

ADCInt Library Module (Interrupt-driven)

1.	Introduction	2
2.	Module Features	2
3.	List of Component Modules.....	3
4.	Using the Library Module in a Project	3
5.	List of Shared Parameters	4
	<i>Shared Data Bytes</i>	<i>4</i>
	<i>Shared Functions</i>	<i>4</i>
	<i>Shared Macros</i>	<i>4</i>
6.	Functions.....	6
7.	Macros.....	8
8.	Port and ADC Reference Voltage Configuration Tables.....	11
9.	Error and Status Flags	13

1. Introduction

ADCInt is a general-purpose library module for processors of PIC18 family (except PIC18F1220 & 18F1320). It configures ADC module for interrupt mode and provides a descriptive interface in the form of macros and functions.

2. Module Features

- User selectable acquisition time to allow for settling of analog signal, before start of AD conversion.
- Provides First In First Out (FIFO) buffer to write ADC results.
- Provides simple functions to read and write the buffers.

3. List of Component Modules

ADCInt.P18.ex.txt	This file demonstrates the usage of the ADCInt library module.
ADCInt.asm	This file contains the functions of the ADCInt library module. <u>User needs to include this file in their project.</u>
ADCInt.inc	This file contains the macros of the ADCInt library module. <u>User needs to include this in their '.asm' file</u> , where the library module macros & functions are utilized.
P18xxx.inc	General purpose processor definition file for the PIC18xxx family

4. Using the Library Module in a Project

Please follow the steps below to use the ADCInt library module in your project.

1. Use Application Maestro™ software to configure the module as required.
2. At the 'Generate Files' step, save the output to the directory where your project code resides.
3. Launch MPLAB® IDE, and open the project's workspace.
4. Verify that the Microchip language tool suite is selected (*Project>Select Language Toolsuite*).
5. In the Workspace view, right-click on the "Source Files" node. Select the "Add Files" option. Select ADCInt.asm and click **OK**.
6. Now right-click on the "Linker Scripts" node and select "Add Files". Add the appropriate linker file (.lkr) for the project's target microcontroller.
7. Add any other files that the project may require. Save and close the project.
8. Include 'ADCInt.inc' in your '.asm' file, where the ADCInt library functions & macros are utilized.

5. List of Shared Parameters

Shared Data Bytes

vADCIntChannelNumber_A	The content of this gives the ADC channel that is being sensed
vADCIntStatus	This holds the error and status flags. . Please refer Section 9 for more information.
vADCIntBufWrPtr	This points to the relative location where the ADC result is written to the buffer.
vADCIntBufRdPtr	This points to the relative location where the ADC result is read from the buffer.
vADCIntResultHigh	The ADC result is read to this location if selected resolution is 8 bits. The ADC high-byte result is read to this location, if selected resolution is 10 bits.
vADCIntResultLow	The ADC result low-byte is read to this location, if selected resolution is 10 bits.
vADCIntResultsCount	This keeps track of the number of ADC results yet to be read from the buffer.
VADCIntBuffer	Starting location of the FIFO buffer.

Shared Functions

ADCIntAcquisitionTime	This provides acquisition time delay before the start of AD conversion. The required acquisition time is entered by the user in the Application Maestro software.
ADCIntISR	This is an Interrupt service routine for the ADC Interrupt. Call it from the Interrupt service routine at proper Interrupt vector(High or Low priority Vector).
ADCIntRead	This function reads the ADC result from FIFO buffer, into the locations vADCIntResultHigh & vADCIntResultLow, if selected resolution is 10 bits, vADCIntResultHigh if selected resolution is 8 bits.
ADCIntBufInit	This function clears ADCIntBufFull, ADCIntBufOverflow flags, sets ADCIntBufEmpty flag, clears vADCIntBufWrPtr, vADCIntBufRdPtr, vADCIntResultsCount.

