example_sci_uart_9600.c

This example code configures SCI and transmits a set of characters. An UART receiver can be used to receive this data. The scilin driver files should be generated with deafult settings.

Execution: Connect the SCI port of the micro to the COM port of the personal computer with MS Windows. Configure the MS Windows Hytermenal as COM port with the following Port Settings: Bits per second : 9600, Data Bits : 8 Parity : None Stop Bits : 2

Step 1:

Create a new project.

Navigate: -> File -> New -> Project

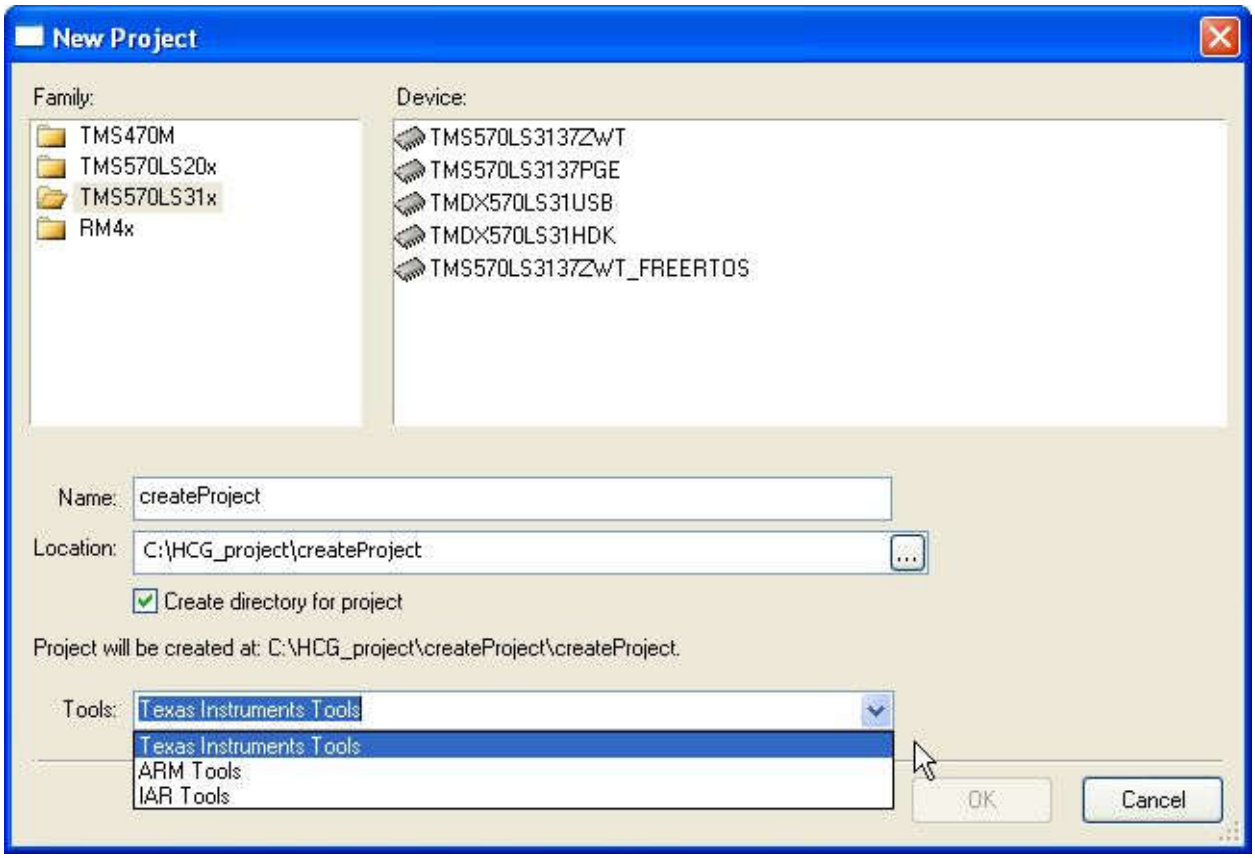

Figure: Create a new Project

Step 2:

Navigate: -> TMS570LSxx /RM4x -> Enable Drivers

Configure driver code generation:

