UNSW SYDNEY

Chapter 6: Analogue to Digital Converter Module (ADC)

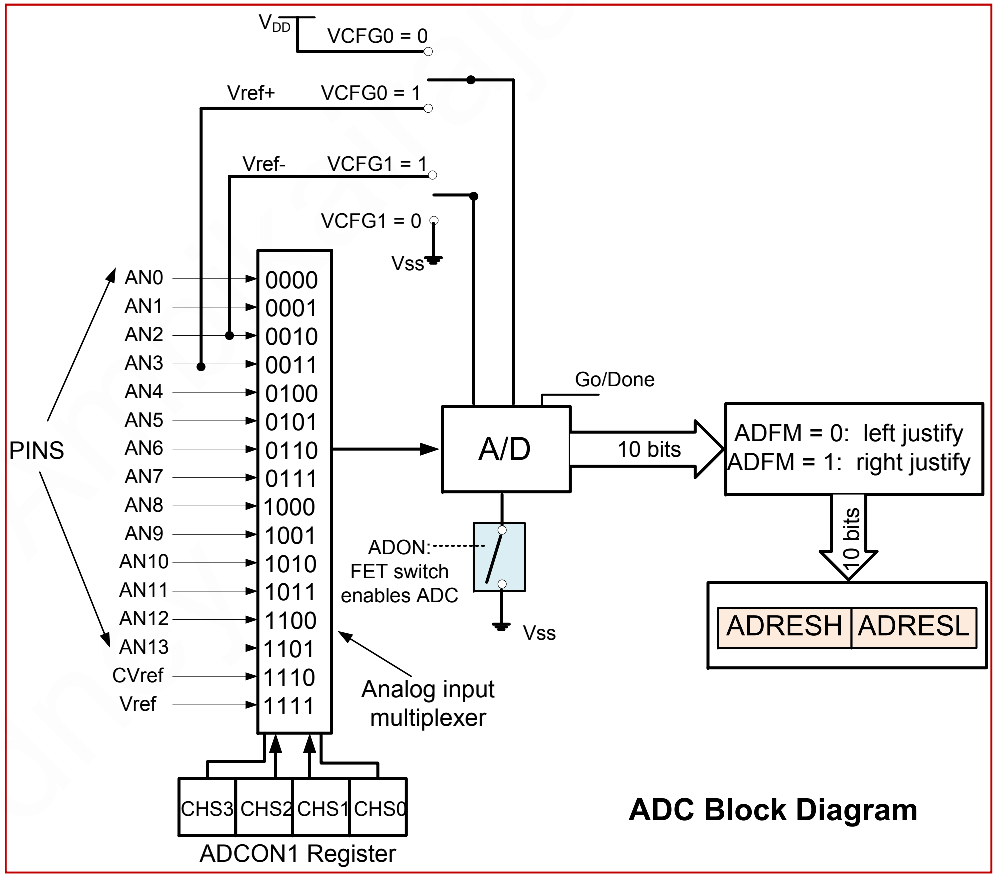
Australia's Global University | Faculty of Engineering | School of Electrical Engineering and Telecommunications

Professor Eliathamby Ambikairajah

Head of School of Electrical Engineering and Telecommunications, UNSW, Sydney

Analogue to digital Converter (ADC) Module

- ✓ The Analogue to Digital Converter allows conversion of an analogue input signal to a 10-bit binary representation of that signal.
- ✓ The conversion result of the ADC is stored in the registers ADRESL (contains low byte conversion result) and ADRESH (contains high byte conversion result).
- ✓ There are 14 separate analogue inputs which are multiplexed in order to select the appropriate input to the ADC
- ✓ The ADC voltage reference is software selectable to either V_{DD} or a voltage applied to the external reference pins
- ✓ The ADC can generate an interrupt upon completion of a conversion or Polling

