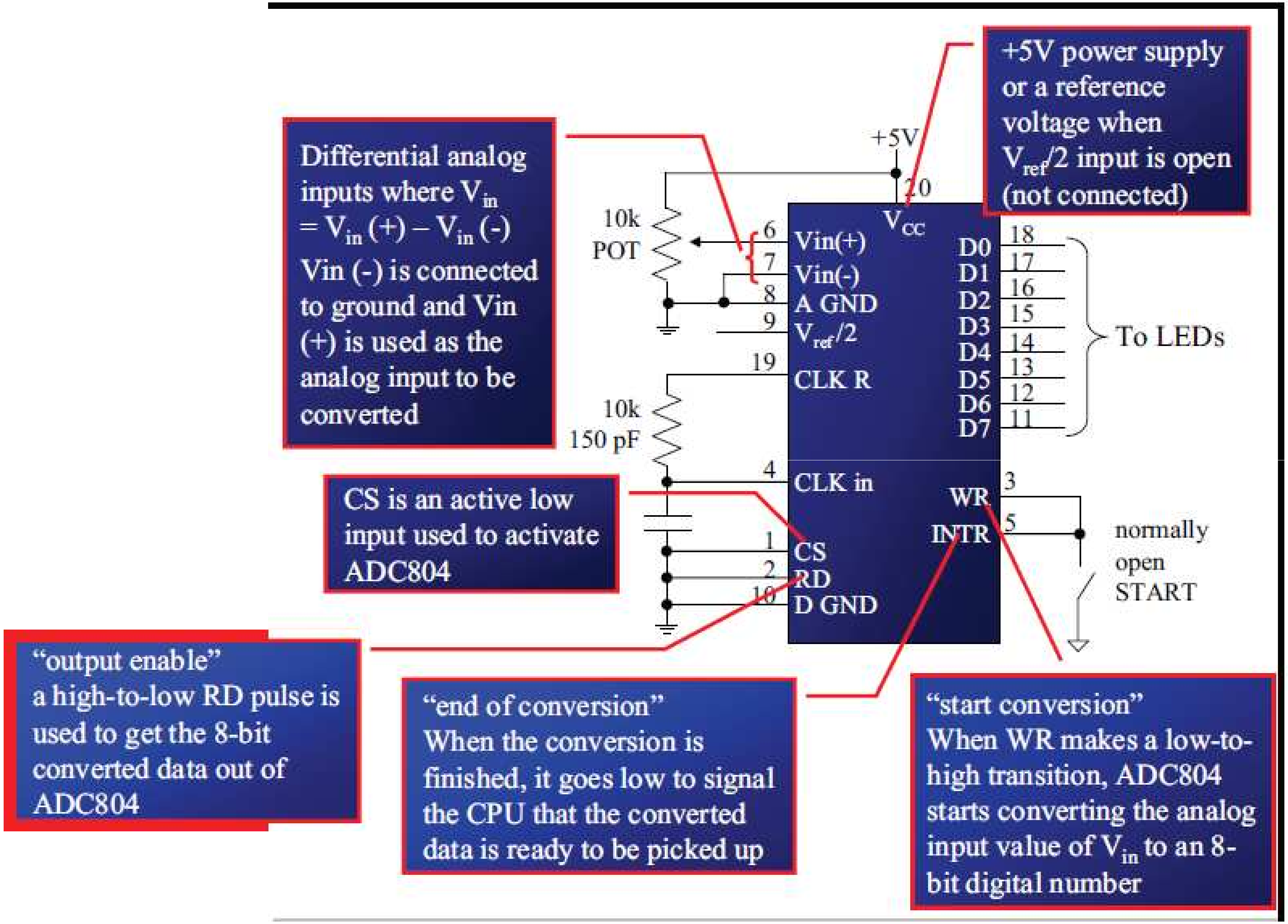
ADC

ADC Devices

- ADCs (analog-to-digital converters) are among the most widely used devices for data acquisition
- A physical quantity, like temperature, pressure, humidity, and velocity, etc., is converted to electrical (voltage, current) signals using a device called a transducer, or sensor
- We need an analog-to-digital converter to translate the analog signals to digital numbers, so microcontroller can read them

