I2C^{TM} Slave Library Module (Interrupt-driven)

1. Introduction

The I2CSInt is a general-purpose library module. It configures SSP/MSSP module in Slave mode and helps in communicating with the I2C**TM** Master.

The module code is linkable and relocatable, which provides the user, the facility to use it without modifications.

It provides interrupt-based operation and has its own Tx/Rx buffer, which provides maximum benefit of parallel processing.

By using this Module one can write his application to interact with any of the I2C Master.
The module allows user to concentrate more on his application's development by providing these library functions.

2. Module Features

It supports following features:-

- It provides simple and primitive functions to communicate with the I2C Master.
- User defined length of Tx/Rx Buffer.
- Interrupt driven transmission and reception.
- It provides error recovery option. It uses, user selectable Timer for this purpose.
- It generates Error flags on the occurrence of an error. All error conditions are passed through the 'I2CSIntStatus' Register.

3. List of Component Modules

`I2CSInt.P16.ex.txt`	This is an example file developed to demonstrate the use of the library functions for PIC16 family.
`I2CSInt.P18.ex.txt`	This is an example file developed to demonstrate the use of the library functions for PIC18 family.
`I2CSInt.asm`	This is the I2C Slave code implementation file. <u>One needs to include this file in their project.</u>
`16I2CSI.asm`	This is the I2C Slave code implementation file for PIC16 family. The `I2CSInt.asm` file will include this file if the PIC16 family processor is used.
`18I2CSI.asm`	This is the I2C Slave code implementation file for PIC18 family. The `I2CSInt.asm` file will include this file if the PIC18 family processor is used.
`I2CSInt.inc`	This file contains the definitions of all the shared parameters and the macros. <u>One needs to include this in the Assembly file</u> where the library functions and macros are called. This file is taking care of definition of all Extern Global parameter so one can directly call library routines in their program.
`P16xxx.inc`	General purpose processor definition file for PIC16 family
`P18xxx.inc`	General purpose processor definition file for PIC18 family

4. Using the Library Module in a Project

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

1. Use the Application Maestro 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, 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 `I2CSInt.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. In your main source (assembler) file, add include directive at the head of the code listing to include `I2CSInt.inc`. By doing so, all files required to make the generated code work in your project will be included by reference when you build the project.
9. To use the module in your application, invoke the functions or macros as needed.

5. List of Shared Parameters

Shared Data Bytes

vI2CSIntStatus — It is the Error/Status register.
The details of each bits of this register is explained in Section 8

Shared Functions

I2CSIntInit — It is used for Synchronous Serial Port Initialization It initializes Port according to the options opted through Application Maestro.

I2CSIntPut — It is used for transmitting a byte on I2C Bus.

I2CSIntGet — It is used for reading the received byte.

I2CSIntISR — It is called from interrupt handler. It transmits/receives data from Master and sets Error/Status flags accordingly.

I2CSIntDiscardRxBuf — It is used for discarding the Rx Buffer contents.

Shared Macros

mI2CSIntDisable — Disables Synchronous Serial Port.

mSetI2CSIntHighPriority — This sets the interrupt priority of SSP High.

mSetI2CSIntLowPriority — This sets the interrupt priority of SSP Low.

6. Functions

Function	`I2CSIntInit`
Preconditions	TRIS bits of the SCL,SDA are to be made inputs and if Timer is used for error recovery, it has to be initialized for the required Time-out period.
Overview	This function is used for initializing the MSSP/SSP module. It initializes the module according to Application Maestro options.
Input	Application Maestro options
Output	None
Side Effects	Bank selection bits and 'W' register are changed
Stack Requirement	1 level deep

Function	`I2CSIntPut`
Preconditions	'`I2CSIntInit`' should have been called.
Overview	This function sends the byte in 'W' Reg. over I2C bus or saves it in the Tx Buffer, to be sent later.
Input	'W' Register.
Output	'`I2CSIntStatus`' Register. '`I2CSIntStatus <I2CSTxBufFull>`' is set if Tx-Buffer becomes full. '`I2CSIntStatus <I2CSTxBufEmpty,I2CSTxBufUnderFlow>`' are cleared.
Side Effects	Bank selection bits and 'W' register are changed
Stack Requirement	1 level deep

