STR7 FAMILY

STR71x DEMONSTRATION SOFTWARE

USER MANUAL

Release 0.3

March 2004

USE IN LIFE SUPPORT DEVICES OR SYSTEMS MUST BE EXPRESSLY AUTHORIZED. STMicroelectronics PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF STMicroelectronics. As used herein:

1.Life support devices or systems are those which (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided with the product, can be reasonably expected to result in significant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can reasonably be expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

STR71x Demonstration Software

USER MANUAL

1 INTRODUCTION

This document describes the demonstration software running on the STR71x Demonstration Board MB393, which you can use to evaluate the capabilities of the microcontroller and the available peripherals.

The demonstration software contains two main applications: the general demonstration (stored in the STR71x embedded flash) and the USB demonstration (stored in an external flash).

In this document, the tool set which is used to program the flash memories is the "ARM RealView Development Kit for ST".

Rev. 0.3

March 2004

Table of Contents

2 DEMONSTRATION SOFTWARE FLASHING PROCEDURE

This section explains how to load the General Demonstration Software and the USB Demonstration Software in the internal and external flash.

This description is divided in two parts: the first explains step-by-step how to program the demo software in the STR71x embedded flash. The second part describes how to program the M28W320CB external flash provided with the demonstration board.

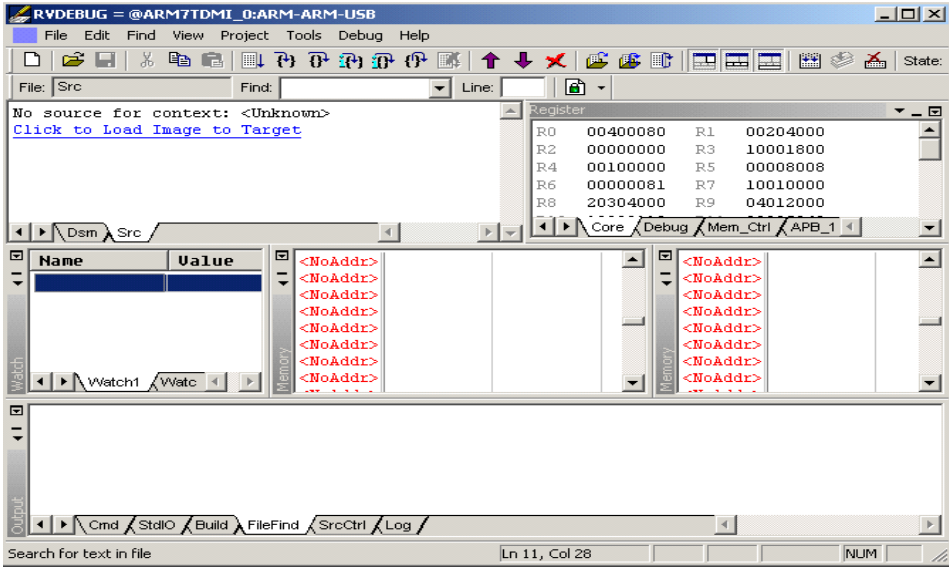
To flash these two memory images, the following AXF files are needed:

"`str71xdemo_gp.axf`", "`str71xdemo_usb.axf`" and "`str71x_extflash.axf`".

2.1 GENERAL DEMONSTRATION SOFTWARE

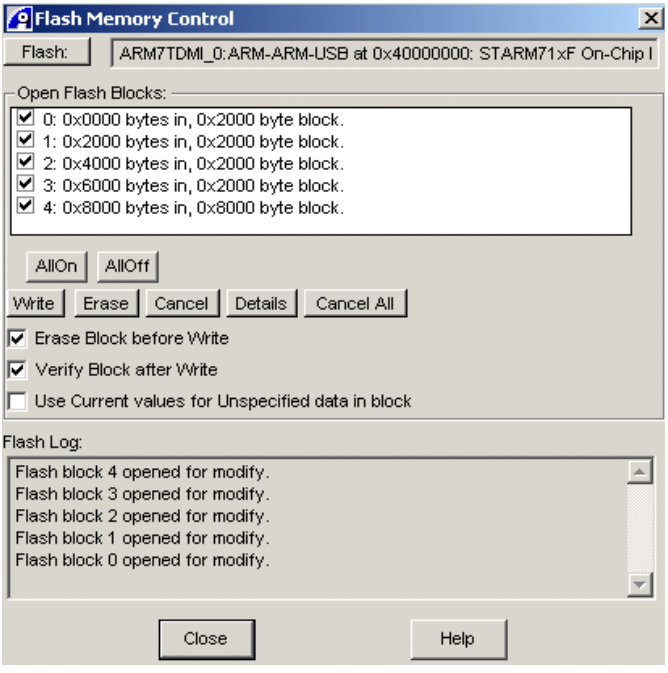
This software has to be programmed in the internal STR71x flash using the integrated flasher provided with the RVDK toolset. Perform the following steps:

1. Open the RVDK Toolset
2. Connect to the Board by selecting **File->Connection->Connect** to target. A Connection Control dialog box is displayed.
3. Select the **STR71x** submenu, then select the **ARM7TDMI_0** Item.
4. "**Click to Load Image to Target**" is displayed in the **Src** window of the Real View debugger editor, as shown on the following figure.



5. Click on this message to display the load **File to Target** dialog box.
6. Select the "`str71xdemo_gp.axf`" file and open it to display the flasher interface shown below.
   If it does not appear, this interface can be also accessed through

Debug->Memory/Register Operation ->Flash Memory Control



- 7. Start to erase the Flash by clicking on the **Erase** button.
  The following messages are displayed on the "**Flash Log**:" part showing the progress of the erase operation.

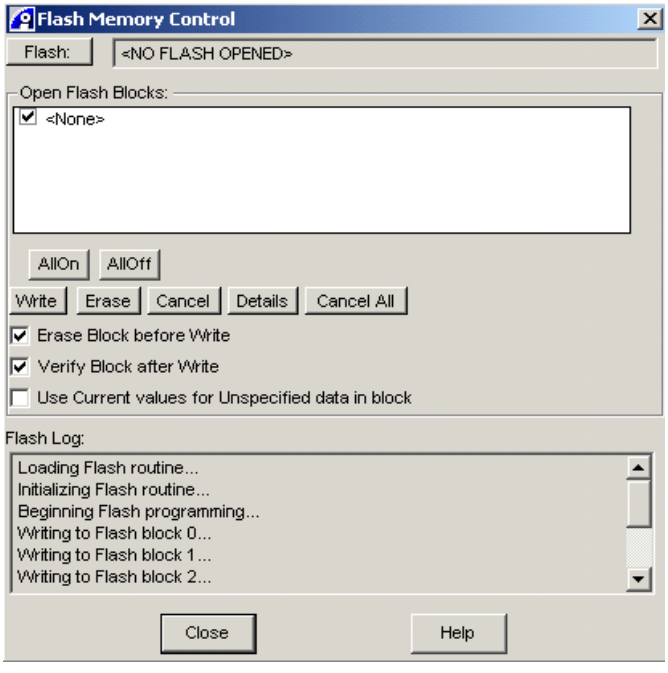
```
"Loading Flash routine...
Initializing Flash routine...
Beginning Flash programming...
Erasing block 0 at address 0x40000000
Erasing block 1 at address 0x40002000
Erasing block 2 at address 0x40004000
Erasing block 3 at address 0x40006000
Erasing block 4 at address 0x40008000
Flash programming complete."
```

- 8. Then click the **Write** button to start writing in the internal flash. The following messages are displayed on the "**Flash Log**:" part showing the progress of the write operation.

```
"Loading Flash routine...
Initializing Flash routine...
Beginning Flash programming...
Writing to Flash block 0...
Writing to Flash block 1...
Writing to Flash block 2...
Writing to Flash block 3...
```

```
Writing to Flash block 4...
Flash programming complete.
Opened Flash closed (last block closed).”
```

Then the Flash Write operation ends and the blocks are closed.



