Memorial University of Newfoundland
Faculty of Engineering and Applied Science

Engineering 4862 Microprocessors

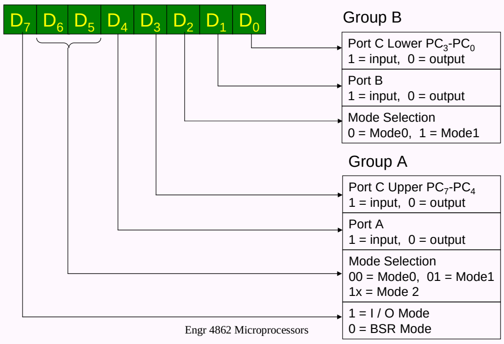
Lecture 22

Cheng Li
EN-4012
licheng@engr.mun.ca

Interface DAC to a PC

- DAC (Digital-to-Analog Converter)
  - Device used to convert digital pulses to analog signals
  - Two methods of making the DAC
    - Binary weighted
    - R / 2R ladder
    - The vast majority of IC use R / 2R since it can achieve a much high degree of precision

Control Word of 8255



Criterion for Judging a DAC: Resolution

- Resolution is a function of the number of binary inputs. ➔ common ones are 8, 10, 12 pins
- The number of analog output levels is equal to 2^n, where n is the number of data inputs

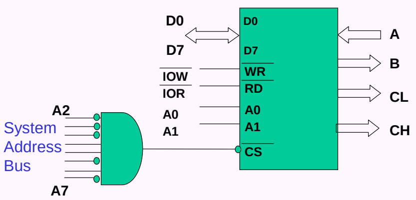
➔ 8-input DAC (MC1408)

gives 256 discrete voltage/current levels of output

➔ 12-input DAC → 4096 voltage/current levels

➔ 16-input DAC → 65,536 voltage/current levels

8255 Design Example



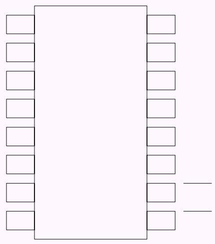
MC1480 DAC (or DAC 808)

- In MC1480, the digital inputs are converted to current (I_{out}) and by connecting a resister to the I_{out} pin, we convert the result to voltage.
- The current provided by I_{out} is a function of binary numbers at D0-D7 and the reference current.
- I_{ref} is generally set to 2.0 mA.
- $I_{out} = I_{ref} \cdot (D_7/2 + D_6/4 + D_5/8 + D_4/16 + D_3/32 + D_2/64 + D_1/128 + D_0/256)$.

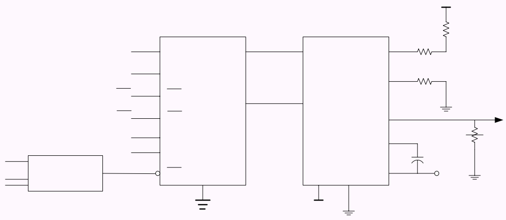
Interface DAC to PC

- Example1
  - Interface MC1480 to Microprocessor through PPI 8255
- Example2
  - Interface AD558 directly to Microprocessor

