UM0002
User manual

Starter kit for ST625x and ST626x MCUs

Hardware features

- Immediate evaluation of all ST625x and ST626x devices, with demonstration examples.
- Software debugging within the user's real application environment.
- Programming of ST62T5x, ST62T6x and ST62E6x sales types (DIL packages).
- In-circuit programming of ST62T5x, ST62T6x and ST62E6x sales types on the user's application board (all packages).

Software features

- Software simulator including I/O read/write.
- Assembler, linker, debugger.
- OTP and EPROM programming utilities.
- Application examples and demonstrations.

Figure 1. Starter kit for ST625x, ST626x MCUs

Contents

List of figures

List of tables

1 Introduction

The ST626x starter kit provides you with all you need to start designing, developing and evaluating programs for ST6253x, ST6255x, ST6260x, ST6263x and ST6265x microcontrollers immediately.

The ST626x starter kit includes:

- The ST6 assembler and linker, AST6 and LST6.
- The ST6 Windows debugger, WGDB6.
- The Windows ST6 microcontroller programmer, Epromer.
- The ST6 starter kit board, which serves as a demonstration board and low-cost debugging tool.
- Some demonstration programs that show how ST6 microcontrollers use the starter kit board resources.
- Some example programs.
- One ST62E60BF1, two ST62E62BF1 and two ST62E65BF1 sales types.
- A complete set of paper documentation and online help.

The demonstration programs, that come pre-loaded on an ST62E65, 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, temperature control circuit and an RS-232 interface.

Using the ST6 assembler and linker, AST6 and LST6, you can assemble and link ST6 programs. The "ST6 Family Software development tools AST6, LST6, WGDB6" User Manual will guide you through the steps of developing, assembling and linking programs for the ST6. The starter kit software includes a set of example programs of typical ST6 applications. These are installed in the directory **C:\st6tools\sk626Xi1\examples**.

For a fast-track solution for developing bug-free programs for the ST6, without the hassle of writing assembler code, try out the ST6-Realizer program.

Once you have developed your ST6 program, you can use the Windows-based ST6 program debugger, WGDB6/SIMULATOR, 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 "ST6 Family Software development tools AST6, LST6, WGDB6" User Manual and online help will lead you through the debugging process using WGDB6.

When 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.

Instructions for use - Warning

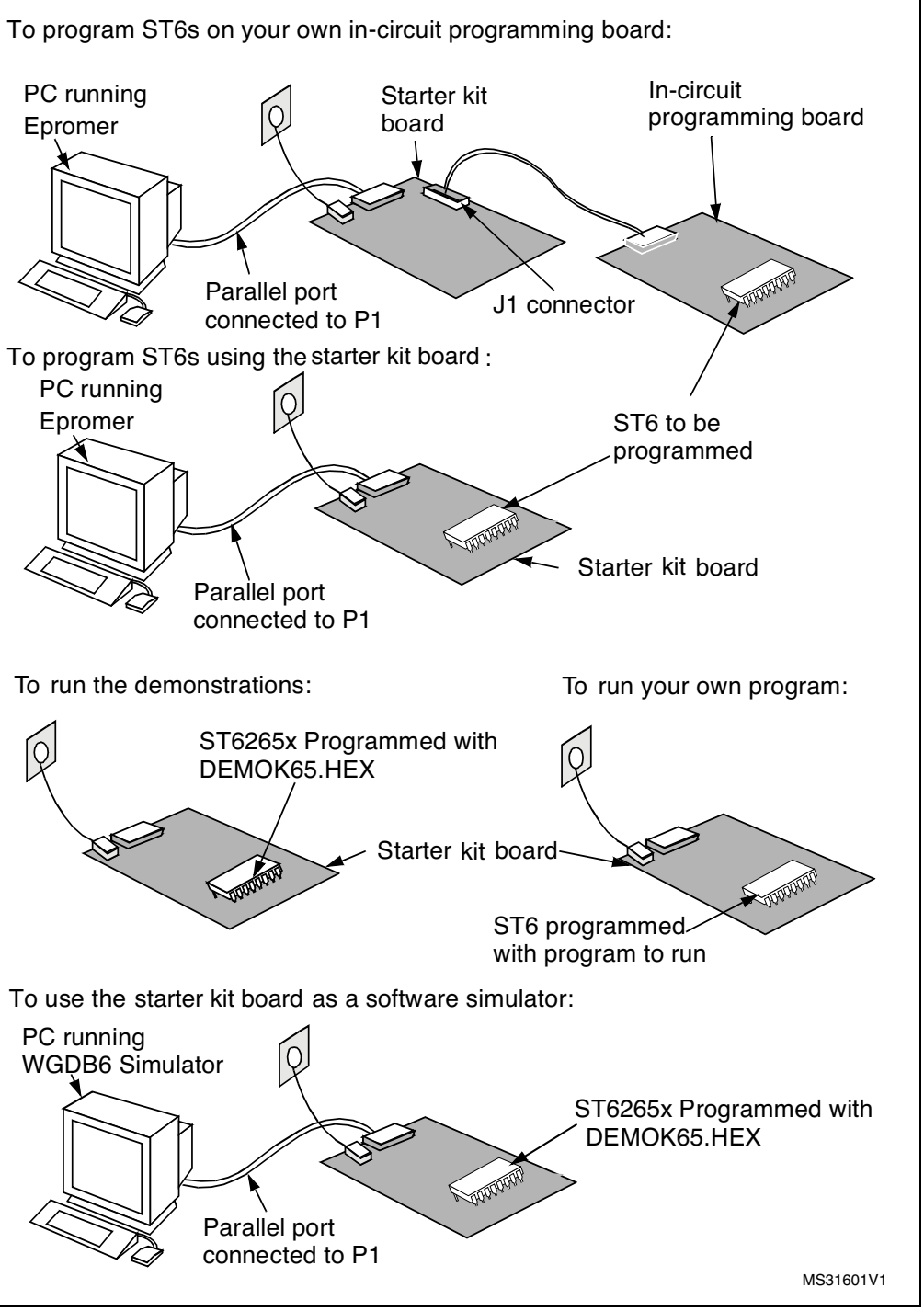
This product conforms with the 89/336/EEC directive; it also complies with the EN55022 emissions standard for ITE, as well as with generic 50082-1 immunity standards.

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**.

The following diagram summarises the possible uses of the starter kit board and the hardware setup required for each one.

Figure 2. Possible uses and hardware setup



1.1 Where to go from here...

The following table directs you to where you should look for further information about using the ST6 starter kit.

Table 1. ST6 starter kit references

To:	Refer to:
Find out about the starter kit board and ST6 microcontrollers provided with the kit.	*The starter kit board on page 10* of this book.
Install the starter kit software, and connect the power supply to the board.	*Installing the starter kit on page 20* of this book.
Find out what the demonstration applications do, and run them.	*Running the demos on page 21* of this book.
Learn how to develop source code for AST6 and LST6.	"ST6 Family Software development tools AST6, LST6, WGDB6" User Manual.
Prepare the starter kit board for use as an ST6 hardware simulator with WGDB6.	*Using the starter kit board as a hardware simulator on page 28* of this book.
Learn how to use WGDB6 for debugging your programs.	"ST6 Family Software development tools AST6, LST6, WGDB6" User Manual.
Prepare the starter kit board for programming ST6 microcontrollers using Epromer.	*Programming ST6 microcontrollers on page 32* of this book.
Prepare the starter kit board for connecting your own in-circuit programming board.	*In-circuit programming on page 36* of this book.
Learn how to use Eprommer for programming ST6 microcontrollers	The Epromer online help.
Connect your own hardware resource to the starter kit board.	*Connecting external resources to the starter kit board on page 26* of this book.
Run your own program on an ST6 using the starter kit board.	*Running your own program on the starter kit board on page 39*

2 The starter kit hardware

This section describes the ST6 microcontrollers and the starter kit board that come with the ST6 starter kit. A full schematic of the starter kit board is provided in *Hardware information on page 41*.

2.1 The ST6 microcontrollers

The starter kit includes the following sales types:

- Two ST62E65BF1
- One ST62E60BF1
- Two ST62E62BF1

The ST6265x microcontroller labelled DEMOK65 is pre-loaded with the code **DEMOK65.HEX**, which includes the demonstration programs (see *Running the demos on page 21*), as well as the communications protocol program, that enables you to use the starter kit board as a simulator (see *Using the starter kit board as a hardware simulator on page 28*). The other ST6 microcontrollers are blank.

The file Demok65.hex is in the directory C:\st6tools\sk626Xi1, so that if you erase it from the ST6265x, you can re-program it following the instructions given in *Programming ST6 microcontrollers on page 32*.

2.2 The starter kit board

The starter kit board includes the following resources:

- Reset and data control buttons
- LED indicators
- Resistance trimmer
- Analog to digital converter
- Audio transducer 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: 16V 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 36* for further details.

