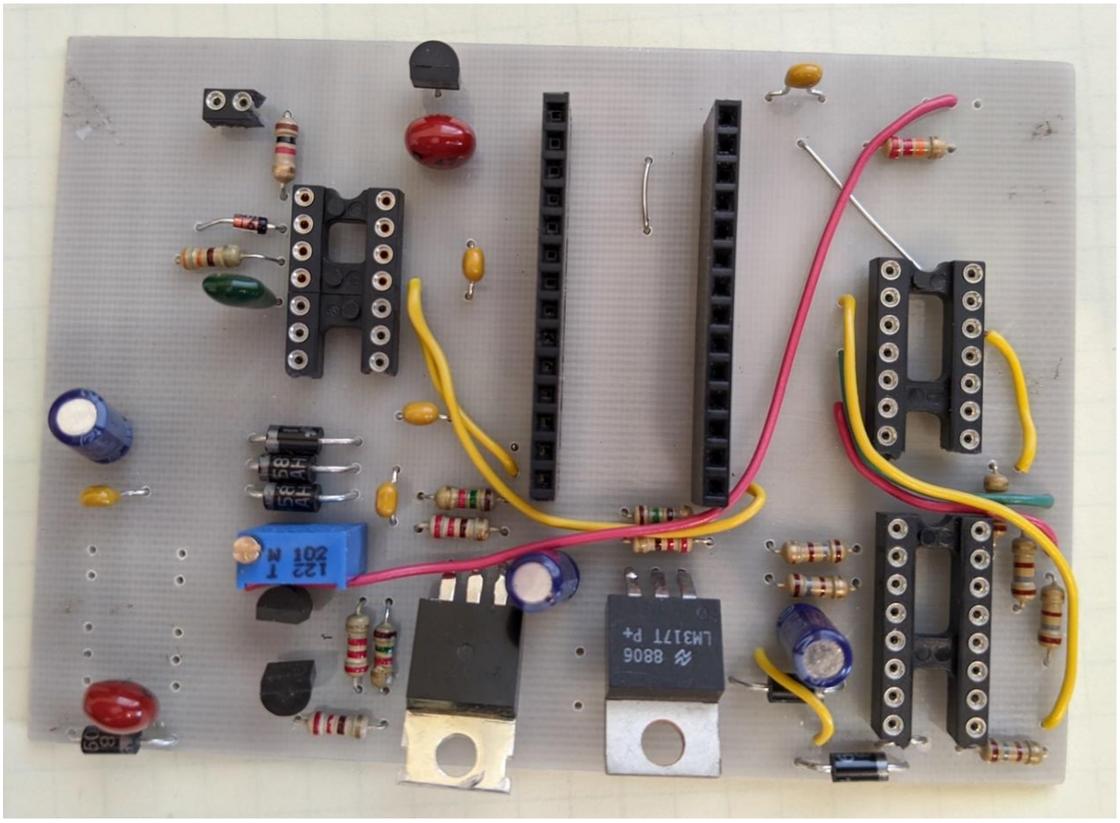
This is the first iteration of the design. The circuit is hand wired using a perf board.



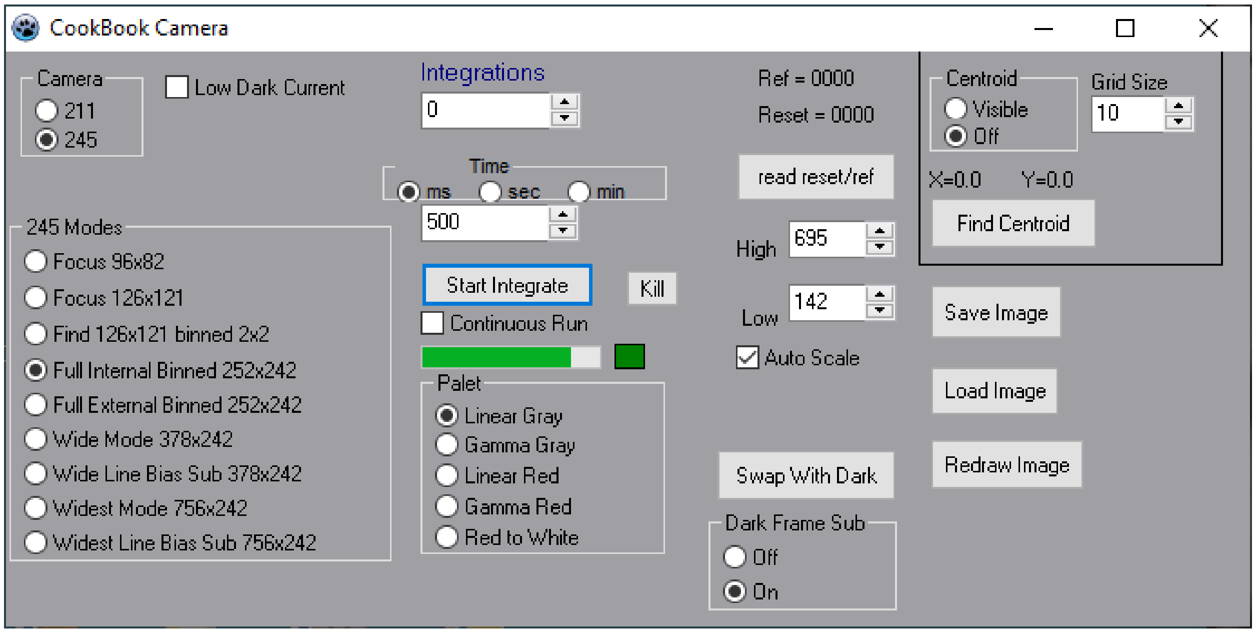
The second iteration uses a CNC mill to cut a single sided board. Note the red jumper wire. Several cuts and jumpers were needed to correct mistakes. The CNC files have been updated.