9. Click on the **Close** button. This ends programming of the STR71x embedded flash.

To verify that flashing has worked, change the boot switches SW14, SW13 and SW15 to boot from STR71 embedded Flash and then test the demonstration by navigating using the LCD.

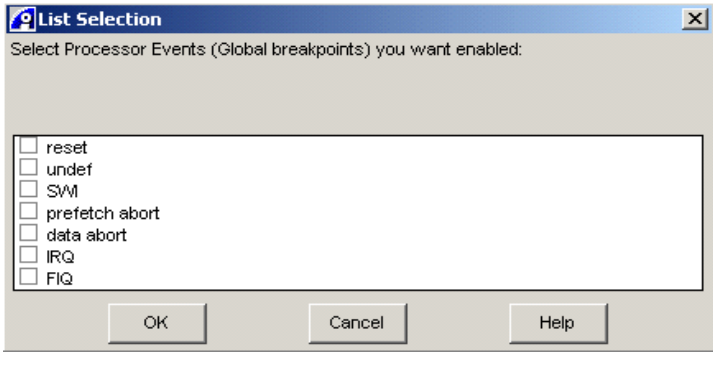
2.2 USB DEMONSTRATION SOFTWARE

To program the external flash, a program is used to copy the USB demonstration image from the internal RAM to the M28W320CB, so the boot configuration jumpers must be set as follows : SW15=2-3, SW13=2-3, SW14=1-2.

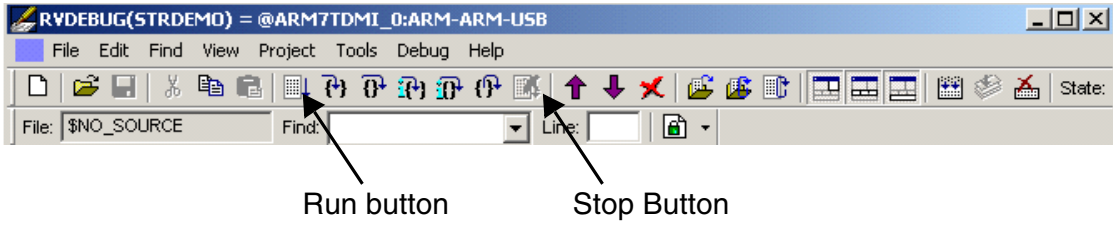
Perform the following steps:

1. Open the RVDK Toolset
2. Connect to the Board by selecting **File->Connection->Connect to target**. A **Connection Control** dialog box is displayed.
3. Select the **STR71x** submenu, then select the **ARM7TDMI_0** Item.
4. The “**Click to Load Image to Target**” message is displayed on the **Src** window of the Real View editor. (Refer to Step4 in Section 2.1)
5. Clicking on this message, a **load File to Target** dialog box is displayed.
6. Select the "`str71xdemo_usb.axf`" file and open it.

8. Close the `Mouse_demo` project using **Project->Close Project**
9. The "**Click to Load Image to Target**" message is displayed on the **Src** window of the Real View editor. (Refer to Step4 in Section 2.1)
10. Click on this message.
11. A **load File to Target** dialog box is displayed (Refer to Step 6 in Section 2.1)
12. Select the "`str71x_extflash.axf`" file and open it.
13. Disable the vector catch by selecting **Debug-> Simple Breakpoint->Processor Events**. A dialog box will be displayed, then uncheck all the checkboxes as shown on the following figure.



- 15. Click on **Run** button on the Realview debugger



Wait until the LEDs (LD3,LD4,LD5,LD6 LD8) on the demonstration board start blinking (after about 15 seconds).

17. Click on the **Stop** button to end the flashing procedure

You can check that the external flash has been correctly programmed by reading the memory at address 0x60000000 with the debugger: it should be set to 0xE59FF018.

To test the progress of the USB demonstration from the external flash you have to change the boot mode to: SW14=2-3, SW13=2-3, SW15=2-3 then press the RESET button. Then plug the USB cable connecting the PC and STR71 demonstration board. The HID device should be recognized by the PC.