Below is a block diagram of the starter kit board:

Figure 3. Starter kit board block diagram

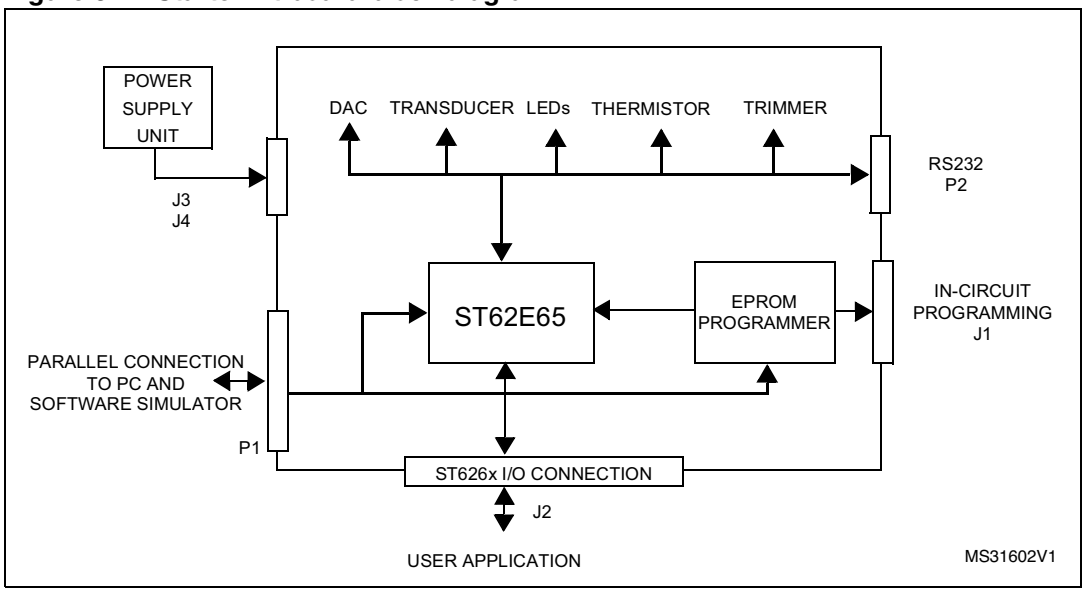
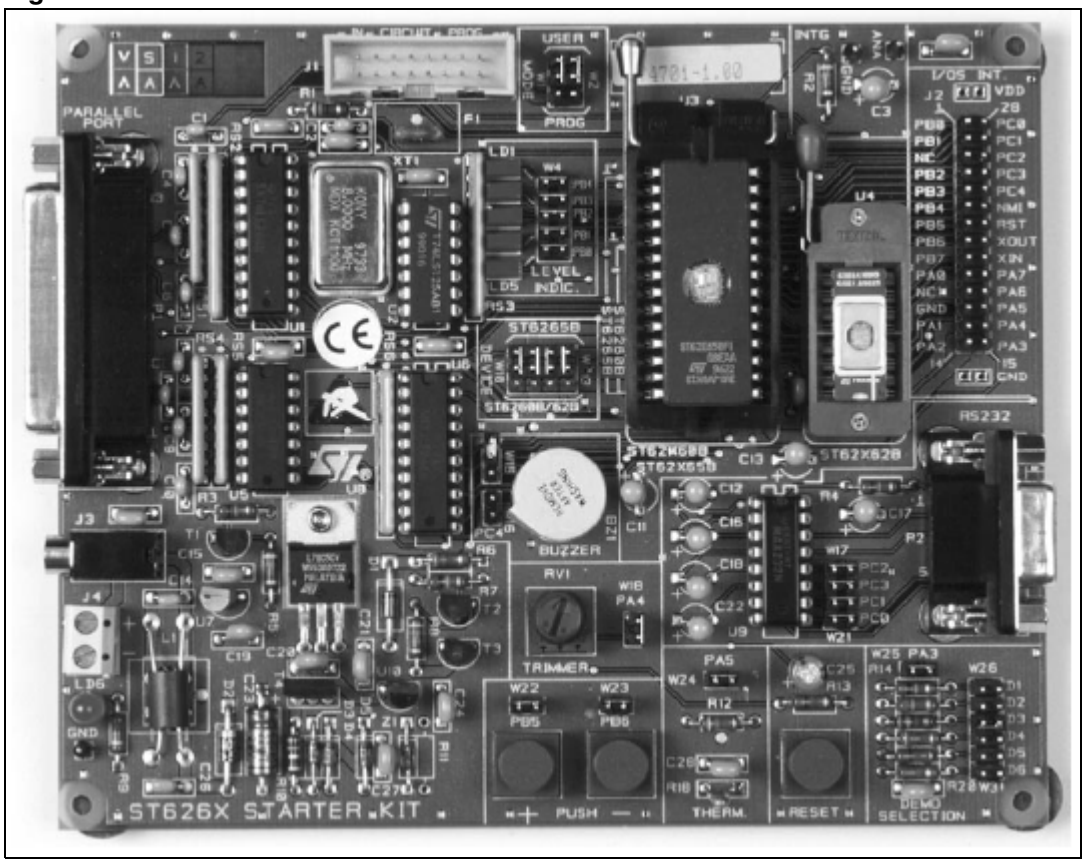


Figure 4. Starter kit board



The following diagram shows the layout of the starter kit board.

Figure 5. Starter kit board diagram

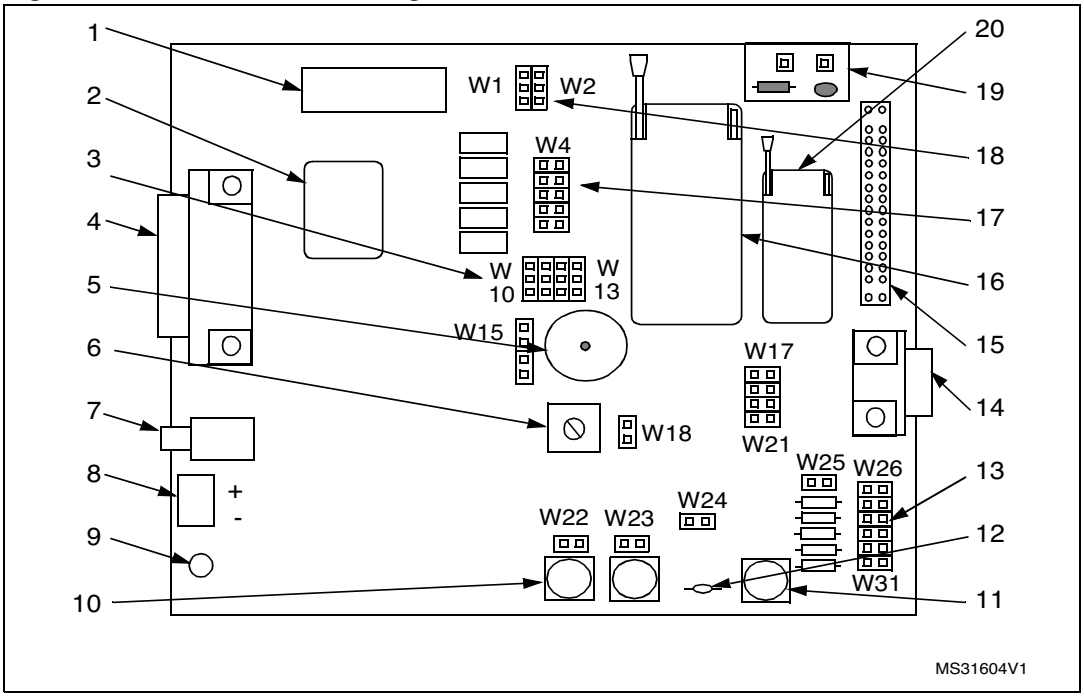


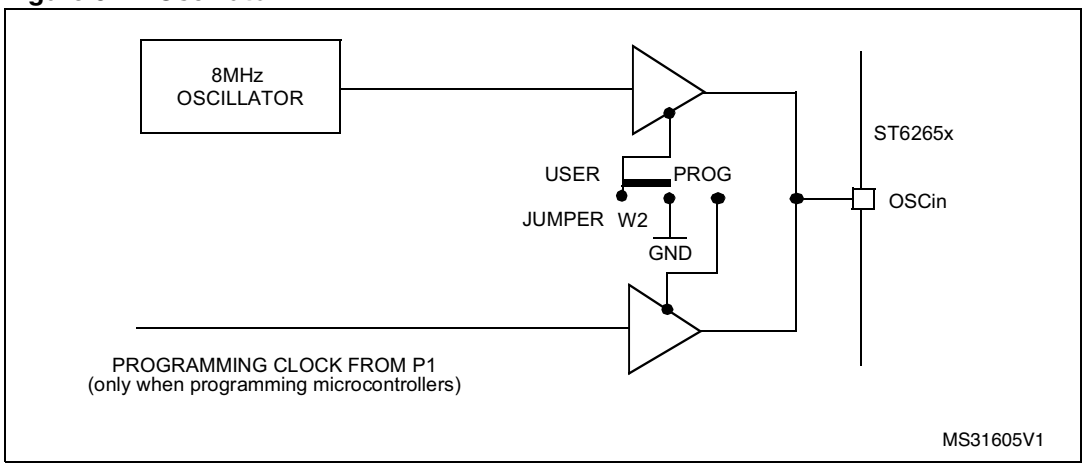
Table 2. Legend

1	In-circuit programming connector (J1)	20	DIL-16 ZIF MCU socket
2	8 Mhz Oscillator	19	Digital to analog conversion circuit
3	“ST6260x/62x” or “ST6265x” device selection jumpers W10 to W13	18	“Programming” or “User” operating mode selection jumpers W1 and W2
4	PC connector P1	17	Five LED level indicators including jumpers W4 to W8
5	Audio Transducer circuit	16	DIL 20-28 ZIF MCU socket
6	10 KΩ trimmer	15	Remote resource I/O interface connector J2
7	Power supply JACK connector J3	14	RS232 interface circuit and connector
8	Power supply connector J4	13	Demonstration routine selector
9	Power supply LED indicator LD6	12	Thermistor including jumper W24
10	“+” and “-” buttons	11	RESET button

2.3 Oscillator

An oscillator feeds the ST6265x OSCIN input with an 8 MHz clock signal. You can disconnect the oscillator by removing the jumper W2 (marked 18 on the *Starter kit board diagram on page 13*).