Shared Macros

mADCIntInit	Configures the ADC as per the user options. This sets channel 0 as the default ADC channel. If the user wants to use a different channel, he can select the required channel by invoking the macro mADCIntChannelSelect.
mADCIntChannelSelect	Selects the ADC channel. This macro calls a function. The purpose of having this macro is to pass parameter for channel number. This macro gives the warning message "WARNING: The processor does not have the selected ADC channel", if the user selects a channel not available for the processor. The channel selection is ignored in the above case and the previous channel will be used for the next conversion.
mADCIntStart	Starts the AD conversion.
mADCIntIsBusy	Checks whether the AD conversion is over or not.

mADCIntReadHigh	This reads the high-byte of the ADC result register (ADRESH) into WREG.
mADCIntReadLow	This reads the low-byte of the ADC result register (ADRESL) into WREG.
mADCIntDisable	Switches off the ADC.
mSetADCIntHighPriority	This macro sets high-priority for ADC interrupt
mSetADCIntLowPriority	This macro sets low-priority for ADC interrupt

6. Functions

Function	ADCIntAcquisitionTime
Pre-conditions	To be used after invoking the macro mADCIntInit & mADCIntChannelSelect
Overview	Provides acquisition time delay before start of AD conversion. The required acquisition time is entered by the user in the Application Maestro software.
Input	None
Output	None
Side Effects	WREG changes
Stack Requirement	1 level deep
Maximum instruction cycles	Processor clock freq.* Acquisition time(u sec)/4000000
Function	ADCIntISR
Pre-conditions	To be used after invoking the macro mADCIntInit.
Overview	This function checks the AD interrupt flag. If flag is not set, control returns. Otherwise the ADC result is written/ not written depending upon the ADCIntBufOverflow flag & user option overwrite/no-overwrite buffer. When the buffer is written, ADCIntBufEmpty flag is cleared. The flag ADCIntBufFull is set, if all the locations of the buffer are written with ADC result but none is read using ADCIntRead. ADCIntBufOverflow flag is set when new ADC result becomes available after ADCIntBufFull is set.
Input	None
Output	Buffer starting at vADCIntBuffer, vADCIntStatus, vADCIntBufWrPtr, vADCIntResultsCount.
Side Effects	WREG, STATUS, BSR changes.
Stack Requirement	1 level deep
Maximum instruction cycles	49
Function	ADCIntRead
Pre-conditions	To be used after invoking the macro mADCIntInit.
Overview	This function reads the ADC results from the buffer. In case of 10-bit ADC resolution, the result is read into the locations vADCIntResultHigh & vADCIntResultLow. In case of 8-bits resolution, the result is read into vADCIntResultHigh only. It sets ADCIntBufEmpty flag when there is no new result to be read from the buffer.
Input	None
Output	vADCIntResultHigh & vADCIntResultLow (10-bits resolution) vADCIntResultHigh (8-bits resolution) ADCIntBufEmpty flag.
Side Effects	WREG, STATUS, BSR changes.
Stack Requirement	1 level deep
Maximum instruction cycles	27

Function	ADCIntBufInit
Pre-conditions	None
Overview	This function clears ADCIntBufFull, ADCIntBufOverflow flags, sets ADCIntBufEmpty flag, clears vADCIntBufWrPtr, vADCIntBufRdPtr, vADCIntResultsCount.
Input	None
Output	vADCIntStatus, vADCIntBufWrPtr, vADCIntBufRdPtr, vADCIntResultsCount.
Side Effects	BSR, STATUS changes
Stack Requirement	1 level deep
Maximum instruction cycles	10