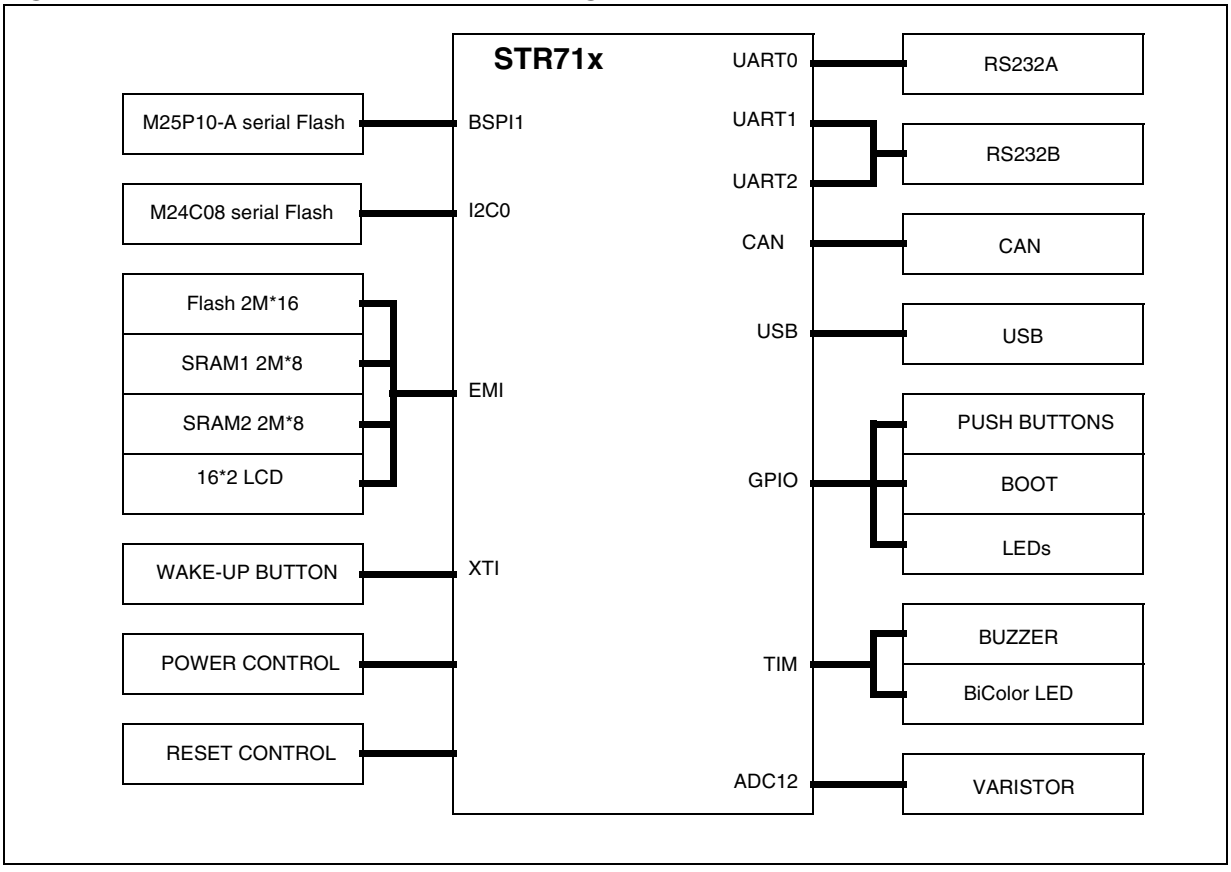
3 HARDWARE FUNCTIONAL DESCRIPTION

The STR71x microcontroller demonstration board (MB393B) provides a development and demonstration platform for the STR71x based application. It is designed to show the major functions of the STR71x microcontroller.

For more details please refer to the board datasheet document.

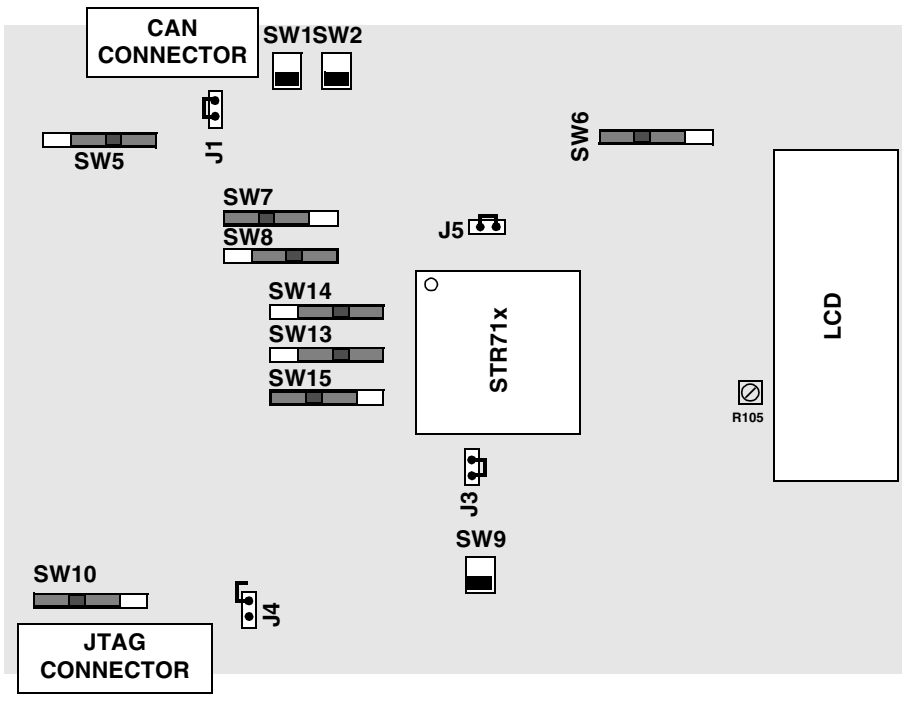
The following block diagram gives an overview of the demonstration board interfaces:

Figure 1. Demonstration Board Block diagram



In order to run the Demonstration Software on the STR71x demonstration board some hardware configurations have to be set up using board jumpers and switches. A typical hardware configuration is described in the following figure:

Figure 2. Board Jumpers and Switches



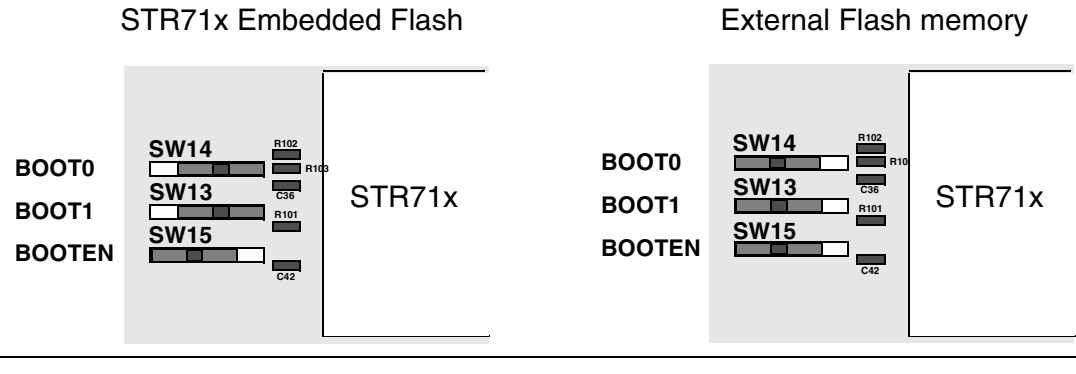
Notes:

– SW13/14/15: these switches are used to select the memory from which the ARM boots the demonstration software. Two modes are available: from embedded Flash (standard demonstration) and from external Flash (USB demonstration). See Figure 3.

– LCD: make sure that the contrast is correctly adjusted with R105 varistor.

– The STR71 demonstration board provides two on-board RS-232 DB9 serial ports that are driven directly by the three embedded UARTs. Each serial port has a DB9 male connector.
UART0 drives RS-232A.
UART1 or UART2 drives RS-232B, depending on the position of switch SW9.

Figure 3. Boot Mode Switches



4 RUNNING THE DEMONSTRATIONS

4.1 SYSTEM INITIALIZATION / BOOT MODES

After reset, the STR71x can optionally boot either from the STR71x RAM, the STR71x internal flash or the external flash connected to EMI Bank 1. This is selected by hardware using three boot configuration switches. The default boot setting is STR71x internal Flash.

Depending on the boot mode, you can choose to execute either the main demonstration or the USB mouse demonstration.

- **Internal flash:** containing the main demonstration software.
- **External flash:** containing the mouse demonstration software.

