EXAMPLE SCI RECEPTION

OBJECTIVE
This example has the following objectives:

- Review the use of serial communication interface (SCI) for reception
- Illustrate the receiving of a character through SCI

PROGRAM SCI_RECEPT
This program is an example of SCI reception. The character T is being sent to SCDR when TDRE is set.

Instructions
(i) Load REGBAS in reg. X
(ii) Initialize BAUD=9600
(iii) Initialize 8-bit word
(iv) Initialize TE and RE

Wait for the reception of a keystroke
(v) check if RDRF is set by loading SCSR into accA and AND with mask for RDRF, %00100000
You are here when reception data reg. is full
(vi) Load SCDR into accB
(vii) Store accB through Port B
(viii) Branch back to (v)
(ix) SWI

Flowchart

The program flowchart is show to the right of the program instructions. Note the initialization block, which contains reg. X initialization and SCI initialization. After initialization, the status of RDRF (reception data register full) flag is checked in a loop. When RDRF is set, the loop is exited and the content of SCDR (serial communication data register) is loaded in accB. This operation automatically resets RDRF. Now, the program loops back to the beginning and waits for another transmission.

The essential code for the program is shown to the right of the program flowchart. This essential code was incorporated into the standard asm template to generate the file SCI_recept.asm.

EXECUTION
a) Press the RUN button. The program should loop on LABEL1 and exit the loop when the condition RDRF=1 is satisfied. This condition is checked with the mask %00100000. The mask is applied on the value found in SCSR. When bit 5 in SCSR is set, the mask senses it and the condition for exiting the loop is satisfied.

b) Type the character T in the serial transmitter window and press Send. The character is sent to the MCU. The MCU takes a few laps to sense it. When the MCU senses it, the RDRF gets set, and the 'check RDRF' loop is exited.

c) When the 'check RDRF' loop is exited, the program gets to the line
   
   LDAB SCDR,X

This line is a breakpoint. At this moment, the screen looks like this:
Note that that SCSR is %11100000, i.e., the RDRF flag (bit 5 of SCSR) is set. This indicates that the character was received. (The TDRE and TE flags, i.e., bits 6 and 7, are also set, but this is due to the reset.) Note also that the ASCII code for the character T, $54 appears in SCDR (serial communication data register). From SCDR it will be loaded into accB.

d) Step manually to the next line, which loads the data from SCDR into accB. This load operation automatically resets RDRF. The screen looks like this:

Note that SCSR is %11000000. This proves that RDRF (bit 5) has been reset by the loading from SCDR; but TDRE and TC remained set, since they are not affected by reception operations.

e) Step again. The content of accB is stored in PORTB. Your screen looks like this:

Note that the ASCII code for T, i.e., $54, appears in PORTB.
f) Switch again to auto run. The program loops now again on LABEL1. Press again the send button in the serial transmitter. The character T is sent again, and after a few laps the program exits the loop and gets to the LDAB SCDR,X breakpoint.

g) Run again automatically. Change the character in the serial transmitter to A. You will notice that the corresponding ASCII code, $41, appear in SCDR, and then, after manual steps, gets loaded in accB and stored in PORTB.

WHAT YOU HAVE LEARNED
In this example, you have learned:

- The use of serial communication interface (SCI) for reception
- The reception of a character through SCI
- New words and notations: RDRF, ASCII code.