ST622x Starter Kit
User Manual

for ST620x, ST621x and ST622x

Release 3.0

November 2000

Ref: DOC-E-ST622XSKUM

www.BDTIC.com/ST

Table of Contents

www.BDTIC.com/ST

Table of Contents

1   INTRODUCTION

The ST622x Starter Kit provides you with all you need to start designing, developing and evaluating programs for ST620x, ST621x and ST622x microcontrollers immediately.

Note: *If you come across any terms or abbreviations you do not understand, you can check their meaning in the Glossary on page 47.*

1.1   Overview

The ST622x Starter Kit includes:

- The ST6 Starter Kit board, which serves as a demonstration board and low-cost debugging tool.

- Two ST62E25, two ST62E20 and one ST62E01 microcontrollers.

- The *MCU on CD* CD-ROM which contains the following:

  - The ST6 assembler and linker, AST6 and LST6.

  - The ST6 Windows debugger, WGDB6.

  - The Windows ST6 microcontroller programmer, Epromer.

  - Some demonstration programs that show how ST6 microcontrollers use the Starter Kit board resources.

  - Some example programs of typical ST6 applications for the ST622x family.

- A complete set of paper documentation and online help.

The demonstration programs, that come pre-loaded on an ST62E25 microcontroller, show how the powerful features of ST6 microcontrollers operate in a real environment. The demonstration programs use the hardware resources provided on the Starter Kit board, which include reset and data control buttons, LED indicators, a resistance trimmer, a temperature control circuit and an RS-232 interface. The source files of these demos are also included on the *MCU on CD* CD-ROM and are installed by default in the following directory:

    C:\Program Files\Stm\ST6toolchain\StarterKit\sk622xli

Using the ST6 assembler and linker, AST6 and LST6, you can assemble and link ST6 programs. The AST6/LST6 User Manual will guide you through the steps of developing, assembling and linking programs for ST6 microcontrollers.

The Starter Kit software includes a set of example programs of typical ST6 applications. These example programs are included in the *MCU on CD* CD-ROM and are installed by default in the following directory:

C:\Program Files\Stm\ST6toolchain\StarterKit\Samples\exos

Once you have developed your ST6 program, you can use the ST6 Windows debugger, WGDB6, together with the Starter Kit board, as a low-cost but powerful debugging tool. WGDB6 includes an ST6 simulator, that simulates the execution of your program, and uses the ST6 that is plugged into the Starter Kit board to emulate all transactions that are performed with the data space. Thus, using the Starter Kit board with WGDB6, you can view how the microcontroller peripherals behave when your program is executed. WGDB6 includes powerful debugging features, such as source-level debugging, instruction and conditional memory access breakpoints and selective trace recording. The *WGDB6 Getting Started* manual and online help will lead you through the debugging process using WGDB6.

In addition to the ST WGDB6 debugger, Raisonance's RIDE-ST6 Windows®-compatible software package is supported by the ST6 Family Starter Kits. Raisonance's software suite is an **integrated development environment** that includes a highly optimized ANSI C Compiler, a powerful Macro-Assembler, a Linker and a fully integrated Simulator/Debugger. The advantage of Raisonance's software suite is that everything you need to evaluate, debug, edit and re-compile your application is integrated in a single, consistent and user-friendly interface. This software is included in the *MCU-on-CD* CD-ROM provided with your emulator kit, or can be downloaded from Raisonance's web site at:

http://www.raisonance.com

Once your program is ready, Epromer provides you with an easy-to-use Windows® interface, which lets you prepare executable code, then write it to the ST6 microcontroller that is plugged into either one of the DIL sockets on the Starter Kit board, or your own in-circuit application board that is connected to the Starter Kit board.

To observe and evaluate the consequences of your program on the resources it controls, you can run it on an ST6 microcontroller that is plugged into the Starter Kit board. If it controls a resource that is not included on the Starter Kit board, you can connect your own resource to the board.

1.2   Possible Uses of the Starter Kit

The following diagram summarizes the possible uses of the Starter Kit board and the hardware setup required for each one.

To program ST6 MCUs on your own in-circuit programming board:

PC running
Epromer

Starter Kit
board

In-circuit
programming board

Parallel port
connected to P1

J1 connector

To program ST6 MCUs using the Starter Kit board:

PC running
Epromer

ST6 MCU to be
programmed

Parallel port
connected to P1

Starter Kit board

To run the demonstrations:

ST6 MCU Programmed with
DEMOK25.HEX

Starter Kit board

To run your own program:

ST6 MCU programmed
with program to run

To use the Starter Kit board as a software simulator:

PC running
WGDB6 Simulator

ST6 MCU Programmed with
DEMOK25.HEX

Parallel port
connected to P1

www.BDTIC.com/ST

1.3   Where to go from here...

The following table directs you to where you should look for further information concerning how to use the ST6 Starter Kit.

To:	Refer to:
Install the Starter Kit software, and connect the power supply to the board.	*Chapter 2: Getting Started* on page 11 of this book.
Find out about the Starter Kit board and ST6 microcontrollers provided with the kit.	*Chapter 3: Hardware Features* on page 15 of this book.
Find out what the demonstration applications do, and run them.	*Chapter 4: Running the Demos* on page 25 of this book.
Connect your own hardware resource to the Starter Kit board.	*Chapter 5: Connecting External Resources to the Starter Kit Board* on page 31 of this book.
Learn how to develop source code for AST6 and LST6.	*AST6/LST6 Assembler and Linker for the ST6 Family -User Guide.*
Install WGDB6, and learn how to use it for debugging your programs.	*WGDB6 Windows Debugger for the ST6 Family - Getting Started.*
Prepare the Starter Kit board for use as an ST6 hardware simulator with WGDB6.	*Chapter 6: Using The Starter Kit Board as a Hardware Simulator* on page 33 of this book.
Learn how to use Epromer for programming ST6 microcontrollers.	The Epromer online help.
Prepare the starter kit board for programming ST6 microcontrollers using Epromer.	*Chapter 7: Programming ST6 Microcontrollers* on page 37 of this book.
Prepare the Starter Kit board for connecting your own in-circuit programming board.	*Section 7.5: Setting Up the Starter Kit Board for In-Circuit Programming* on page 42 of this book.
Run your own program on an ST6 microcontroller using the Starter Kit board.	*Chapter 8: Running Your Own program on the Starter Kit Board* on page 45.

1.4 Getting Assistance

For more information, application notes, FAQs and software updates on all the ST microcontroller families, check out the CD-ROM or our website:

http://mcu.st.com

For assistance on all ST microcontroller subjects, or if you need help with using your emulator, refer to *Product Support* on page 49. We'll be glad to help you!

1.5 Instructions for use - Warning

This product conforms with the 89/336/EEC directive; it also complies with the EN55022 emissions standard for ITE, as well as with generic 50082-1 immunity standards.

The product is a **Class A apparatus**. In a residential environment this device may cause radioelectrical disturbances which may require that the user adopt appropriate precautions.

The product is not contained in an outer casing, and cannot therefore be immune against electrostatic discharge (ESD): **it should therefore only be handled at static safe work stations**.

2   GETTING STARTED

2.1   Your System Requirements

To run/install the software provided on the *MCU on CD* CD-ROM, you must have a PC running either Microsoft® Windows® 95, Windows® 98 or Windows® NT®.

2.2   Installing the Hardware

### 2.2.1   Connecting the Power Supply

If you have an AC mains supply, connect the jack plug on the power supply cable provided to the J3 input socket, then connect the mains plug to a mains source.

If you have DC mains supply, connect the male plug on the power supply cable provided to the J3 input socket, then connect the mains plug to a mains source with the following characteristics:

- Voltage: 16 V min./20 V max.

- Current: 100 mA min.

Note: *To avoid a short circuit, always connect the power input cable to the starter kit board before connecting it to a mains power supply.*

If you use your own 3.5 mm power supply plug, its polarity must be as follows:
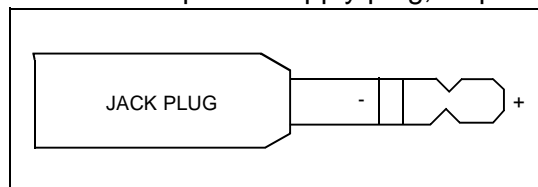
2.3   Installing WGDB6 and other ST6 Software Tools

Your starter kit comes with the *MCU on CD* CD-ROM which contains a number of ST6 software tools. To install them, follow these steps:

1   Place the *MCU on CD* CD-ROM in your CD-ROM drive. The CD-ROM's autorun feature opens up a welcome screen on your PC.

   If the autorun feature does not work, use Windows® Explorer to browse to the CD-ROM's root folder, and double-click on welcome.exe.

2   Select ***Install Your Development Tools*** from the list of options. A new screen appears listing the different families of STMicroelectronics MCUs.

3   Use your mouse to place the cursor over the ***ST6 TOOLS*** option. Choose ***ST TOOLS*** and ***ST6 TOOLCHAIN*** from the lists that appear.

4   The install wizard is launched. Follow the instructions that appear on the screen.

You can choose the package you wish to install. To install the complete ST6 Toolchain for your starter kit, select **Complete Toolchain for Starter Kit**. This option installs the WGDB6 debugger version for your starter kit, as well as a Windows Epromer and ST6 Assembler-Linker software.

Alternatively, you can choose to perform a custom installation where you choose which of the available software applications you wish to install.

Note: *In order to configure your starter kit, you must, as a minimum, install the **ST6 WGDB6 for Starter Kit***
*If you do not choose any options, but click **Next>**, the ST6 Assembler-Linker will be installed by default.*

5   Follow the instructions that appear on your screen. You will be prompted to select the parallel port you wish to connect the emulator to, as well as the program folder that the software will be installed to.

6   If you are installing WGDB6 on a Windows® NT® platform, you must install the Windows® NT® parallel port driver supplied on the CD-ROM.
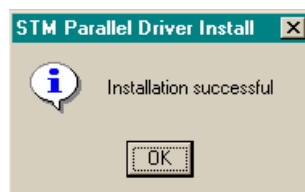
A window pops up if you have not already installed this driver (***parstm.sys***).
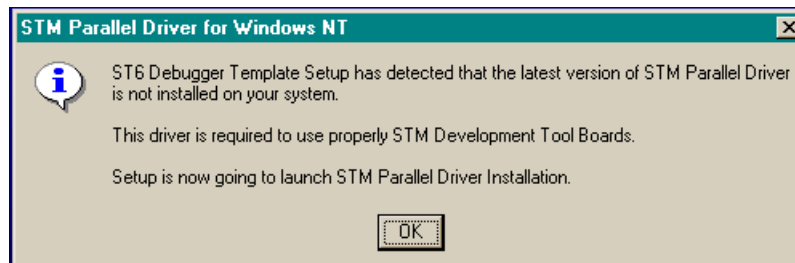
7   Click **OK**. The following window appears:
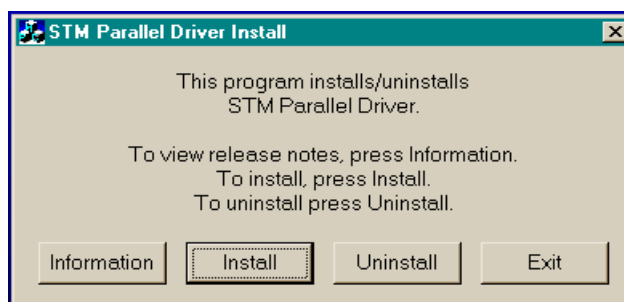
8   Click **Install**. The following window appears:

9   Click **OK**, the installation is now complete. The following window appears.
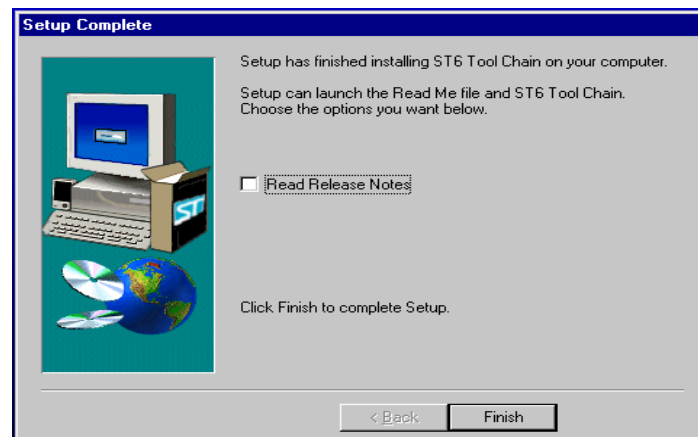
10 Choose to read the **Release Notes** or start WGDB6, then click the **Finish** button.