4.1.1 Menu

1st Level	2nd Level	3rd Level	4th Level
Welcome screen " WELCOME TO THE STR71 DEMONSTRATION"			
Low pw modes	Stand-by	RTC OFF	
		RTC ON	
		Back	
	WFI	TIM	
		RTC	
		Back	
	Run / Slow	RUN 48MHz	
		SLOW /16	
		32KHz	
		Back	
	Back		
RS232 comm	Choose UART	UART0	
		UART1	
		UART2	
	Settings	Baudrate	1200
			2400
			4800
			9600
			19200
			38400
			57600
			115200
		Parity	Even
			Odd
			None
		Stop bits	0.5
			1
			1.5
			2
		Mode	8bit data
			9bit data
			7bit parity
			8b parity
			8b wake-up
		Back	
	Start		
	Stop		
	Back		
CAN	Settings	Bitrate	100K
			125K
			250K
			500K
			1M
		Id Type	Standard
			Extended
		Mode	Ext Loop Back
			Int Loop Back
	Start		
	Back		

1st Level	2nd Level	3rd Level	4th Level
USB Demo			
Play Sound	Sound Select	Music 1	
		Music 2	
		Back	
	Play	Single	
		Single repeat	
		All repeat	
		Back	
	Stop		
	Back		
LEDs adjust	Bicolor	Green	
		Red	
		Back	
	Luminosity		
	Back		
Settings	Date	Adjust	
		Show	
		Back	
	Time	Adjust	
		Show	
		Back	
	Alarm		
	Back		
Board test	Automatic		
	Autotest Report		
	Manual		
	Back		
Data transfer	Data write		
		STR7 RAM	
		SPI EEPROM	
		I2C FLASH	
		EMI FLASH	
		EMI SRAM	
		Back	
	Data read		
		STR7 RAM	
		SPI EEPROM	
		I2C FLASH	
		EMI FLASH	
		EMI SRAM	
		Back	
	Back		

4.1.2 Welcome screen

The following table summarizes which peripherals are used in this software part:

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through the menus)
GPIOs	Blinking the LEDs.
EMI	Driving the LCD.
RTC	Configuring the real time clock for the time and the date.

After the board RESET, a message is displayed indicating the initialization of the demonstration *"Initializing"*.

A scrolling message appears on the top line of the LCD: *"Welcome to the STR71x demonstration"*.

Then the LCD displays date and time as shown on the picture below.

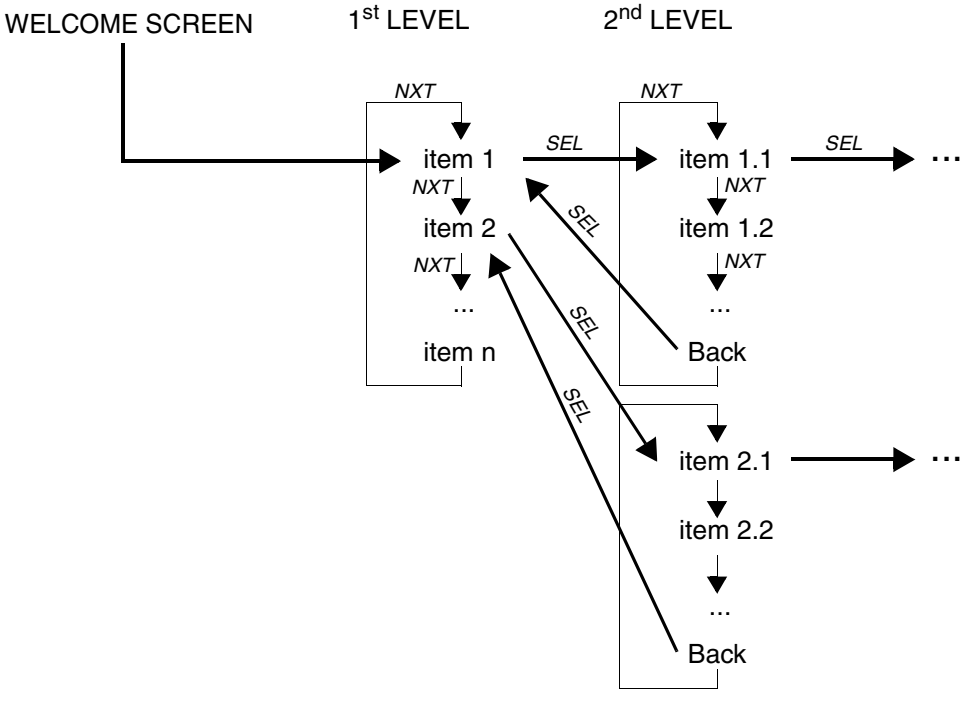
```
STR71x   DD/MM/YY
HH:MM:SS
```

When the board is powered up for the first time, you have to configure the date and time in the Settings menu.

When you press the "NXT" or "SEL" button the application enters the main menu and displays the first menu item *"Low power modes"*.

4.1.3 Navigation

The demonstration menu is based on circular navigation, sub-menu and item selection and back capability as follows:

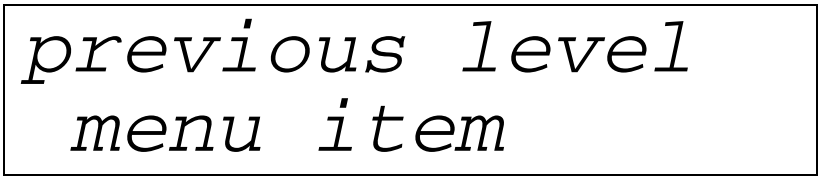


From the 2nd LEVEL, the last item of each sub-menu is *"Back":* it allows you to exit from the current menu level.

You navigate using the “NXT” and “SEL” push buttons located on the demonstration board or using the "N" and "S" keys via the HyperTerminal application.

- “NXT” and “SEL” push buttons: These buttons allow you to the navigate in the demonstration menu. “NXT” button performs a circular navigation in the current menu items, and the “SEL” button selects the current item.
- HyperTerminal navigation: If you connect the board is connected to a PC through the RS232/UART interface and start communication using the *"RS232 Comm"* menu, you can start HyperTerminal and navigate using the PC keyboard. The “N” (or "n") key is equivalent to the "NXT" button and the “S” (or "s") key is equivalent to the "SEL" button. Enabling the serial communication allows you to see application output on the PC screen as well as on the LCD.

When the demonstration menu is activated the following message is displayed on the LCD (and on the HyperTerminal after starting the RS232/UART communication)



If no user input is received for 30 seconds after displaying the menu, a software RESET is performed by the WATCHDOG. When an item is selected, this watchdog feature is disabled.

4.2 APPLICATION

The following part is a detailed description of each part of the demonstration.

Notes:

- This description assumes that the last item in each sub-menu from the 2nd LEVEL to the 4th LEVEL is *"Back"*. This item is always used to return to the previous menu level.
- All messages displayed on the LCD can be also shown on the HyperTerminal if the UART/RS232 communication is enabled.

4.2.1 Low-power modes

The STR71x microcontroller provides various low power modes. The purpose of this menu is to show the behaviour of the microcontroller in low-power mode. Stand-by and Wait For Interrupt modes are taken as examples.