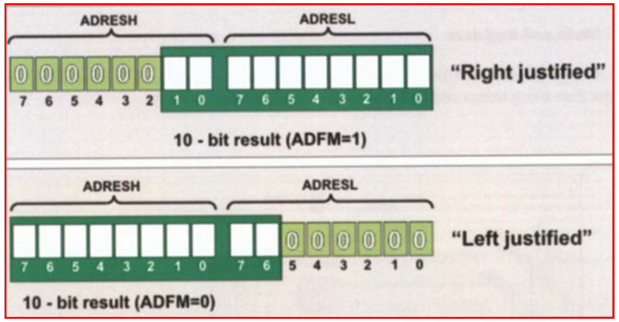


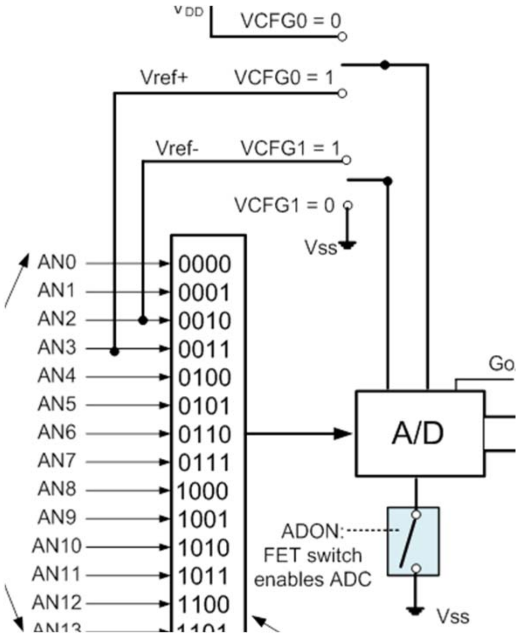
ADC Block Diagram

A/D acquisition Requirements: After selecting (or changing) the analogue input and before starting conversion it is necessary to provide at least 20 µs time delay to enable the ADC module (A/D converter module) to provide maximal conversion accuracy.

ADRESH and ADRESL Registers

- ✓ The ADC output can be stored in two formats: left justified and right justified
- ✓ The ADFM bit (bit 7) of ADCON1 control register determines the format of the conversion result





ADCON1 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADFM	-	VCFG1	VCFG0	-	-	-	-
R/W(0)		R/W(0)	R/W(0)				

R/W Read/Writable bit; (0) After rest, bit is cleared; (-) bit is unimplemented

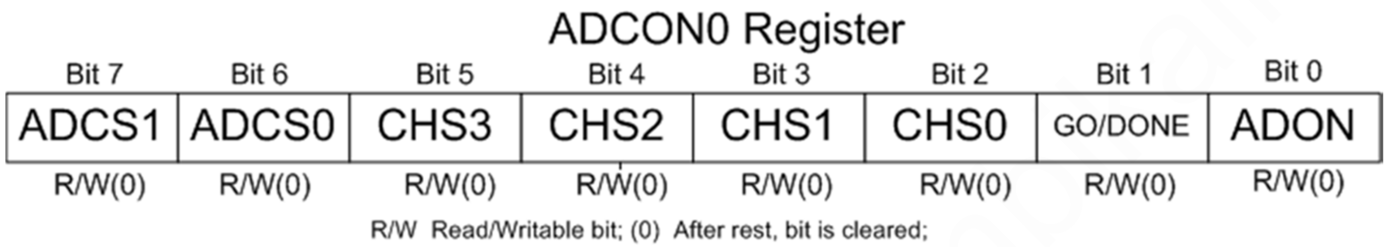
ADFM – A/D result Format Select bit
1 – conversion result right justified; six most significant bits of the ADRESLH are not used
0 – Conversion result is left justified. Six least significant bits of the ADRESL are not used.