2.4   Launching WGDB6

From Windows$^{®}$ 95, 98 or Windows$^{®}$ NT, click the **Start** button, point to **Programs** -> **ST6 Tool Chain -> Development Tools**, then choose **WGDB6 StarterKit**.

Refer to the *WGDB6 Windows Debugger for the ST6 Family User Manual* for full instructions on how to use WGDB6. This manual is available in PDF format on the *MCU on CD* CD-ROM.

www.BDTIC.com/ST

3 HARDWARE FEATURES

This section describes the ST6 microcontrollers and the Starter Kit board that come with the ST6 Starter Kit. *Figure 3* on page 18 shows a full schematic of the Starter Kit board.

3.1 The ST6 Microcontrollers

The Starter Kit includes two ST62E25, two ST62E20 and one ST62E01 microcontrollers.

The ST62E25 microcontroller labelled DEMOK25 is pre-loaded with the **Demok25.hex** code, which includes the demonstration programs (see *Chapter 4: Running the Demos* on page 25), as well as the communications protocol program, that enables you to use the Starter Kit board as a simulator (see *Chapter 6: Using The Starter Kit Board as a Hardware Simulator* on page 33). The other ST6 microcontrollers are blank.

The **Demok25.asm** file is also provided on the *MCU on CD* CD-ROM and is installed in the **...\ST6toolchain\StarterKit\sk622xli** directory, so that if you erase it from the ST62E25, you can re-program it following the instructions given in *Chapter 7: Programming ST6 Microcontrollers* on page 37.

3.2 The Starter Kit Board

The Starter Kit board includes the following resources (described in more detail later in this chapter):

- Oscillator.
- Reset button.
- Data control buttons.
- LED indicators.
- Resistance trimmer.
- Temperature control circuit.
- RS-232 interface.
- Demonstration program selector jumpers.

It comes with its own power supply unit that can be plugged into an AC mains source, or a DC source with the following characteristics:

- Voltage: 16 V min./20 V max., Current: 100 mA min.

It includes the following connectors:

- A parallel port connector (P1) for connection to the host PC when it is used as a hardware simulator or for programming.

- A remote resource I/O interface connector (J2) to which you can connect your own hardware resource.

- An RS-232 connector, which you can use for observing RS-232 communication control using an ST6.

- A connector for your own in-circuit ST6 programming board. See *Application Board Connections* on page 40 for further details.

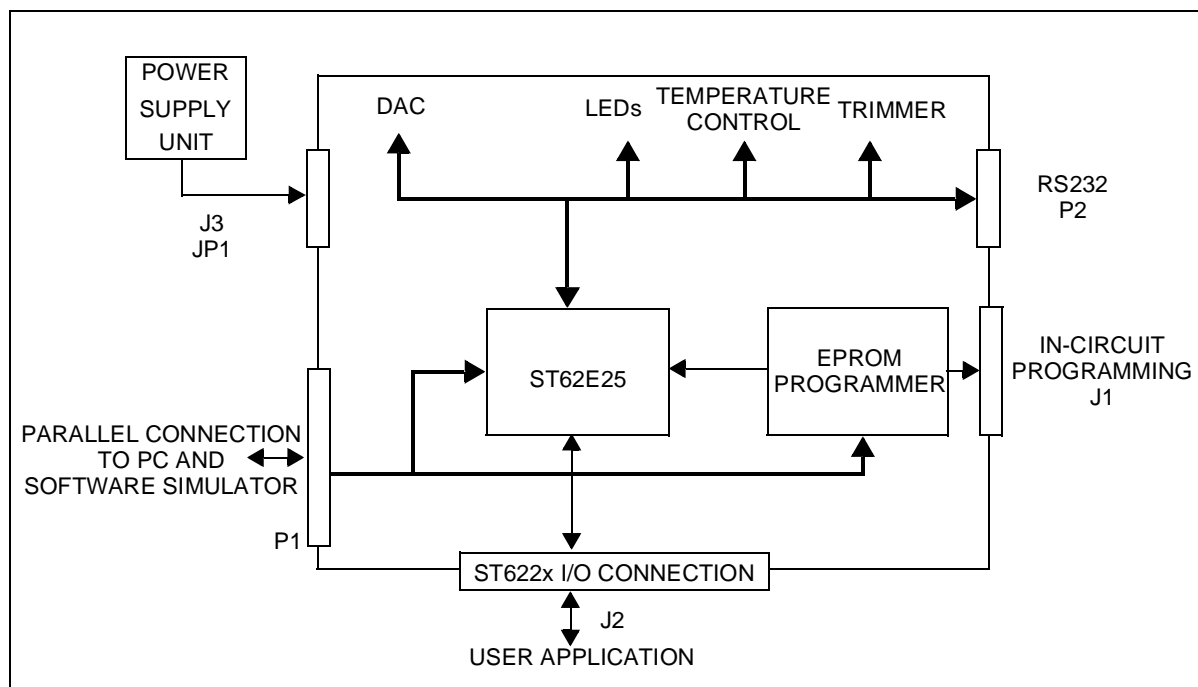A block diagram of the Starter Kit board is shown in *Figure 1.*.

Figure 1: Block diagram of Starter Kit board
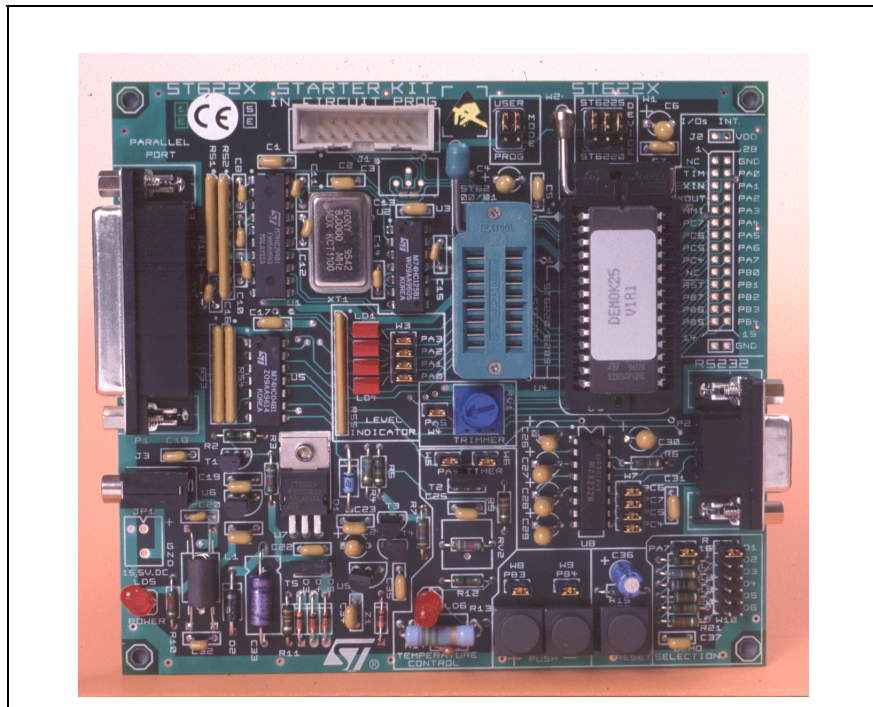
A photo of the Starter Kit board is shown in *Figure 2*.

Figure 2: Starter Kit board

The layout of the Starter Kit board is shown in *Figure 3*.
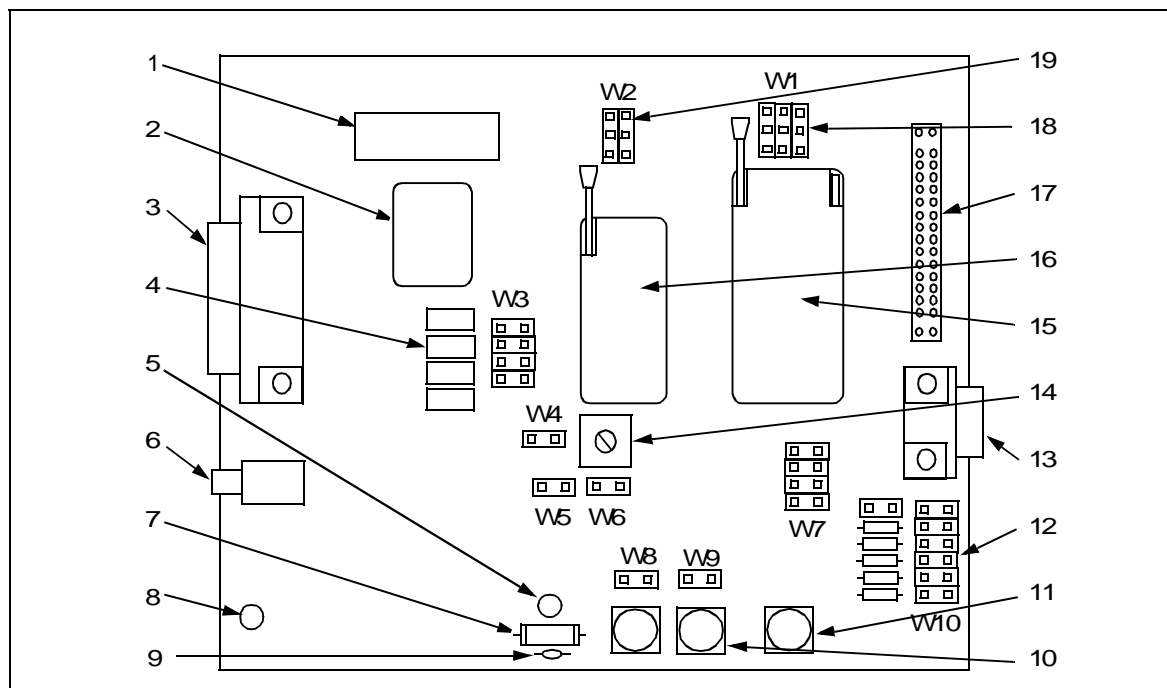
Figure 3: Starter Kit board layout

1	In-circuit programming connector (J1).	11	RESET button.
2	8 Mhz oscillator.	12	Demonstration routine selector.
3	PC connector P1.	13	RS232 interface circuit and connector.
4	4 LEDs.	14	10 KΩ trimmer.
5	Heater resistor LED indicator LD6.	15	DIL 20-28 ZIF MCU socket.
6	Power supply JACK connector J3.	16	DIL 16 ZIF MCU socket.
7	Heater resistor.	17	Remote resource I/O interface connector J2.
8	Power supply LED indicator LD5.	18	ST6220 or ST6225 device selection jumpers W1.
9	Thermistor.	19	"Programming" or "User" operating mode selection jumpers W2.
10	"+" and "-" buttons.		