Figure 6. Oscillator



2.4 Reset button

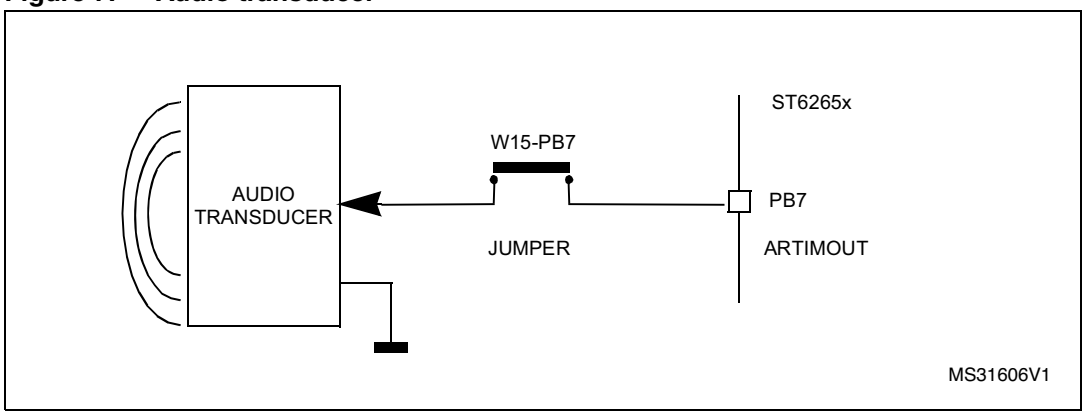
This activates the ST6265x RESET input when pressed. A power-on reset circuit is also provided.

2.5 Audio transducer

A piezoelectric audio transducer is connected to the ST6265x Audio Reload Timer (PB7 pin). It is provided to demonstrate and evaluate the sound generation capabilities.

The transducer can be disconnected from PB7 pin by removing the W15-PB7 jumper.

Figure 7. Audio transducer



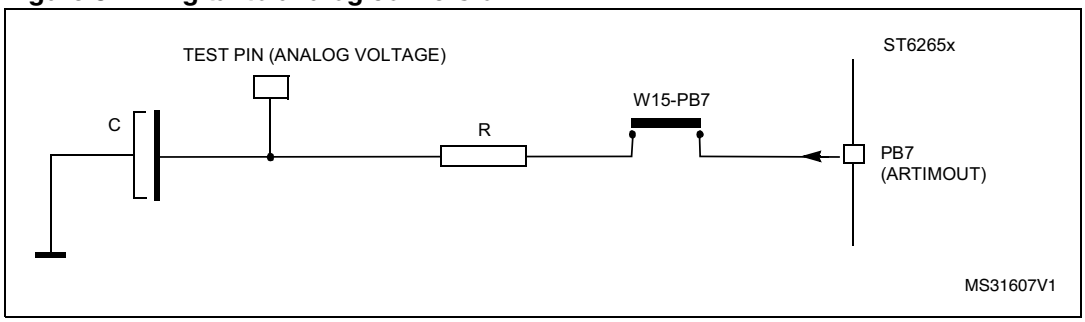
2.6 Digital to analog conversion

Digital to analog conversion is performed using an RC integrator circuit connected to the ST6265x auto reload timer output (PB7).

A PWM signal is generated by the ARTimer at PB7 output. The voltage value at the output of the RC integrator circuit is controlled by the PWM duty cycle.

The RC circuit input can be disconnected from PB7 output by removing the W15-PB7 jumper.

Figure 8. Digital to analog conversion

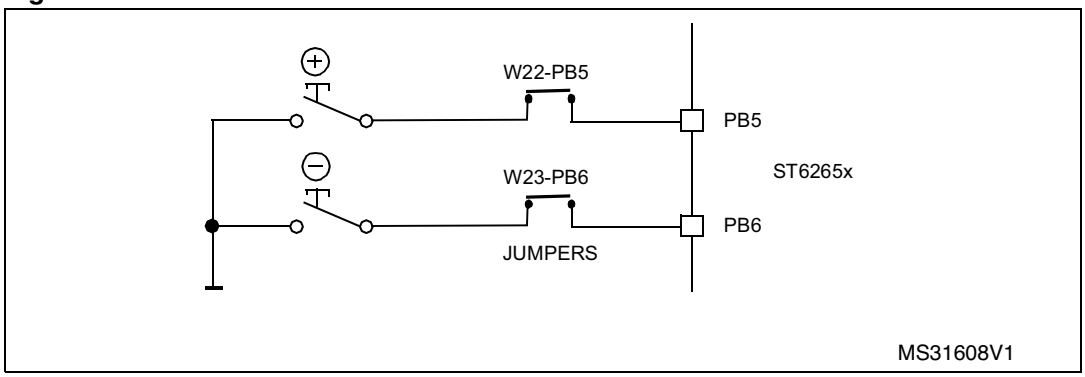


2.7 + and - buttons

These are connected to PB5 and PB6 pins on the microcontroller respectively. They drive the PB5 and PB6 inputs down to GND when pressed.

You can disconnect PB5 or PB6 by removing the appropriate W22 or W23 jumper.

Figure 9. + and - buttons

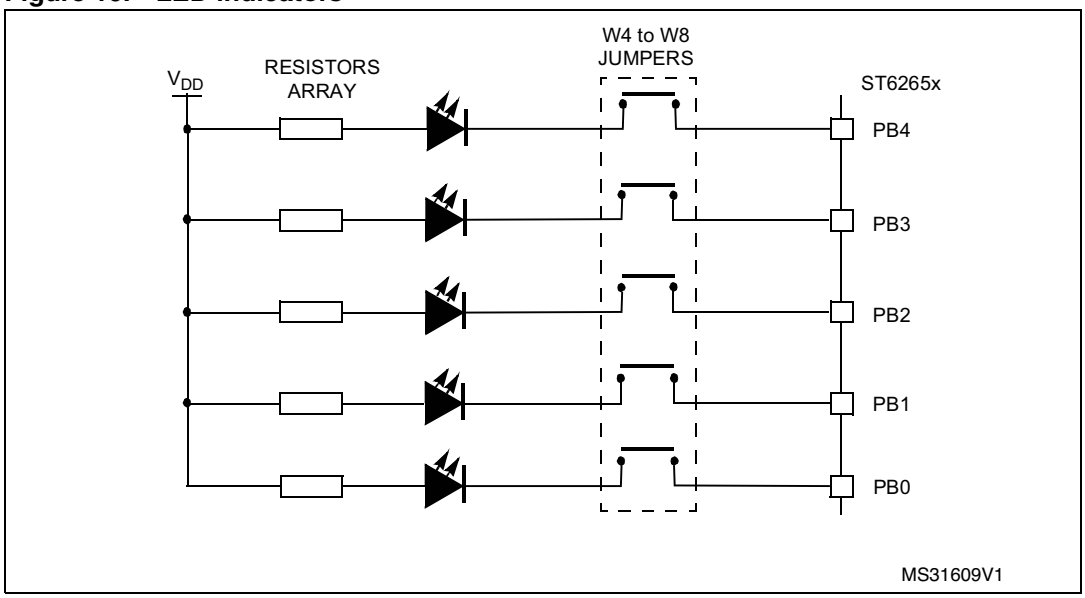


2.8 LED indicators

Five LEDs are used for level indication in the demonstration routines. They are connected to the ST6265x pins: PB0 to PB4 (when programmed as outputs) to demonstrate direct LED-driving capability.

The five LEDs can be disconnected by removing the W4 to W8 (PB4 to PB0) jumpers.

Figure 10. LED indicators

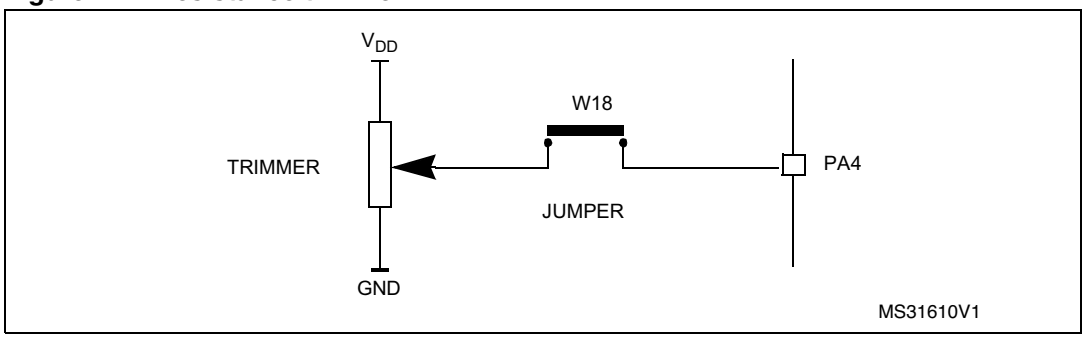


2.9 Resistance trimmer

A 10 KΩ resistance trimmer feeds the ST6265x PA4 I/O pin (when programmed as an A/D Converter input) with a variable voltage (0 to 5V DC). It is used for A/D conversion demonstration/evaluation.

The trimmer can be disconnected from the I/O pin by removing the W18-PA4 jumper.

