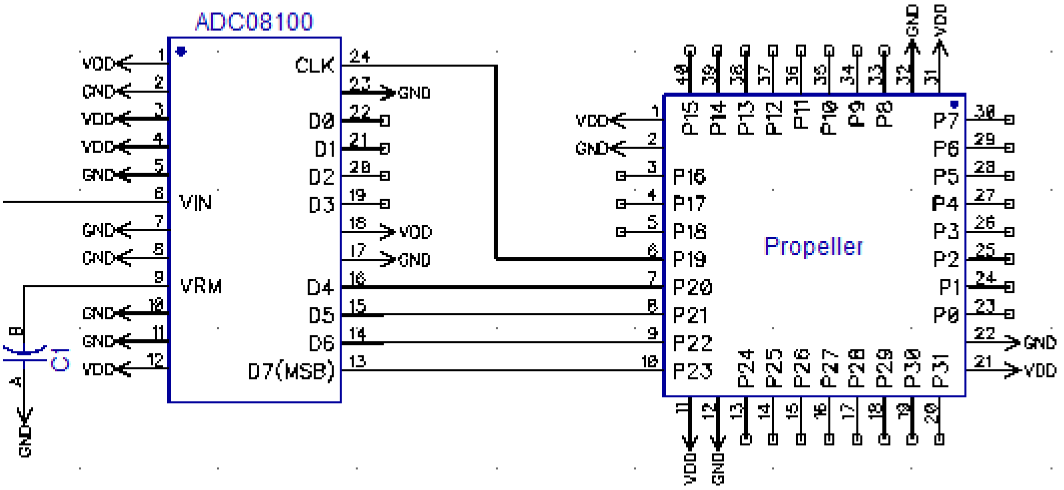
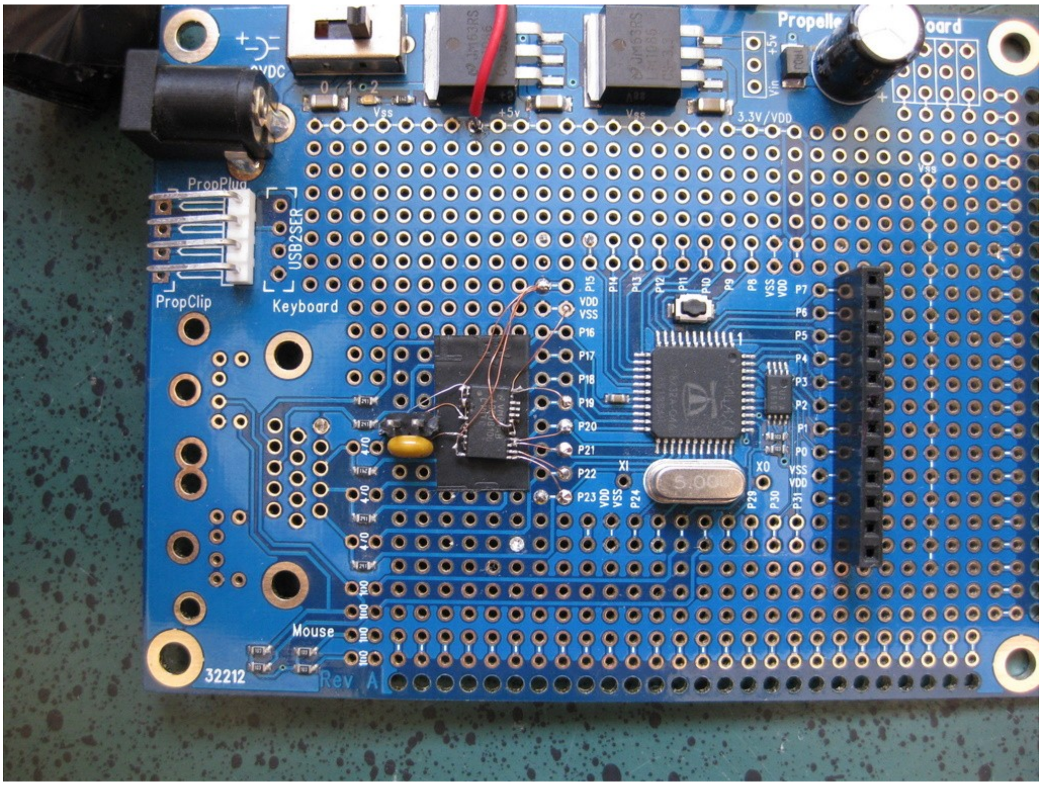
Using the ADC08100, a Propeller ProtoBoard and ViewPort to capture video
Hanno Sander
http://hannoware.com