7. Macros

Macro	<code>mADCIntInit</code>
Overview	Configures the ADC as per the user options. Enables the ADC interrupt. This macro sets channel 0 as the default ADC channel. If user wants to use a different channel, he can select the required channel by invoking the macro <code>mADCIntChannelSelect</code> .
Input	This macro takes one or two arguments depending upon the processor used. For processors not having ADCON2 register, ADC port configuration name is the only argument required. Chose the appropriate port configuration name from Table 1. For processors having ADCON2 register, this macro requires two arguments. The first argument is the port configuration name & the second argument is the ADC reference voltage configuration name. Chose these arguments from Table 2 & 3 respectively.
	<p>Example1: Single Argument</p> <pre>; AN0-AN4 analog ports, VREF+ is V_{DD}, VREF- is V_{SS} mADCIntInit ADCPORTCONFIG2</pre> <p>Example 2: Two arguments</p> <pre>; AN0-12 analog ports, External VREF+ at AN3, VREF- is AV_{ss} mADCIntInit ADCPORTCONFIG2,ADCREERENCECONFIG1</pre>
Output	None
Side Effects	WREG, STATUS, BSR changes
Stack Requirement	1 level deep
Maximum instruction cycles	20
Macro	<code>mADCIntChannelSelect</code>
Overview	Selects the ADC channel. This macro calls a function. The purpose of having this macro is to pass parameter for channel number. This macro gives the warning message "WARNING: The processor does not have the selected ADC channel", if user selects a channel not available for the processor. The channel selection is ignored in the above case and the previous channel will be used for next conversion.
	<p>Example: To select the ADC channel 11, write</p> <pre>mADCIntChannelSelect D'11' or mADCIntChannelSelect .11</pre>
Input	ADC channel number as the macro argument.
Output	Stores the selected channel number in the location <code>ADCIntChannelNumber_A</code>
Side Effects	WREG, STATUS changes
Stack Requirement	1 level deep
Maximum instruction cycles	12

Macro	mADCIntStart
Overview	Starts the AD conversion.
Input	None
Output	None
Side Effects	None
Stack Requirement	None
Maximum instruction cycles	1

Macro	mADCIntIsBusy
Overview	Checks whether the AD conversion is over or not.
Input	None
Output	WREG contains 0x0 if the AD conversion is over, else 0x1.
Side Effects	None
Stack Requirement	None
Maximum instruction cycles	3

Macro	mADCIntReadHigh
Overview	This reads the high-byte of the ADC result register (ADRESH) into WREG.
Input	None
Output	WREG
Side Effects	Status changes
Stack Requirement	None
Maximum instruction cycles	1

Macro	mADCIntReadLow
Overview	This reads the low-byte of the ADC result register (ADRESL) into WREG
Input	None
Output	WREG
Side Effects	Status changes
Stack Requirement	None
Maximum instruction cycles	1

Macro	mADCIntDisable
Overview	Switches off the ADC.
Input	None
Output	None
Side Effects	None
Stack Requirement	None
Maximum instruction cycles	2

Macro	mSetADCIntHighPriority
Overview	This macro sets high priority for ADC interrupt
Input	None
Output	None
Side Effects	None
Stack Requirement	None
Maximum instruction cycles	2

Macro	mSetADCIntLowPriority
Overview	This macro sets low priority for ADC interrupt
Input	None
Output	None
Side Effects	None
Stack Requirement	None
Maximum instruction cycles	2

