(20) Question 1. First, make a look up table (more points could be added, but these points were specifically given in the problem. Each point is given as the integer part of the fixed-point value

```
const unsigned short Input[]={   1,  10,100,254,1000,2000,10000};
const short Output[]        ={-200,-100,  0, 41, 100, 130,  200};
```

We then can use the same lookup/interpolation software developed for lab 7

(20) Question 2. Implement an SPI output channel with PT7 as its clock and PT6 as its data.

Part a) Show the ritual that initializes the interface. Be friendly. Show comments.

```
void SPI2_Init(void){
  DDRT |= 0xC0; // outputs
  PTT &= ~0x80; // CLK=0
}
```

Part b) Data is stable on rising edge of clock

```
void SPI2_Out(unsigned char data){
unsigned int bit;
  for(bit=8; bit; bit--){
    if(data&0x80){  // bits 7,6,5,4,3,2,1,0
      PTT |= 0x40;  // data bit high
    } else{
      PTT &= ~0x40; // data bit low
    }
    PTT |= 0x80;    // CLK=1, rising edge
    data = data<<1; // next bit
    PTT &= ~0x80;   // CLK=0
  }
}
```

(30) Question 3. connect a 74HC374 octal D flip-flip directly to the address/data bus if the 6811.

Part a) The address decoder for the 74HC374 and the EEPROM

$0000 to $01FF	0000,000X,XXXX,XXXX	9 X's
$1000 to $103F	0001,0000,00XX,XXXX	6 X's
$4000	0100,0000,0000,0000	no X's
$B600 to $B7FF	1011,011X,XXXX,XXXX	9 X's
$BF00 to $BFFF	1011,1111,XXXX,XXXX	8 X's
$E000 to $FFFF	111X,XXXX,XXXX,XXXX	13 X's

Choose address lines A15,A14:

OutputSelect = not(A15)*A14 EEPROMSelect = A15*A14

Part b) /OE is grounded because the output is always available. Rising edge of CLK at 500ns.

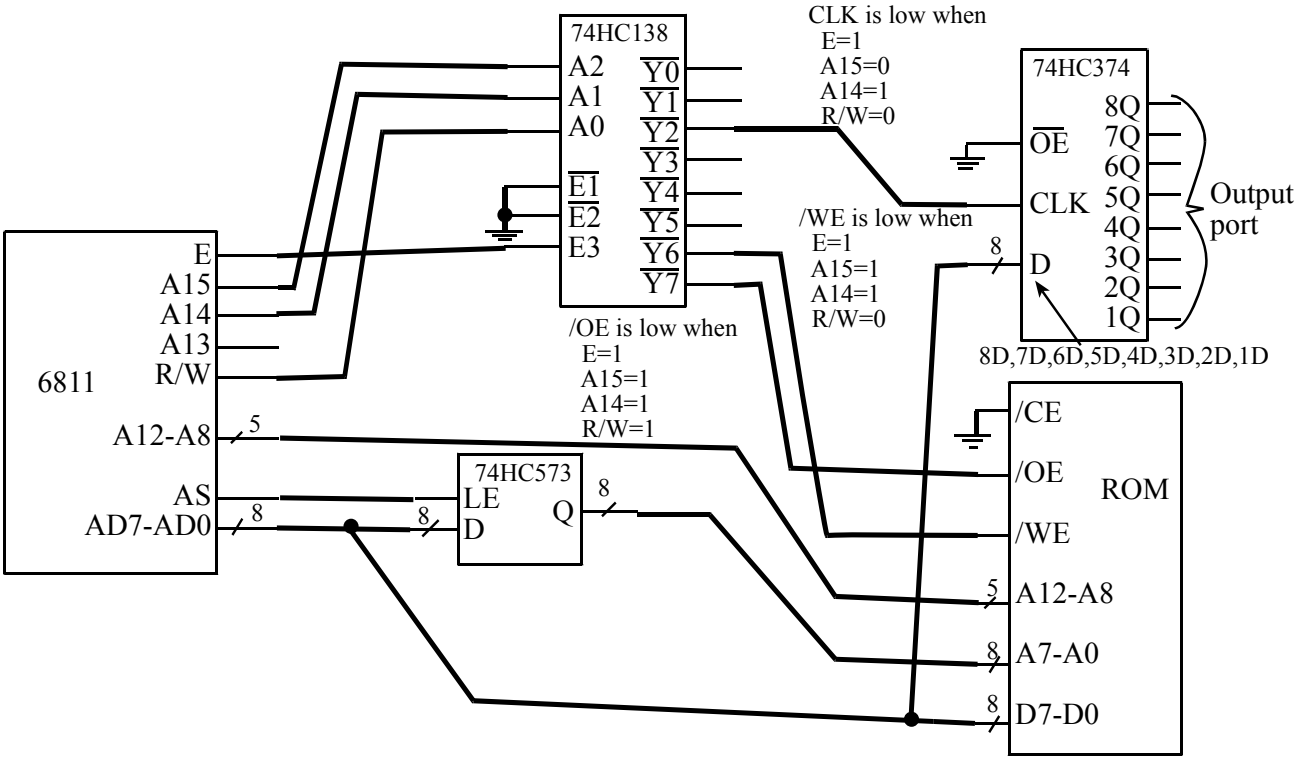
Select	R/W	E	CLK	/OE	action
0	0	0	1	0	Wrong address
0	0	1	1	0	Wrong address
0	1	0	1	0	Wrong address
0	1	1	1	0	Wrong address
1	0	0	1	0	
1	0	1	0	0	Clock on rising edge
1	1	0	1	0	Ignore read cycles
1	1	1	1	0	Ignore read cycles