Figure 11. Resistance trimmer

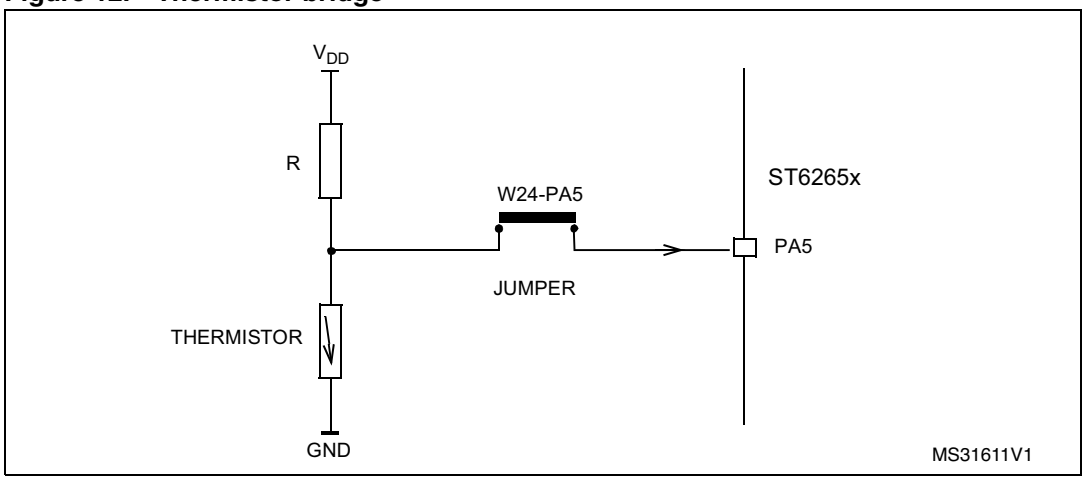


2.10 Thermistor bridge - Temperature control

A thermistor bridge (with a negative coefficient) is connected to the ST6265x I/O pin PA5, which is defined as an analog to digital converter (ADC) input. The voltage value at the ADC input decreases when the temperature is increased. Demonstration 4 shows how the thermistor is used to indicate temperature levels on the five LEDs (see *Running the demos on page 21*)

The thermistor bridge can be disconnected from the pin PA5 by removing the W24-PA5 jumper.

Figure 12. Thermistor bridge



2.11 RS-232 interface

The RS-232 interface enables you to communicate with the pre-programmed ST6265x 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:

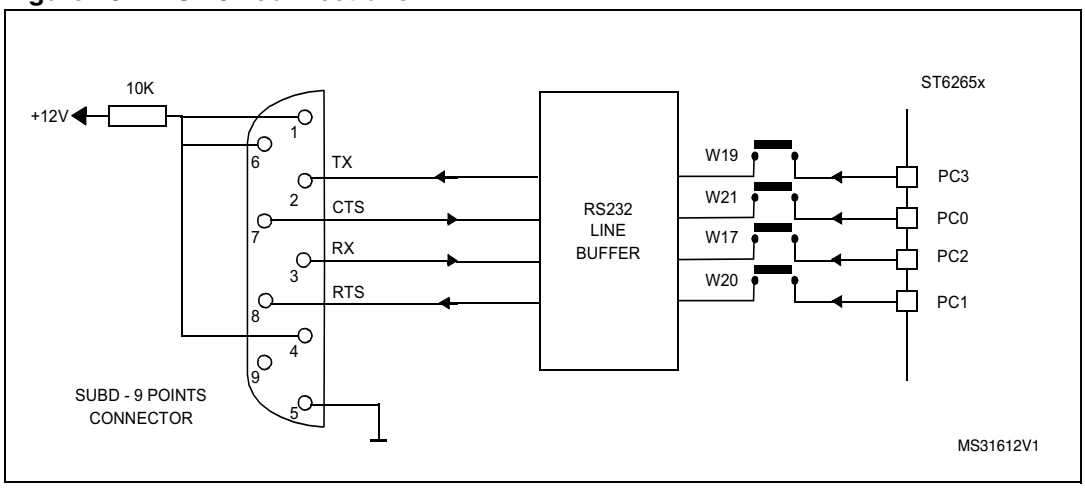
Table 3. RS-232/ST-6 pin connections

Signal name	SUBD-9 Pin	ST6 Pin
Data Transmission (TX)	2	PC3
Data Reception (RX)	3	PC2
Request to Send (RTS)	8	PC1
Clear to Send (CTS)	7	PC0

You can disconnect these by removing the corresponding jumpers from W17 to W21.

The following diagram shows the RS-232 connections and line allocations:

Figure 13. RS-232 connections



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.

2.12 Demonstration selector jumpers

The demonstration selector is made up of 6 jumpers W26 to W31 (marked 13 on the *Starter kit board diagram on page 13*), 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 ST6265x. PA3 is programmed as an A/D converter. The program installed on the ST6265x uses the input from PA3 to select the appropriate demo. The following table lists the voltage values generated by each resistor:

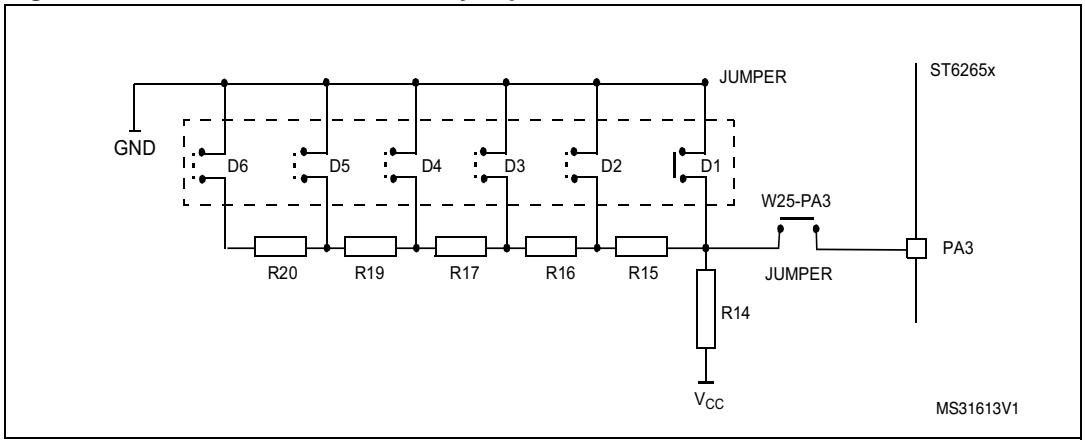
Table 4. Resistor and voltage values

Resistor value	Theoretical voltage value
R14: 10 KΩ	No JUMPER: 5 V
R15: 680 Ω	JUMPER-D1: 0 V
R16: 750 Ω	JUMPER-D2: 333 mV
R17: 820 Ω	JUMPER-D3: 666 mV
R19: 1 KΩ	JUMPER-D4: 999 mV
R20: 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 application note: “AN431: Using ST6 Analog Inputs for Multiple Key Decoding”.

You can disconnect the demonstration selector from PA3 by removing the PA3 jumper from W25.

Figure 14. Demonstration selector jumpers



3 Installing the starter kit

3.1 Hardware and software requirements

To be able to install and run the ST6 starter kit, you need a PC with:

- A 3 1/2” Floppy Disk Drive
- A CDROM Disk Drive
- 1.5 Mbytes free memory space
- A free Centronics compatible parallel port connector
- MS-WindowsTM 3.11, NT or 95.

3.2 Installing the software

If diskettes are provided, you must install the software with them in order to have the latest release:

1. Place the SK626XI1 diskette into your floppy disk drive.
2. In Windows Explorer or File Manager, view the contents of the diskette, then double-click the Setup file or icon.
3. Follow the instructions as they appear on screen.

If only the ST62 CDROM is provided, then:Place the ST62 CDROM provided into your CDROM disk drive.

1. In Windows Explorer or File Manager, view the contents of the CDROM, browse to st62oncd\ftools\sk626Xi1 and double-click the Setup file or icon.
2. Follow the instructions as they appear on screen.

3.3 Connecting the power supply

If you have 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.

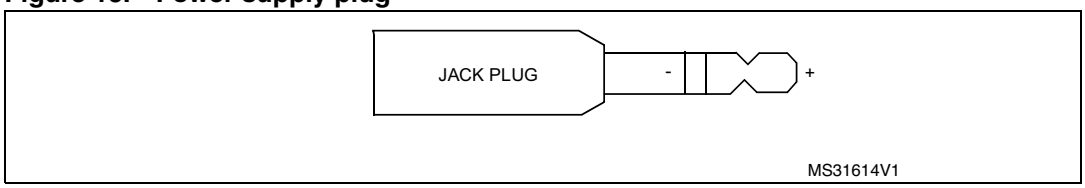
If you have DC mains supply, connect the male plug on the power supply cable provided to the J3 input socket, then connect the mains plug to a mains source with the following characteristics:

- 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:

Figure 15. Power supply plug



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 24* below for details on how to select and run a demo.

The source files of these demos are provided with the starter kit software in the file **C:\st6tools\sk626Xi1\sk626Xli\DEMOK65.ASM**.

4.1.1 Demo 1 - Sound generation

After RESET, this program generates a PWM signal at the output of the AUTO RELOAD TIMER peripheral (PB7), which is connected to the Audio Transducer.

The frequency of the PWM signal can be adjusted by pressing the + (increase) or the - (decrease) pushbuttons.

An oscilloscope probe can be positioned on the W15-PB7 jumper to observe the PWM signal.