VCFG1 – Voltage Reference bit selects negative voltage reference source needed for A/D converter operating
1 – Negative voltage reference is applied on the Vref- pin
0 – Voltage power supply VSS is used as negative voltage reference source

VCFG0 – Voltage Reference bit selects positive voltage reference source needed for A/D converter operating
1 – positive voltage reference is applied on the Vref+ pin
0 – Voltage power supply VDD is used as positive voltage reference source

Channel selection

✓ The CHS bits of the ADCON0 control register determine which channel is connected to the A/D converter (ie connected to the Sample and Hold circuit)



ADCON0 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCS1	ADCS0	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON
R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

R/W Read/Writable bit; (0) After rest, bit is cleared;

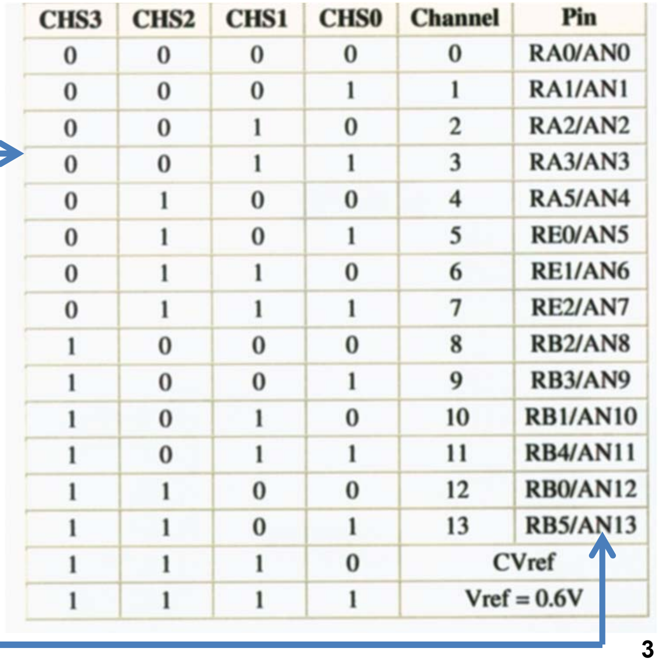
ADON– A/D enable bit
1 – A/D converter is enabled
0 – A/D converter is disabled

ADCS1, ADCS0 – A/D conversion Clock select bits. Select clock frequency used for internal synchronisation of A/D converter. It also affects the duration of conversion

GO/DONE: A/D conversion status bit determines current status of conversion;
1– A/D conversion is in progress;
0 - A/D conversion is complete

ADCS1	ADCS2	Clock
0	0	$F_{osc}/2$
0	1	$F_{osc}/8$
1	0	$F_{osc}/32$
1	1	F_{RC}*

CHS0 to CHS3: Analogue select bits select a pin/analogue channel for analogue to digital conversion



CHS3	CHS2	CHS1	CHS0	Channel	Pin
0	0	0	0	0	RA0/AN0
0	0	0	1	1	RA1/AN1
0	0	1	0	2	RA2/AN2
0	0	1	1	3	RA3/AN3
0	1	0	0	4	RA5/AN4
0	1	0	1	5	RE0/AN5
0	1	1	0	6	RE1/AN6
0	1	1	1	7	RE2/AN7
1	0	0	0	8	RB2/AN8
1	0	0	1	9	RB3/AN9
1	0	1	0	10	RB1/AN10
1	0	1	1	11	RB4/AN11
1	1	0	0	12	RB0/AN12
1	1	0	1	13	RB5/AN13
1	1	1	0	CVref	
1	1	1	1	Vref = 0.6V	

F_{RC}* = clock derived from a dedicated internal oscillator = 500kHz max)

There are 14 analogue input select pins available:
PA0 to PA5, PE0 to PE2 and PB0 to PB5 (Ports A, B and E)