www.BDTIC.com/ST

3.2.1   Oscillator

An oscillator feeds the ST6225 OSCIN input with an 8 MHz clock signal. You can disconnect the oscillator by removing the two jumpers W2 (marked 19 on the Starter Kit board diagram on *page 18*).
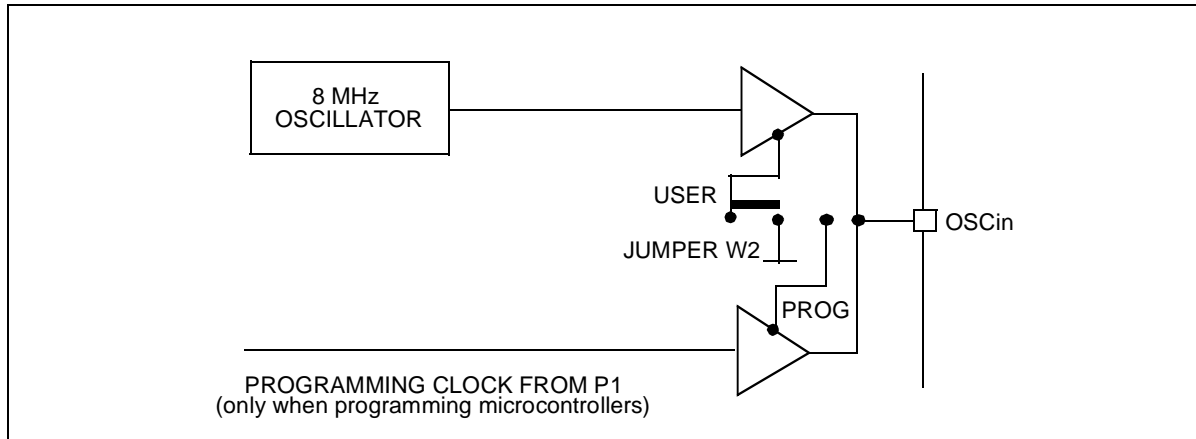
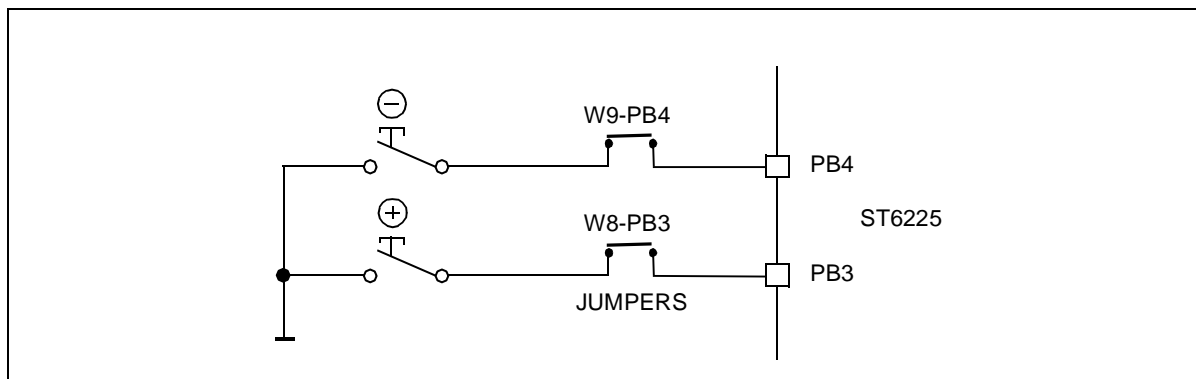### 3.2.2   Reset Button

The Reset button activates the ST6225 RESET input when pressed. A power-on reset circuit is also provided.

3.2.3   + And - Buttons

The + and - buttons are connected to PB3 and PB4 pins on the microcontroller respectively. They drive the PB3 and PB4 inputs down to GND when pressed.

You can disconnect PB3 by removing the W8 jumper, and PB4 by removing the W9 jumper.

3.2.4   LED Indicators

Four LEDs are used for level indication in the demonstration routines. They are connected to the ST6225 pins: PA0 to PA4 (when programmed as outputs) to demonstrate direct LED-driving capability.

The 4 LEDs can be disconnected by removing the W3-PA0 to 3 jumpers.
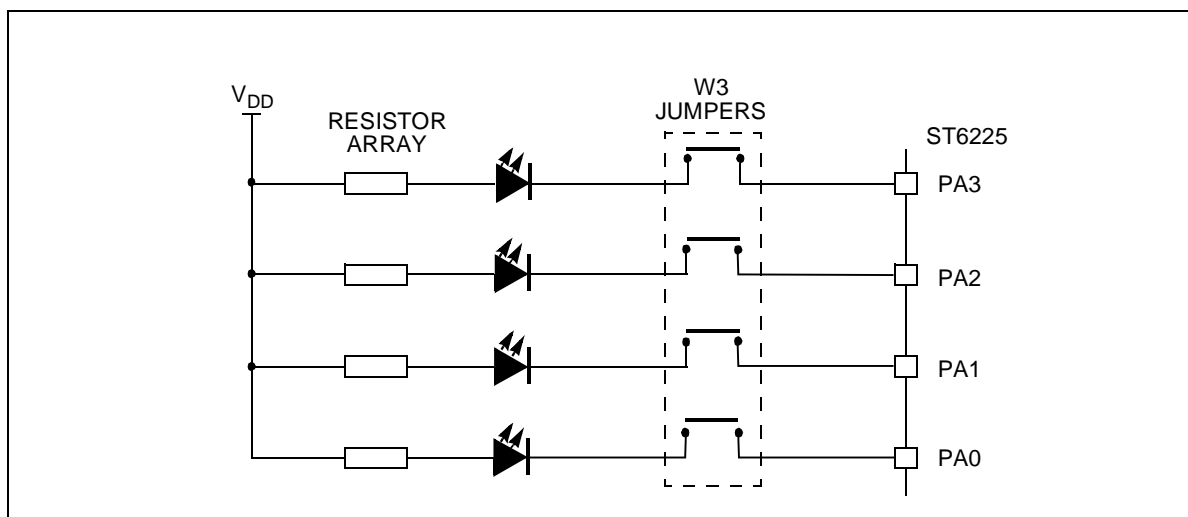
3.2.5   Resistance trimmer

A 10 KΩ resistance trimmer feeds the ST6225 PA5 I/O pin (when programmed as an A/D Converter input) with a variable voltage (0 V to 5 V DC). It is used for A/D conversion demonstration/evaluation.

The trimmer can be disconnected from the I/O pin by removing the W4-PA5 jumper.

www.BDTIC.com/ST

3.2.6 Temperature Control Circuit

This circuit demonstrates temperature control, using the on-chip A/D converter.

A heater resistor circuit (150 Ω, 1 Watt) is driven by the TIMER output of the ST6225 microcontroller. The heat level can be controlled by varying the duty cycle of a PWM signal present on the TIMER output.
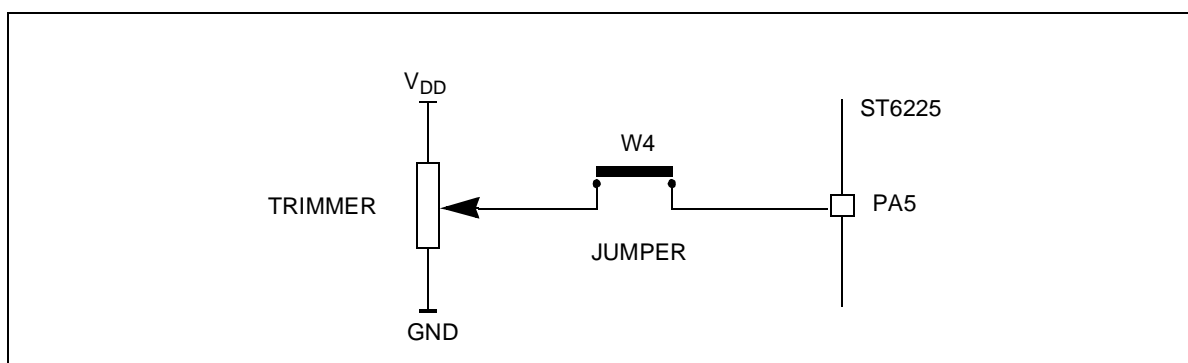
The heater circuit can be disconnected from the TIMER output pin by removing the W6-TIMER jumper.

A thermistor bridge (Negative Temperature Coefficient) is connected to the PA4 I/O pin, which is programmed as an A/D converter input. The voltage value at the A/D converter input decreases as the thermistor temperature increases.

The thermistor circuit can be disconnected from PA4 by removing the W5 jumper.

The thermistor is located in close proximity to the Heater resistor on the Starter Kit board, so that it can be used as a sensor to detect the temperature of the resistor body.

Demonstration 6 shows how this circuit is used as a feedback loop that controls the temperature level (see *Running the Demos* on page 25).
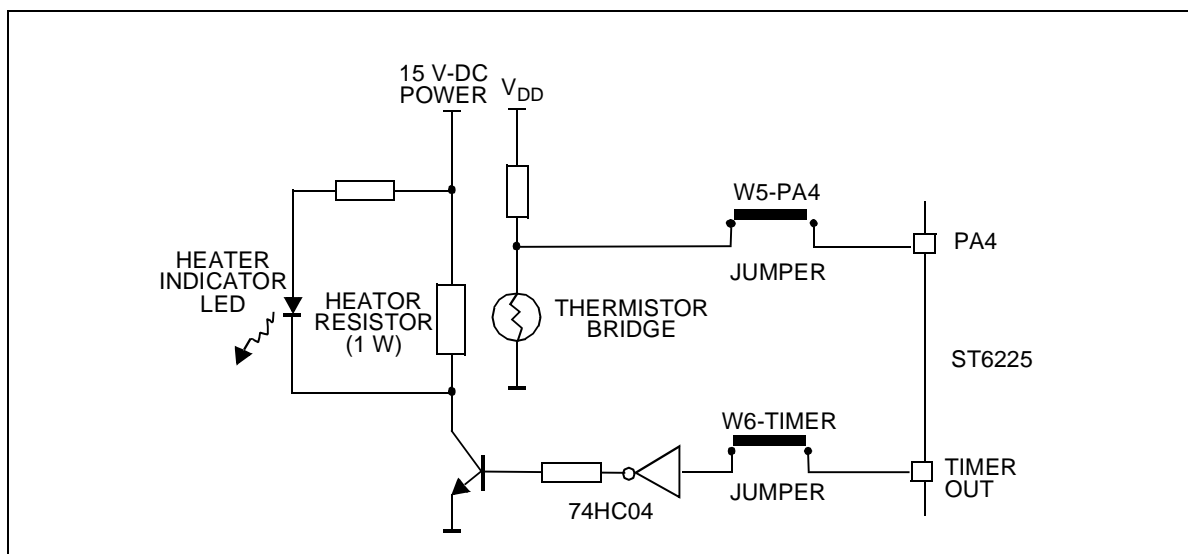
3.2.7 RS-232 Interface

The RS-232 interface enables you to communicate with the pre-programmed ST6225 microcontroller provided with the Starter Kit. It includes an RS-232 buffer circuit that is connected to a standard PC-compatible RS-232 SUBD-9 connector.

The following table lists the RS-232/ST6 pin connections:

Signal Name	SUBD-9 Pin	ST6 Pin
Data Transmission (TX)	2	PC7
Data Reception (RX)	3	PC6
Request to Send (RTS)	8	PC5
Clear to Send (CTS)	7	PC4

You can disconnect these by removing the corresponding jumpers from W7.