The following table summarizes which peripherals are used in this software part:

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through the menus)
EMI	Driving the LCD.
RTC	Using RTC Alarm to wake-up the system
PCU	Entering Low power mode
RESET	Resetting the system
Wake-up and ITC	Waking-up the system
TIM	Waking-up the system
RTC	Waking-up the system

4.2.1.1 Stand-by

In this mode, the STR71x consumes the least possible power. This mode is entered by a software selection. Stand-by mode has the following characteristics:

– All system clocks are stopped
– All I/O pads are forced to High impedance
– V_{18} power supply is switched off
– A separate power supply is activated for a backup block
– System status monitored by the nSTDBY pad.

Exit from Stand-by mode triggered internally by a RTC wake-up alarm or externally using the wake-up pin. These two exit modes are handled in two sub-menus:

- ***"RTC OFF"*:** when you select this item, the STR71x enters STAND-BY mode without setting any RTC alarm. The following message is displayed on the LCD:

```
STAND-BY RTC=OFF
Press RESET
```

Only an external wakeup or reset can perform an exit from STAND-BY in this case.

- ***"RTC ON"***: When you select this item, the STR71x enables the RTC wake-up Alarm and enters STAND-BY mode. A reset is performed either after 5 seconds using the RTC wake-

up Alarm or using the external wakeup or reset push button. The following message is displayed on the LCD:

```
STAND-BY  RTC=ON
RESET in 5 Secs
```

4.2.1.2 WFI

Wait For Interrupt mode has the following characteristics:

- The CPU is forced into Wait mode
- The CPU clock is software configurable: internal slow clock or external RTC clock
- Integrated peripherals are still running (RAM and register contents are kept). They may generate an interrupt.

Returning from this low-power mode can be triggered by a peripheral interrupt. In this application two exit modes are used in two sub-menus:

- ***"RTC"*:** This item shows how to configure the RTC to exit from WFI mode. The counter displayed on the LCD or on the HyperTerminal shows the number of interrupts generated by the RTC (one interrupt every 5 seconds).

The following message is displayed on the LCD:

```
Wait For Intr
 RTCCount=  0
```

The RTCCount value is the number of RTC interrupts that wake up the system.

- ***"TIM"*:** When you select this item, the STR71 enters WFI mode that and can exit from WFI using the Timer 1 interrupt. The following message is displayed on the LCD:

```
Wait For Intr
 TIMCount=  0
```

The TIMCount value is the number of timer interrupts that wake up the system.

These counters are incremented until you press the "SEL" or the "NXT" keys.

You must hold down the "NXT" key for a few seconds in order to return to the previous menu level.

4.2.1.3 Run/Slow

Selecting this item shows you how to run an application with different clock frequencies. An incrementing counter shows the effect of changing the clock.

- ***"Run 48MHz"***: the application runs at the default frequency 48 MHz. Press the "NXT" button to exit from this mode.
- ***"Slow/16"*:** the application is clocked by Clock2 divided by 16 (Clock2 = 16 MHz). Press the "NXT" button to exit from this mode.
- ***"32KHz":*** the application is clocked by the RTC clock. Hold down the "NXT" key for a few seconds in order to return to the previous menu level.

4.2.2 RS232 communication

The STR71x provides 3 UARTs supporting full duplex, asynchronous, communications with external devices and independently programmable TX and RX baud rates up to 250 kbauds. The purpose of this menu is to show how to use these UARTs to communicate between the demonstration board and a PC. If allows you to configure the baudrate, parity, stop bits and the mode of the selected UART and start the communication. The communication can be performed at 115 kbauds.

The following table summarizes which peripherals are used in this software part :

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through the menus)
EMI	Driving the LCD.
UART0,1,2	Communicating with a PC.

After establishing the communication, you can navigate in the menus and the applications via the HyperTerminal and the PC keyboard in addition to the push buttons ("NXT" and "SEL"). The ‘N’ (or ‘n’) and the ‘S’ (or ‘s’) keys replace the "NXT" and the "SEL" push buttons respectively.

This menu is divided into 4 sub-menus.

4.2.2.1 Choose UART

Selecting this sub-menu allows you to select the UART used to communcate with the PC. By default, UART0 is used.

- ***"UART0"*:** Select the UART0
- ***"UART1"*:** Select UART1. In this case the SW9 switch has to be configured properly (refer to section 2.9.1 for more details).

- ***"UART2"*:** Select UART2. In this case the SW9 switch has to be configured properly (refer to section 2.9.1 for more details).

After selecting the UART, you must select the "Start" item to start communication with the new settings.

4.2.2.2 Settings

In this section, the selected UART has to be configured.

- ***"Baudrate"***

Selects the communication baudrate. The possible values are the following : 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200 baud. The default baudrate value is 9600 baud.

- ***"Parity"***

Selects the communication parity. You can disable the parity feature or choose either odd or even parity. The default parity settings is No parity.

- ***"Stop bits"***

Selects the number of stop bits configured for communication. The possible values are the following : 0.5, 1, 1.5 and 2 bits. The default number of stop bits is 1.

- ***"Mode"***

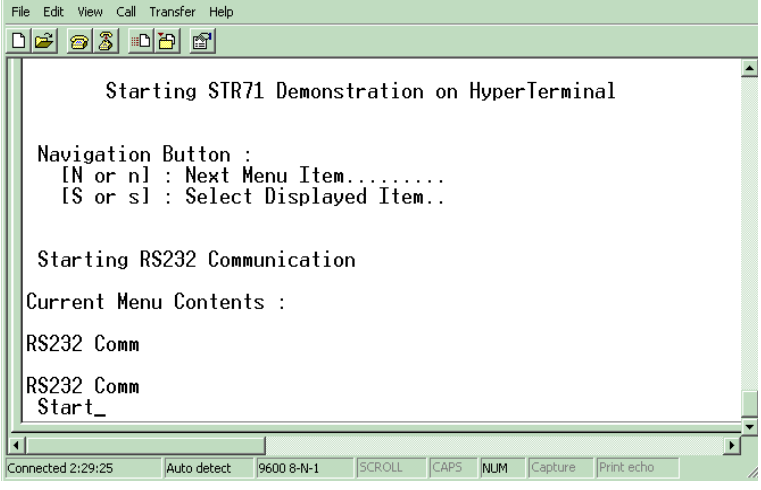
Selects the communication mode from the following :

  – *"8bit data"*: referring to 8 bit data without parity mode.
  – *"9bit data"*: referring to 9 bit data without parity mode.
  – *"7bit parity"*: referring to 7 bit data with parity mode.
  – *"8b parity"*: referring to 8 bit data with parity mode.
  – *"8b wake-up"*: referring to 8 bit data with wakeup bit mode.

The default mode is 8 bit data without parity.

4.2.2.3 Start

This sub-menu is used to enable the communication between the demonstration board and the PC via the selected UART and with the configured settings. After pushing the Select button a prompt message is displayed on the HyperTerminal :



Then, the dispaly on the HyperTerminal is the same as the LCD display. You can use the PC keyboard to navigate through the demonstration menu as described in the Navigation section.