Interface AD558 to 8088
8-bit DAC Voltage Output



Interface MC1480 to Microprocessor through PPI 8255

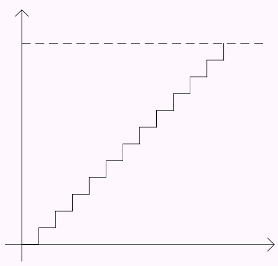


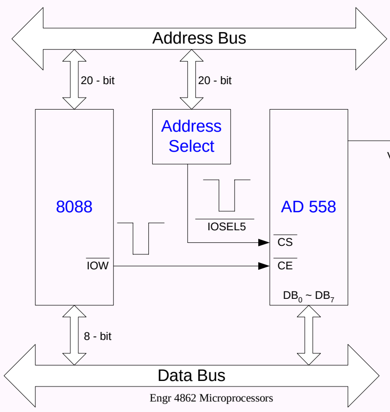
Interface ADC and Sensors to a PC

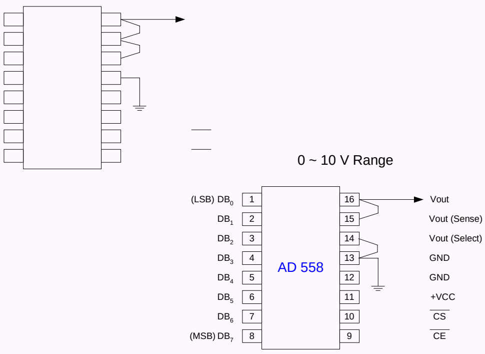
- AD558 is configured as "write only"
- VCC range +4.5V ~ + 16.5 V, normally +5V
- Vout Range: 0 ~ 2.56 V, or 0 ~ 10 V
- Digital Input Code / Output Voltage

Binary	Hex	Decimal	2.56V	10V
00000000	00	0	0	0
00000001	01	1	0.010V	0.039V
00001111	0F	15	0.150V	0.586V
11111111	FF	255	2.55V	9.961V


```
      MOV AL, 80H
      OUT PCtrl, AL
      MOV AL, 0
Cont: OUT PA, AL
      INC AL
      CMP AL, 0
      JZ Stop
      MOV CX, 0FFFFH
Here: LOOP Here
      JMP Cont
Stop: INT 6
```







Data Conversion Procedure

- Steps for data conversion by the AD804 chip
  - Make CS = 0 and send a low-to-high pulse to the WR pin to start the conversion
  - Keep monitoring the INTR pin. If INTR is low, the conversion is finished and we can go to the next step. If the INTR is high, keep polling until it goes low
  - After INTR become low, we make CS = 0 and send a high-to-low pulse to the RD pin to get the data out of the AD804 chip
  - Example1: AD804, Example2: AD7574

Interface ADC and Sensors to a PC

- ADC (Analog-to-Digital Converter)
  - Most widely used device for data acquisition
  - ADC converts the analog input to its binary equivalent and holds it in an internal register
  - Transducers: device that convert physical quantity to electrical signals, also called sensors
  - For ADC, in additional to resolution, conversion time (the time it takes the ADC to convert the analog input to a digital number), is another major factor in judging an ADC

0 ~
1
2
nse)
3
ect)
DB_3 4 AD 558 13 GND
DB_4 5 12 GND
6
7
8

Pin Description

- RD: active low input signal
  - RD is used to get the converted data out of the chip
  - When CS = 0, if a high-to-low pulse is applied to the RD pin, the 8-bit digital output shows up at the D0 – D7 data pins
  - Thus, RD is also referred to as output enable
- WR: active low input signal ➔ Start Conversion
  - If CS = 0, when WR makes a low-to-high transition, the ADC 804 starts converting the analog input value to an 8-bit digit number

8253 / 8254 Timer

- A.k.a. PIT (programmable Interval Timer), used to bring down the frequency to the desired level
- Three counters inside 8253/8254. Each works independently and is programmed separately to divide the input frequency by a number from 1 to 65536
- There are 4 port address needed for a single 8253/8254, given by A0, A1, and CS

CS	A1	A0	Select
0	0	0	Counter 0
0	0	1	Counter 1
0	1	0	Counter 2
0	1	1	Control Reg.

8253 / 8254 Timer

- Each of the three counter has 3 pins associated
  - CLK: input clock frequency
    - A square wave of 33% duty cycle
    - 8253: 0 ~ 2 MHz,   8254: 0 ~ 8 MHz
  - OUT: can be square wave, or one shot
  - GATE: Enable (high) or disable (low) the counter
- Data Pins: (D0 ~ D7)
  - Allow the CPU to access various registers inside the 8253/54 for both read and write operations. $\overline{RD}$ and $\overline{WR}$ are connected to $\overline{IOR}$ and $\overline{IOW}$ of control bus.

8253 / 8254 Timer

- Task1: program counter 0 for binary counter for mode 3 to divide CLK0 by number 4282 (BCD)

```
MOV AL, 0011 0111B
OUT 97H, AL
MOV AX, 4282H (BCD needs H)
OUT 94H, AL (Low Byte)
MOV AL, AH
OUT 94H, AL (High Byte)
```

- OUT0 = CLK0 / 4282

8253 / 8254 Timer

- Each of the three counters must be programmed separately
- Control byte must be first written into the control register. The 8253/54 must be initialized before use
- The programmer can not only write the value of the divisor into the 8253/54, but read the content of the counter at any given time as well
- All counters are down counters.