ADC804 Chip

- ADC804 IC is an analog-to-digital converter
  - It works with +5 volts and has a resolution of 8 bits
  - Conversion time is another major factor in judging an ADC
    - Conversion time is defined as the time it takes the ADC to convert the analog input to a digital (binary) number
    - In ADC804 conversion time varies depending on the clocking signals applied to CLK R and CLK IN pins, but it cannot be faster than 110 μs



+5V power supply or a reference voltage when $V_{ref}/2$ input is open (not connected)
Differential analog inputs where $V_{in} = V_{in}(+) - V_{in}(-)$
Vin (-) is connected to ground and Vin (+) is used as the analog input to be converted
+5V
20
V_{CC}
10k POT
6 Vin(+)
7 Vin(-)
8 A GND
9 $V_{ref}/2$
19 CLK R
10k
150 pF
4 CLK in
1 CS
2 RD
10 D GND
D0 18
D1 17
D2 16
D3 15
D4 14
D5 13
D6 12
D7 11
To LEDs
WR 3
INTR 5
normally open START
CS is an active low input used to activate ADC804
"output enable"
a high-to-low RD pulse is used to get the 8-bit converted data out of ADC804
"end of conversion"
When the conversion is finished, it goes low to signal the CPU that the converted data is ready to be picked up
"start conversion"
When WR makes a low-to-high transition, ADC804 starts converting the analog input value of V_{in} to an 8-bit digital number

- CLK IN and CLK R
  - CLK IN is an input pin connected to an external clock source
  - To use the internal clock generator (also called self-clocking), CLK IN and CLK R pins are connected to a capacitor and a resistor, and the clock frequency is determined by

$$f = \frac{1}{1.1RC}$$

    - Typical values are R = 10K ohms and C = 150 pF
    - We get f = 606 kHz and the conversion time is 110 μs

- $V_{ref}/2$
  - It is used for the reference voltage
    - If this pin is open (not connected), the analog input voltage is in the range of 0 to 5 volts (the same as the Vcc pin)
    - If the analog input range needs to be 0 to 4 volts, $V_{ref}/2$ is connected to 2 volts

$V_{ref}/2$ Relation to V_{in} Range

$V_{ref}/2$(v)	V_{in}(V)	Step Size (mV)
Not connected*	0 to 5	5/256=19.53
2.0	0 to 4	4/255=15.62
1.5	0 to 3	3/256=11.71
1.28	0 to 2.56	2.56/256=10
1.0	0 to 2	2/256=7.81
0.5	0 to 1	1/256=3.90

Step size is the smallest change can be discerned by an ADC

- D0-D7
  - The digital data output pins
  - These are tri-state buffered
    - The converted data is accessed only when CS = 0 and RD is forced low
  - To calculate the output voltage, use the following formula