A/D Conversion Procedure

- ✓ **Configure the port:**
  - ▪ Write logic one to the corresponding bit of the TRIS register to configure it as input;
  - ▪ Write logic one to the corresponding bit of the ANSEL register to configure it as analog input.
- ✓ **Configuring ADC module:**
  - ✓ Configure voltage reference in the ADCON1 register;
  - ✓ Select ADC conversion clock in the ADCON0 register
  - ✓ Select one of the input channels CH0-CH13 of the ADCON0 register
  - ✓ Select data format using the ADFM bit of the ADCON1 register
  - ✓ Enable A/D converter by setting the ADCON bit of the ADCON0 register
- ✓ **Configuring ADC Interrupt (optional):**
  - ✓ Clear ADIF bit (interrupt flag)
  - ✓ Set the ADIE,PEIE and GIE bits (enable ADC, peripheral and Global interrupts)
- ✓**Wait for the Required acquisition time (approx 20 μs) to pass**
- ✓**Start conversion by setting the GO/DONE bit of the ADCON0 register –** if GO/DONE bit is set, the A/D conversion will start
- ✓**Wait for ADC conversion to complete by one of the following :**
  - ✓Check in program loop to see if the GO/DONE bit is cleared (completion of a conversion)
  - ✓Wait for an A/D interrupt
- ✓**Read ADC results (i.e. read the ADRESH and ADRESL registers)**
- ✓**Clear the ADC interrupt flag (required if interrupt is enabled)**

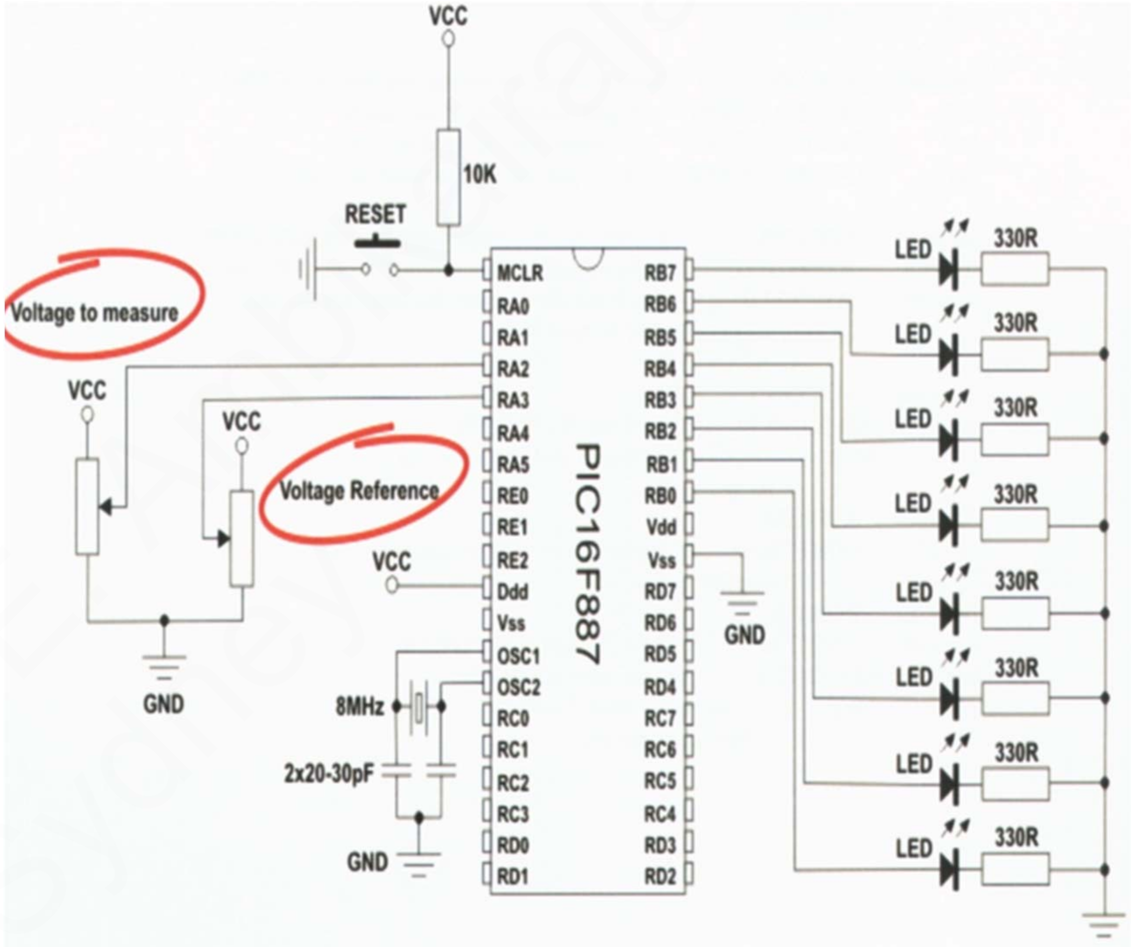
A/D Converter Example 1

✓ Variable analogue signal is applied on the AN2 (PA2) pin while the result of conversion is shown on PORTB as binary number.