Assembly:
Solder wires to the ADC08100 and connect them as follows directly on the ProtoBoard:

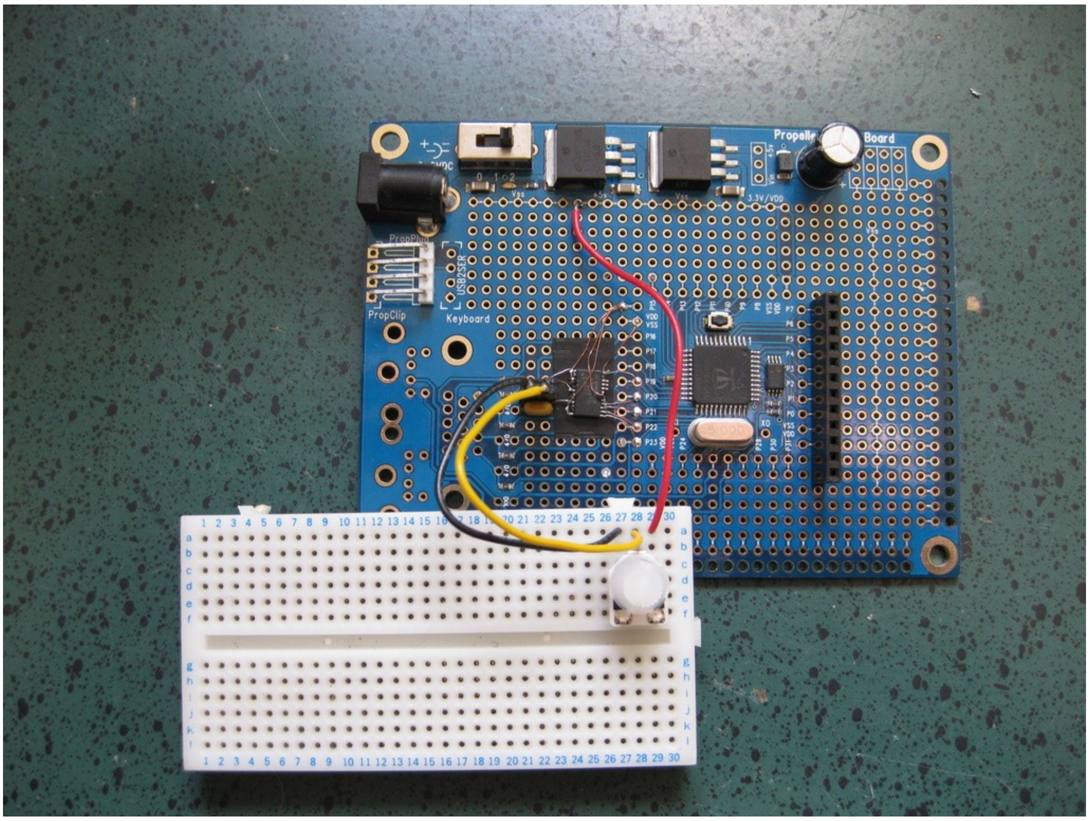


Here's a photo:



This circuit uses the ADC as a high speed ADC- the analog voltage on Vin is converted into a digital equivalent feeding Propeller P20..P23. P19 is used to clock the ADC.