$$D_{out} = \frac{V_{in}}{step\ size}$$

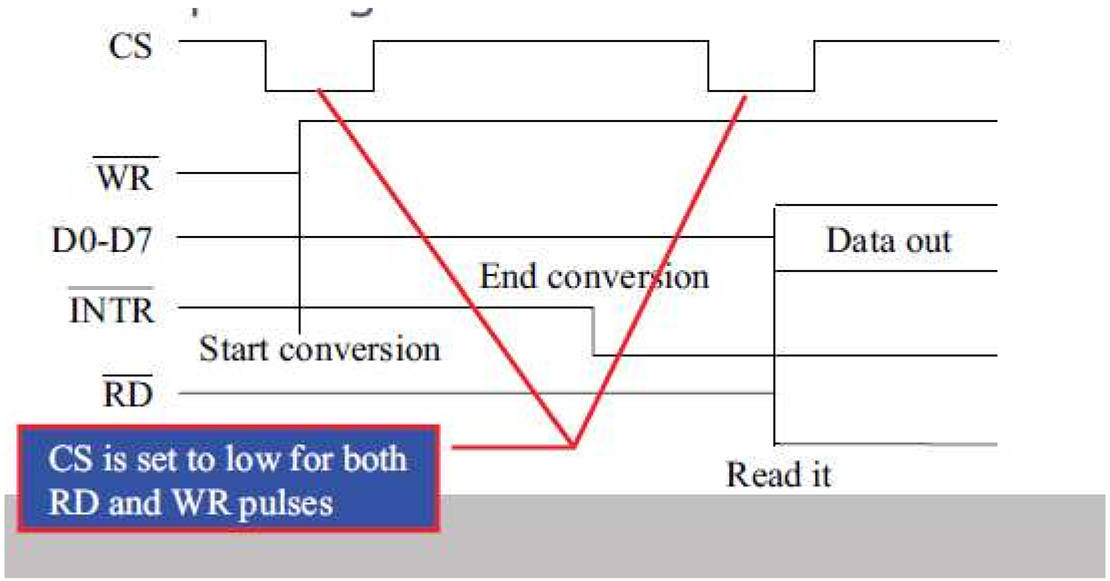
    - *Dout* = digital data output (in decimal),
    - *Vin* = analog voltage, and
    - *step size* (resolution) is the smallest change

ADC804 Chip

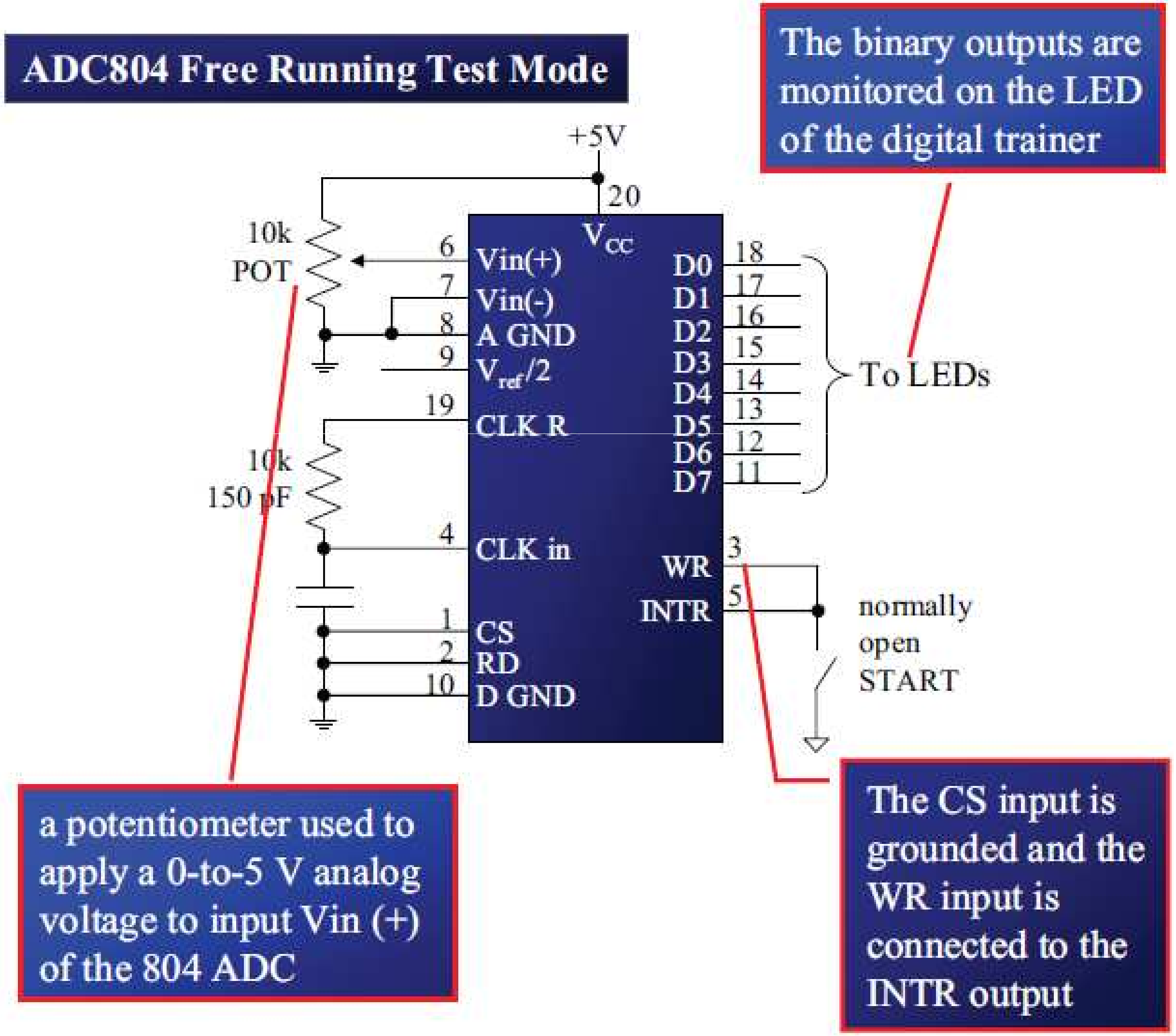
- Analog ground and digital ground
  - Analog ground is connected to the ground of the analog Vin
  - Digital ground is connected to the ground of the Vcc pin
- To isolate the analog Vin signal from transient voltages caused by digital switching of the output D0 – D7
  - This contributes to the accuracy of the digital data output