8. Port and ADC Reference Voltage Configuration Tables

Table1: Port Configuration

PORT CONFIGURATION NAME	PORT AN7	PORT AN6	PORT AN5	PORT AN4	PORT AN3	PORT AN2	PORT AN1	PORT AN0	ADC VREF+	ADC VREF-
ADCPORTCONFIG0	A	A	A	A	A	A	A	A	VDD	VSS
ADCPORTCONFIG1	A	A	A	A	VREF+	A	A	A	AN3	VSS
ADCPORTCONFIG2	D	D	D	A	A	A	A	A	VDD	VSS
ADCPORTCONFIG3	D	D	D	A	VREF+	A	A	A	AN3	VSS
ADCPORTCONFIG4	D	D	D	D	A	D	A	A	VDD	VSS
ADCPORTCONFIG5	D	D	D	D	VREF+	D	A	A	AN3	VSS
ADCPORTCONFIG6	A	A	A	A	VREF+	VREF-	A	A	AN3	AN2
ADCPORTCONFIG7	D	D	A	A	A	A	A	A	VDD	VSS
ADCPORTCONFIG8	D	D	A	A	VREF+	A	A	A	AN3	VSS
ADCPORTCONFIG9	D	D	A	A	VREF+	VREF-	A	A	AN3	AN2
ADCPORTCONFIG10	D	D	D	A	VREF+	VREF-	A	A	AN3	AN2
ADCPORTCONFIG11	D	D	D	D	VREF+	VREF-	A	A	AN3	AN2
ADCPORTCONFIG12	D	D	D	D	D	D	D	A	VDD	VSS
ADCPORTCONFIG13	D	D	D	D	VREF+	VREF-	D	A	AN3	AN2

Table 2: Port Configuration

PORT CONFIGURATION NAME	AN15	AN14	AN13	AN12	AN11	AN10	AN9	AN8	AN7	AN6	AN5	AN4	AN3	AN2	AN1	AN0
ADCPORTRCONFIG0	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
ADCPORTRCONFIG1	D	D	A	A	A	A	A	A	A	A	A	A	A	A	A	A
ADCPORTRCONFIG2	D	D	D	A	A	A	A	A	A	A	A	A	A	A	A	A
ADCPORTRCONFIG3	D	D	D	D	A	A	A	A	A	A	A	A	A	A	A	A
ADCPORTRCONFIG4	D	D	D	D	D	A	A	A	A	A	A	A	A	A	A	A
ADCPORTRCONFIG5	D	D	D	D	D	D	A	A	A	A	A	A	A	A	A	A
ADCPORTRCONFIG6	D	D	D	D	D	D	D	A	A	A	A	A	A	A	A	A
ADCPORTRCONFIG7	D	D	D	D	D	D	D	D	A	A	A	A	A	A	A	A
ADCPORTRCONFIG8	D	D	D	D	D	D	D	D	D	A	A	A	A	A	A	A
ADCPORTRCONFIG9	D	D	D	D	D	D	D	D	D	D	A	A	A	A	A	A
ADCPORTRCONFIG10	D	D	D	D	D	D	D	D	D	D	D	A	A	A	A	A
ADCPORTRCONFIG11	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A	A
ADCPORTRCONFIG12	D	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
ADCPORTRCONFIG13	D	D	D	D	D	D	D	D	D	D	D	D	D	D	A	A
ADCPORTRCONFIG14	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	A

Table 3: ADC Reference Voltage Configuration

CONFIGURATION NAME	ADC VREF+	ADC VREF-
ADCREERENCECONFIG0	AVDD	AVSS
ADCREERENCECONFIG1	External VREF+ at AN3	AVSS
ADCREERENCECONFIG2	AVDD	External VREF- at AN2
ADCREERENCECONFIG3	External VREF+ at AN3	External VREF- at AN2

9. Error and Status Flags

<code>ADCIntBufFull</code>	This bit is set when the buffer is full. It gets reset when data is read from the buffer by the function <code>ADCIntRead</code> .
<code>ADCIntBufEmpty</code>	This bit is set when there is no new data to be read from the buffer. It gets reset, when a new data is written to the buffer by the function <code>ADCIntISR</code>
<code>ADCIntBufOverflow</code>	This bit is set when buffer is full & a new data is ready to be written into the buffer. <u>User needs to clear this bit in their firmware</u> . User can make use of the function <code>ADCIntBufInit</code> to clear <code>ADCIntBufOverflow</code> flag and to initialize the buffer.

MPLAB is a registered trademark of Microchip Technology INC. in the U.S.A and other countries. Application Maestro is a trademark of Microchip Technology INC. in the U.S.A and other countries. All other trademarks mentioned herein are property of their respective companies.