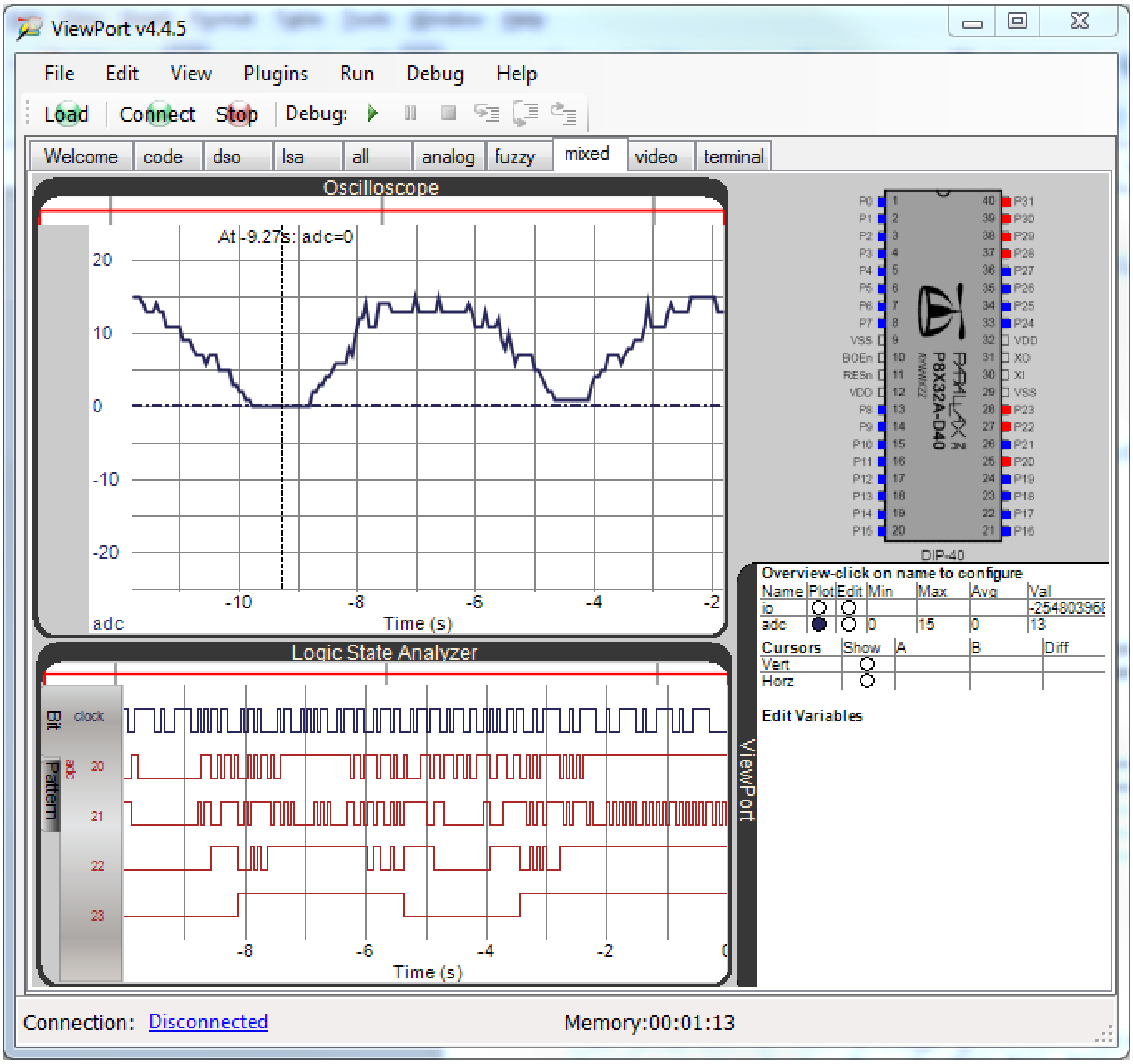
You can feed any analog signals into P6 of the ADC08100- ideally between 0 and 3V. Here's a photo of a poti connected to the ADC:



The code to read this into ViewPort is simple:

```
CON
 _clkmode         = xtal1 + pll16x
 _xinfreq         = 5_000_000
OBJ
 vp   :         "Conduit" 'transfers data to/from PC
pub demoADC|io
 vp.config(string("var:io(bits=[19clock,adc[20..23]]),adc(decode=io[20..23])"))
 vp.share(@io,@io)   'share just the io variable (vp will decode the adc channel)
 dira[19]~~          'output clock on pin 19
 repeat
  io:=ina            'read io pins- including adc
  !outa[19]
```

Here's a ViewPort screenshot showing both analog and digital representations of the resulting signal. The upper “DSO” graph shows the analog value of the ADC- it corresponds to the potentiometer's position as I turned it. The “LSA” graph shows the actual digital input to the Propeller's pins- notice the clock on pin 19 and the ADC's outputs on P20-P23:



Overview-click on name to configure

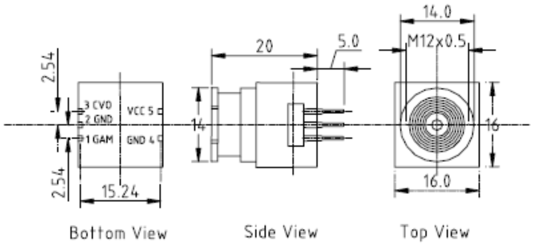
Name	Plot	Edit	Min	Max	Avg	Val
io	○	○				-25480396E
adc	●	○	0	15	0	13

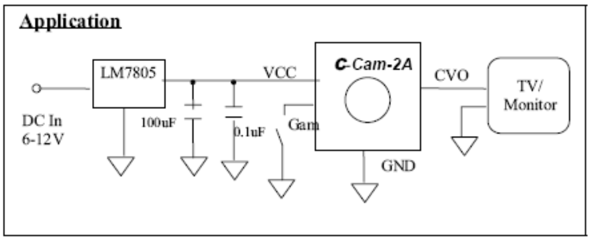
Cursors	Show	A	B	Diff
Vert	○			
Horz	○			

Edit Variables

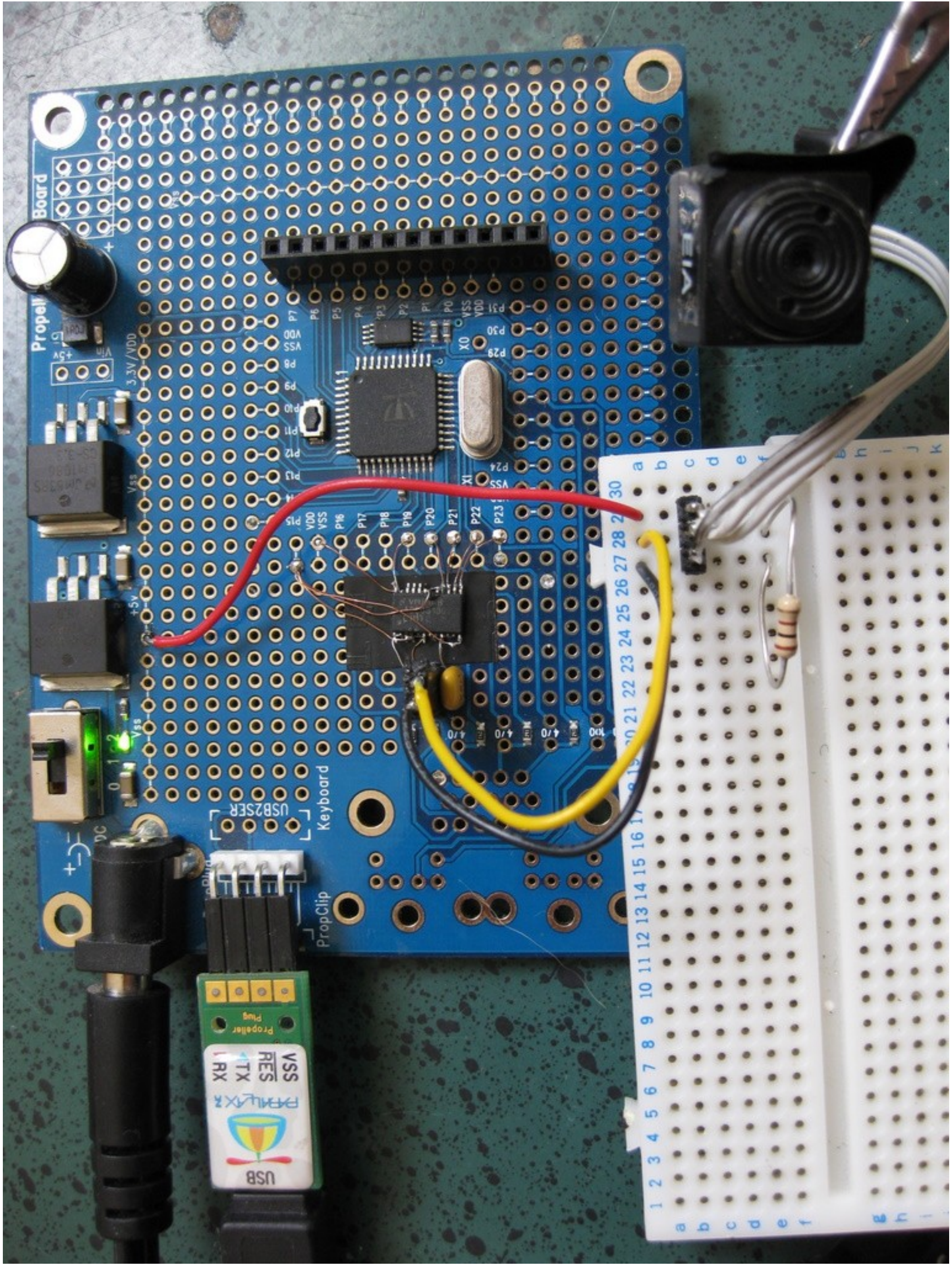
Connection: Disconnected Memory:00:01:13