Figure 4 shows the RS-232 connections and line allocations:
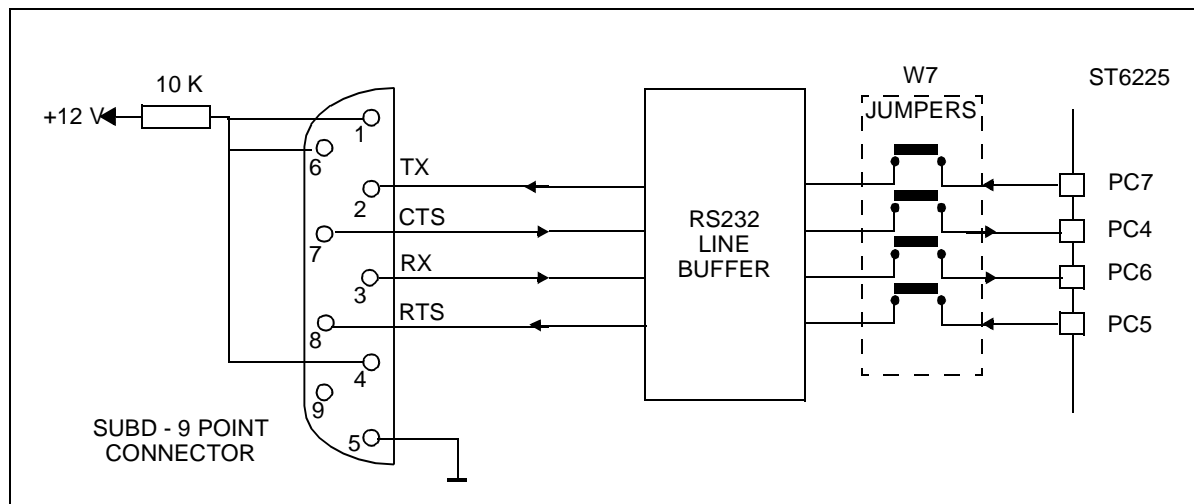
Figure 4: RS-232 connections and line allocations

TX, RX, CTS and RTS are defined so that the board is used as a slave. To use the board as a master, swap both the RX and TX and the CTS and RTS pin connections on the cable.

3.2.8   Demonstration Selector Jumpers

The demonstration selector is made up of a 6-position jumper W10 (marked 12 on the Starter Kit board diagram on *page 18*), with each jumper connected to a resistor. Each resistor generates a different voltage.

After reset, the voltage value generated by the resistor whose jumper is installed is sent to PA3 on the ST6225. PA3 is programmed as an A/D converter. The program

installed on the ST6225 uses the input from PA3 to select the appropriate demo. The following table lists the voltage values generated by each resistor:

Resistor value:	Theoretical voltage value:
RT: 10 KΩ	No JUMPER: 5 V
R1: 680 Ω	JUMPER-D1: 0 V
R2: 750 Ω	JUMPER-D2: 333 mV
R3: 820 Ω	JUMPER-D3: 666 mV
R4: 1 KΩ	JUMPER-D4: 999 mV
R5: 1.2 KΩ	JUMPER-D5: 1.332 V

The same principle can be used for keyboard decoding. For a complete example of this, refer to the STMicroelectronics application note: *AN431: Using ST6 Analog Inputs for Multiple Key Decoding*.

You can disconnect the demonstration selector from PA7 by removing the PA7 jumper from W10 (marked 12 on the Starter Kit board diagram on *page 18*).
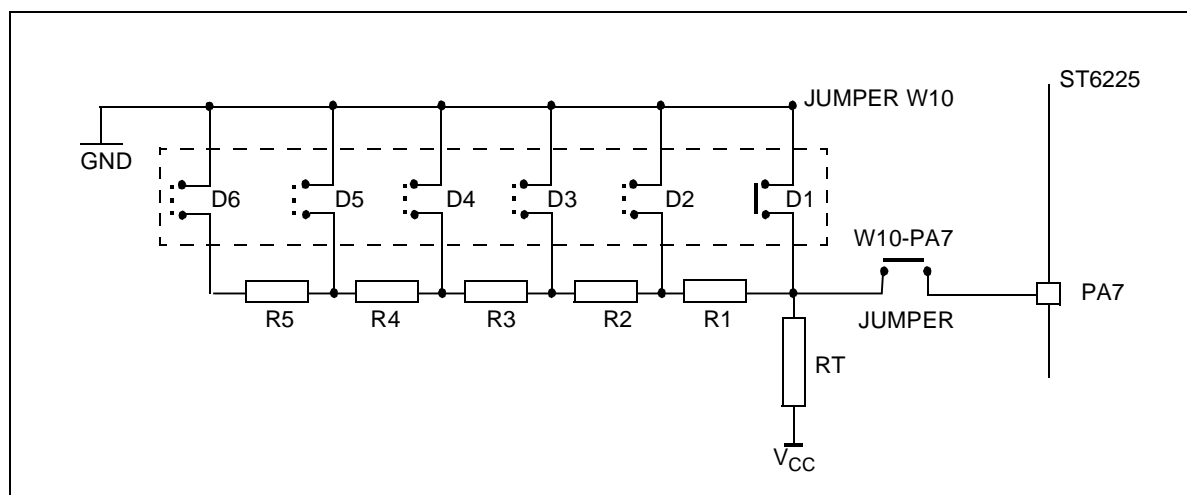
4   RUNNING THE DEMOS

This section describes the demonstration programs that are provided with the Starter Kit and explains how to run them.

4.1   What the Demos Do

The following paragraphs describe the demos that come pre-loaded with the ST6 Starter Kit demos. See *Running the Demonstration Programs* on page 27 for details on how to select and run a demo.

The source files of these demos are also included on the *MCU on CD* CD-ROM and are installed in the following directory:

    ...\Stm\ST6toolchain\StarterKit\sk622xli.

4.1.1   Demo 1 - Button/LED Indicator Control

1   Initializes the pins as follows:

This pin:	Is initialized as:
PB3	Input with pull-up and interrupt. Connected to + button.
PB4	Input with pull-up and interrupt. Connected to - button.
PA0 to PA3	Push-pull outputs. Connected to the four LEDs (as marked on board).

2   Performs a task relating to your actions:

When you do this:	The program does this:
Press the + button	Jumps to an interrupt subroutine that lights up each led in turn, from LD4 to LD1.
Press the - button	Jumps to an interrupt subroutine that lights up each led in turn, from LD1 to LD4.

4.1.2   Demo 2 - Voltage trimming and LED level indication

1   Initializes the pins as follows:

This pin:	Is initialized as:
PA5	Analog input. Connected to the trimmer.
PA0 to PA3	Push-pull outputs. Connected to the four LEDs (as marked on board).

2   Reads the A/D converter data register, and turns on a LED according to the Voltage value input by the trimmer:

This voltage:	Turns this LED on:
0 to 1	none
1 to 2	LD1
2 to 3	LD2
3 to 4	LD3
4 to 5	LD4

Adjusting the voltage trimmer (marked 9 on the Starter Kit board diagram on *page 18*) turns on the appropriate LED. Pressing the + or - buttons has the same effect as increasing or decreasing the voltage.

4.1.3   Demo 3 - Temperature Control

1   Initializes the pins as follows:

This pin:	Is initialized as:
PA4	Analog input. Connected to the thermistor circuit.
TIMER	Push-pull output. Connected to the Heater Resistor circuit.
PB3	Input with pull-up and interrupt. Connected to + button.
PA0 to PA3	Push-pull outputs. Connected to the four LEDs (as marked on board).

2   Reads and stores the A/D converter data register value. This value indicates the temperature at reset.

3   Reads and stores the A/D converter data register value at regular intervals. If this value exceeds the value that was stored at reset, a LED is turned on indicating the difference between the two values. The higher the difference is between the stored value and the read value, the higher LED number is turned on (roughly in steps of LD(n+1) for each additional degree difference).

You can either increase the temperature by touching the thermistor (marked 14 on the Starter Kit board diagram on *page 18*) or pressing the + button.

4.1.4   Demo 5 - Fuzzy Logic Heater Control

This demonstration shows how a heater control application can be developed using a Fuzzy Logic description. The application was developed using the **fuzzyTECH** explorer development tool. For more information about this demonstration, see the **Fuzzykit** text file, this is supplied on the *MCU on CD* CD-ROM and is installed in the following directory:

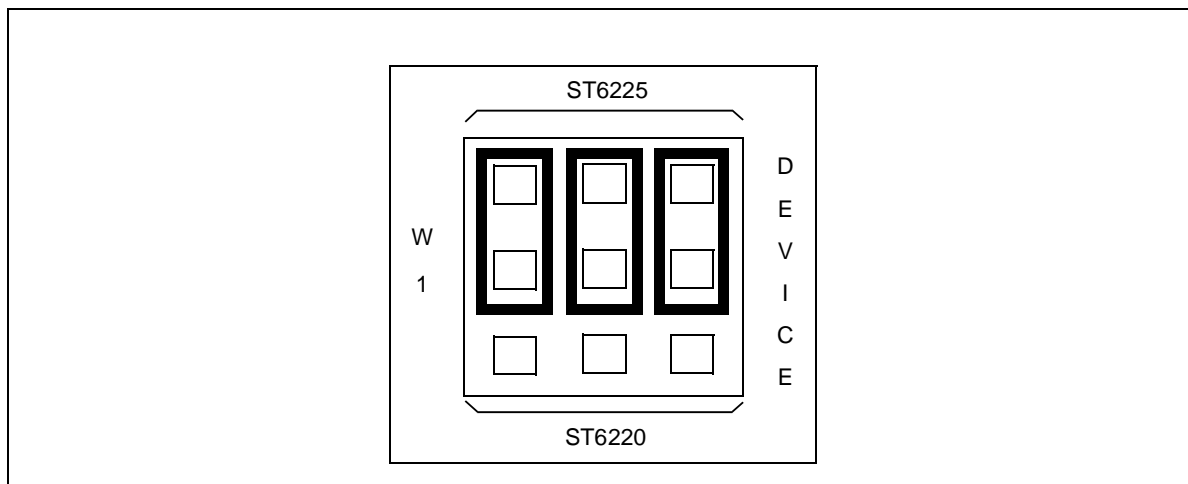   **...\Stm\ST6toolchain\StarterKit\sk622xli**.

