IO and Peripheral

General Concepts of IO and Peripheral
Digital to Analog Conversion
Analog to Digital Conversion
Handshake between devices

1

1

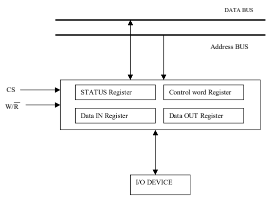
IO and Peripheral

General Concepts of IO and Peripheral
Digital to Analog Conversion
Analog to Digital Conversion
Handshake between devices

2

2

Basic Concept



- There is an interface circuit between the buses and IO/Peripheral
- The CPU read/write data to the device through the Data IN/OUT registers.
- The status register usually is a read only register that reflects the status of the IO device.
- The CPU write control words to the control word register to control the operation modes of the IO device.
- Each register occupies an address.

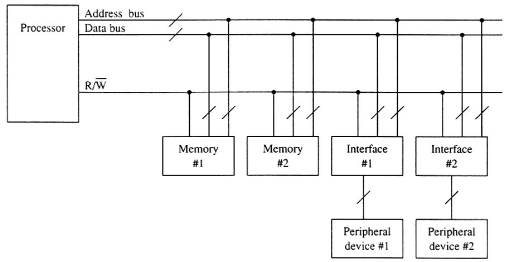
For the PIC18 case, IO PORTs are most simple examples.
TRISA, TRISB, ..are used to control the ports for input or output
Accessing PORTA, PORTB can read/output the data to the Peripheral.

3

3

Memory Mapped I/O

IO devices are addressed and selected by decoders as if they were memory devices.



4

4

Memory Mapped I/O

Advantages :
- Addressed and selected like a memory device. i.e. all the memory reference operations and addressing modes can be used for I/O devices.
- e.g.

```
MOVWF TRISC
MOVF  PORTB, W
CLRF  TRISB
```

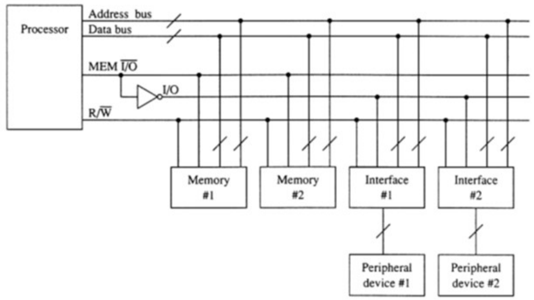
Disadvantages :
- Some system memory addresses are used up for IO and are therefore not available for memory, thus the total space for memory is reduced.

5

5

Direct I/O

Using a separate address space with separate instructions and spearate control signals for I/Os



6

6

Direct I/O

Advantages:
- Not using the system memory address space.

Disadvantages:
- Only special IN and OUT instructions can be used for I/O operations.

e.g. In 8086

IN AL, iox ; ***iox address***, range 00-FFh

OUT iox, AL ; 8-bit io address

Note: some uPs, such as 8051, support only memory mapped I/Os and some support both. The Intel 8086 supports both.

7

7

IO and Peripheral

General Concepts of IO and Peripheral

Digital to Analog Conversion

Analog to Digital Conversion

Handshake between devices

8

8

Data Conversion

- Analog signals are continuous, with infinite values in a given range.
  - Examples: A clock face with hands, a voltmeter with a needle, and audio signals.
- Digital signals have discrete values such as on/off or 0/1.
  - Examples: A digital clock or a digital voltmeter.

9

9

Data Conversion

- Limitations of analog signals
  - Analog signals pick up noise as they are being amplified.
  - Analog signals are difficult to store.
- Advantages of digital systems (signals)
  - Noise can be reduced by converting analog signals in 0s and 1s.
  - Binary signals of 0s/1s can be easily stored in memory.
- The major limitation of a digital system is how accurately it represents the analog signals after conversion.

10

10

Embedded System

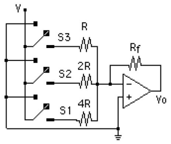
- A typical system that converts signals from analog to digital and back to analog includes:
  - A transducer that converts non-electrical signals into electrical signals
  - An A/D converter that converts analog signals into digital signals
  - A digital processor that processes digital data
  - A D/A converter that converts digital signals into equivalent analog signals
  - A transducer that converts electrical signals into non-electrical signals

11

11

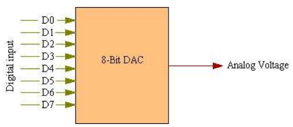
Digital to Analog

- Use current source with digital switches



$$V0 = -Rf\left(S3\frac{V}{R} + S2\frac{V}{2R} + S1\frac{V}{4R}\right)$$
$$= \frac{-Rf}{4R}V\left(4S3 + 2S2 + S1\right)$$

