

```
*****
'* Name      : ADCGilbert.BAS          *
'* Author    : Marc Bertrand          *
'* Notice    : Copyright (c) 2017 Marc Bertrand *
'*           : All Rights Reserved     *
'* Date      : 11/17/2017             *
'* Version   : 1.0                    *
'* Notes     : 8 Channel ADC          *
'*           :                         *
*****
Device=16F767
Config1 INTRC_IO, WDT_OFF
Symbol GO_DONE = ADCON0.2 ' A/D Conversion Enable
Symbol INT0IF = INTCON.1 ' RBO External Interrupt Flag
Symbol INT0IE = INTCON.4 ' RBO External Interrupt Enable
Symbol GIE = INTCON.7 ' Global Interrupt Enable
Dim ADCResultMSB[8] As Byte
Dim ADCResultLSB[8] As Byte
Dim ADC_Channel As Byte
Dim ChannelSelected As Byte
On Hardware Interrupt GoTo ISR_OutputADCbyte
GoTo Main ' Jump over the interrupt subroutine
' -----
ISR_OutputADCbyte:
Context Save ' Save any variables or SFRs before the interrupt starts
If INT0IF = 1 Then ' Is it RBO that caused the interrupt (CS_not)?
ChannelSelected=PORTB & 0x60 'Get Bits 6 and 5 Address lines A2, and A1
ChannelSelected.4=PORTA.7 'Get Address line A0
Select ChannelSelected
Case 0x10
If PORTB.7=1 Then 'Do we need to output the MSB or LSB of A/D conversion?
PORTC=ADCResultMSB[1]
Else
PORTC=ADCResultLSB[1]
EndIf
Case 0x20
If PORTB.7=1 Then
PORTC=ADCResultMSB[2]
Else
PORTC=ADCResultLSB[2]
EndIf
Case 0x30
If PORTB.7=1 Then
```

```
        PORTC=ADCResultMSB[3]
    Else
        PORTC=ADCResultLSB[3]
    EndIf
Case 0x40
    If PORTB.7=1 Then
        PORTC=ADCResultMSB[4]
    Else
        PORTC=ADCResultLSB[4]
    EndIf
Case 0x50
    If PORTB.7=1 Then
        PORTC=ADCResultMSB[5]
    Else
        PORTC=ADCResultLSB[5]
    EndIf
Case 0x60
    If PORTB.7=1 Then
        PORTC=ADCResultMSB[6]
    Else
        PORTC=ADCResultLSB[6]
    EndIf
Case 0x70
    If PORTB.7=1 Then
        PORTC=ADCResultMSB[7]
    Else
        PORTC=ADCResultLSB[7]
    EndIf
Case Else
    If PORTB.7=1 Then
        PORTC=ADCResultMSB[0]
    Else
        PORTC=ADCResultLSB[0]
    EndIf
EndSelect
While PORTA.6=1
    'Wait until Output Enable (OE_NOT) goes LOW
Wend
Clear TRISC    'Config PORTC as Output
While PORTA.6=0
    'Wait until OE_NOT goes back HIGH
Wend
EndIf
```

```
Set TRISC      'Config PORTC as Input
Clear INT0IF
Context Restore ' Restore any variables or SFRs and exit the interrupt
' -----
Main:
OSCCON=%01110010 'Set for 8 MHz internal RC system clock

'Configure analog pins, voltage reference and digital I/O
'[ADFM | ADCS2 | VCFG1 | VCFG0 | PCFG3 | PCFG2 | PCFG1 | PCFG0]
'bit 7 ADFM: A/D Result Format Select bit
'1 = Right justified. Six Most Significant bits of ADRESH are read as '0'.
'bit 6 ADCS2: A/D Clock Divide by 2 Select bit
'0 = Disabled
'bit 5 VCFG1: Voltage Reference Configuration bit 1
'0 = VREF- is connected to VSS
'bit 4 VCFG0: Voltage Reference Configuration bit 0
'0 = VREF+ is connected to VDD
'bit 3-0 PCFG<3:0>: A/D Port Configuration bits
'0011 = AN0 to AN11 set as analog inputs
ADCON1=%10000011

Clear OPTION_REG.6 'Interrupt on falling edge of RB0/INT pin
Set INT0IE         'Enable RB0 External Interrupt
Set GIE           'Enable Global Interrupt
GetADconversion:
If PORTA.4=0 Then
  Clear ADCON1.4
Else
  Set ADCON1.4
EndIf
For ADC_Channel = 0 To 7
  Select ADC_Channel
  Case 1
    ADCON0=%11001001
  Case 2
    ADCON0=%11010001
  Case 3
    ADCON0=%11100001
  Case 4
    ADCON0=%11010011
  Case 5
    ADCON0=%11000011
  Case 6
```

```
    ADCON0=%11001011
  Case 7
    ADCON0=%11011011
  Case Else
    ADCON0=%11000001
EndSelect
DelayUS 50
Set GO_DONE 'Start A/D conversion
While GO_DONE=1
  'Wait for A/D conversion to complete
Wend
  'Save A/D Result registers
  ADCResultMSB[ADC_Channel]=ADRESH
  ADCResultLSB[ADC_Channel]=ADRESL
Next
GoTo GetADconversion
```