4.2.2.4 Stop

Selecting this item ends the serial communication.

4.2.3 CAN

The STR71 provides a CAN module which can handle, completely autonomously, the transmission and reception of CAN frames in accordance with the CAN specification V2.0 part B (active).

The CAN demonstration is a basic standalone CAN application. It consists of sending and receiving data through the same CAN peripheral using either the internal loopback feature or an external cap connected to the DB9 connector.

The CAN demo menu contains three sub-menus:

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through the menus)
EMI	Driving the LCD.
CAN	Sending and receiving data.

4.2.3.1 Settings

Selecting this menu allows you to configure the CAN communication settings in terms of baudrate, ID type and mode. The CAN default parameters are: 100 kbit/s, standard ID type, internal loopback mode.

This submenu contains two child menus:

- **"Bitrate":** Selects one of the following the bitrates : 100k, 125k, 250k, 500k and 1M bit per second.
- **"Id Type":** Selects the identifier type : standard (11-bit) or extended (29-bit).
- **"Mode":** Selects the CAN transmission mode, you are asked to choose the transmission mode between External loopback (you need to put a cap on the DB CAN connector) and Internal Loopback (no cap needed, the CAN module is running in self-test mode). These modes are selected by *"Ext Loop Back"* or *"Int Loop Back*" items.

4.2.3.2 Start

The application starts sending data through the CAN peripheral, and, due to the loopback (external or internal) all the data sent is received by the same peripheral. This transmission is ended when all the data has been sent or when an error occurs.

During the transmission the following message is displayed on the LCD

```
CAN is active
 nbr messages
```

The "nbr" value is the number of messages that have been sent and received correctly.

If an error occurs during the communication the CAN is stopped and the following message is displayed on the LCD,

```
CAN is stopped
Receive error
```

The CAN demo stops when all the messages are received, or when there is an error (bad ID type, ID value or data contents). There is no time-out handling when the transmission is done in external loopback mode, so the program may wait indefinitely for the receive message if the you forget to put the cap on.

You must press the "NXT" or "SEL" buttons to return to the "Play" Item.

4.2.4 Playing sounds

The STR71 TIM may be used for a variety of purposes. One of these purposes is generating a PWM waveform with pulse lengths and waveform periods that can be modulated from a few microseconds to several milliseconds.

The following table summarizes which peripherals are used in this software part :

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through the menus).
EMI	Driving the LCD.
TIM	Using the timer PWM option (generating PWM wave)
ADC	Acting on the music sound through the potentiometer (Changing PWM duty cycle).

This sub-menu is divided into three items.

4.2.4.1 Selecting Sounds

Two music sounds are saved in a special sector of the SPI serial flash. Through this sub-menu you have to choose one of the two sounds to be played. In this application, the potentiometer is used to configure the sound volume.

- **"Music1":** when selected, music1 is configured to be played after selecting the "Play" item.
- **"Music2":** when selected, music2 is configured to be played after selecting the "Play" item.
- **"Play":** you can play either a single sequence or in a continuous mode while other demonstration features are running.
  - *"Single"*: The selected music is run only once.
  - *"Single repeat"*: The selected music is repeated continuously until you select the Stop sub-menu.
  - *"All repeat"* : The two provided Music (Music 1 and Music2) are repeated continuously until you select the Stop sub-menu.
- **"Stop":** Stops the playing sound.

4.2.5 Adjusting the LEDs

The following table summarizes which peripherals are used in this software part :

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating in menu).
EMI	Driving the LCD.
TIM	Using the timer PWM option (generating PWM wave)
ADC	Changing the bicolor led color through the potentiometer

The LEDs adjust menu controls the bicolor LED. This menu is divided in two sub-menus.

4.2.5.1 Bicolor Selection

This allows you to select the color of the bicolor LED. This sub-menu is divided in two items:

- ***"Green"***: Selecting green color.
- ***"Red"***: Selecting red color.

4.2.5.2 Luminosity

You can vary intensity of the bicolor LED using this sub-menu. Control is done by modulating the PWM waveform pulse lengths generated by the TIM timer. This is done by acting on the potentiometer linked to the ADC analog pin. The color of the LED varies depending on the analog voltage adjusted by the potentiometer. It will change from red to green.

4.2.6 Settings

This sub-menu is used to configure some miscellaneous functions such as the time, date and alarm.

The following table summarizes which peripherals are used in this software part :

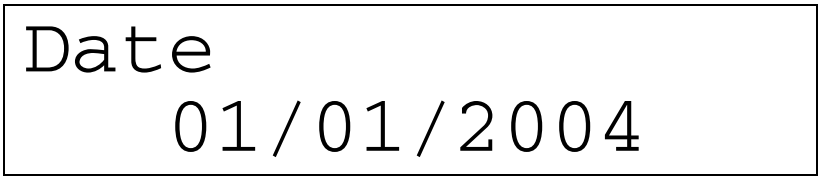
STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through the menus).
EMI	Driving the LCD.
RTC	Setting the real time clock
ADC12	Adjusting the digit value through the potentiometer

4.2.6.1 Date

The STR71 provides a Real Time Clock (RTC) which has a set of continuously running counters that can be used, with suitable software, to implement a clock-calendar function. The counter values can be written to set the current time/date of the system. This sub-menu is divided in two items.

- ***"Adjust"*:** You have to select this item at least once after each power-up in order to set the current date. You are asked to enter the current date to be stored in the application memory. The date is displayed on 8 digits: MM/DD/YYYY. 01/01/2004 is displayed when you enter this menu for the first time after power-up. The first digit of the month field is ready to be changed. Pressing "NXT" button will display the actual value plus one. To change the digit value, press "NXT". After choosing the digit value, press "SEL", the cursor jumps automatically to the next digit. When all the date digits are set, you need to press the "SEL" button to exit from the date sub menu. Some digit values are limited depending on the field (month, day or year). When the board is connected to the PC via a serial cable, you can set the value via the terminal using a PC keyboard corresponding to the "SEL" and "NXT" buttons.

- ***"Show"*:** Selecting this item allows you to see the current date. The default date displayed after power up and, before using the Adjust item application, is 01/01/2004. The following message is displayed on the LCD the first time you power-up:



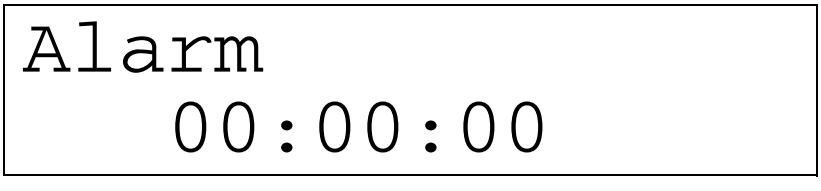
4.2.6.2 Time

This sub-menu is divided in two items allowing the user to view or to adjust the current time**.**

- **Adjust:** You have to select this sub-menu mainly after a power up in order to correct the time value which is by default 00:00:00. You are asked to fill the current time to be stored in the application memory. The first digit of the hour field is ready to be changed. Pressing the "NXT" button will display the current value plus one. To change the digit value, press "NXT". To choose the digit value press SEL, the cursor jumps automatically to the next digit. When all the time digits are set, press the "SEL" button to exit from the *"Time"* sub menu. Some digits values are limited depending on the field (hours, minutes or seconds). When the board is connected to the PC via a serial cable, you can set the value via the terminal using a PC keyboard.
- **Show:** Select this item to view the current time. The default time displayed after power up and before using the Adjust item application is 00:00:00. The following message is displayed on the LCD the first time you power-up:

4.2.6.3 Alarm

- You can use this item to configure the time at which an alarm can be activated. A signal is emitted when you reach an appointment. The signal used is the Music 2 sound from the Play Sound menu. The following message is displayed on the LCD the first time you power-up:



Refer to the Time paragraph for details on how to set the hours, minutes and seconds.