	S3	S2	S1
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1



12

12

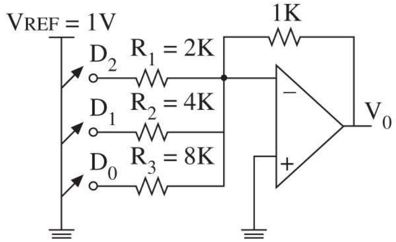
Digital to Analog Conversion

- D/A, DAC, or D-to-A
  - Converting discrete signals into discrete analog values that represent the magnitude of the input signal compared to a standard or reference voltage
  - The output of the DAC is discrete analog steps.
  - By increasing the resolution (number of bits), the step size is reduced, and the output approximates a continuous analog signal.

13

13

D/A Converter Circuits



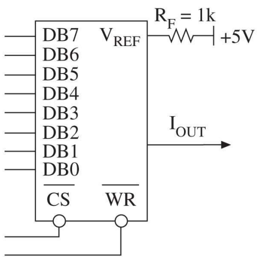
$$I_o = I_T = I_1 + I_2 + I_3 = \frac{V_{REF}}{R_1} + \frac{V_{REF}}{R_2} + \frac{V_{REF}}{R_3} = \frac{V_{REF}}{1k}\left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8}\right) = 0.875\text{mA}$$

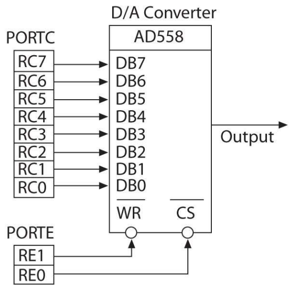
$$V_O = -R_f I_T = -(1k)\times(0.875\text{ mA}) = -0.875\text{ V} = \left|\tfrac{7}{8}\text{V}\right|$$

14

14

IC D/A Converters





$$I_O = \frac{V_{REF}}{R_{REF}}\left(\frac{DB7}{2} + \frac{DB6}{4} + \frac{DB5}{8} + \ldots\ldots + \frac{DB0}{2^8}\right)$$

$$I_O = 5\text{ mA}\left(\frac{1}{2} + \frac{0}{4} + \frac{0}{8} + \frac{1}{16} + \frac{0}{32} + \frac{0}{64} + \frac{0}{128} + \frac{1}{256}\right) = 2.832\text{ mA}$$

15

IO and Peripheral

General Concepts of IO and Peripheral

Digital to Analog Conversion

Analog to Digital Conversion

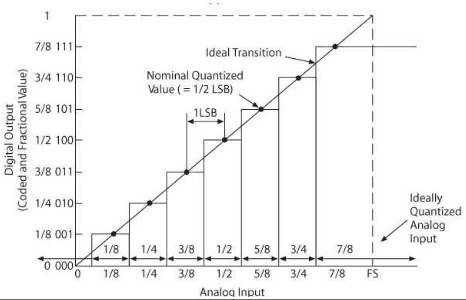
Handshake between devices

16

16

Analog-to-Digital Conversion

- A/D, ADC, or A-to-D
  - Process of converting a continuous varying signal, such as voltage or current, into discrete digital quantities that represent the magnitude of the signal compared to standard or reference voltage



17

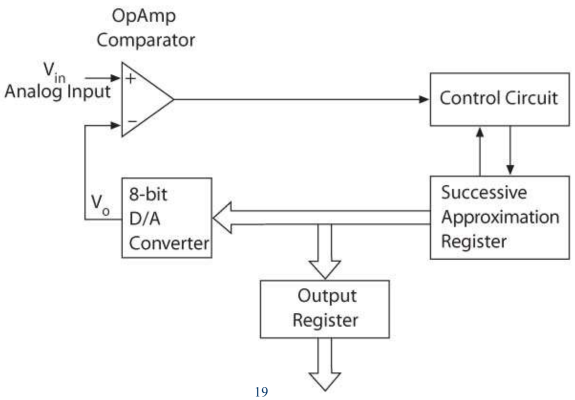
A/D Conversion

- Flash
  - Uses multiple comparators in parallel
  - High-speed, high cost converter
- Integrator
  - Charges a capacitor for a given amount of time using the analog signal
  - Slow, but high accuracy and low noise
- Successive Approximation
  - Effective compromise among resolution, speed, and cost
- Counter
  - Similar to successive approximation circuit
  - Slower, with variable conversion times