4.1.2 Demo 2 - Music box

After RESET, this program produces 5 tunes that are played by the Audio Transducer. The sound frequencies are generated at the AUTO RELOAD TIMER output peripheral (PB7), used in PWM mode.

The LED that is turned on indicates the tune to be played (1 through 5). To select the tune to play, press the + button.

The music starts playing when the - button is pressed.

The tempo of the music can be modified using the voltage trimmer (marked 6 on the *Starter kit board diagram on page 13*). This is connected to PA4 I/O programmed as Analog input.

Once the music has finished playing, another tune can be selected and played the same way.

4.1.3 Demo 3 - Voltage trimming and LED level indication

1. Initialises the pins as follows:

This pin:	Is initialised as:
PA4	Analog input. Connected to the trimmer.
PB0 to PB4	Push-pull outputs. Connected to the five 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 1V	LD5
1 to 2V	LD4
2 to 3V	LD3
3 to 4V	LD2
4 to 5V	LD1

Adjusting the voltage trimmer (marked 6 on the *Starter kit board diagram on page 13*) turns on the appropriate LED.

4.1.4 Demo 4 - Temperature control

1. Initialises the pins as follows:

This pin:	Is initialised as:
PA5	Analog input. Connected to the thermistor circuit.
PB0 to PB4	Push-pull outputs. Connected to the five 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 increase the temperature by touching the thermistor (marked 12 on the *Starter kit board diagram on page 13*).

4.1.5 Demo 5 - Digital to analog conversion

Digital to Analog Conversion (DAC) is performed using an RC integrator circuit connected to the ST6265x Auto Reload Timer output (PB7).

A PWM signal is generated by the ARTimer at PB7 output. The voltage value at the output of the RC integrator circuit is controlled by the PWM duty cycle.

After RESET, the program:

- Generates a 64 KHz frequency PWM signal at the Auto Reload Timer output. The duty cycle is initialized at 50%.
- Decreases/increases the duty cycle value each time the - or + button is pressed (2% steps). This modifies the analog voltage at the integrator circuit output.

For more precise observation, position an oscilloscope probe on the ANA test point, located in the Digital to Analog circuit (marked 19 on the *Starter kit board diagram on page 13*).

4.1.6 Demo 6 - RS-232 communications

This demonstration shows how an RS-232 communication line buffer can be managed using an ST6265x microcontroller.

To run this demonstration:

1. Connect the RS-232 connector on the starter kit board to a serial port on your PC using the RS-232 cable provided.
2. On the host PC, in MS-DOS, execute the program: ST6K232.EXE which is in the st6tools\sk626Xi1 directory.
3. Follow the instructions as they appear on screen.

4.2 Running the demonstration programs

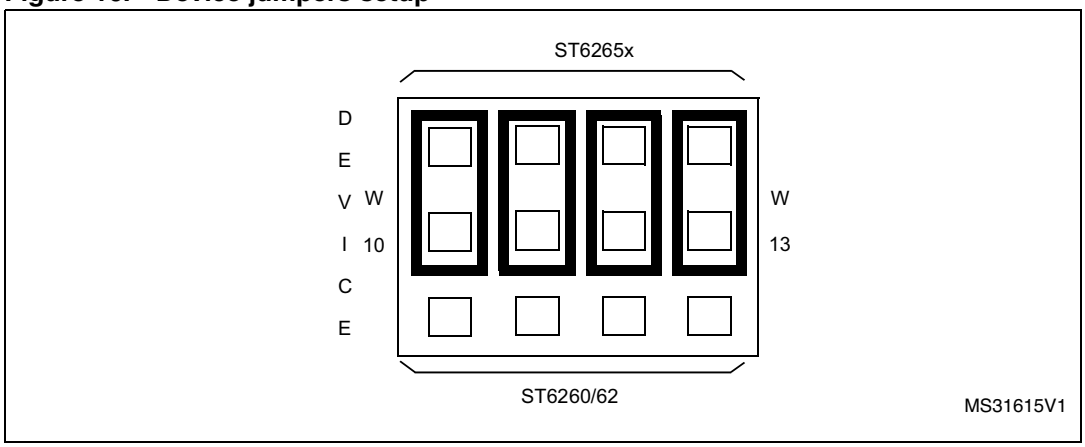
The ST6265x microcontroller labelled DEMOK65 is programmed with the demonstration software.

If this software has been erased from the microcontroller, you can reprogram it from the file **DEMOK65.HEX** (the file is in the st6tools\sk626Xi1 directory). For details of how to program microcontrollers refer to *Programming ST6 microcontrollers on page 32*.

To run the demonstrations:

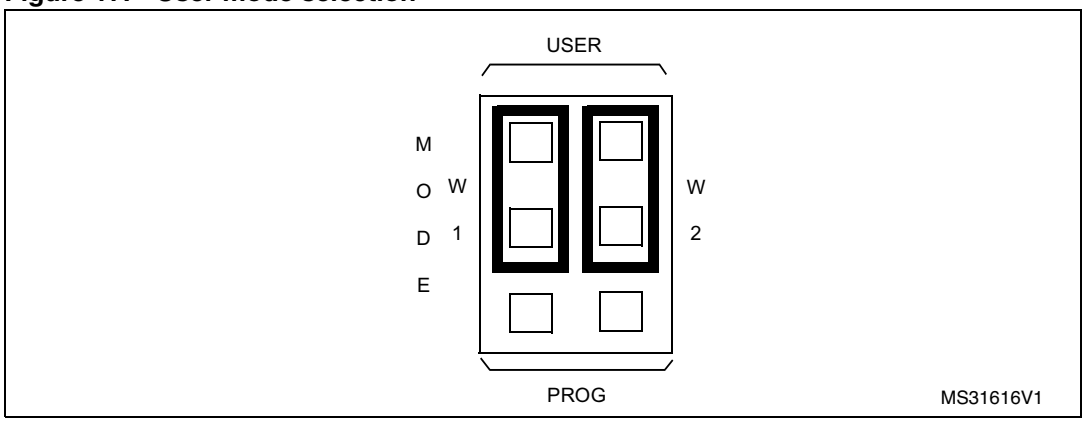
1. Power down the starter kit board.
2. Make sure that the pre-programmed ST62E65 is plugged into the DIL connector, and that the DEVICE jumpers W10 to W13 (marked 3 on the *Starter kit board diagram on page 13*) are set to ST6265x as shown in the following diagram:

Figure 16. Device jumpers setup



3. Select the USER mode using the jumpers marked W1 and W2 (marked 18 on the *Starter kit board diagram on page 13*), as shown in the diagram below:

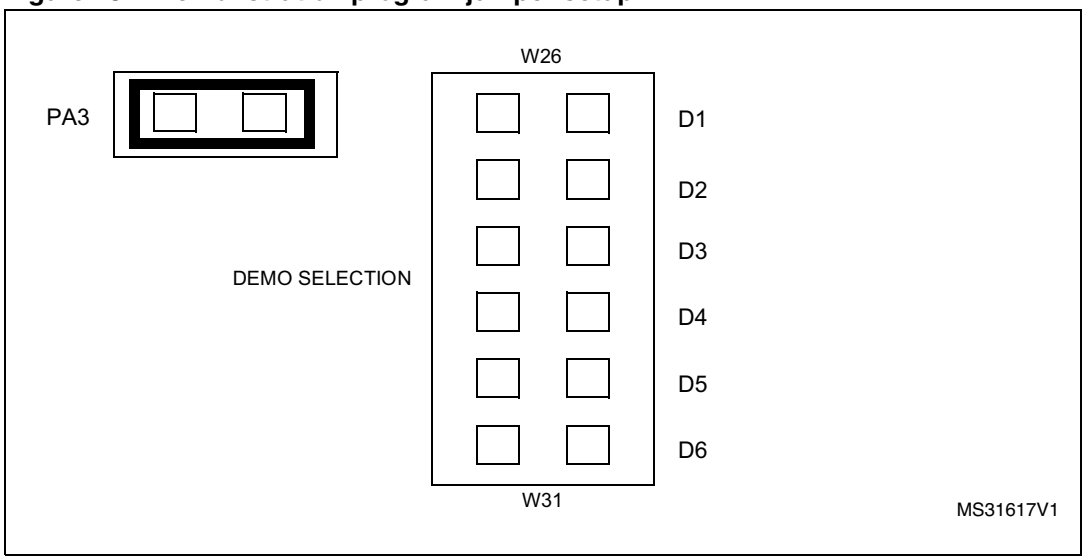
Figure 17. User mode selection



4. Disconnect the cable from the parallel port (P1) connection, if it is connected.
5. Power up the starter kit board.

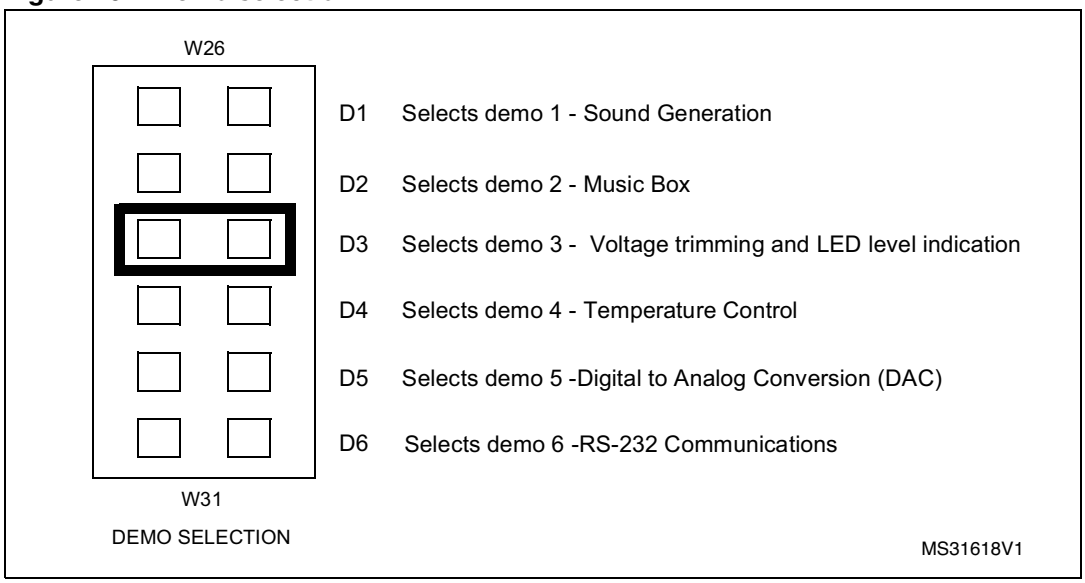
6. Install the demonstration program jumper marked PA3, as shown in the diagram below:

Figure 18. Demonstration program jumper setup



7. Select the demo you want to run, by installing the appropriate jumper W26 to W31 (marked 13 on the *Starter kit board diagram on page 13*), as indicated on the diagram below:

Figure 19. Demo selection



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

8. Press the reset button.

The selected demo is now ran.

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 ST6265x to debug or evaluate your programs, using the connector J2 (marked 15 on the *Starter kit board diagram on page 13*). To be able to connect your own resources to the starter kit board, you must disconnect the resources that are already connected to the ST6265x, to avoid external resource/ starter kit board resource conflicts. The following table lists the starter kit board resources and the corresponding J2 connections, and indicates the jumper that disconnects each resource.

Table 5. Board resources and connections

On-board resource	JP	ST6265x Connector	PIN	PIN	ST6265x Connector	JP	On-board resource
LED level indicator	W8 PB0	PB0	1	28	PC0	W21 PC0	RS232 CTS
LED level indicator	W7 PB1	PB1	2	27	PC1	W20 PC1	RS232 RTS
–	–	nc	3	26	PC2	W17 PC2	RS232 RX
LED level indicator	W6 PB2	PB2	4	25	PC3	W19 PC3	RS232 TX
LED level indicator	W5 PB3	PB3	5	24	PC4	W16 PC4	Not used
LED level indicator	W4 PB4	PB4	6	23	NMI(1)	–	System Tasks
Pushbutton +	W22 PB5	PB5	7	22	RESET/(1)	–	Reset push. Power-on
Pushbutton -	W23 PB6	PB6	8	21	OSCOUT	–	System Tasks
Audio Transducer	W15 PB7	PB7	9	20	OSCIN	–	8MHz Osc.
System Tasks	–	PA0(1)	10	19	PA7		None
–	–	nc	11	18	PA6	–	Not used
–	–	GND	12	17	PA5	W24 PA5	Thermistor
System Tasks	–	PA1(1)	13	16	PA4	W18 PA4	Trimmer
System Tasks	–	PA2(1)	14	15	PA3	W25 PA3	Demonstrati on Selector

1. 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 conencted 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 ST6252x, ST6253x, ST6255x, ST6260x, ST6262x, ST6263x or ST6265x microcontroller. Note, however that you must use the pre-programmed ST6265x microcontroller, labelled DEMOK65 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.

6.1 The data transmission driver

Data is transferred between the simulated peripheral registers and the ST626x registers via the host PC's parallel port. The DEMOK65.HEX program, with which the ST6265x microcontroller that is plugged into the starter kit board must be loaded includes the transmission driver.

The data transfer driver uses the following bits:

Table 6. Data transmission

PC parallel port	ST6265x MCU	WGDB6 Use
D2	PA2	Synchronisation
D3	PA1	Write data to MCU
D4	RESET	Hardware reset of peripherals
D6	NMI	Initiates data transfer
SDOP	PA0	Read data from MCU

Note: Do not connect any external resources to the corresponding J2 connector pins when using the starter kit board as a peripheral emulator.

6.2 Technical limitations

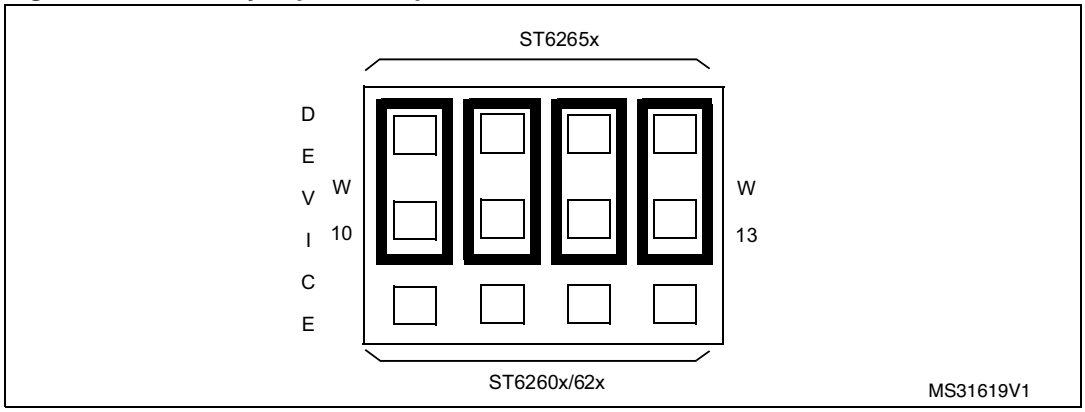
The starter kit board has the following limitations when used with WGDB6 as a hardware simulator:

- Real-time program execution is not supported.
- Resetting the ST6265x 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 ST6265x microcontroller are not supported by the WGDB6 simulator.
- The pins: NMI, PA0, PA1 and PA2 on the ST6265x microcontroller are used for communications with the host PC, and are thus not available for simulation.
- You cannot modify the D0, D1 and D2 bits of the DDRA, ORA and DRA registers.

To use the starter kit board as a hardware simulator:

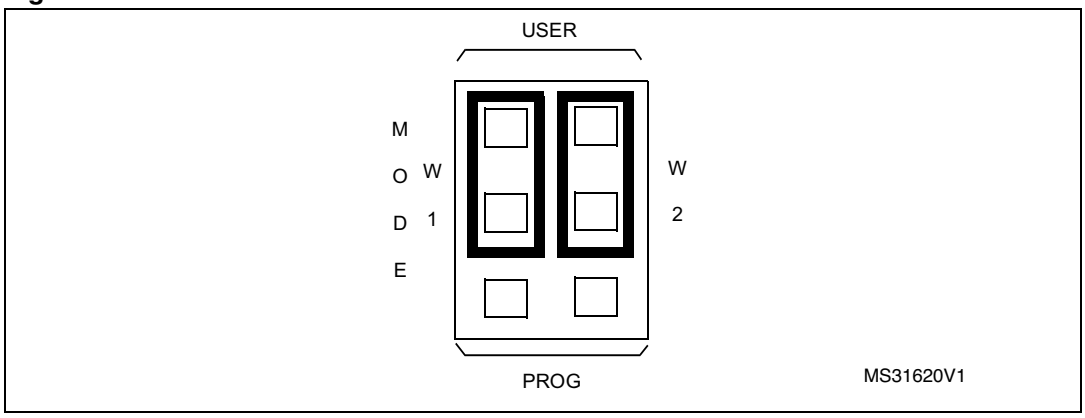
1. Power down the starter kit board.
2. Make sure that the pre-programmed ST6265x is plugged into the DIL socket U3, and that the DEVICE jumpers W10 to W13 (marked 3 on the *Starter kit board diagram on page 13*) are set to ST6265x as shown in the following diagram:

Figure 20. Device jumpers setup



3. Select the USER mode using the jumpers marked W1 and W2 (marked 18 on the *Starter kit board diagram on page 13*), as shown in the diagram below:

Figure 21. User mode selection



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:

• If you are using Windows 95, click the **Start** button, point to **Programs**, then **ST6 Tools**, then click **WGDB6/Simulator**.

• If you are using Windows 3.x, double-click the appropriate **WGDB6/Simulator** icon in the **ST6 Tools** program group.

Refer to "WGDB6 User Guide" in the "ST6 Family Software Development Tools AST6, LST6, WGDB6" User Manual for full instructions on how to use WGDB6.

6.3 Error messages

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

Table 7. Error messages

Error message	Description
Error 116 Port A protected when using board.	This means that WGDB6 tried to access the PORT A registers. These are used for communications with the board.
Error 117 Communication error with ST626x board.	This means that a problem occurred during communcations between the host PC and the board. Perform the checks listed below.

6.4 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 shoud check the following:

- That the starter kit board is correctly powered up.
- That the parallel port cable is correctly connected.
- That the device jumpers W1and W2 are in the USER position.
- That the device type selection jumpers W10 to W13 are in the ST6265x position.
- That an ST6265x is plugged into the starter kit board, and it is programmed with DEMOK65.HEX.

7 Programming ST6 microcontrollers

You can use the starter kit board, in conjunction with the program Epromer, to program ST6252x, ST6253x, ST6255x, ST6260x, ST6262x, ST6263x or ST6265x microcontrollers. You can also perform in-circuit programming of ST6252x, ST6253x, ST6255x, ST6260x, ST6262x, ST6263x or ST6265x 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 13*).