Shape of the 8253/54 Output

- Given CLK = 1.193 MHz, the clock period of input frequency is 838 ns
- If the number N loaded into the counter is even, both high and low pulse are the same length, which is N/2 * 838 ns
- If the number N loaded into the counter is odd, the high pulse is (N+1)/2 * 838 ns and the low pulse is (N–1)/2 * 838 ns
- ➔ If N is odd, the high portion of the output square wave is slightly wider than the low portion

8253 / 8254 Timer

- To program a given counter to divide the CLK input frequency, one must send the divisor to that specific counter's register.
- Although all three counters share the same control register, the divisor registers are separate for each counter
- Example: given the port addresses for 8253/54:
  Counter 0: 94H   Counter 1: 95H
  Counter 2: 96H   Control Reg: 97H

8253/54 Operation Modes

- Mode 0: Interrupt on terminal count
  - The output is initially low, and remain low for the duration of the count if GATE=1. When the terminal count is reached, the output will go high and remain high until a new control word or new count number is loaded
    - Width of low pulse = N * T, where T is clock period
  - Example: GATE=1 and CLK = 1 MHz
    Clock count N = 1000

8253/54 Operation Modes

- Mode 0: Interrupt on terminal count
  - If GATE becomes low at the middle of the count, the count will stop and the output will be low. The count resumes when the GATE becomes high again ➔ This in effect adds to the total time the output is low.
- Mode 1: HW triggered / programmable one shot
  - The triggering must be done through the GATE input by sending a 0-to-1 pulse to it.
  - Steps: 1) Load the count register
    2) A 0-to-1 pulse must be sent to the GATE input to trigger the count

8253/54 Operation Modes

- Mode 1: HW triggered / programmable one shot
  - In Mode 1, after sending the 0-to-1 pulse to GATE, OUT becomes low and stays low for a duration of N*T, then becomes high and stays high until the GATE is triggered again
  - If during the activation, a retriggered happened, then restart the down counting
- Mode 2: Rate Generator (Divide-by-N counter)
  - In Mode2, if GATE=1, OUT will be high for N*T, goes low only for one clock pulse, then counter is reloaded automatically, and the process continues indefinitely. ➔ Whole period: (N+1) * T

8253/54 Operation Modes

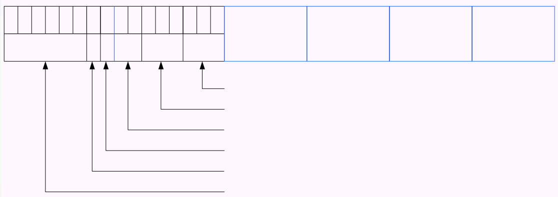
- Mode 3: Square wave rate generator
  - Most commonly used
- Mode 4: Software triggered strobe
  - Similar to Mode2, except that the counter is not reloaded automatically
  - In Mode4, if GATE=1, the output will go high when loading the count, it will stay high for duration N*T. After the count reaches zero, it becomes low for one clock pulse, then goes high again and stays high until a new command word or new count is loaded
  - To repeat the strobe, the count must be reloaded

8253/54 Operation Modes

- Mode 5: Hardware triggered strobe
  - Similar to Mode4, except that the triggering must be done with the GATE input
  - The count starts only when a 0-to-1 pulse is sent to the GATE input
  - If GATE retriggered during the counting, it will restart the down counting

Instruction and Machine Code

- Typical 8086/8088 Machine Instruction Format



Encoding (Instruction → Machine Code)

- E.g1. MOV CL, [BX + 39A2H]
  - Book P. 3-121

  (1)Memory/Register Operand to/from Register Operand

  (2) 1 0 0 0 1 0 d w mod reg r/m

  d = 1: SRC = EA, DEST = REG

  d = 0: SRC = REG, DEST = EA ➔ d = 1

  (3)Byte operation: ➔ w = 0
  - Book P. 6-55, Table 6-20

  (4) MOD = 10: memory mode, 16-bit disp follows

Encoding (Instruction → Machine Code)