18

18

Successive Approximation



19

19

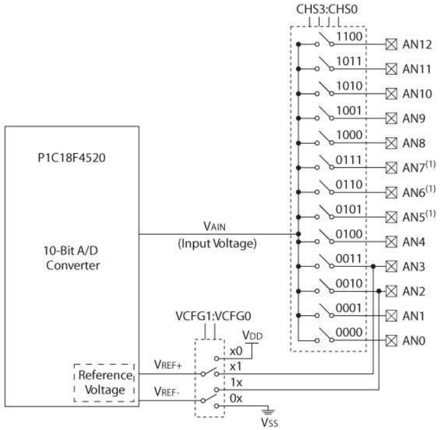
PIC18F A/D Converter Module

- The PIC18F4520 microcontroller includes:
  - 10-bit A/D converter
  - 13 channels AN0 - AN12
  - Three control and status registers: **ADCON0, ADCON1, and ADCON2**
  - Data register: **ADRESH, ADRESL**

20

20

PIC18F4520 A/D Converter



21

PIC18F4520 A/D Converter

A/D converter requires a low reference voltage (VREF-) and a high reference voltage (V_{REF+}) to perform conversion. Most A/D converters are ratiomertic:

1.An analog input of V_{REF-} is converted to digital code 0.
2.An analog input of V_{REF+} is converted to digital code 2n – 1.
3.An analog input of V_{in} is converted to digital code

$$k=(2^n - 1) \times (V_{in} - V_{REF-}) \div (V_{REF+} - V_{REF-})$$

Given k, the measured voltage is given by

$$V_{in}=V_{REF-} + (V_{REF+} - V_{REF-}) \times k \div (2^n - 1)$$

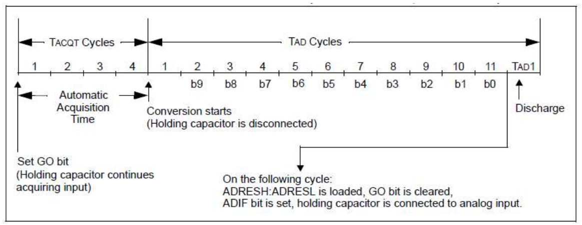
Most systems use VDD and 0V as V_{REF+} and V_{REF-}, respectively.

The output of a transducer should be scaled and shifted to the range of 0V ~ VDD in order to achieve the best accuracy

22

22

PIC18F4520 A/D Converter



T_{AD} is the conversion time per bit
T_{ACT} is the setup time for ADC.
Both timing parameters can be programed.

23

23

PIC18F4520 A/D Converter

Param No.	Symbol	Characteristic		Min	Max	Units	Conditions
130	TAD	A/D Clock Period	PIC18**F**XXXX	0.7	25.0(1)	µs	TOSC based, VREF ≥ 3.0V
			PIC18**LF**XXXX	1.4	25.0(1)	µs	VDD = 2.0V; TOSC based, VREF full range
			PIC18**F**XXXX	TBD	1	µs	A/D RC mode
			PIC18**LF**XXXX	TBD	3	µs	VDD = 2.0V; A/D RC mode
131	TCNV	Conversion Time (not including acquisition time) **(Note 2)**		11	12	TAD	
132	TACQ	Acquisition Time **(Note 3)**		1.4 TBD	— —	µs µs	-40°C to +85°C 0°C ≤ to ≤ +85°C
135	TSWC	Switching Time from Convert → Sample		—	**(Note 4)**		
TBD	TDIS	Discharge Time		0.2	—	µs	

T_{AD} is the conversion time per bit
T_{ACQ} is the setup time for ADC.
Both timing parameters can be programed.

24

24

ADCON0

- ADCON0 Register
  - Select a channel for input analog signal
  - Start a conversion
  - Indicate the end of the conversion

B7	B6	B5	B4	B3	B2	B1	B0
----	---	CHS3	CHS2	CHS1	CHS0	GO/ DONE	ADON

Channel Select

0000 = Channel 0 (AN0)
0001 = Channel 1 (AN1)

1100 = Channel 12 (AN12)

1 = Set it to Start Conversion
0 = It is Reset at the End of Conversion

1 = A/D Powered Up
0 = A/D Shut-off

25

25

ADCON1

- ADCON1 Register
  - Set up the I/O pins either for analog signal or for digital signals
  - Select V_{REF} voltages

B7	B6	B5	B4	B3	B2	B1	B0
----	---	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0

V_{REF} Voltage Selection

I/O Port Configuration

$1 = V_{REF-}$ (AN2)
$0 = V_{SS}$

$1 = V_{REF+}$ (AN3)
$0 = V_{DD}$

See Table 12.1

26

26

ADCON1

	7	6	5	4	3	2	1	0
	--	--	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0
value after reset	0	0	0	0	0	0	0	0