7.1 Programming signals

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

Table 8. Programming signals

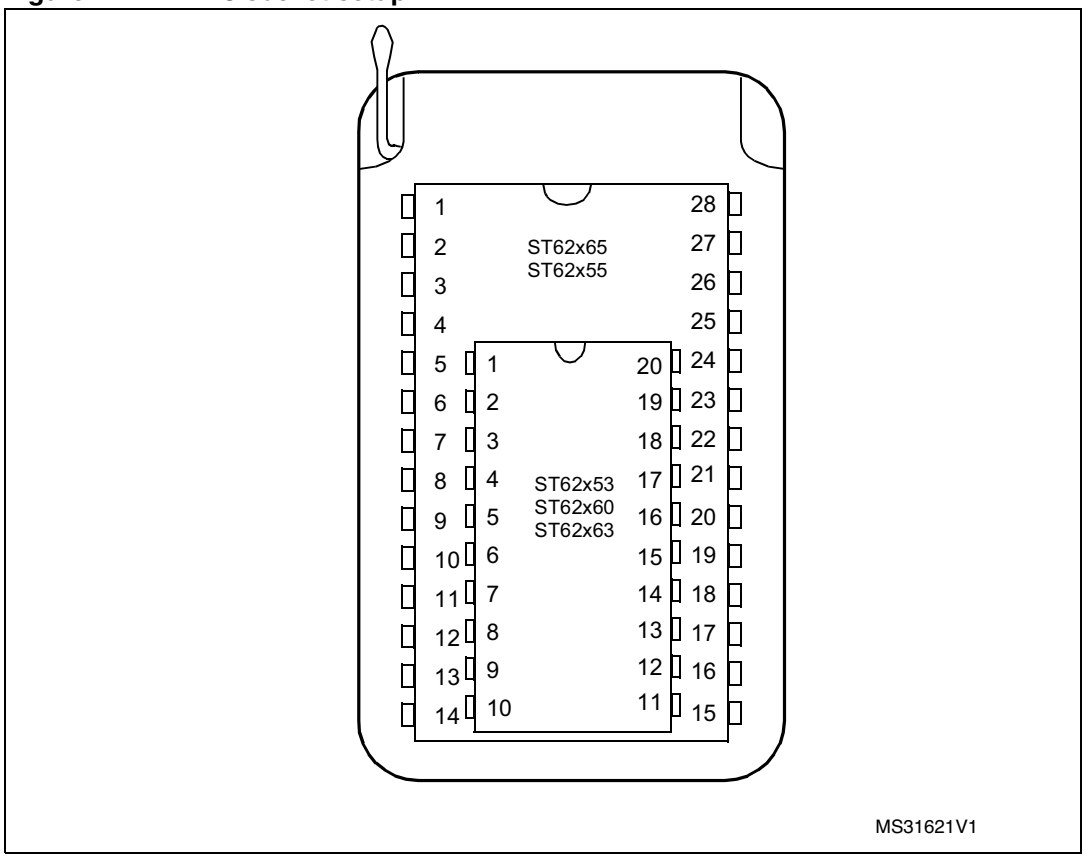
Programming signals	Programming states	ST62x65 MCU		ST62x60 MCU		ST62x62 MCU	
		MCU pin	J2 connector	MCU pin	J2 connector	MCU pin	J2 connector
PB0	High	Pin 1	Pin 1	Pin 1	Pin 5	Pin 1	Pin 5
PB2	High/Low	Pin 4	Pin 4	Pin 4	Pin 8	Pin 3	Pin 8
PB3	High/Low	Pin 5	Pin 5	Pin 5	Pin 9	Pin 4	Pin 9
TEST	5V/12V	Pin 3	n/c	Pin 3	Pin 7	Pin 2	Pin 7
OSCIN	High/Low	Pin 20	Pin 20	Pin 14	Pin 18	Pin 11	Pin 18
OSCOUT	High	Pin 21	Pin 21	Pin 15	Pin 19	Pin 12	Pin 19
RESET	High/Low	Pin 22	Pin 22	Pin 16	Pin 20	Pin 13	Pin 20

Note: The PB2, PB3, OSCIN and RESET signals on the J2 connector are used during ST6 programming, thus if you have connected an external resource to J2, you must disconnect these signals before using the starter kit board for programming.
The PB2, PB3, OSCIN and RESET pins are used by the system during programming. The programming signals are therefore switched to different pins of the DIL-28 socket.

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 DIL-28 socket (U3) or the DIL-16 socket (U4). The DIL-16 socket (U4) is only used to program ST6252x and ST6262x devices. Use the DIL-28 socket (U3) according to the following diagram:

Figure 22. DIL-18 socket setup



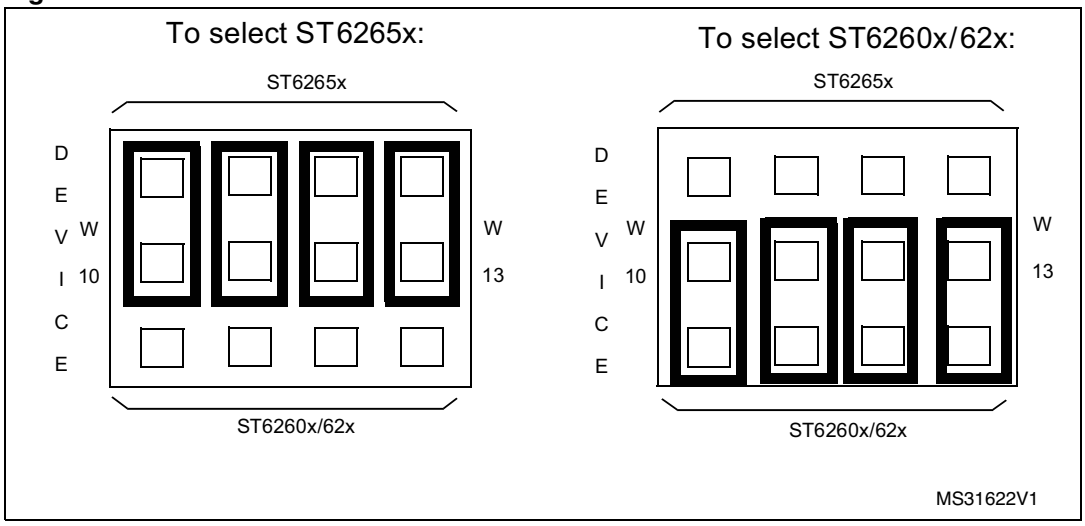
3. Set the DEVICE jumpers W10 to W13 (marked 3 on the *Starter kit board diagram on page 13*) for the microcontroller that you want to program according to the following table:

Table 9. Setting the device jumpers W10 to W13

Device Type	W10 to W13 Position
ST6252x	ST6260x/62x
ST6253x	ST6260x/62x
ST6255x	ST6265x
ST6260x	ST6260x/62x
ST6262x	ST6260x/62x
ST6263x	ST6260x/62x
ST6265x	ST6265x

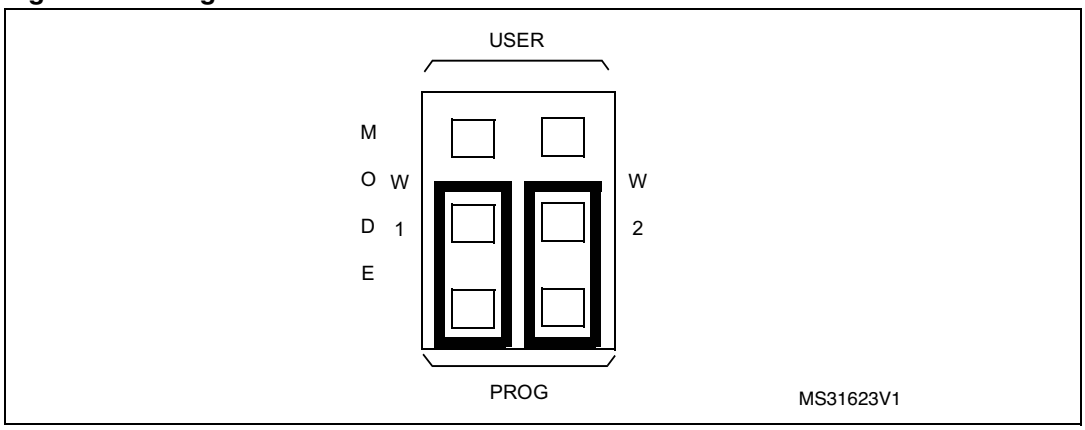
4. As shown in the following diagram:

Figure 23. Device selection



5. Select the PROG mode using the jumpers marked W1 and W2 (marked 18 on the *Starter kit board diagram on page 13*), as shown in the diagram below:

Figure 24. Prog mode selection



6. 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.
7. Power up the starter kit board.

You can now use Epromer to program the microcontroller that is plugged into the starter kit board.

Note: Epromer does not work under Windows NT.
To run Epromer from Windows 3.x, double-click the Epromer icon in the ST6 Tools group.
*To run Epromer from Windows 95, click **Start**, **Programs**, **ST6 Tools**, then **Epromer**.*
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 ST6252x, ST6253x, ST6255x, ST6260x, ST6262x, ST6263x or ST6265x 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 13*).

7.3.1 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 13*) on the starter kit board.

The following table shows the required pin connections:

Table 10. Pin connections

ST626x/5x	Connector
PB3	Pin 1
Connected to P1 Pin 7	Pin 3
OSCin	Pin 5
PB2	Pin 7
RESET	Pin 9
VPP/TM	Pin 13
VDD	Pin 14, 16
VSS	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 mF.