- E.g1 (cont'd) MOV CL, [BX + 39A2H]
  - (5) Format: __ __ A2 39
  - (6) w = 1, CL ➔ REG = 001
  - (7) MOD = 10, (BX + Disp) ➔ R/M = 111

  Therefore, the final code is

  10001010 10001111 A2 39 ➔ 8A 8F A2 39

Decoding (Machine Code → Instruction)

- E.g3. 88 95 00 02
  - Book P. 6-61, Table 6-23

  (1) 88 → MOV Reg8 / Mem8, Reg8
  10001000 Mod Reg R/M Disp_Lo Disp_Hi
  → d = 0: SRC = Reg, DEST = EA (P. 3-121)
  → w = 0: byte operation
  → Displacement: 0200 H
  → Mod Reg R/M = 1 0 0 1 0 1 0 1
  - Book P. 6-55

  (2) Reg: 010 → DL, R/M: 101 → (DI) + D16
  ➔ Therefore, MOV BYTE PTR [DI + 0200H], DL

Encoding (Instruction → Machine Code)

- E.g2. **ADD WORD PTR ES: [BX + SI + 1053H], AX**
  - (1)Override operation: we need to put a overrode prefix before the machine code
  - Book P. 6-61
  - (2)0 0 1 reg 1 1 0 SEGMENT = override prefix
  - Book P. 6-56
  - (3) ES: 00 CS: 01 SS: 10 DS:11
  - Book P. 6-61
  - (4) Therefore the prefix is: 0 0 1 0 0 1 1 0 (26H)

Decoding (Machine Code → Instruction)

- E.g4. 36 81 8C 8E 00 F4 00
  - Book P. 6-61, Table 6-23

  (1) 36 → Segment override prefix: SS
  (2) 81 → many choices: ADD, OR, ADC, SBB, ……
  xxx Reg16/Mem16, Immed16
  (3) 8C → help to explain:
  Mod Reg R/M: 1 0 0 0 1 1 0 0
  → Reg: 001 → OR Reg16/Mem16, Immed16
  10000001 mod 001 r/m Disp-Lo Disp-Hi Data-Lo Data-Hi
  10 110 Disp (008E) Data (00F4H)
  P. 6-55: mod: 10, r/m: 110 → (BP) + D16
  ➔ Therefore, OR WORD PTR [BP + 008EH], 00F4H

Encoding (Instruction → Machine Code)

- E.g2. **ADD WORD PTR ES: [BX + SI + 1053H], AX**
  - Book P. 3-64
  - (5) Memory/register Operand with Register Operand
    0 0 0 0 0 0 d w mod reg r/m
  - d = 1: LSRC=REG, RSRC=EA, DEST=REG
  - d = 0: LSRC=EA, RSRC=REG, DEST=EA ➔ d = 0
  - (6) Word operation ➔ w = 1
  - Book P. 6-55, Table 6-18
  - (7) 16-bit Disp ➔ mod = 10
  - (8) Reg: AX ➔ Reg = 000, R/M = 000
  - Therefore, 00100110 00000001 10000000 53 10

Decoding (Machine Code → Instruction)

- Practice Question:

  C7 C7 A9 12 3B 47 F4
  MOV DI, 12A9H
  CMP AX, [BX - 12]

  B8 00 02 8E D8 B9 08 00 E2 FE
  MOV AX, 0200H
  MOV DS, AX
  MOV CX, 0008
  here: LOOP here

About Test2

- Friday 9:00 – 9:50, EN – 1040
- Bring the Intel 8086/88 User's Manual
- Format:
  - 1. Analysis
  - 2. Design
  - 3. Address encoding and decoding
  - 4. Timing

Memory and Memory Interfacing

- Speed
  - The speed of a memory chip is commonly referred to as its access time, varied from a few *ns* to hundreds of *us*.
- Characteristics
  - Capacity, Organization, Speed
- Examples
  - 256 K memory chip with 8 data pins
    - Organization: 32K * 8 / Address: 15pins
  - A memory chip has 13 address lines and 4 data lines
    - Organization: 2^{13} * 4 = 8K * 4 = 32Kbits