✓A positive reference voltage is applied to the AN3 (PA3) pin.

✓Difference between two voltage levels is converted to a binary (10 bit) number.

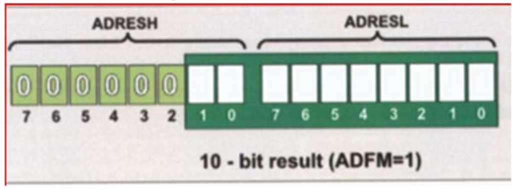
✓Only 8 lower bits of the result of conversion are shown.



A/D Converter Program

```
BANKSEL    TRISB
clrf       TRISB                 ;all portB pins are output
movlw      B’00001100’
movwf      TRISA                 ;pins PA2 and PA3 as configured as inputs

BANKSEL    ANSEL
movlw      B’00001100’           ;inputs AN2 and AN3 are analogue while others are digital
movwf      ANSEL  ; To configure a pin as an analog input, the appropriate bit of the ANSEL must be set  to 1
clrf       ANSELH  ; set the port B lines as digital

BANKSEL    ADCON1
bsf        ADCON1,ADFM           ; right justification of result →
bcf        ADCON1,VCFG1          ;Voltage Vss (GND) is used as Vref-
bsf        ADCON1,VCFG0          ;RA3 pin voltage is used as Vref+

BANKSEL    ADCON0
movlw      B’00001001’           ;A/D converter uses clock Fosc/2
movwf      ADCON0                ;on RA2 pin is used for conversion and A/D converter is enabled
```

Right Justified



ADCON1 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADFM	-	VCFG1	VCFG0	-	-	-	-
R/W(0)		R/W(0)	R/W(0)				

R/W Read/Writable bit; (0) After rest, bit is cleared; (-) bit is unimplemented

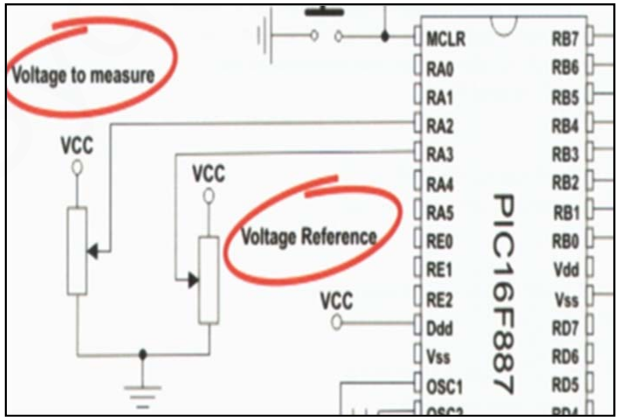
ADCON0 Register

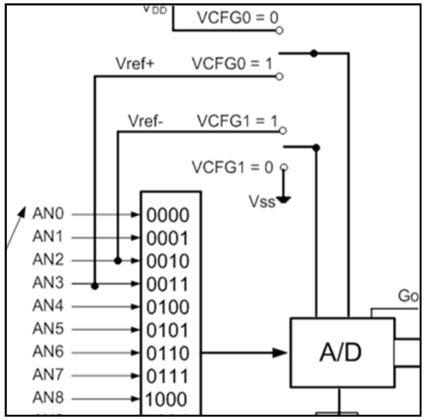
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCS1	ADCS0	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON
R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

R/W Read/Writable bit; (0) After rest, bit is cleared;