4.2   Running the Demonstration Programs

The ST62E25 microcontroller labelled DEMOK25 is programmed with the demonstration software. The **Demok25.asm** file is also provided on the *MCU on CD* CD-ROM and is installed in the **...\Stm\ST6toolchain\StarterKit\sk622xli** directory, so that if you erase it from the ST62E25 microcontroller, you can re-program it following the instructions given in *Chapter 7: Programming ST6 Microcontrollers* on page 37.
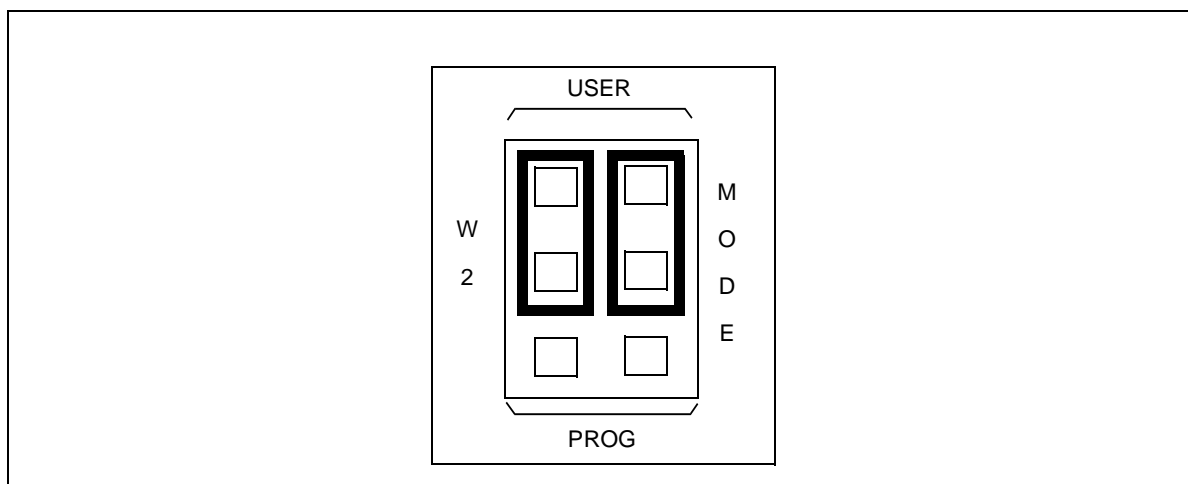
To run the demonstrations:

1   Power down the Starter Kit board.

2   Make sure that the pre-programmed ST62E25 is plugged into the DIL connector, and that the DEVICE jumpers W1 (marked 18 on the Starter Kit

board diagram on *page 18*) are set to ST6225 as shown in the following diagram:

```
              ST6225
         ┌──────────────┐
         │ ▢  ▢  ▢      │ D
    W    │ ▢  ▢  ▢      │ E
         │              │ V
    1    │              │ I
         │ ▢  ▢  ▢      │ C
         │              │ E
         └──────────────┘
              ST6220
```

3 Select the USER mode using the jumpers marked W2 (marked 19 on the Starter Kit board diagram on *page 18*), as shown in the diagram below:

```
           USER
      ┌──────────┐
      │ ▢   ▢    │ M
  W   │ ▢   ▢    │ O
      │          │ D
  2   │ ▢   ▢    │ E
      └──────────┘
           PROG
```
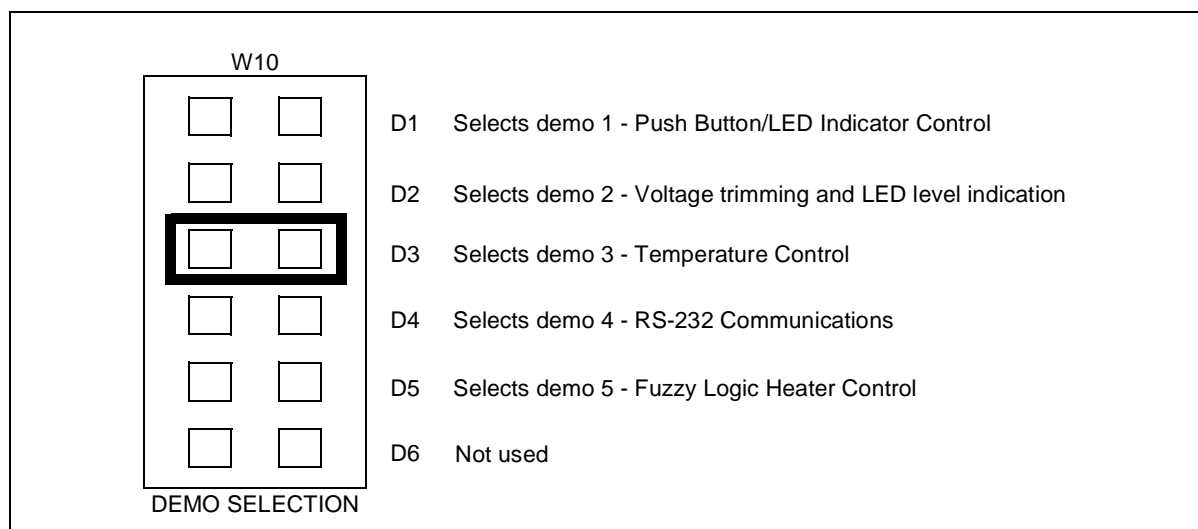
4 Disconnect the cable from the parallel port (P1) connection, if it is connected.

5 Power up the Starter Kit board.

6 Install the demonstration program jumper marked PA7, as shown in the diagram below:
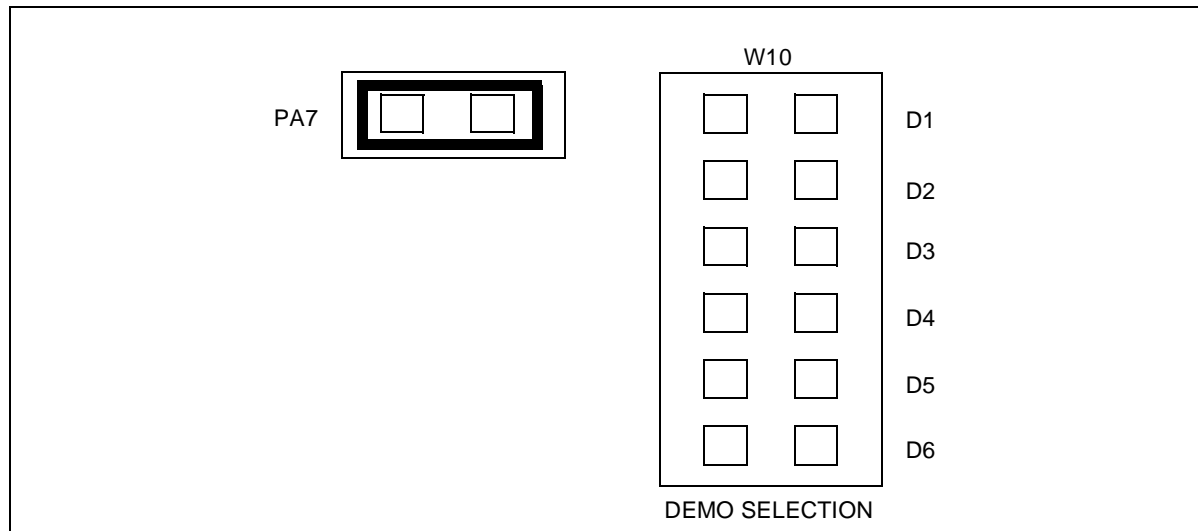
7 Select the demo you want to run, by installing the appropriate jumper on W10 (marked 12 on the Starter Kit board diagram on *page 18*), as indicated on the diagram below:

For example, in the above diagram demo 3 is selected.

8 Press the reset button.

The selected demo is now run.

9 To run a different demo, repeat steps 7 and 8.

5   CONNECTING EXTERNAL RESOURCES TO THE STARTER KIT BOARD

You can connect your own external resources to the pre-programmed ST62E25 to debug or evaluate your programs, using the connector J2 (marked 17 on the Starter Kit board diagram on *page 18*).

To be able to connect your own resources to the Starter Kit board, you must disconnect the resources that are already connected to the ST62E25 (to avoid conflicts between external resources and Starter Kit board resources).

The following table lists the Starter Kit board resources and the corresponding J2 connections, and indicates the jumper that disconnects each resource.

On-board resource	Jumper to disconnect	ST6225 I/O	PIN	PIN	ST6225 I/O	Jumper to disconnect	On-board resource
+5 V Supply	-	V_{DD}	1	28	V_{SS}	-	GND
Heater resis. control	W6-TIMER	TIMER	2	27	PA0	W3-PA0	LED Indicator
8 MHz OSC.	W2	OSCIN	3	26	PA1	W3-PA1	LED Indicator
none	-	OSCO	4	25	PA2	W3-PA2	LED Indicator
System tasks (Simulator link)	(*)	NMI	5	24	PA3	W3-PA3	LED Indicator
RS232 Driver TX Output	W7-PC7	PC7	6	23	PA4	W5-PA4	Thermistor
RS232 Driver RX Input	W7-PC6	PC6	7	22	PA5	W4-PA5	Trimmer
RS232 Driver RTS Output	W7-PC5	PC5	8	21	PA6	-	none
RS232 Driver CTS Input	W7-PC4	PC4	9	20	PA7	W10-PA7	Routine selector
System tasks (Programming)	W2	V_{PP}	10	19	PB0	(*)	System task (Simul. link)
Reset Button	-	RESET/	11	18	PB1	(*)	System task (Simul. link)

On-board resource	Jumper to disconnect	ST6225 I/O	PIN	PIN	ST6225 I/O	Jumper to disconnect	On-board resource
System tasks (Programming)	W2	PB7	12	17	PB2	(*)	System task (Simul. link)
System tasks (Programming)	W2	PB6	13	16	PB3	W8-PB3	+ Button
System tasks (Programming)	W2	PB5	14	15	PB4	W9-PB4	- Button

*)   This is not available if the Starter Kit board is connected to a host PC.

Note: *Some of the signals on the J2 connector are used during ST6 programming, thus you must disconnect any external resource that is connected to J2 before using the Starter Kit board for programming.*

6   USING THE STARTER KIT BOARD AS A HARDWARE SIMULATOR

WGDB6, the ST6 debugger that runs under Windows, lets you test your programs without having to program the EPROM of your target ST6. Depending how much information you want, and how close to real life you want your test environment to be, you can use WGDB6 in one of three ways:

- As a software simulator. If you use WGDB6 as a simulator, you need not attach any additional hardware to your PC. The ST6 simulator program, that comes with WGDB6 and is run when you run WGDB6/Simulator, simulates the execution of your program, letting you step through the code and see what happens as the program runs. WGDB6 simulator includes Wave Form Editor, which simulates the output of the pins on your target ST6 in relation to inputs that you define, enabling you to see how its peripherals react to the inputs they receive.

- With an ST6 hardware emulator. Emulators are hardware systems that act as your target microcontroller, at the same time capturing detailed information, such as which areas of memory are accessed by the program and what happens when they are accessed. In this case, WGDB6/Emulator provides an interface between the emulator and your PC, displaying data captured by the emulator and letting you implement the WGDB6 features in the emulator, such as software or hardware breakpoints.

- With the Starter Kit board as a hardware simulator. This is a cross between the above two. The WGDB6 software simulator simulates the execution of your program, but each time the data space is accessed, it accesses that of the ST6 that is plugged into your Starter Kit board. Thus, using the Starter Kit board with WGDB6, you can view how the real microcontroller peripherals behave when your program is executed.

This section describes the third option, how to use the Starter Kit board as a hardware simulator.

You can use the Starter Kit board to emulate any ST620x, ST621x or ST622x microcontroller. Note, however that you must use the pre-programmed ST62E25 microcontroller, labelled DEMOK25 supplied with the kit for hardware simulation. Thus, when simulating programs designed for other microcontrollers, make sure that you do not use resources that are not available on the microcontroller your application is designed for.

ST

6.1   The Data Transmission Driver

Data is transferred between the simulated peripheral registers and the ST620x, ST621x and ST622x registers via the host PC's parallel port. The **Demok25.hex** program, with which the ST62E25 microcontroller that is plugged into the Starter Kit board must be loaded includes the transmission driver.