Memory and Memory Interfacing

- Memory Fundamentals
  - In all computer designs, semiconductor memories are used as primary storage for code and data
  - Requirement of primary memory ➔ Fast in responding to CPU
  - Types: RAM and ROM
  - Memory capacity
    - The capacity of a memory IC chip is always given in bits
      - Chip capacity: the number of bits that a chip can store: Kbits, Mbits
    - The capacity of a computer is given in bytes
    - Example: A 4M chip – 4M bits, A 4M computer – 4M bytes

Memory and Memory Interfacing

- Summary
  - Each memory chip contains 2^x locations, *x* is the number of address pins on the chip
  - Each location contains *y* bits, *y* is the number of data pins
  - The entire chip will contain $2^x * y$ bits
- Memory Types
  - ROM (non-volatile)
    - PROM (Programmable ROM), OTP, need burner or programmer
    - EPROM (Erasable Programmable ROM), UV-radiation to erase
    - EEPROM (Electrically erasable programmable ROM)
      - » Advantage: 1. Much quicker, 2. One can select byte to be erased, 3. One can program/erase while still on board

Memory and Memory Interfacing

- Memory organization
  - Memory chips are organized into a number of locations within the IC
  - The number of bits that each location can hold is always equal to the number of data pins on the chip
  - How many locations exist inside a memory map?
    - ➔ That depends on the number of address pins
    - ➔ Given *x* the number of address pins → 2^x locations
  - The total number of bits that a memory chip can store is equal to the number of locations times the number of data bits per location

Memory and Memory Interfacing

- Memory Types
  - ROM (non-volatile)
    - Flash Memory EPROM
      - » Since 1990s
      - » Advantage: the process of erasure of the entire content takes less than a second, erasure method is electrical
      - » Widely used as a way to upgrade the BIOS ROM of the PC
    - Mask ROM
      - » The kind of ROM whose contents are programmed by the IC manufacturer ➔ Low cost
  - RAM (volatile)
    - Three types: SRAM, DRAM, NV-RAM

Memory and Memory Interfacing

– Memory Types
  - RAM (volatile)
    – SRAM (Static RAM)
      » Storage cells in SRAM are made of flip-flops and therefore do not require refreshing in order to keep their data
      » The problem is that each cell requires at least 6 transistors to build and the cell holds only one bit data
      » The capacity of SRAM is far below DRAM
      » SRAM is widely used for cache memory
    – DRAM (Dynamic RAM)
      » The use of a capacitor as a means to store data
      » Cuts down the number of transistors needed to build cell
      » However, it requires constant refreshing due to leakage

Memory Chip

– Ex1: Find the organization and chip capacity for ROM
  - 14 Address pins, 8 data pins
  - 12 Address pins, 8 data pins

– Ex2: Find the organization and chip capacity for RAM
  - 17 address pins, 8 data pins, SRAM
  - 9 address pins, 4 data pins, DRAM

– Ex3: Find the capacity and # of address/data pins for the following memory chip
  - 256K × 4 SRAM
  - 32K × 8 EPROM
  - 1M × 1 DRAM

Memory and Memory Interfacing

– Memory Types
  - RAM (volatile)
    – DRAM (Dynamic RAM)
      » Advantage:
      » 1. High density (capacity)
      » 2. Cheaper cost per bit
      » 3. Lower power consumption per bit
      » Disadvantage:
      » 1. Must be refreshed periodically
      » 2. While it is being refreshed, the data can not be accessed

ALE Timing in 8088 Based System

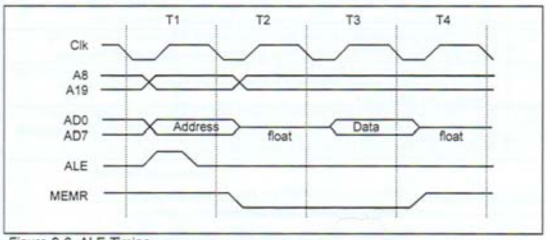


Figure 9-6. ALE Timing

Memory and Memory Interfacing

– Packaging in DRAM
  - To reduce the number of pins needed for address, multiplex / demultiplexing is used
  - Method is to split the address into half and send in each half of the address through the same pins ➔ requires fewer pins
  - Internally, DRAM is divided into a square of rows and columns, the first half of the address is called the row and the second half is called the column

– Organization of DRAM
  - Most DRAM are x 1 and x 4

– NV-RAM (Non-volatile RAM)

Memory Read Bus Timing in 8088