4.2.7 Board test

You can choose to start the board test. This test consists of checking the different components of the board and the peripherals of the STR71x device.

The following table summarizes which peripherals are used in this software part :

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through menus).
EMI	Driving the LCD.
EMI	Communicating with the external memories.
TIM	Using the timer PWM option (generating PWM wave)
ADC	Getting voltage value through the varistor.
PCU	Entering Stand-By low power mode.
Wake-up	Resetting the system.
UART	Communicatiing with a PC.

The board test demo is made of the following sub-menus:

4.2.7.1 Automatic

The first board test sub-menu is the automatic test; when selected an automatic test procedure is started. In this mode, not all the features and the peripherals are tested, it performs checks which need no action from the board user. These tests are referred by:

"STR7 RAM": Testing Read and write operations in the STR7 embedded RAM.

"STR7 BSPI": Testing Read and write operation in the SPI serial flash.

"STR7 I2C": Testing Read and write operation in the I2C EEPROM.

"EMI Flash": Testing Read and write operation in the External Flash mapped on the EMI.

"EMI SRAM": Testing Read and write operation in the External SRAM mapped on the EMI.

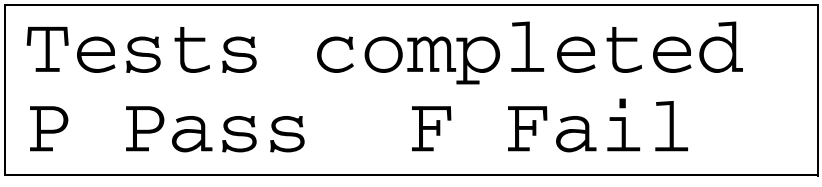
"CAN" :Testing send and receive of a single data frame with ID in internal loopback mode

During the test the following information is displayed on the LCD

At the End of each test, the test result is shown at the last Character in the second LCD line

R can be either "P" for Passed test or "F" for Failed Test .

At the end of all these automatic tests a message is displayed indicating the number of passed test (P) and the number of the failed test (F). The following message is displayed on the LCD.

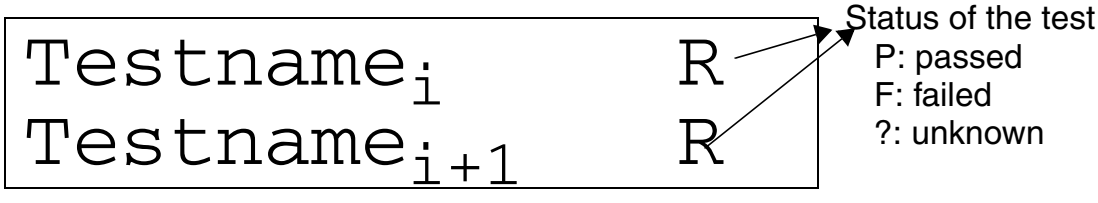


All these tests result are saved and can be shown when selecting the *"Autotest Report"* item on the *"Board test"* menu

4.2.7.2 Autotest Report

This sub-menu allows you to view all the automatic test results.

After selecting this sub-menu, the report display starts on the LCD:



Push "NXT" to show the following test status.

Push "SEL" to exit from this report and return to the *"AutotestReport"* item.

If UART communication is active, the report is displayed on the HyperTerminal at the same time.

4.2.7.3 Manual Test

The second board test sub-menu is the manual test. An interactive test is started allowing you test addtional features. A report is also generated at the end of the test. The manual test is divided in 3 parts (LEDs and Buzzer, UART connection and Wake-up).The demonstration allows you to select or to ignore each test.

LEDs and Buzzer

After selecting this Sub-menu the following message will appear on the LCD.

```
LEDs and Buzzer
SEL:ok NXT:skip
```

You can select either option:

- If you push the "SEL" Button, the "LEDs and Buzzer" test is selected. This test has three parts:
  - Testing the 16 LEDs: you are asked to adjust the varistor. The following message is displayed on the first line of the LCD to indicate the test is ongoing: *"Adjust varistor to light on/off".* All the 16 LEDs of the demo board are either Switched on or off.They are switched on if a lower voltage is provided to the ADC through the varistor. They are switched off in the opposite case. To proceed to the next part of the test, push the "NXT" or "SEL" buttons.
  - Testing the Bicolor LED (red and green): You are asked to adjust the varistor. The following message is displayed on the first line of the LCD : *"Adjust varistor to change color".* The bicolor LED is either Red or Green. Green is selected if a lower voltage is provided to the ADC through the varistor. Red is selected in the opposite case. To proceed to the next part of the test, push the "NXT" or "SEL" button.
  - Testing the modulation of the sound: You are asked to adjust the varistor in order to modulate the sound. The following message is displayed on the first line of the LCD to indicate the test is in progress *"Adj. varistor to modulate sound".* To pass to the next test *"UART connection"* push the the "NXT" or "SEL" button.
- Pushing "NXT": The current test is ignored and you go to the next test *"UART connection"*

UART connection

After exiting from the previous test the following message is displayed on the LCD:

```
UART connection
SEL:ok NXT:skip
```

The current test is used to test the communication between the PC and the demonstration board through the UART

You can select either option:

- Pushing "SEL" Button: The "UART connection" test is started by displaying a message on the LCD asking you to connect UART0 to the serial port of a PC and to configure a HyperTerminal communication at 9600 baud, 8 bits, no parity and 1 stop bit. After setting this

configuration, press the "NXT" or "SEL" button. Then a message is displayed on the HyperTerminal to confirm the successful result of this test. A message is sent to the Hyperterminal, asking you to send back a string to the board and press Return to close the connection. The user entry is echoed on the LCD screen. Then push the SEL or the NXT button to proceed to the "*Wake-up*" test.

- Pushing "NXT": The current test is ignored and pass to the following test *"Wake-up"*

Wake-up Test

This test checks the operation of Wake-Up from Stand-By low power mode. This test is usually ended by resetting the system.