The data transfer driver uses the following bits:

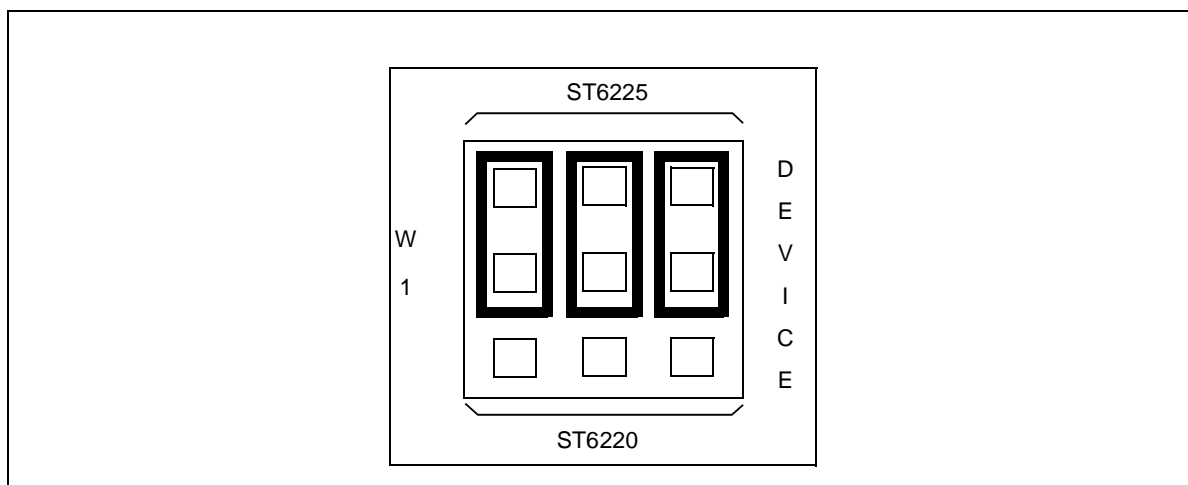| PC parallel port	ST6225	WGDB6 Use
D2	PB2	Synchronization
D3	PB1	Write data to MCU
D4	RESET	Hardware reset of peripherals
D6	NMI	Initiates data transfer
SDOP	PB0	Read data from MCU

6.2   Technical Limitations

The Starter Kit board has the following limitations when used with WGDB6 as a hardware simulator:

- Real-time program execution is not supported.

- Resetting the ST6225 by power on, pressing the Reset button or external reset does not reset the simulated ST6 core. To perform a complete simulated reset, use the WGDB6 reset command instead.

- Interrupts sent by the ST62E25 microcontroller are not supported by the WGDB6 simulator.

- The pins: NMI, PB0, PB1 and PB2 on the ST62E25 microcontroller are used for communications with the host PC, and are thus not available for simulation.
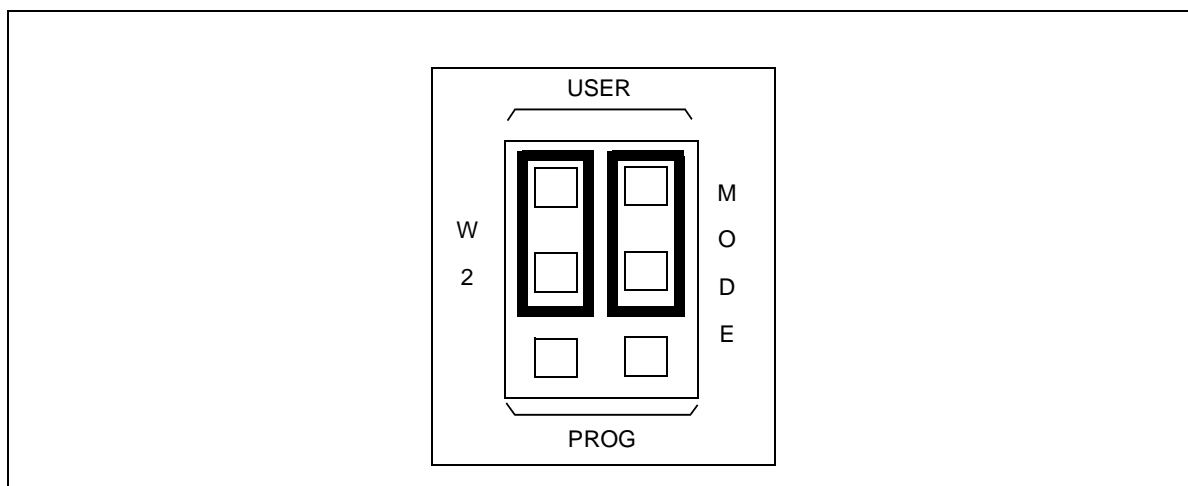
Note: *Do not connect any external resources to the corresponding J2 connector pins when using the Starter Kit board as a peripheral emulator.*

6.3   To use the Starter Kit board as a hardware simulator:

1   Power down the Starter Kit board.

2   Make sure that the ST62E25 labelled DEMOK25 is plugged into the DIL28 socket, and that the DEVICE jumpers W1 (marked 18 on the Starter Kit board diagram on *page 18*) are set to ST6225 as shown in the following diagram:

```
              ST6225
       ┌───┐ ┌───┐ ┌───┐      D
       │ ▣ │ │ ▣ │ │ ▣ │      E
  W    │   │ │   │ │   │      V
       │ ▣ │ │ ▣ │ │ ▣ │      I
  1    └───┘ └───┘ └───┘      C
        ▢     ▢     ▢         E

              ST6220
```

3   Select the USER mode using the jumpers marked W2 (marked 19 on the Starter Kit board diagram on *page 18*), as shown in the following diagram:

```
           USER
       ┌───┐ ┌───┐      M
       │ ▣ │ │ ▣ │      O
  W    │   │ │   │      D
       │ ▣ │ │ ▣ │      E
  2    └───┘ └───┘
        ▢     ▢

           PROG
```

4   Connect the Parallel port P1 on the Starter Kit board to a spare parallel port on your PC using the cable provided with the Starter Kit.

5   Power up the Starter Kit board.

To run WGDB6:

From Windows 95, 98 or NT click **Start > Programs** > **ST6 Tool Chain > Development Tools** > **Wgdb6 StarterKit**.

Refer to the *WGDB6 Windows Debugger for the ST6 Family User Manual* for full instructions on how to use WGDB6. This manual is available in PDF format on the *MCU on CD* CD-ROM.

6.4 Error Messages

The following table lists the error messages you may encounter when using WGDB6 with the Starter Kit board:

Error message	Description
Error 116 Port B protected when using board.	This means that WGDB6 tried to access the PORT B registers. These are used for communications with the board.
Error 117 Communication error with ST620x/ST621x/ST622x board.	This means that a problem occurred during communications between the host PC and the board. Perform the checks listed below.

6.5 Troubleshooting

If there is a communications problem between WGDB6 and the Starter kit board, the title "WGDB6 Simulator" appears in the WGDB6 title bar. In this case, you should check the following:

- That the Starter Kit board is correctly powered up.

- That the parallel port cable is correctly connected.

- That the device jumpers (W2) are in the USER position.

- That the device type selection jumpers (W1) are in the ST6225 position.

- That an ST6225 is plugged into the Starter Kit board, and it is programmed with **Demok25.hex**.

7   PROGRAMMING ST6 MICROCONTROLLERS

You can use the Starter Kit board, in conjunction with the ST6 Epromer, to program ST620x, ST621x or ST622x microcontrollers. You can also perform in-circuit programming of ST620x, ST621x or ST622x OTP/EPROM microcontrollers using your own board, connected to the Starter Kit board via the connector J1 (marked 1 on the Starter Kit board diagram on *page 18*).

7.1   Programming Signals

The following table shows the programming signals and states and their corresponding pin numbers.

|		MCU DIL28		MCU DIL20		MCU DIL16	
Programming signals	Programming states	MCU pin	J2 connector	MCU pin	J2 connector	MCU pin	J2 connector
PB7	High	Pin 12	Pin 12	Pin 8	Pin 12	Pin 7	Pin 12
PB6	High/Low	Pin 13	Pin 13	Pin 9	Pin 13	Pin 8	Pin 13
PB5	High/Low	Pin 14	Pin 14	Pin 10	Pin 14	Pin 9	Pin 14
TEST	5 V/12 V	Pin 10	NC	Pin 6	Pin 10	Pin 5	Pin 10
OSCIN	High/Low	Pin 3	Pin 3	Pin 14	Pin 7	Pin 2	Pin 7
RESET	High/Low	Pin 11	Pin 11	Pin 16	Pin 11	Pin 6	Pin 11
PB3	High	Pin 16	Pin 16	Pin 12	Pin 16	NC	NC

Note: *Some of the signals on the J2 connector are used during ST6 programming, thus if you have connected an external resource to J2, you must disconnect it before using the Starter Kit board for programming.*

This section describes how to set up the Starter Kit board for programming microcontrollers, and lists the connection requirements for in-circuit application boards.

7.2   Setting Up the Starter Kit Board

1 Power down the Starter Kit board.

2 Plug the ST6 microcontroller you want to program into the U3 or U4 ZIF connector, according to the following diagram:
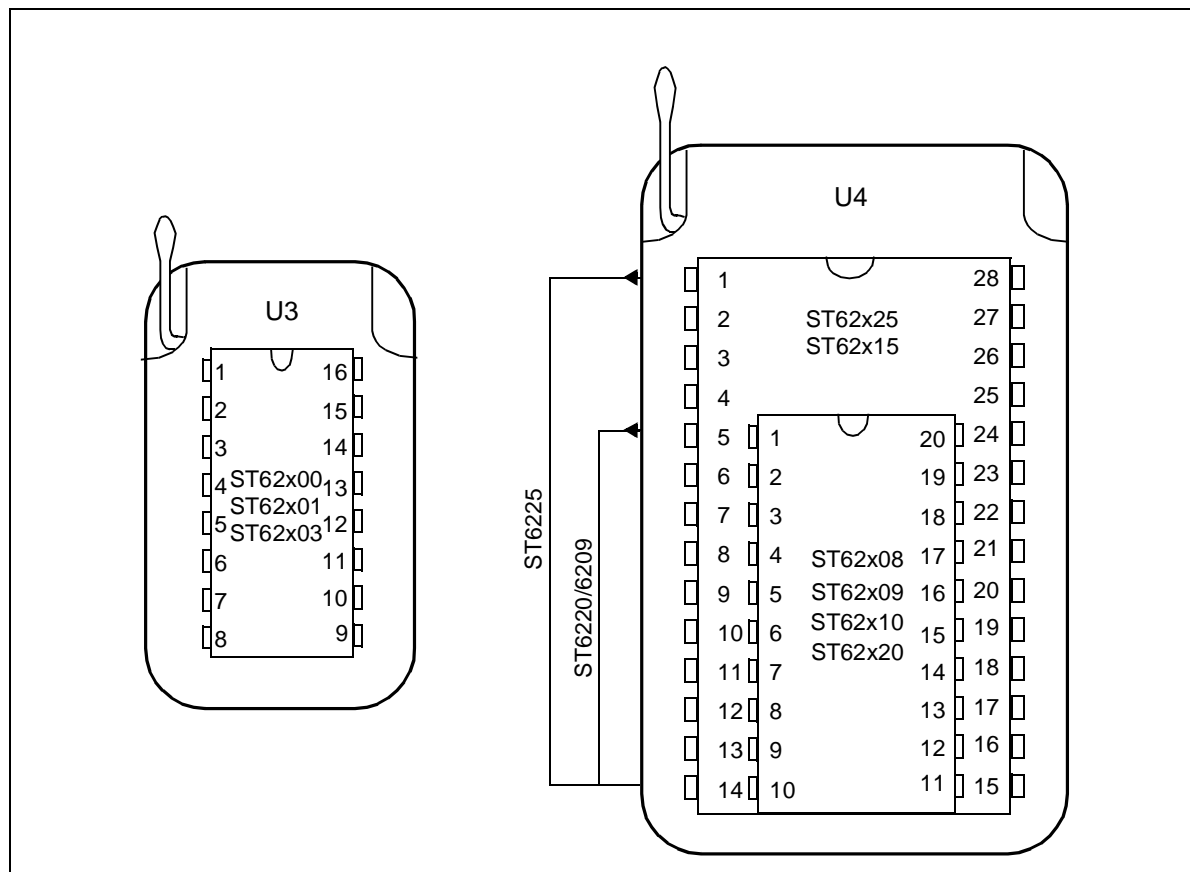