ADCS1	ADCS2	Clock
0	0	$F_{osc}/2$
0	1	$F_{osc}/8$
1	0	$F_{osc}/32$
1	1	F_{RC}*

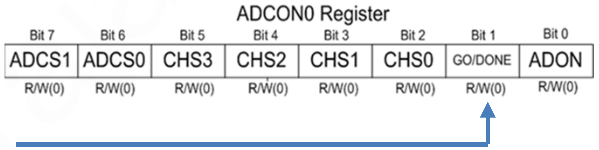
ADON– A/D enable bit
1 – A/D converter is enabled
0 – A/D converter is disabled





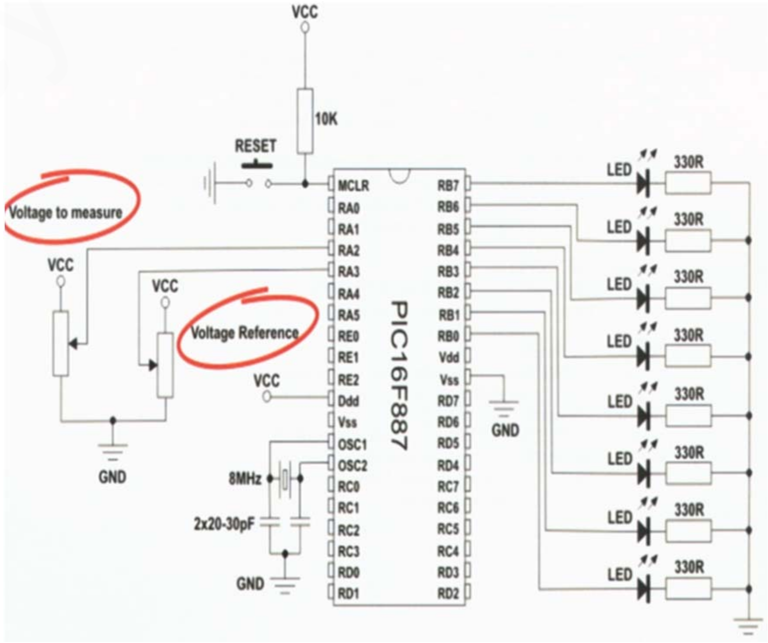
A/D Converter Program contd..

```
        call          acquisition_delay       ; Delay of 20 µs

        BANKSEL  ADCON0
        bsf           ADCON0,1 ;start conversion by setting GO/DONE bit

loop    nop
        btfsc         ADCON0,1 ;Tests bit Go/Done; Is this bit = 0 (at the end of A/D conversion ADC module clears this bit)
        goto          loop          ; conversion in progress, stay in loop
                                    ;At the end of conversion ADRESH:ADRESL registers are updated with the new
                                     ;conversion result
        BANKSEL  ADRESL
        movf          ADRESL,w ;lowerbyte of conversion result is copied to W

        banksel      PORTB
        movwf        PORTB       ;byte is copied to PORTB
        bsf           ADCON0,1 ;starts new conversion
        goto          loop
        END
```



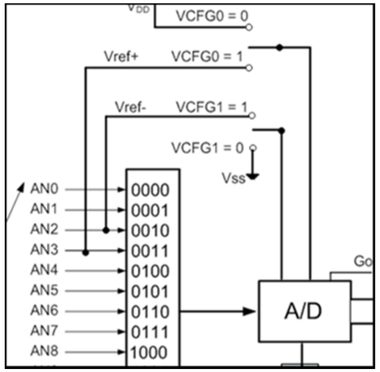
ADCON0 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCS1	ADCS0	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON
R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)



A/D Converter Example 2

- An analogue voltage source is connected to AN0 (RA0) of PORTA of a PIC16F886 microcontroller which is set up to sample the analogue signal and stores the 10 bit digital output in the locations ‘DIGITALHI’ (upper two bits) and ‘DIGITALLO’ (lower 8 bits). You need to configure the ADC for polling, V_{DD} and V_{SS} as reference, select F_{RC} clock and AN0 as analogue input.
- Write an assembly program to achieve the above.