A partially populated circuit board showing the various jumpers needed for a single sided board. There were a few changes for the final version.



The user interface is written using the Lazarus IDE (free pascal compiler). It has basic functions to set the camera mode, options, integration time, display and file system.



CookBook Camera

Camera: 211 [✓ 245]

☐ Low Dark Current

245 Modes:
- Focus 96x82
- Focus 126x121
- Find 126x121 binned 2x2
- [✓ Full Internal Binned 252x242]
- Full External Binned 252x242
- Wide Mode 378x242
- Wide Line Bias Sub 378x242
- Widest Mode 756x242
- Widest Line Bias Sub 756x242

Integrations: 0

Time: [✓ ms] sec min — 500

☐ Continuous Run

Palet:
- [✓ Linear Gray]
- Gamma Gray
- Linear Red
- Gamma Red
- Red to White

Ref = 0000

Reset = 0000

High: 695

Low: 142

☑ Auto Scale

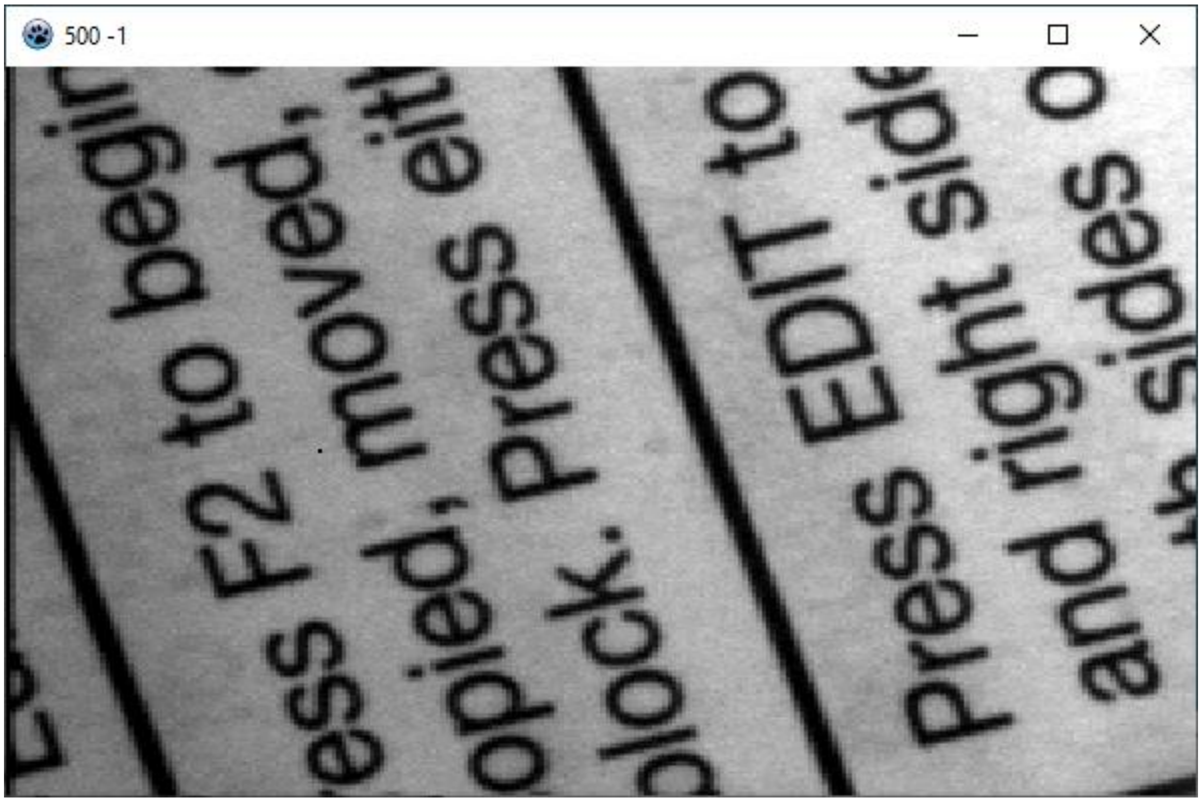
Dark Frame Sub: Off [✓ On]

Centroid: Visible [✓ Off]

Grid Size: 10

X=0.0 Y=0.0

Start Integrate
Kill
read reset/ref
Swap With Dark
Find Centroid
Save Image
Load Image
Redraw Image



500 -1