3 Set the DEVICE jumpers W1 (marked 18 on the Starter Kit board diagram on *page 18*) for the microcontroller that you want to program according to the following table:
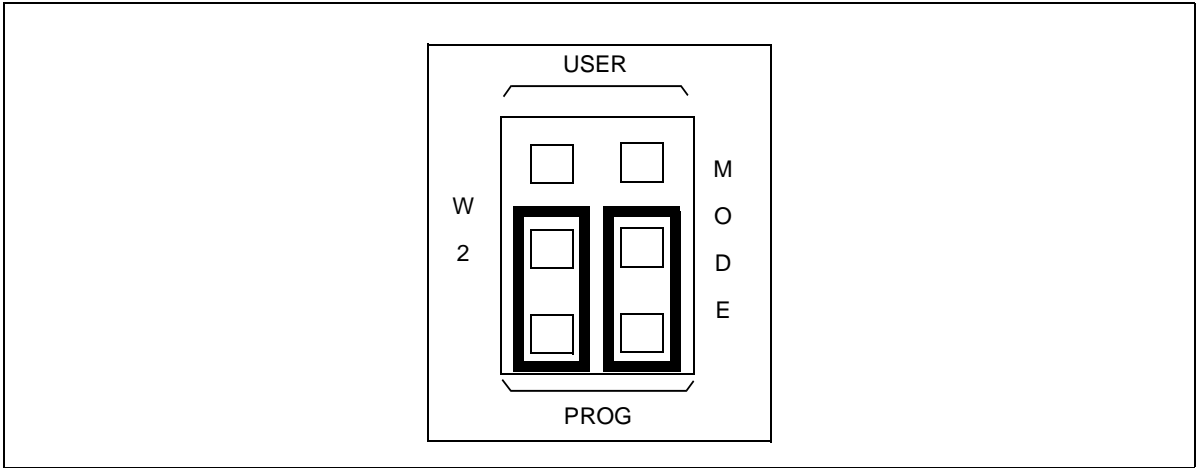
Device Type	W1 position
ST6200	ST6220
ST6201	ST6220
ST6203	ST6220
ST6208	ST6220
ST6209	ST6220
ST6210	ST6220

Device Type	W1 position
ST6220	ST6220
ST6215	ST6225
ST6225	ST6225

As shown in the following diagram:
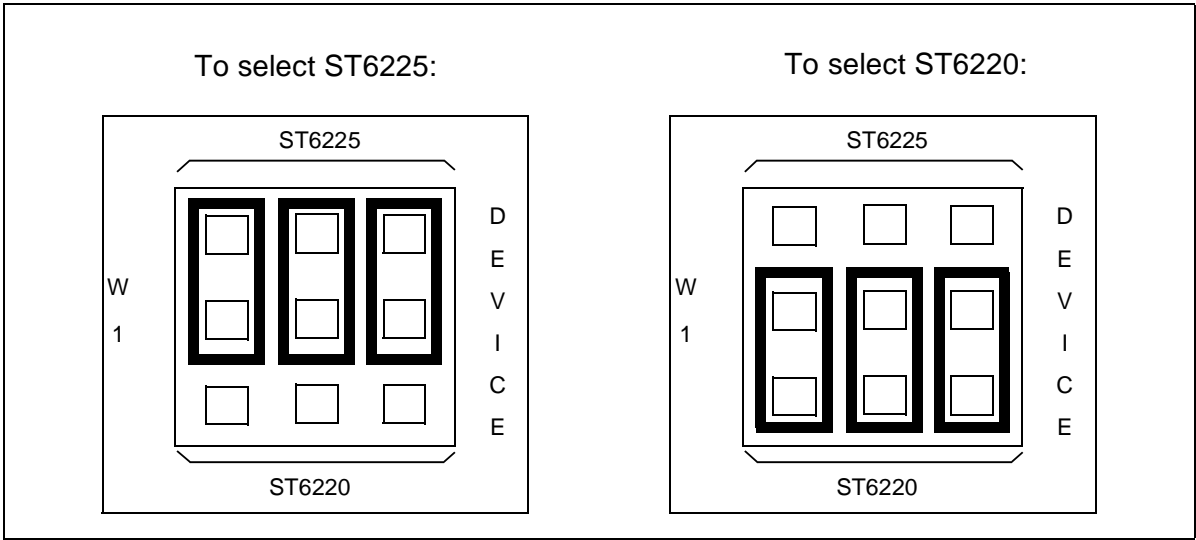
4 Select the PROG mode using the jumpers marked W2 (marked 19 on the Starter Kit board diagram on *page 18*), as shown in the following diagram:

5   Connect the Parallel port P1 on the Starter Kit board to a spare parallel port on your PC using the cable provided with the starter kit.

6   Power up the Starter Kit board.

You can now use Epromer to program the microcontroller that is plugged into the Starter Kit board.

To run Epromer from Windows 3.x, double-click the Epromer icon in the ST6 Tools group.

To run Epromer from Windows 95, 98 or NT click **Start > Programs** > **ST6 Tool Chain > Development Tools** > **Winee**.

For instructions on how to operate Epromer, click **Help** in the Epromer main window.

7.3   In-Circuit Programming

You can perform in-circuit programming of ST620x, ST621x or ST622x OTP/ EPROM microcontrollers using your own board, connected to the Starter Kit board via the connector J1 (marked 1 on the Starter Kit board diagram on *page 18*).

7.4   Application Board Connections

The following paragraphs specify the connection requirements between your application board and the Starter Kit board.

The application board must have a suitable 16-way connector (8x2 header HE10) to be connected via a 16-way cable to connector J1 (marked 1 on the Starter Kit board diagram on *page 18*) on the Starter Kit board.

The following table shows the required pin connections:

ST620x, ST621x and ST622x Pins	Connector Pins
PB6	Pin 1
PB5	Pin 3
OSCin	Pin 5
PB7	Pin 7
RESET	Pin 9
V_{PP}/TM	Pin 13
V_{DD}	Pin 14, 15

ST620x, ST621x and ST622x Pins	Connector Pins
V_{SS}	Pin 2, 4, 6, 8, 10

V_{DD}

Use of the V_{DD} connection is optional, depending on whether the application board supply can or cannot be disconnected. If the application board supply is disconnected, you can supply it through pins 14 an 16 of the connector, as long as the total load current does not exceed 100 mA, and the capacitive load is less than 50 μF.

If the application board has its own power supply, its voltage must be set to 5 V, so that logic levels are compatible with those of the Starter Kit board.

OSCin

Synchronizes the programming operations using a clock generated by the programming tool. OSCin is located on the application board, and must be directly connected to Pin 5 on the 16-way connector. No isolation is needed as long as a quartz crystal or ceramic resonator is used in the application. If an external clock generator is used in the application, it must be disconnected during in-circuit programming.

RESET

Controls the programming mode entry. To prevent signal level contention, RESET must be directly connected to Pin 9 on the 16-way connector, and must be isolated from other nodes on the application board. Any direct connection to V_{DD}, V_{SS} or an output must be avoided. This pin can be connected to a CMOS input, a 2 KΩ pull-up, a 10 KΩ pull-down or left open (Internal pull-up). The capacitive load of the RESET pin should not exceed 1 μF.

Pins 1 and 7 on the 16-way connector are used to establish communications between the programming tool and the microcontroller.

To prevent signal-level contention, Pins 1 and 7 must be directly connected to PB6 and PB7 on the 16-way connector, and must be isolated from other nodes on the application board. Any direct connection to V_{DD}, V_{SS} or an output must be avoided. These pins may be connected to a CMOS input, a 2 KΩ pull-up, a 10 KΩ pull-down or left open (Internal pull-up).

If pin 3 on the 16-pin connector is connected to the target device, the same applies. Connection to pin 3 is not necessary if a high voltage level is guaranteed by the board design.

Some I/O pins are not connected to the 16-way connector and must be set to a high level during programming. This is normally achieved by the RESET signal sent by the programming tool through the 16-way cable, setting the I/O pins as inputs with an internal 300 KΩ pull-up. To keep these I/O lines high, direct connection of these pins to GND or to any other signal at low level (even temporarily) must be avoided. Only connections to another CMOS input, to an external pull-up or a 10 MΩ pull-down is allowed.

The signals on PB3 and PB5 (if not directly biased through pin 3 of the 16-way connector) must be kept at a high voltage level.

The V_{pp}/TM pin must not be directly connected to GND/V_{SS} on the application board, to avoid any conflict with the programming voltage provided by the programming tool via pin 13 on the connector. This pin should be pulled down by a resistor with minimum value of 10 KΩ. You must add a 100 nF ceramic capacitor between V_{pp}/Test and V_{SS}.

7.5   Setting Up the Starter Kit Board for In-Circuit Programming

1 Power down the Starter Kit board.

2 Set the DEVICE jumpers W1 (marked 18 on the Starter Kit board diagram on *page 18*) to ST6225, as shown in the following diagram:
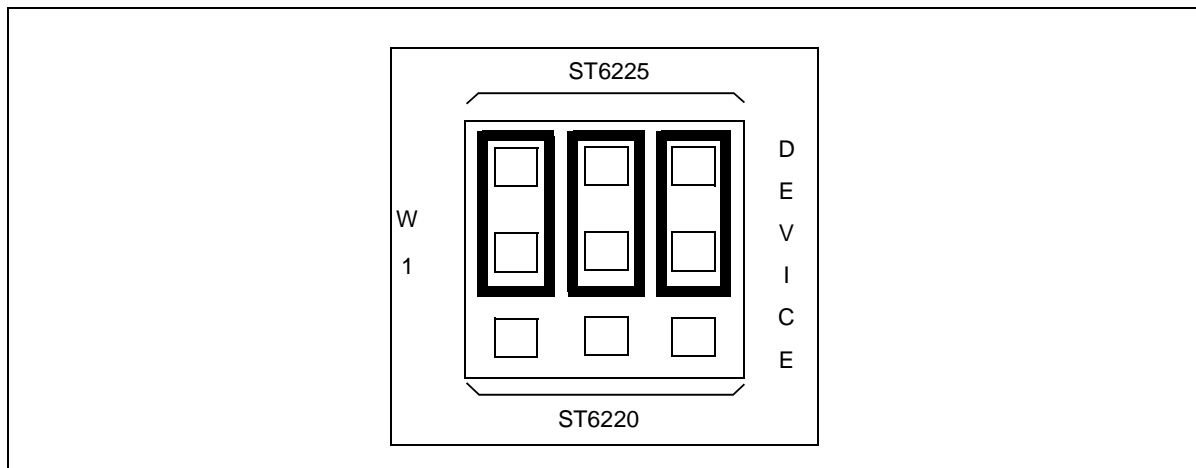
3 Select the PROG mode using the jumpers marked W2 (marked 19 on the Starter Kit board diagram on *page 18*), as shown in the diagram below:
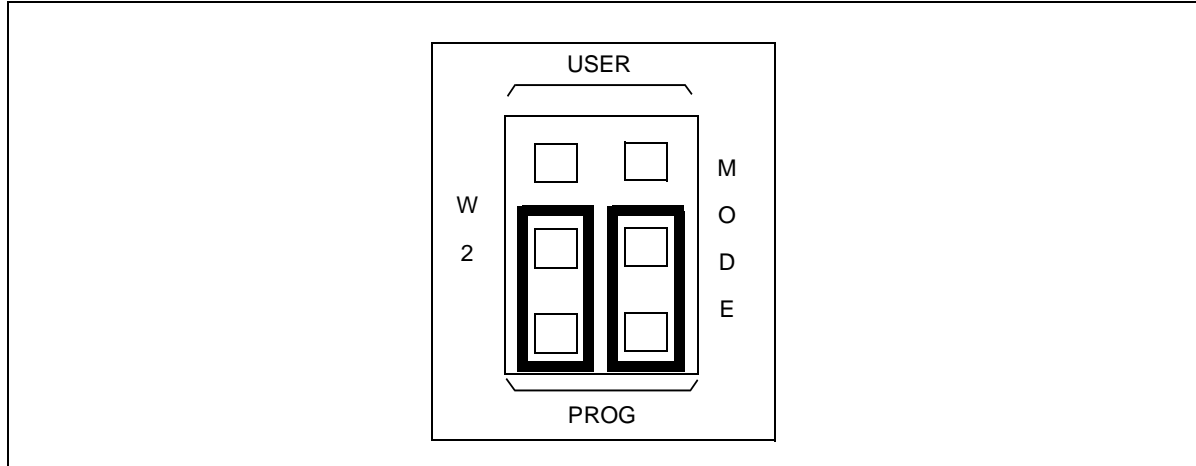
4 Connect the Parallel port P1 on the Starter Kit board to a spare parallel port on your PC using the cable provided with the starter kit.