Part c) Interface both the EEPROM and the 74HC374 to the 6811.

CLK = not(E*not(R/W)*not(A15)*A14) so rising edge of CLK occurs at end of write cycle to $4000

/OE = not(E*R/W*A15*A14) so data driven second half of read cycle from $E000-FFFF

/WE = not(E*not(R/W)*A15*A14) so rising edge of /WE occurs at end of write cycle to $E000-FFFF



Part d) The rising edge of CLK stores into the 374 with a setup of 30 and a hold of 5ns
WDA = Write Data Available = (2 + 4 + 19 , 1 + 21)
= (230 + 20 + 128 , 500 + 33) = (378 , 533)
Write Data Required = (↑CLK – setup , ↑CLK+hold) = (↑CLK – 30 , ↑CLK+5)
= (↓E+10 - 30 , ↓E+10+5) = (500+10 - 30 , 500+10+5) = (480 , 515)

(30) Question 4. The analog hardware for a data acquisition system to measure displacement
Part a) Fill in the following design table. The bridge output (V2-V1) is a differential voltage.

Displacement (mm)	R1 (Ω)	R2 (Ω)	R3 (Ω)	R4 (Ω)	V2 (V)	V1 (V)	V2-V1 (V)	V3 (V)	ADC result
-1	1010	990	990	1010	2.475	2.525	-0.05	0.00	0
-0.5	1005	995	995	1005	2.4875	2.5125	-0.025	1.25	256
0	1000	1000	1000	1000	2.5	2.5	0	2.50	512
0.5	995	1005	1005	995	2.5125	2.4875	0.025	3.75	768
1	990	1010	1010	990	2.525	2.475	0.05	5.00	1023

Need a gain of 5/(0.05 - -0.05)= 50 and an offset of 2.5V
V3 = 50(V2-V1) + 2.50
Part b) Use an instrumentation amp to get differential gain and good CMRR is required. AD627 gain is 5+200k/Rg, for this amp Rg = 200k/45= 4.44k. AD623 gain is 1+100k/Rg, for this amp Rg = 100k/49= 2.04k.