After finishing the previous test or skipping it, the following message is displayed on the LCD to indicate the wake-up part on the manual test:

```
Wake-up mode
SEL:ok NXT:skip
```

You can select either option:

- Pushing "SEL" Button: The "wake-up" test is selected and the system enters Stand-By low power mode. You are is asked to press the wake-up button to exit this mode. A reset is performed after you push the wake-up button, and the demonstration is restarted.
- Pushing "NXT": The current test is ignored and you return to the "Manual" Sub-menu.

4.2.8 Data transfer

The following table summarizes which peripherals are used in this software part :

STR71x IP used	Purpose
GPIOs	NXT/SEL button (navigating through the menus)
EMI	Driving the LCD.
I2C0	Communicatiing with I2C serial flash.
BSPI1	Communicatiing with SPI serial flash.
EMI	Communicating with External memory.
ADC	Adjusting the Address or the Data value

This application allows you to perform read & write operations in the different memory areas on the demonstration board or on the STR71x device. You can access the STR71x embedded flash, STR71x embedded SRAM, I2C serial EEPROM, SPI serial Flash, EMI Flash and the EMI SRAM. The various operations are allowed in only in special address locations of these memories because the some locations are used by other applications.

The following table summarizes the authorized address ranges:.

Memory	Start address	End address
STR71x embedded RAM	0x20004000	0x2000CFFF
I2C EEPROM	0x00	0xFF
SPI Flash	0x18000	0x1FFFF
EMI Flash	0x60068000	0x6006FFFF
EMI SRAM	0x62000000	0x623FFFFF

This sub-menu is divided in two main items:

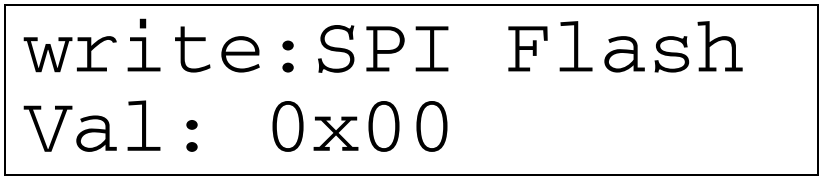
- Data write: Using this application you can write data in the different memories. You are asked to choose the target memory:
  - STR71 embedded Flash
  - STR71 embedded RAM
  - I2C EEPROM
  - SPI Flash
  - EMI Flash
  - EMI SRAM

After selecting the target memory you are asked to enter the address. Depending on the memory type the address or the data can be 8 or 20 or 32 bits wide. For example in the case of the SPI flash, the address is displayed 4 digits:

```
write:SPI Flash
Add: 0x00000
```

The first time you are asked to enter the address value, then the default data value, 0x0 is displayed (the number of digits depends on the memory type). The first digit of the address field is ready to be changed and the cursor blinks on this digit. Pressing "NXT" button will display the current value got from the ADC. Change the value of the digit by rotating the potentiometer then press "NXT" to confirm your choice. The cursor blinks and jumps automatically to the next digit. When all the address digits are set, you need to press the "SEL" button. Then the

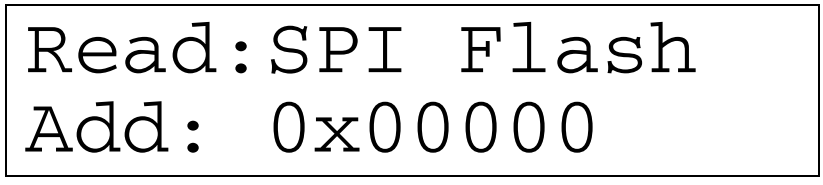
you are asked to enter the data to be written on the selected address and the following message is displayed on the LCD:



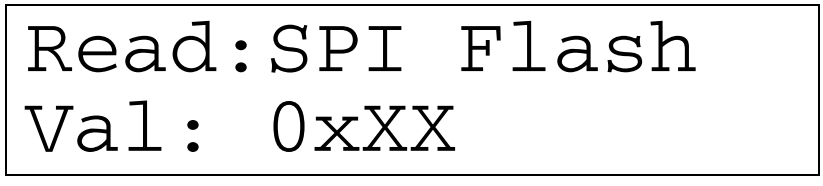
After finishing data acquisition with the same method as the address acquisition, press "SEL" to return to the previous menu level.

- **Data Read:** Through this application you can read data from the various types of memory. You are asked to choose the target memory like for the data write.

After selecting the target memory you are asked to enter the address of the data to be read. Depending on the memory type the address can be 8 or 20 or 32 bits wide. As an example, in the case of the SPI flash, the address is displayed as 5 digits:



When you are asked to enter the address value, 0x0 is displayed as default value. The first digit of the address field is ready to be changed. Pressing "NXT button will display the current value plus one. Choose the value of the digit using the potentiometer then press "NXT" to confirm the choice. The cursor jumps automatically to the next digit. When all the digits are set, you need to press the "SEL" button to confirm the address. Then the data value is displayed on the LCD:



Push the "SEL"or "NXT" buttons to exit from this sub-menu.

4.2.9 USB Demo

The mouse demonstration can be considered as a sample application of the USB. This application is located in the external flash so you have to reboot from the external flash. This configuration is done by changing the boot switches which are configured to boot from the external flash by default (Refer to Section 2.2 for more details).

When this sub-menu is selected the following message is displayed on the LCD and the HyperTerminal “change boot pin and reset”.

After this reset the USB demonstration is started. Besides the USB peripheral, the demonstration uses the UART0 to communicate with the PC through a serial cable.

This demonstration is a USB application using an endpoint programmed as interrupt type providing a channel to a PC for reading the status of a virtual HID mouse. The Mouse demonstration consists of emulating left, right, up and down keys by means of a PC terminal connected via a serial cable to the RS232A in the board. Each possible event is associated with a key. When the main program detects a valid key coming from the serial cable, it updates 3 cells related to the mouse status. The content of these cells is copied into the transmission buffer of the endpoint.

The PC during the enumeration phase has detected the presence of the mouse device and it has loaded the associated software driver and from this moment it periodically polls the endpoint with IN tokens. The new ready data are sent after the first coming IN.

The first time the device is connected to computer the driver of this new hardware is loaded. The related Windows driver is: USB Human Interface Device

To test this mouse tutorial, a minimum configuration is required:

- Connect a PC serial port to the UART0 of the STR71x (RS232A of the demonstration board)
- On the PC, run a terminal emulation program (i.e. Hyperterminal) configured with the following parameters:
  - baud rate = 9600
  - data bits = 8
  - parity = no
  - stop bits = 1
  - flow control = no

Mouse motion events are emulated with the following PC keyboard keys:

[a] = right.

[d] = left.

[w] = up.

[x] = down.

To exit the Mouse demonstration, you should reboot from the internal flash. For this reason, you have to change the boot switches and reset the board. The system boots from the flash, and the general demonstration is started.

THE SOFTWARE INCLUDED IN THIS MANUAL IS FOR GUIDANCE ONLY. STMICROELECTRONICS SHALL NOT BE HELD LIABLE FOR ANY DIRECT, INDIRECT OR CONSEQUENTIAL DAMAGES WITH RESPECT TO ANY CLAIMS ARISING FROM USE OF THE SOFTWARE.