ADCON1 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADFM	-	VCFG1	VCFG0	-	-	-	-
R/W(0)		R/W(0)	R/W(0)				

ADCON0 Register

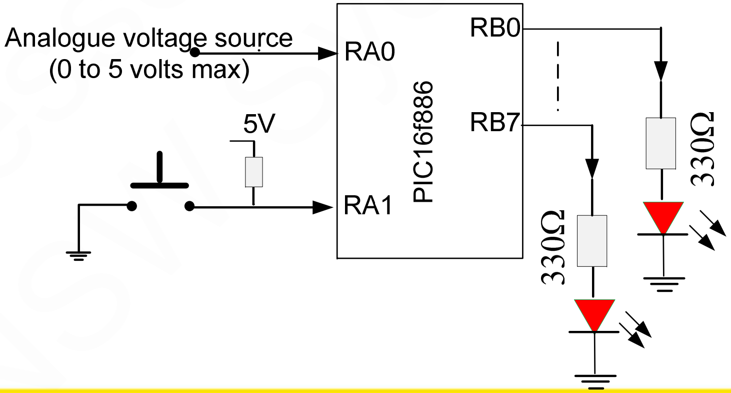
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCS1	ADCS0	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON
R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)	R/W(0)

```
     DIGITALHI   equ     H’20’
     DIGITALLO   equ     H’21’

     BANKSEL             ADCON1
     MOVLW               B’10000000’  ; result right justified ADFM =1
     MOVWF               ADCON1       ;VCFG1=0 (select Vss – ground); VCFG0=0 (select VDD)
     BANKSEL             TRISA
     BSF                 TRISA,0      ; Set RA0 to input
     BANKSEL             ANSEL
     BSF                 ANSEL,0      ; Set RA0 to analogue

     BANKSEL             ADCON0
     MOVLW               B’11000001’  ; FRC clock: ADCS1=1; ADCS0 =1; CHS0 to CHS3 = 0 (select AN0 for A/D conversion)
     MOVWF               ADCON0        ; ADON=1 (A/D converter is enabled)
     CALL                Delay        ; Delay of 20 µs
     BSF                 ADCON0,1     ; start conversion by setting GO/DONE bit
LOOP NOP
     BTFSC               ADCON0,1     ;Tests bit Go/Done; Is this bit = 0 (at the end of A/D conversion ADC module clears this bit)
     GOTO                LOOP         ; No try again

     BANKSEL             ADRESH
     MOVF                ADRESH,w     ;Read upper two bits
     MOVWF               DIGITALHI    ; store in memory

     BANKSEL             ADRESL
     MOVF                ADRESL,w     ;Read lower 8 bits
     MOVWF               DIGITALLO    ; store in memory
     END
```

Laboratory Activity

Activity 9: Analogue to Digital conversion

- A variable analogue voltage source is connected to PA0 (RA0) of PORTA of a PIC16F886 microcontroller and a switch is connected to PA1(RA1) of PORTA of the same microcontroller.
- Write an assembly code such that when the switch is pressed (high-to-low transition), the microcontroller samples the analogue input and the result of the conversion is shown as an 8-bit binary number on PORTB. Only the least significant 8-bits (of the 10-bit ADC) need to be displayed using the eight LEDs connected to PORTB.
- You need to clear VCFG1, VCFG0, ADSC0 and ADSC1 for the desired operation of the ADC module.
- You need to write a 20µs acquisition delay sub-routine (acquisition_delay) for you to call in your code. That is , after selecting (or changing) the analogue input and before starting conversion it is necessary to provide at least 20 µs time delay to enable the ADC module to provide maximal conversion accuracy.



ELEC2117: References

1. Designing Embedded Systems with PIC Microcontrollers – Tim Wilmshurst, Elsevier, 2010
2. PIC Microcontrollers –Free online book – mikroElektronika ; http://www.mikroe.com/products/view/11/book-pic-microcontrollers/
3. PIC 16F886 Data Sheet (2007), Microchip Technology; www.microchip.com