Function	`I2CSIntGet`
Preconditions	'`I2CSRxBufEmpty`' bit of '`I2CSIntStatus`' should be '0'.
Overview	This function reads the byte received.
Input	None
Output	'W' Register and '`I2CSIntStatus`' Register. 'W' Register' will have received Data. '`I2CSIntStatus <I2CSRxBufEmpty>`' is set if Rx-Buffer becomes empty. '`I2CSIntStatus <I2CSRxBufFull,I2CSRxBufOverFlow>`' are cleared.
Side Effects	Bank selection bits and 'W' register are changed
Stack Requirement	1 level deep

Function	I2CSIntISR
Preconditions	Must be called from interrupt handler.
Overview	If the SSP interrupt has occurred then, it transmits/receives data from Master, sets Error/Status flags accordingly, clears the Timer and enables the Timer interrupt. If Timer interrupt has occurred then, it disables SSP Module, releases the clock, re-enable the SSP Module and disables the timer interrupt.
Input	None
Output	'I2CSIntStatus' Register. 'I2CSIntStatus <I2CSTx>' is set if Master wants to read from Slave. 'I2CSIntStatus <I2CSRxBufEmpty>' is cleared if data is received from Master. 'I2CSIntStatus <I2CSRxBufFull>' is set if Rx-Buffer becomes full. 'I2CSIntStatus <I2CSRxBufOverFlow>' is set if a data byte is received when Rx-Buffer is full. 'I2CSIntStatus <I2CSTxBufFull>' is cleared if data is sent to the Master. 'I2CSIntStatus <I2CSTxBufEmpty>' is set if Tx-Buffer becomes empty. 'I2CSIntStatus <I2CSTxBufUnderFlow>' is set if the Master wants to read a data byte when Tx-Buffer is empty.
Side Effects	Bank selection bits and 'W' register are changed
Stack Requirement	2 level deep

Function	I2CSIntDiscardRxBuf
Preconditions	None
Overview	This discards the received data bytes.
Input	None
Output	'I2CSIntStatus' Register. 'I2CSIntStatus <I2CSRxBufEmpty>' is set. 'I2CSIntStatus <I2CSRxBufFull, I2CSRxBufOverFlow >' are cleared.
Side Effects	Bank selection bits are changed
Stack Requirement	1 level deep

7. Macros

Macro   `mI2CSIntDisable`
Overview   Disables the SSP/BSSP/MSSP module.
Input   None
Output   None
Side Effects   Bank selection bits are changed.
Stack Requirement None

Macro   `mSetI2CSIntHighPriority` (Valid only for PIC18 family devices).
Overview   This sets the interrupt priority of SSP High.
Input   None
Output   None
Side Effects   Bank selection bits are changed.
Stack Requirement None

Macro   `mSetI2CSIntLowPriority` (Valid only for PIC18 family devices).
Overview   This sets the interrupt priority of SSP Low.
Input   None
Output   None
Side Effects   Bank selection bits are changed.
Stack Requirement None

8. Error and Status Flags

All errors/status are set as a content of 'I2CSIntStatus' Register. Individual errors/status are unique. Please refer below list for the information.

I2CSTx	This indicates that, the Master wants to read data bytes from this device.
I2CSRxBufFull	This indicates that, the Rx-Buffer is full..
I2CSRxBufEmpty	This indicates that, the Rx-Buffer is empty.
I2CSRxBufOverFlow	This indicates that, a byte of data has been received while the Rx-Buffer is full.
I2CSTxBufEmpty	This indicates that, the Tx-Buffer is empty.
I2CSTxBufFull	This indicates that, the Tx-Buffer is full..
I2CSTxBufUnderFlow	This indicates that, a byte of data is demanded by the Master while the Tx-Buffer is empty.