5 Connect your application board to the connector J1 (marked 1 on the Starter Kit board diagram on *page 18*) on the Starter Kit board.

6 Power up your Starter Kit board.

You can now use Epromer to program the microcontroller that is plugged into the Starter Kit board.

To run Epromer from Windows 3.x, double-click the Epromer icon in the ST6 Tools group.

To run Epromer from Windows 95, 98 or NT, click **Start**, **Programs**, **ST6 Tools**, then **Epromer**.

For instructions on how to operate Epromer, click **Help** in the Epromer main window.

Note: *If your application board is not powered by the Starter Kit, you must connect it to a 5 V DC power supply before you start programming.*

8   RUNNING YOUR OWN PROGRAM ON THE STARTER KIT BOARD

You can run your own programs on the Starter Kit board, using any of the Starter Kit resources:

- 8 Mhz oscillator

- 10 KΩ trimmer

- + and - buttons

- Thermistor bridge

- Heater resistor control circuit

Note: *You can only run applications on the Starter kit board using ST6225 microcontrollers. If your application is designed for another microcontroller, you must change its port definitions to match those of the ST6225.*
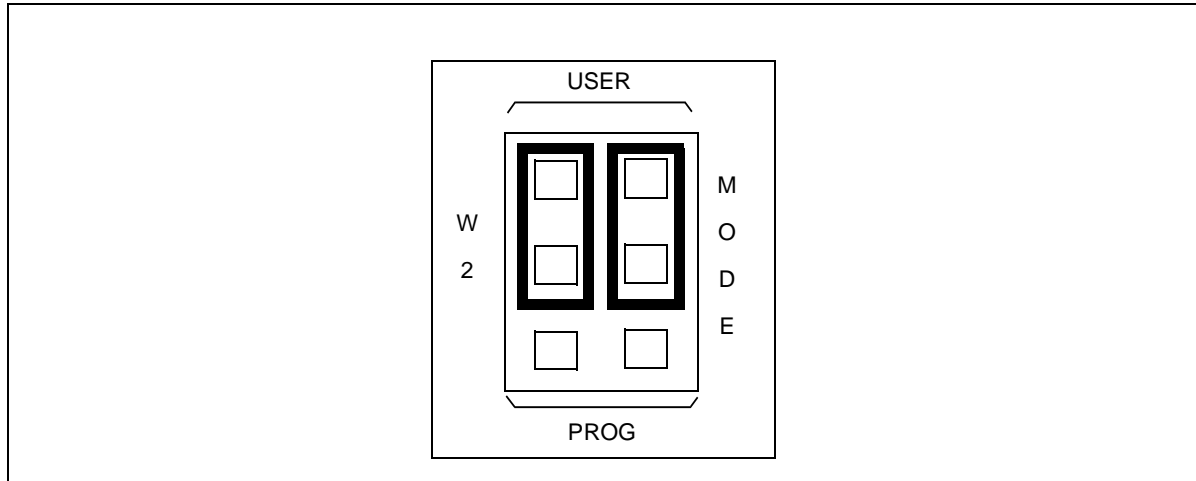
You can also use your own hardware resource by connecting it to the J2 connector (see *Connecting External Resources to the Starter Kit Board* on page 31).

To run your own program on the Starter Kit board:

1   Power down the Starter Kit board.

2   Program the microcontroller with the application you want to run following the instructions given in *Programming ST6 Microcontrollers* on page 37.

3   Make sure that the programmed microcontroller is plugged into the DIL connector, and that the DEVICE jumpers W1 (marked 18 on the Starter Kit board diagram on *page 18*) are set to ST6225, as shown in the following diagram:
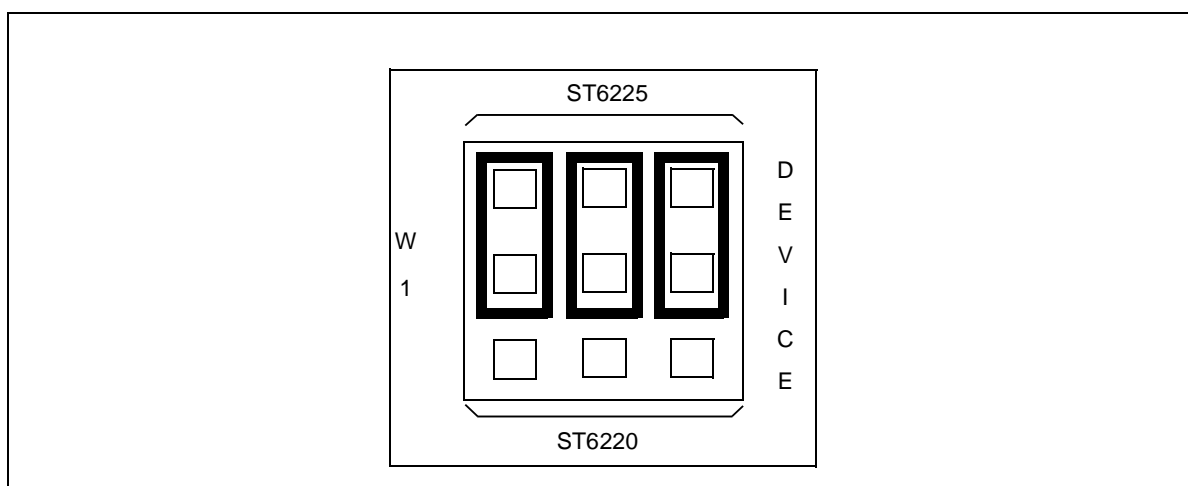
4   Select USER mode using the jumpers marked W2 (marked 19 on the Starter Kit board diagram on *page 18*), as shown in the diagram below:

```
                   USER
            ┌─────────────┐
        ┌───────────────────┐
        │  ┌───┐   ┌───┐    │ M
   W    │  │   │   │   │    │ O
        │  └───┘   └───┘    │ D
   2    │  ┌───┐   ┌───┐    │ E
        │  │   │   │   │    │
        │  └───┘   └───┘    │
        │  ┌───┐   ┌───┐    │
        │  │   │   │   │    │
        │  └───┘   └───┘    │
        └───────────────────┘
            └─────────────┘
                   PROG
```

5   Disconnect the cable from the parallel port (P1) connection, if it is connected

6   Disconnect the demonstration program selector by removing the jumper marked PA7 in the Demonstration Selector circuit marked 12 in the Starter Kit board diagram on *page 18*.

7   If you are using your own hardware resources connected to J2 (marked 17 on the Starter Kit board diagram on *page 18*), disconnect any Starter Kit board resources that use the same pins, following the instructions given in *Connecting External Resources to the Starter Kit Board* on page 31.

8   Power up the Starter Kit board.

APPENDIX A: GLOSSARY

AST6

AST6 is a macro-assembler that translates files that are written in assembler language into either executable files or object files.

DIL or DIP

Dual In Line, also known as Dual In Line Package. These two abbreviations designate a type of device package with two rows of pins for thru-hole mounting.

EPROM

Erasable Programmable Read-Only Memory. A non-volatile type of memory that can be erased by exposure to an ultra-violet light source. MCUs that have EPROM are easily recognized because the package has a quartz window to allow exposure to the UV light. If the EPROM MCU is packaged in an opaque plastic package, it is called a OTP (One-time Programmable) MCU because there is no way to expose the EPROM to a UV light source.

EPROMER

Epromer is a ST6 microcontroller program for Windows that lets you view, edit and save executable files in the Hex (Intel) and .S19 (Motorola) formats, generated by the Assembler, Linker or C Compiler for ST microcontrollers. It also enables you to program executable files into ST microcontrollers and to view and verify a microcontroller's memory contents.

GNU

Acronym for Not UNIX. Name of a project developed by the Free Software Foundation.

LST6

The ST6 family linker. It links relocatable objects (assembled source file modules) into a single, executable file that can be loaded into the ST6 memory.

MCU

Microcontroller Unit. Otherwise referred to as the "target device" throughout this manual. This is the core product (or family of products) for which the Starter Kit is designed to act as an emulator or simulator. In general terms, an MCU is a complete computer system, including a CPU, memory, a clock oscillator and I/O on a single integrated circuit.

OTP

> One Time Programmable. Also referred to as OTPROM (One Time Programmable Read-Only Memory). A non-volatile type of memory that can be programmed but cannot be erased. An OTP ROM is an EPROM MCU that is packaged in an opaque plastic package—it is called a one-time programmable MCU because there is no way to expose the EPROM to a UV light source.

RS-232 Interface

> An interface that enables you to communicate with the pre-programmed ST6225 microcontroller provided with the Starter Kit.

Starter Kit

> A printed circuit board for microcontroller evaluation and programming. It has the ability to program one DIP EPROM or OTP part at a time. Different starter kits are available for programming different ST6 devices.

WGDB6

> Windows GNU Debugger for ST6. This is a software tool running under Windows. It is the main user interface when operating the ST6-HDS2 emulator, and is part of the ST6 software development tool chain (editor, assembler, linker and debugger). WGDB6 is supplied on the CD-ROM with the ST6-HDS2 emulator and must be installed on your development PC.

PRODUCT SUPPORT

If you experience any problems with this product or if you need spare parts or repair, contact the distributor or ST sales office where you purchased the product:

Getting prepared before you call

Collect the following information about the product before contacting ST or your distributor:

1   Name of the company where you purchased the starter kit.

2   Date of purchase.

3   Order Code: Refer to the side of your starterkit box. The order code will depend on the region for which it was ordered (i.e. the UK, Continental Europe or the USA).

4   Serial Number: The serial number is located on a label on the starter kit board.

5   Target Device: The sales type of the ST6 microcontroller you are using in your development.

Contact List

Note: *For **American and Canadian customers** seeking technical support the US/Canada is split in 3 territories. According to your area, contact the following sales office and ask to be transferred to an 8-bit microcontroller Field Applications Engineer (FAE).*

Canada and East Coast

STMicroelectronics
Lexington Corporate Center
10 Maguire Road, Building 1, 3rd floor
Lexington, MA 02421
Phone: 781-402-2650

Mid West

STMicroelectronics
1300 East Woodfield Road, Suite 410
Schaumburg, IL 60173
Phone: 847-517-1890

West coast

STMicroelectronics, Inc.
30101 Agoura Court
Suite 118
Agoura Hills, CA 91301
Phone: 818-865-6850

Europe

France (33-1) 47407575
Germany (49-89) 460060
U.K. (44-1628) 890800

Asia/Pacific Region

Japan (81-3) 3280-4120
Hong-Kong (852) 2861 5700
Sydney (61-2) 9580 3811
Taipei (886-2) 2378-8088

Index

www.BDTIC.com/ST

Index

www.BDTIC.com/ST