VCFG1:VCFG0: Voltage reference configuration bits (see Table 12.2)
PCFG3:PCFG0: A/D port configuration control bits

	AN15	AN14	AN13	AN12	AN11	AN10	AN9	AN8	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0
0000	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
0001	D	D	A	A	A	A	A	A	A	A	A	A	A	A	A	A
0010	D	D	D	A	A	A	A	A	A	A	A	A	A	A	A	A
0011	D	D	D	D	A	A	A	A	A	A	A	A	A	A	A	A
0100	D	D	D	D	D	A	A	A	A	A	A	A	A	A	A	A
0101	D	D	D	D	D	D	A	A	A	A	A	A	A	A	A	A
0110	D	D	D	D	D	D	D	A	A	A	A	A	A	A	A	A
0111	D	D	D	D	D	D	D	D	A	A	A	A	A	A	A	A
1000	D	D	D	D	D	D	D	D	D	A	A	A	A	A	A	A
1001	D	D	D	D	D	D	D	D	D	D	A	A	A	A	A	A
1010	D	D	D	D	D	D	D	D	D	D	D	A	A	A	A	A
1011	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A	A
1100	D	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
1101	D	D	D	D	D	D	D	D	D	D	D	D	D	D	A	A
1110	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	A
1111	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D

27

27

ADCON2

- ADCON2 Register
  - Select an acquisition time and clock frequency
  - Right or left justify output reading

B7	B6	B5	B4	B3	B2	B1	B0
ADFM	---	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0

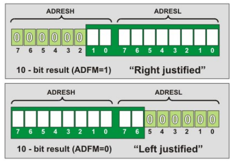
Acquisition Time Select Bits

Clock Select Bits

1 = Conversion Reading Right Justified
0 = Conversion Reading Left Justified

See Table 12-3

See Table 12-2



ACQT2:ACQT0: A
000 = 0 TAD(1)
001 = 2 TAD
010 = 4 TAD
011 = 6 TAD
100 = 8 TAD
101 = 12 TAD
110 = 16 TAD
111 = 20 TAD

ADCS2:ADCS0: A/D conversion clock select bits
000 = FOSC/2
001 = FOSC/8
010 = FOSC/32
011 = FRC (clock derived from A/D RC oscillator)
100 = FOSC/4
101 = FOSC/16
110 = FOSC/64
111 = FRC (clock derived from A/D RC oscillator)

28

28

Selecting ADC conversion time

- The programming value of T_{AD} (ADCS2-ADCS0) must be greater than the minimum value of T_{AD}
- The programming value of T_{ACQ} (ACQT2-ACQT0) must be greater than the minimum of T_{ACQ}

Let
the minimum value of T_{AD} = 1.6 µs
the minimum value of T_{ACQ} = 13 µs
fosc = 32 MHz

What are (ADCS2-ADCS0) and (ACQS2-ADCS0) ?

- f_{OSC} = 32 MHz, the A/D clock source must be set to 64 f_{OSC}, which makes T_{AD} =2µs. (ADCS2-ADCS0)=110.
- For T_{AD} =2µs, T_{ACQ} must be set to at least 8 TAD. (ACQT2-ACQT0)=100

29

29

Selecting ADC conversion time

Assembly instruction sequence that achieve the desired setting:

```
movlw 0x01          ; select channel AN0 and enable A/D
movwf ADCON0,A
movlw 0x0E          ; configure only channel AN0 as analog port,
movwf ADCON1,A      ; select VDD and VSS as reference voltage
movlw 0xA6          ; set A/D result right justified, set acquisition
movwf ADCON2,A      ; time to 8 TAD, clock source FOSC/64
```

30

30

Procedure

Procedure for Performing A/D Conversion

- Configure the A/D module
  1. Configure analog pins, reference voltages
  2. Select A/D input channel
  3. Select A/D acquisition time (if available)
  4. Select A/D conversion clock
  5. Enable A/D module
- Configure A/D interrupt
  1. Clear ADIF flag
  2. Set ADIE bit (if desired)
  3. Set GIE bit (if desired)
- Wait for the desired acquisition time (if required)
- Start conversion by setting the GO/DONE bit
- Wait for A/D conversion to complete
- Read the A/D result registers; clear the ADIF flag

31

31

Example

Example 12.6 Assume that the AN0 pin of a PIC18 running with a 32 MHz crystal oscillator is connected to a potentiometer. The voltage range of the potentiometer is from 0V to 5V. Write a program to measure the voltage applied to the AN0 pin, convert it, and retrieved the conversion result and place it in PRODH:PRODL. Let the minimum value of T_{AD} = 1.6 µs and the minimum value of T_{ACQ} = 13 µs