ADC804 Chip

- The following steps must be followed for data conversion by the ADC804 chip
  - Make CS = 0 and send a low-to-high pulse to pin WR to start conversion
  - Keep monitoring the INTR pin
    - If INTR is low, the conversion is finished
    - If the INTR is high, keep polling until it goes low
  - After the INTR has become low, we make CS = 0 and send a high-to-low pulse to the RD pin to get the data out of the ADC804



Testing ADC804

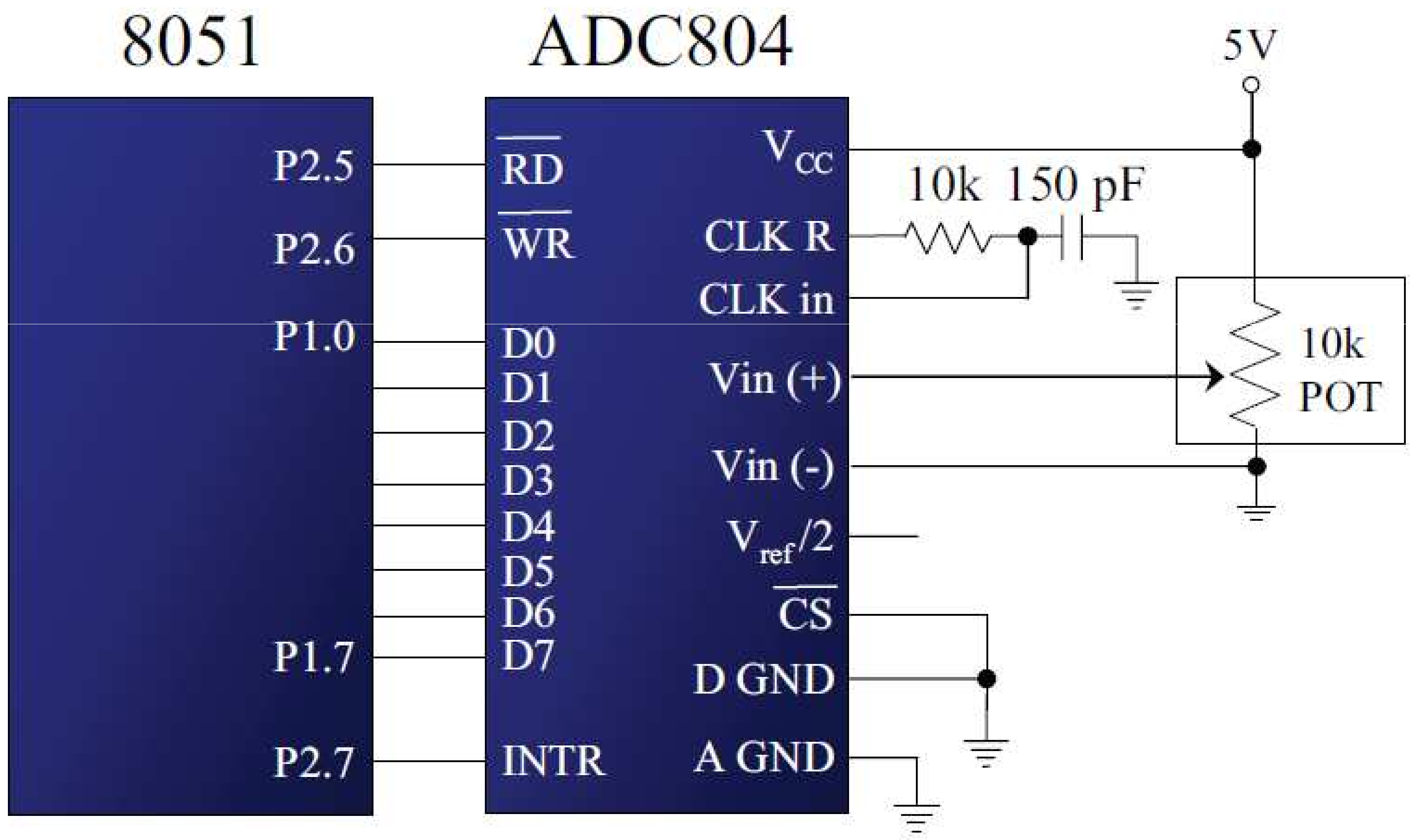


Testing ADC804

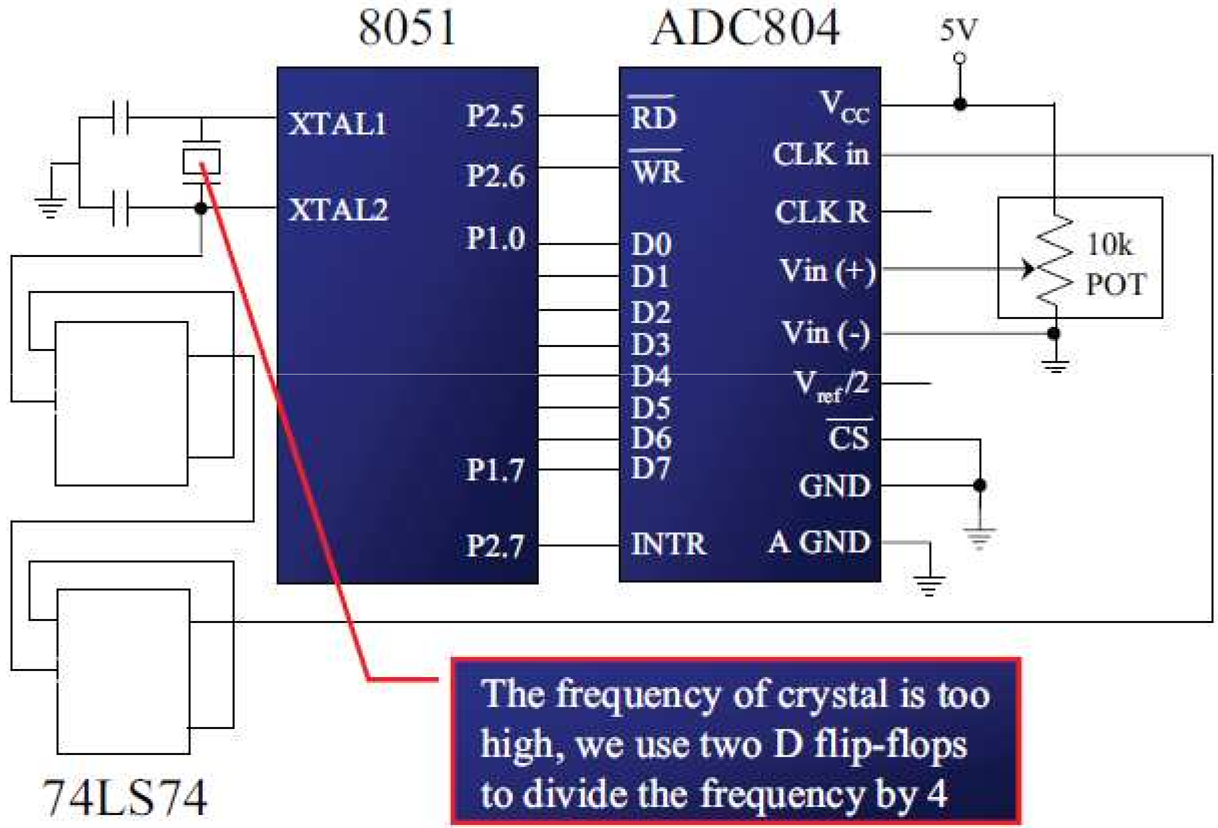
Examine the ADC804 connection to the 8051 in Figure 12-7. Write a program to monitor the INTR pin and bring an analog input into register A. Then call a hex-to ACSII conversion and data display subroutines. Do this continuously.

```
;p2.6=WR (start conversion needs to L-to-H pulse)
;p2.7 When low, end-of-conversion)
;p2.5=RD (a H-to-L will read the data from ADC chip)
;p1.0 - P1.7= D0 - D7 of the ADC804
;
      MOV   P1,#0FFH      ;make P1 = input
BACK: CLR   P2.6          ;WR = 0
      SETB  P2.6          ;WR = 1 L-to-H to start conversion
HERE: JB    P2.7,HERE     ;wait for end of conversion
      CLR   P2.5          ;conversion finished, enable RD
      MOV   A,P1          ;read the data
      ACALL CONVERSION    ;hex-to-ASCII conversion
      ACALL DATA_DISPLAY;display the data
      SETB  p2.5          ;make RD=1 for next round
      SJMP  BACK
```

