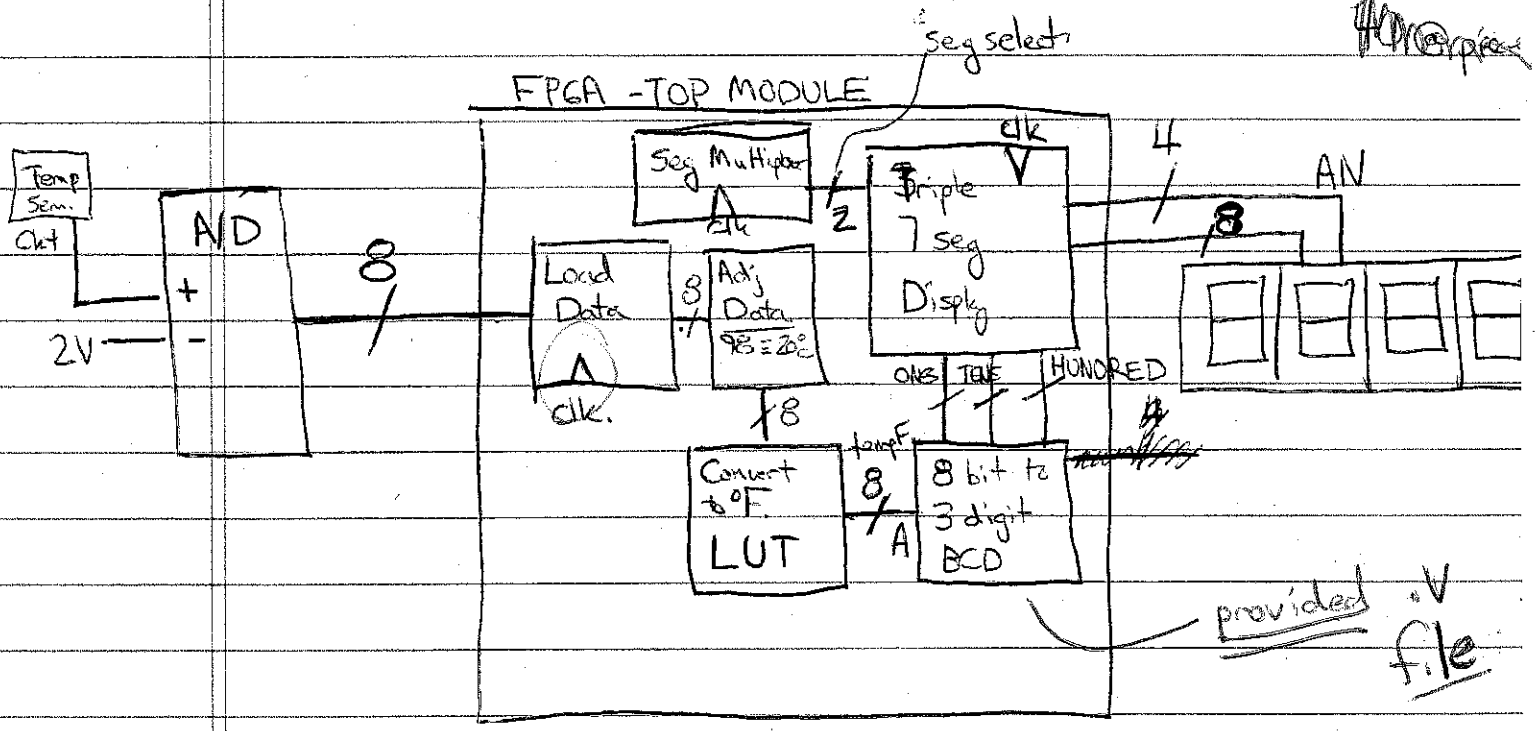
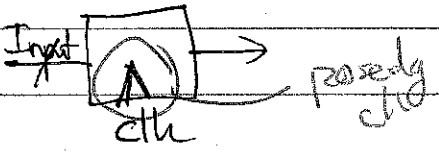
Lab #9



OTHER NOTES up to V-Log

Load Data – Latch in Data
always @(posedge clk) ← 8 bits
Data ⟸ Input

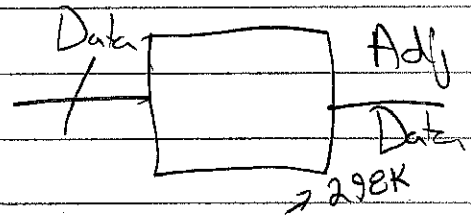
explain



→ Insures stable inputs (since A/D is asynchr & may change values)

Adjust Data

assign AdjDate = Data − #



with → V_ = 2 volts ≈ 25°C ⇒ 2.98V.