Here's a schematic of a grayscale NTSC camera- we can feed it's output (CVO) to Vin with a 100ohm resistor to ground:





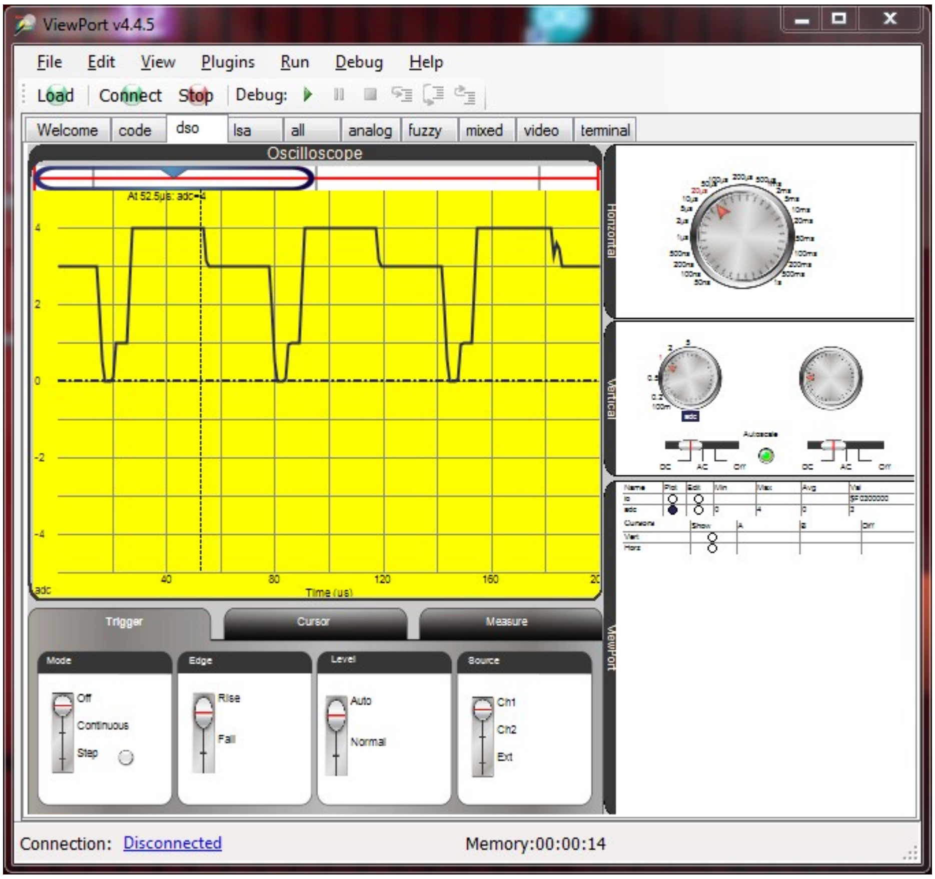
Here's a photo of the NTSC camera, ADC, and ProtoBoard:



Here's code to clock to read the NTSC signal at 10Msps:

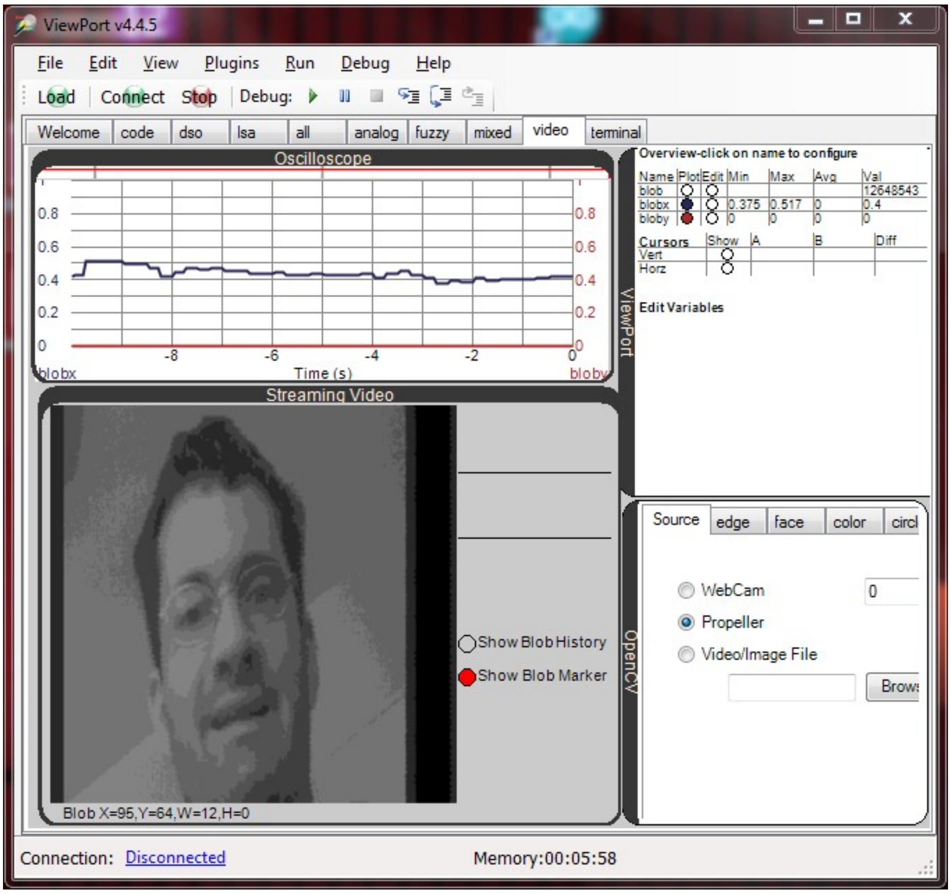
```
CON
 _clkmode         = xtal1 + pll16x
 _xinfreq         = 5_000_000
OBJ
 vp   :        "Conduit" 'transfers data to/from PC
 qs   :        "QuickSample"     'samples INA up to 80Mhz
 Freq :        "Synth"
pub demoADC|a,frame[1600+6] 'frame stores 1600 samples+configuration
 vp.register(qs.sampleINA(@frame,1 )) 'sample INA with 1 cogs up to 20Mhz4
 vp.config(string("var:io,adc(decode=io[20..23])"))
 vp.config(string("dso:view=adc,timescale=2ms"))
 vp.share(0,0)
 Freq.Synth("A",19, 10_000_000)
```

Here's a ViewPort screenshot showing several NTSC scanlines- each scanline encodes a horizontal line of pixels. Since we're using a grayscale camera, a pixel's brightness corresponds to the ADC's measurement at that time.



The PropCVCapture object looks for the horizontal and vertical sync marks in the pattern and then samples the data into memory. In one of its modes it continually grabs 240hx200v pixels at 16 grayscales into 24KB of HUB ram. ViewPort uses the Conduit object to stream this data so it can display roughly 10frames/second using a 2Mbps connection.

Here's the end result: a self portrait:



Overview-click on name to configure

Name	Plot	Edit	Min	Max	Avg	Val
blob	○	○				12648543
blobx	●	○	0.375	0.517	0	0.4
bloby	●	○	0	0	0	0

Cursors	Show	A	B	Diff
Vert	○			
Horz	○			

Edit Variables

○ Show Blob History

● Show Blob Marker

Blob X=95,Y=64,W=12,H=0

○ WebCam 0

◉ Propeller

○ Video/Image File

Connection: Disconnected

Memory:00:05:58