8051 Connection to ADC804 with Self-Clocking



8051 Connection to ADC804 with Clock from XTAL2 of 8051



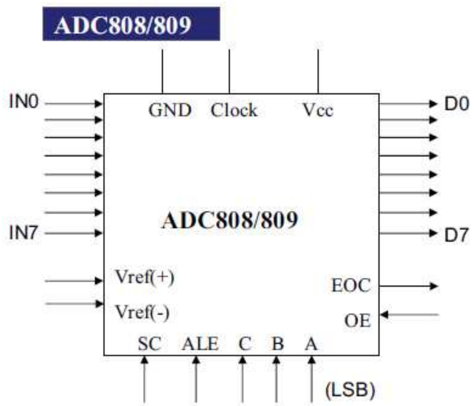
The frequency of crystal is too high, we use two D flip-flops to divide the frequency by 4

ADC808/809

- ADC808 has 8 analog inputs
  - It allows us to monitor up to 8 different transducers using only a single chip
  - The chip has 8-bit data output just like the ADC804
  - The 8 analog input channels are multiplexed and selected according to table below using three address pins, A, B, and C

ADC808 Analog Channel Selection

Selected Analog Channel	C	B	A
IN0	0	0	0
IN1	0	0	1
IN2	0	1	0
IN3	0	1	1
IN4	1	0	0
IN5	1	0	1
IN6	1	1	0
IN7	1	1	1



Steps to Program ADC808/809

- Select an analog channel by providing bits to A, B, and C addresses
- Activate the ALE pin
  - It needs an L-to-H pulse to latch in the address
- Activate SC (start conversion) by an H-to-L pulse to initiate conversion
- Monitor EOC (end of conversion) to see whether conversion is finished
- Activate OE (output enable) to read data out of the ADC chip
  - An H-to-L pulse to the OE pin will bring digital data out of the chip