- Enable SCILIN driver
- Disable others

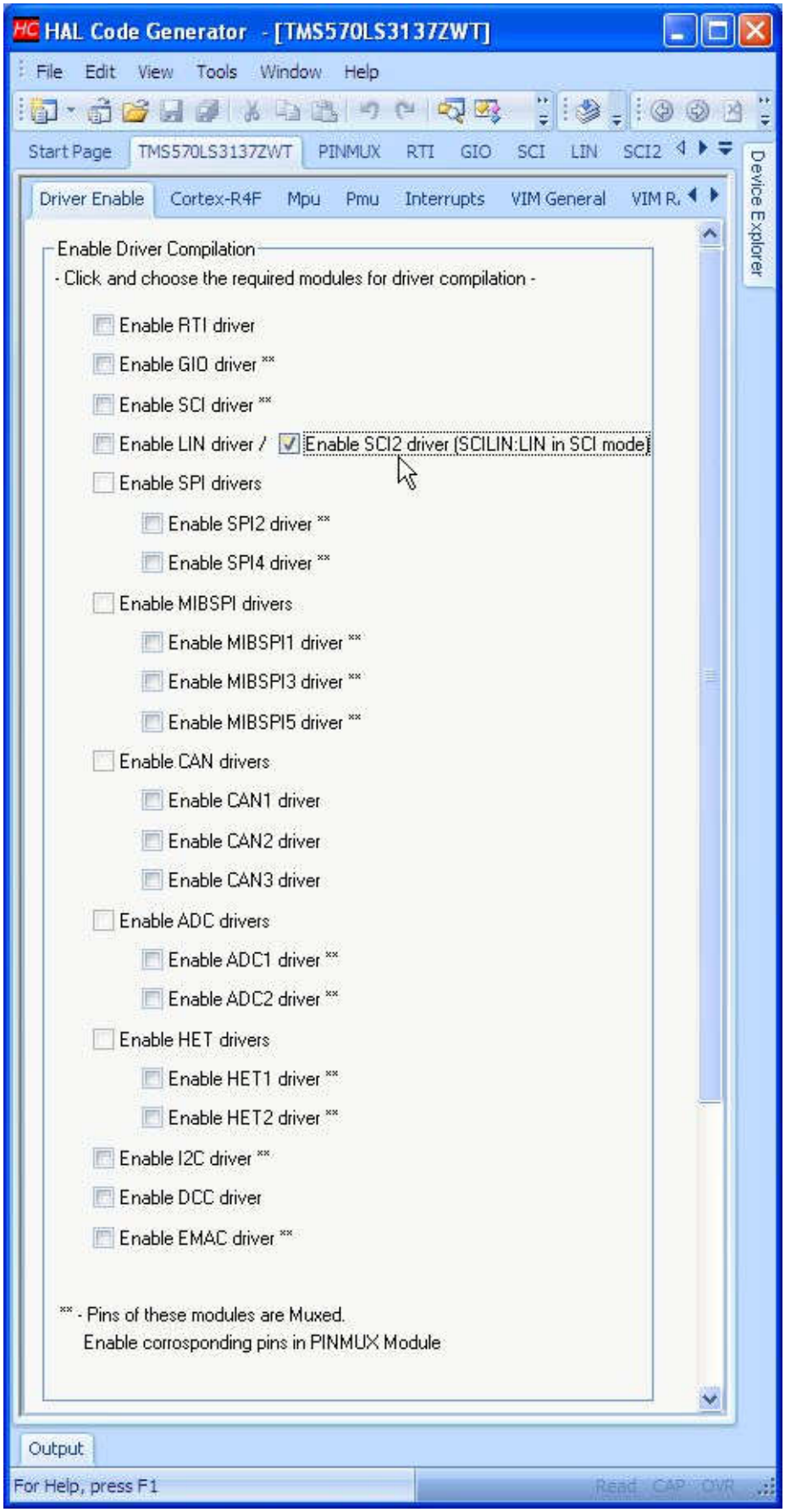

Figure: SCI Enable Driver

Step 3:

Configure SCI:

Navigate: -> TMS570LSxx /RM4x -> SCI

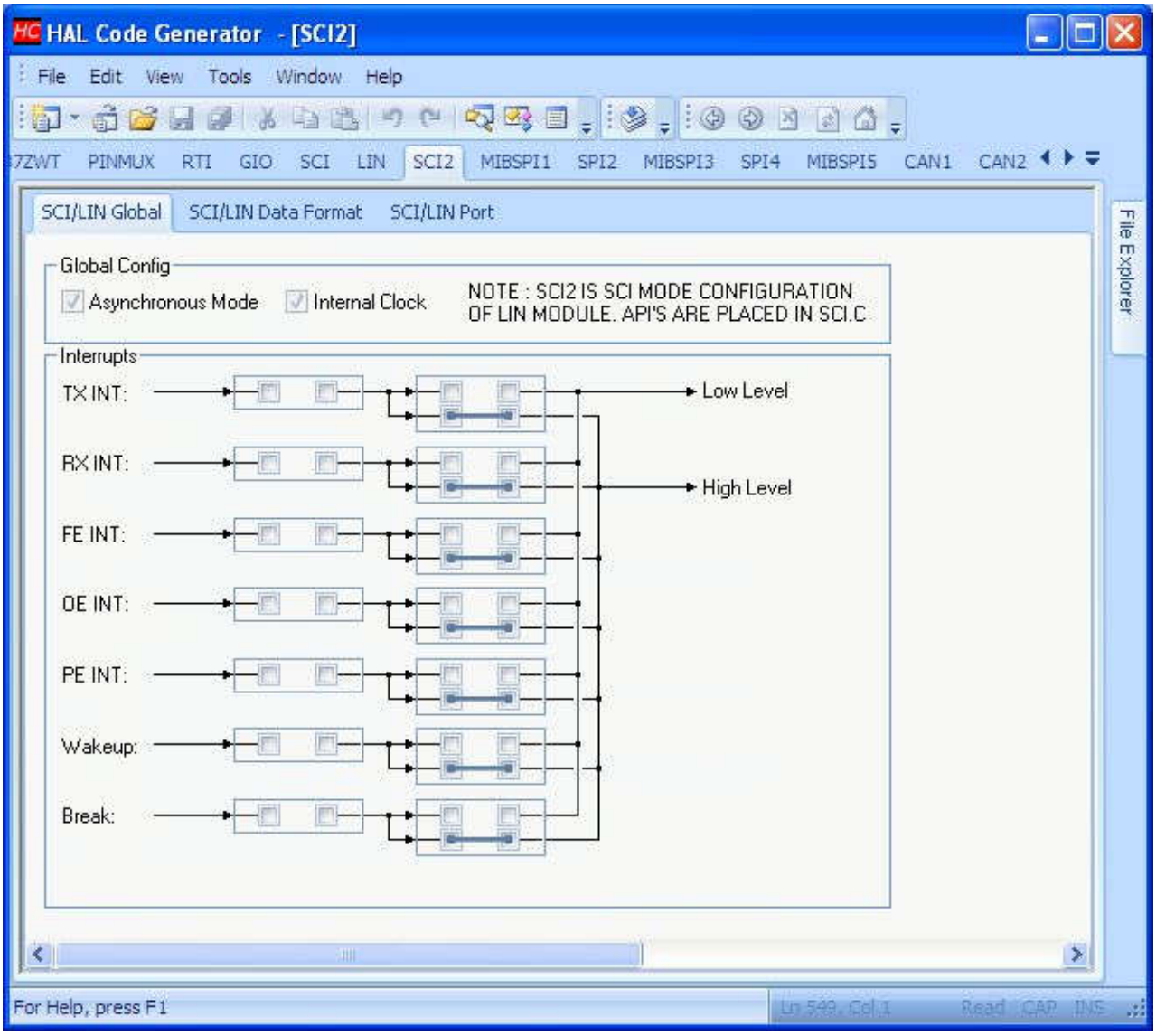