So 25°C corresponds to ≈ 98 in binary 01100010

273K = 0C

So easiest way is to subtract some constant so binary corresponds to temp.

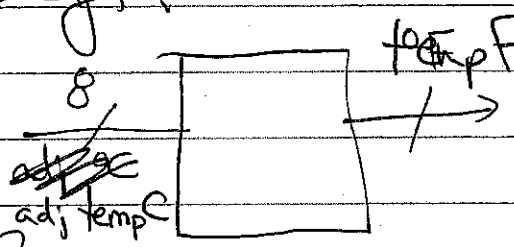
Data −73;

easiest (on A/D make V_ = 2.73)

but 2.73 + 1.25 × 2

5.23 > 5 lose 180 to 212°

Convert to DegF.



°F = 9/5 × °C + 32

unfortunately - Division not native & fractional div. is complicated Luckily, LUT free.

combinl always@(tempC)

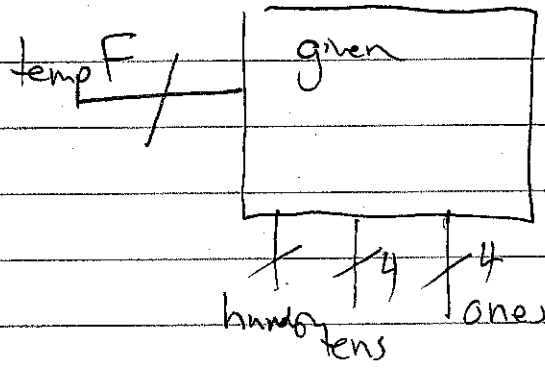
begin case(tempC)

15: tempF = 59;

16: tempF = 61;

⋮

8 bit to 3 digit BCD.
(provided 2 modules)



Segment Time Multiplexing Ctrl

outputs 2 bit # segselect

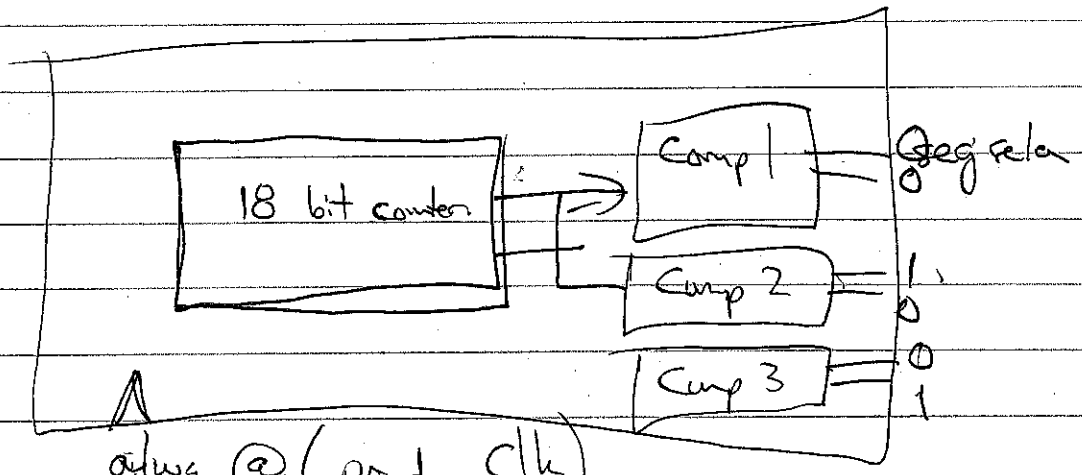
Mine
- 00 - seg Zero ones
- 01 - seg 1 tens
- 10 - seg 2 hundreds
- 11 - unused or for display data.

50 MHz
T = .02 μs
1 ms each
50,000

- 1 to 3 milliseconds → seg 1
- 1 to 3 milliseconds → seg 2
- 1 to 3 milliseconds → seg 3

outputs:	8 segs
1/3 time → 1110	1's
1/3 time → 1101	10's
1/3 time → 1011	100's

AN
~~seg select~~



always @(posedge clk)
begin
if (count < 65535)
seg select = 2'b00
else if (count < 131,071) ⇒ 01
else if (count < 196,606) ⇒ 10
{ other 1/4 off

Tired Sequential Execution

seg select 11

unused

hund | tens | ones

2

8 | 4

Triple Display

always @(posedge clk)
begin if (seg select == 2'b00)
begin AN = 4'b1110;
Case (Ones)
0: seg = 8'b00000011;
1
2
3
4
5
6
7
8
9
endcase
else if (seg select == 2'b01)