32

32

Example

```
       org 0x00
       goto start
       org 0x08
       retfie
       org 0x18
       retfie
start  movlw 0x01     ; select channel AN0 and enable A/D
       movwf ADCON0,A ; "
       movlw 0x0E     ; use VDD & VSS as reference voltages &
       movwf ADCON1,A ; configure channel AN0 as analog input
       movlw 0xA6     ; select FOSC/64 as conversion clock,
       movwf ADCON2,A ; 8 TAD for acquisition time,
                      ;right-justified
       bsf ADCON0,GO,A ; start A/D conversion
wait_con
       btfsc ADCON0,DONE,A ; wait until conversion is done
       bra wait_con
       movff ADRESH,PRODH ; save conversion result
       movff ADRESL,PRODL ; "
end
```

33

33

Scaling Circuit

Used to amplify the transducer output from a range of 0V ~ V_Z to 0 ~ V_{DD}.

•- Usually V_Z is smaller than V_{DD}.

•- Voltage gain:

$$A_V = V_{OUT} \div V_{IN} = (R_1 + R_2) \div R_1 = 1 + R_2/R_1$$

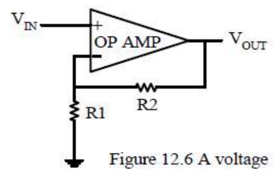


Figure 12.6 A voltage scaler

34

34

Scaling Circuit

Suppose the transducer output voltage ranges from 0V to 200 mV.

Choose the appropriate values for R1 and R2 to scale this range to 0~5V.

Solution:

•**R2/R1** = $(V_{OUT}/V_{IN}) - 1 = 24$

•Choose 240 KΩ for R2 and 10 KΩ for R1.

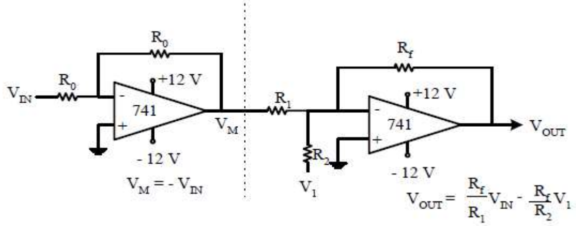
35

35

Scaling Circuit

Voltage Translation Circuit

Needed to shift and scale the transducer output in a range of $-V_X \sim V_Z$ to 0V ~ V_{DD}



36

36

Scaling Circuit

Choose appropriate resistor values and the adjusting voltage so that the circuit shown in the last page can shift the voltage from the range of –1.2V ~ 3.0V to the range of 0V ~ 5V.

Solution:

$0 = -1.2 \times (Rf/R1) - (Rf/R2) \times V1$
$5 = 3.0 \times (Rf/R1) - (Rf/R2) \times V1$

Choose R0 = R1 = 10 KΩ and V_1 = -5V, solve R_2 = 50 KΩ, and R_f = 12KΩ

37

37

IO and Peripheral

General Concepts of IO and Peripheral
Digital to Analog Conversion
Analog to Digital Conversion
Handshake between devices

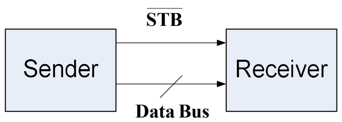
38

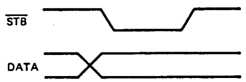
38

Simple Strobe I/O

The sender outputs a strobe signal, STB, to indicate the data is ready.

Works well for low rates data transfer but fails at higher rate due to no acknowledge of the received signal.



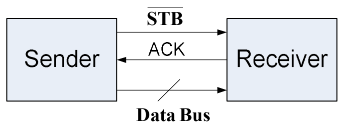


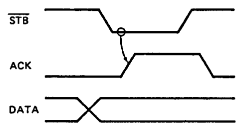
39

39

Single Handshake I/O

The sender outputs the data and sends an STB signal to the receiver. After taking the data, the receiver sends an ACK signal, to acknowledge the receive of the data.



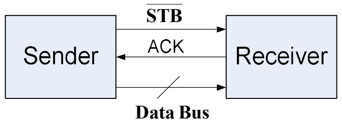


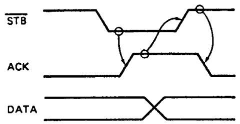
40

40

Double Handshake I/O

(1) The sender asserts its STB line low to ask the receiver if it is ready. (2) The receivers, if ready, will respond by sending the ACK signal. (3) The sender then sends the data and raises its STB line high. (4) After taking the data, the receiver acknowledges it by letting ACK low.





41

41