Figure: SCI Global settings

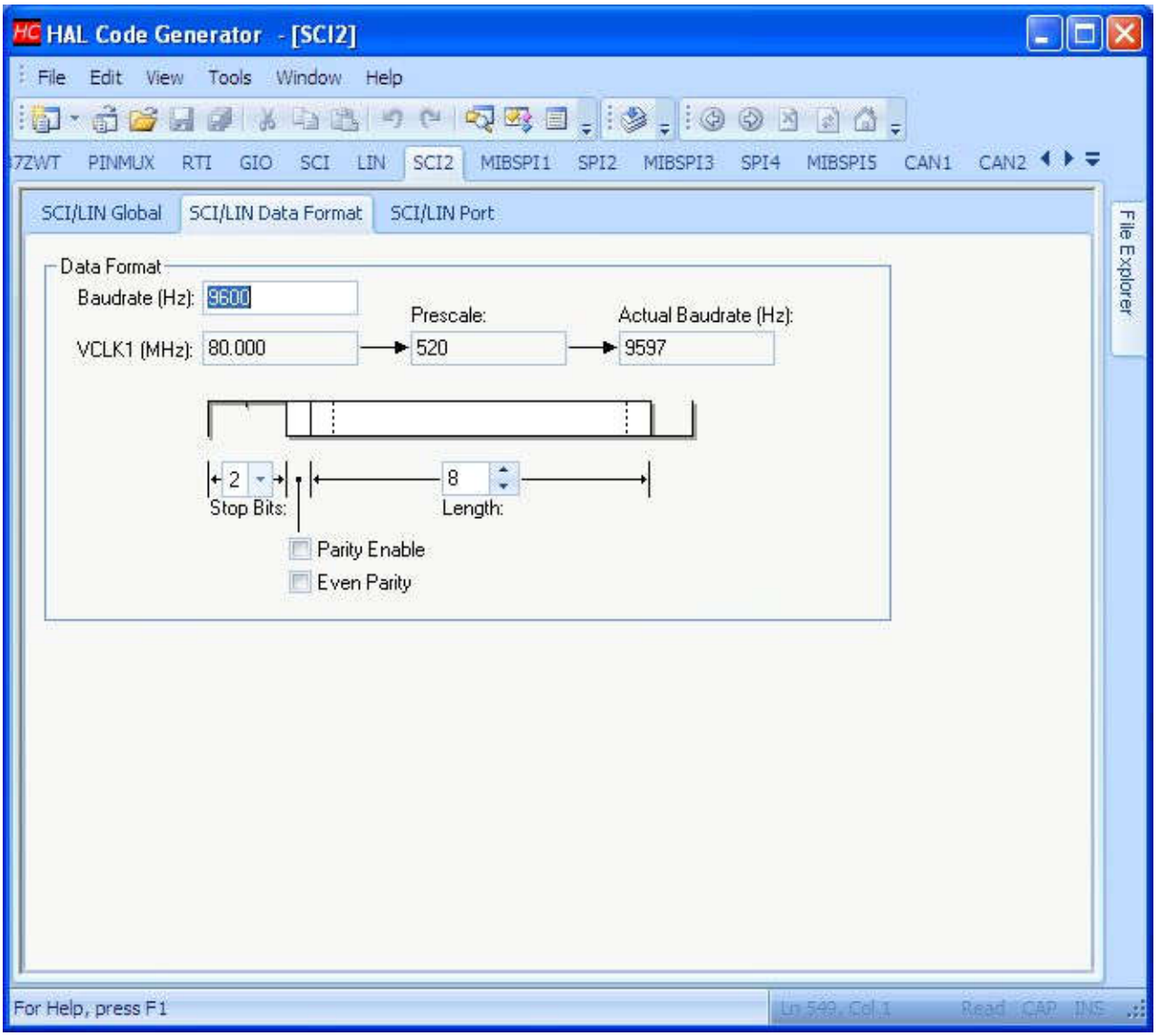

Figure: SCI Data Format

Step 4:

Click on Generate code icon. Navigate: -> File -> Generate Code

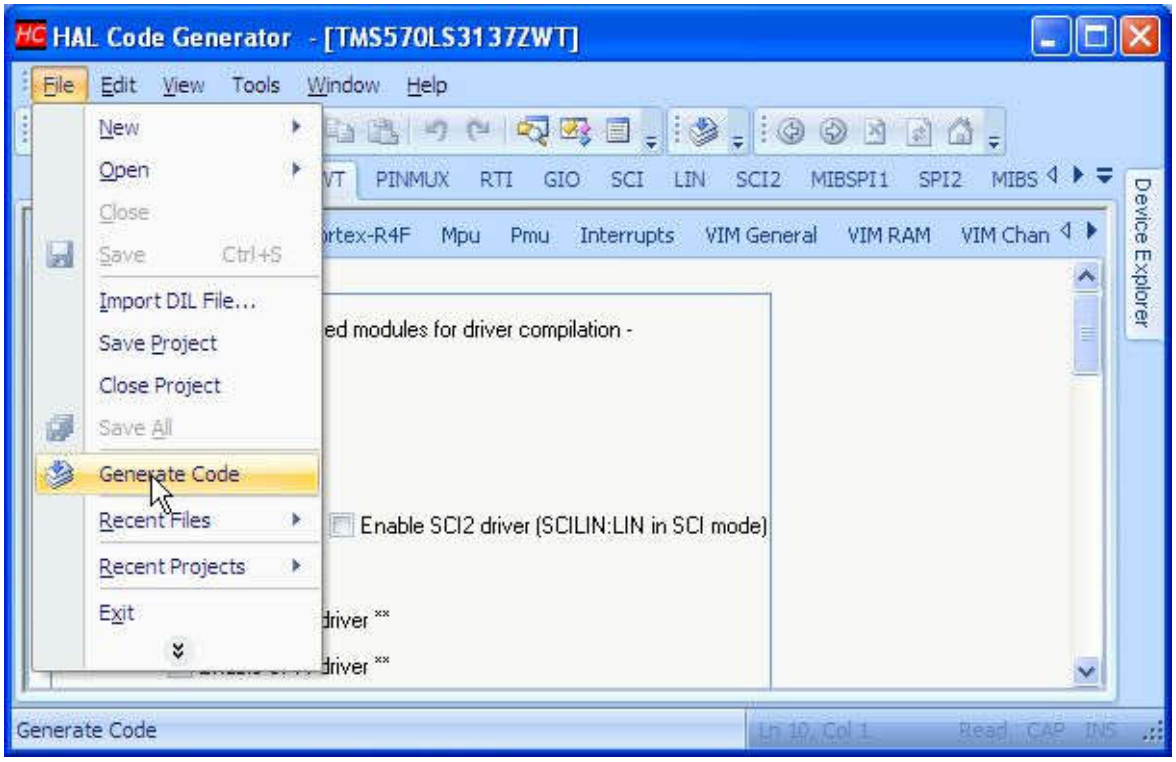

Figure: Click Generate Code

Step 5:

Copy the source code below into your **sys_main.c** (or) replace **sys_main.c** with this file.

The example file can also be found in the examples folder: ../HALCoGen/examples

Note

HALCoGen generates an empty main function in **sys_main.c**,

```
/*
* Copyright (C) 2009-2015 Texas Instruments Incorporated - www.ti.com
*
*
*  Redistribution and use in source and binary forms, with or without
*  modification, are permitted provided that the following conditions
*  are met:
*
*    Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
*    Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the
*    distribution.
*
*    Neither the name of Texas Instruments Incorporated nor the names of
*    its contributors may be used to endorse or promote products derived
*    from this software without specific prior written permission.
*
*  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
*  "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
*  LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
*  A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
*  OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
*  SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES INCLUDING, BUT NOT
*  LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
*  DATA, OR PROFITS; OR BUSINESS INTERRUPTION HOWEVER CAUSED AND ON ANY
*  THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
*  INCLUDING NEGLIGENCE OR OTHERWISE ARISING IN ANY WAY OUT OF THE USE
*  OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*
```

```
*/

/* USER CODE BEGIN (0) */
/* USER CODE END */

/* Include Files */

#include "sys_common.h"
#include "system.h"

/* USER CODE BEGIN (1) */
#include "sci.h"


#define  TSIZE1 10
uint8  TEXT1[TSIZE1]= {'H','E','R','C','U','L','E','S',' ',' '};
#define  TSIZE2 18
uint8  TEXT2[TSIZE2]= {'M','I','C','R','O','C','O','N','T','R','O','L','L','E','R','S',' ',' '};
#define  TSIZE3 19
uint8  TEXT3[TSIZE3]= {'T','E','X','A','S','
        ','I','N','S','T','R','U','M','E','N','T','S','\n','\r'};

void sciDisplayText(sciBASE_t *sci, uint8 *text, uint32 length);
void wait(uint32 time);


#define UART scilinREG
/* USER CODE END */

/* USER CODE BEGIN (2) */
/* USER CODE END */


void main(void)
{
/* USER CODE BEGIN (3) */

    sciInit();      /* initialize sci/sci-lin    */
                    /* even parity , 2 stop bits */

    while(1)        /* continious desplay        */
    {
      sciDisplayText(UART,&TEXT1[0],TSIZE1);   /* send text code 1 */
      sciDisplayText(UART,&TEXT2[0],TSIZE2);   /* send text code 2 */
      sciDisplayText(UART,&TEXT3[0],TSIZE3);   /* send text code 3 */
      wait(200);
    };

/* USER CODE END */
}

/* USER CODE BEGIN (4) */
void sciDisplayText(sciBASE_t *sci, uint8 *text,uint32 length)
{
    while(length--)
    {
        while ((UART->FLR & 0x4) == 4); /* wait until busy */
        sciSendByte(UART,*text++);      /* send out text   */
    };
}


void wait(uint32 time)
{
    time--;
}
/* USER CODE END */
```