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

OSCin

Synchronises 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 KOhm pull-down or left open (Internal pull-up). The capacitive load of the RESET pin should not exceed 1 mF.

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 PB3 and PB2 on the 16-way connector, and must be isolated from other nodes on the application board. Any direct connection to V_{DD}, GND or an output must be avoided. These pins may be connected to a CMOS input, a 2 KΩ pull-up, a 10 KOhm 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.

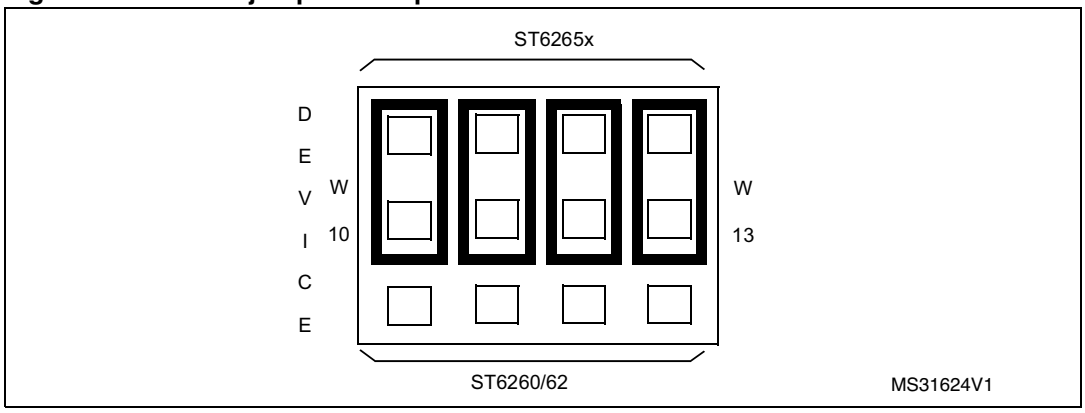
The signals on PB3 and PB5 (if not directly biased through pin 3 of the 16-way connector) must be kept at a high voltage level.

The **Vpp/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 Vpp/Test and V_{SS}.

7.4 Setting up the starter kit board for in-circuit programming

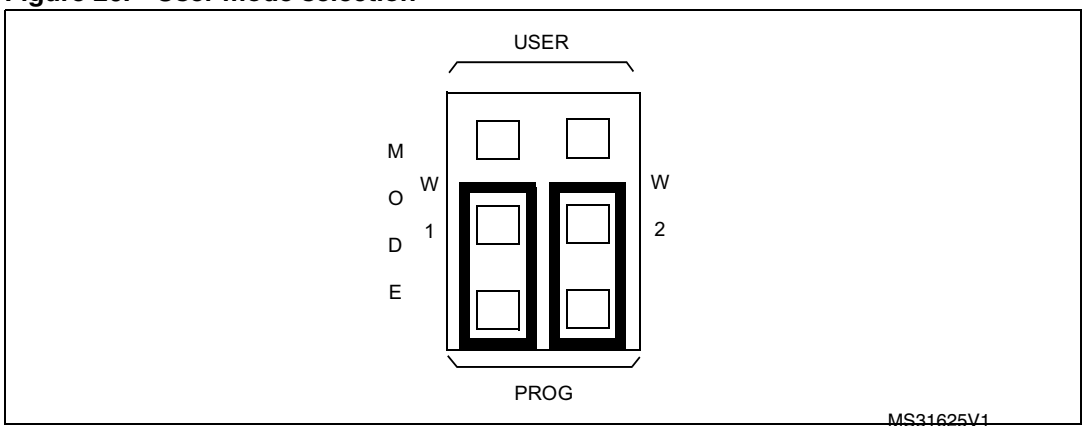
1. Power down the starter kit board.
2. Set the DEVICE jumpers W10 to W13 (marked 3 on the *Starter kit board diagram on page 13*) to ST6265x, as shown in the following diagram:

Figure 25. Device jumpers setup



3. Select the PROG mode using the jumpers marked W1 and W2 (marked 18 on the *Starter kit board diagram on page 13*), as shown in the diagram below:

Figure 26. User mode selection



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 13*) on the starter kit board.
6. Power up your starter kit board.

You can now use Epromer to program the microcontroller that is on your own board.

Note: Epromer does not work under Windows NT.
To run Epromer from Windows 3.x, double-click the Epromer icon in the ST6 Tools group.
*To run Epromer from Windows 95, click **Start**, **Programs**, **ST6 Tools**, then **Epromer**.*
*For instructions on how to operate Epromer, click **Help** in the Epromer main window.*
If your application board is not powered by the starter kit, you must connect it to a 5V 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 Kohm trimmer
- Audio Transducer
- + and - buttons
- Thermistor bridge
- Heater resistor control circuit
- Five LED level indicator

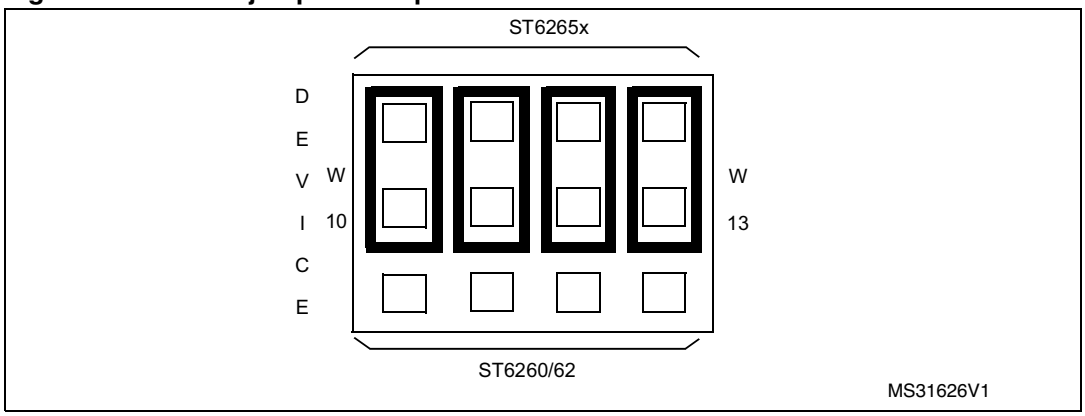
Note: You can only run applications on the starter kit board using ST6265x microcontrollers. If your application is designed for another microcontroller, you must change its port definitions to match those of the ST6265x.

You can also use your own hardware resource by connecting it to the connector J2 (see *Connecting external resources to the starter kit board on page 26*).

To run your own program on the starter kit board:

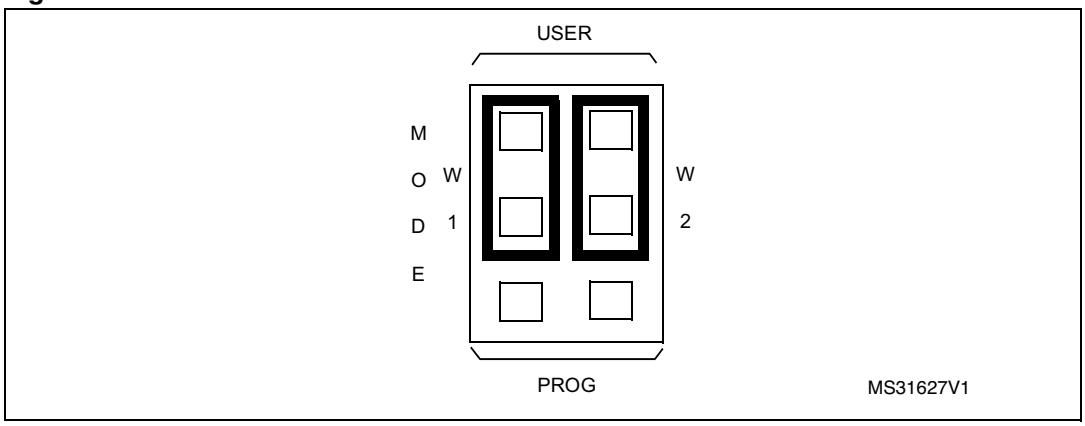
1. Power down the starter kit board.
2. Program the ST6265x with the application you want to run following the instructions given in *Programming ST6 microcontrollers on page 32*.
3. Set the DEVICE jumpers W10 to W13 (marked 3 on the *Starter kit board diagram on page 13*) to ST6265x, as shown in the following diagram:

Figure 27. Device jumpers setup



4. Select USER mode using the jumpers marked W1 and W2 (marked 19 on the *Starter kit board diagram on page 13*), as shown in the diagram below:

Figure 28. User mode selection



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 PA3 in the Demonstration Selector circuit (marked 12 in the *Starter kit board diagram on page 13*)
7. If you are using your own hardware resources connected to J2 (marked 17 on the *Starter kit board diagram on page 13*), 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 26*.
8. Power up the starter kit board.

9 Hardware information

9.1 Parts list

Table 11. Board resources and connections

Part	Device	Part	Device
BZ1	BUZZER	R15	680Ω
C1,C2,C4,C5,C6,C7, C8, C9, C10	100pF	R16	750Ω
C3	4.7MF	R17	820Ω
C11,C12,C13,C16, C17,C18,C22	10MF	R18	CTN 4,7K
C14,C26	1.0nF	RS1, RS4	150Ω SIL8 4R
C15,C19,C20,C21, C24, C27, C28, CD21,CD22,CD23, CD24, CD25, CD26, CD27, CD28, CD37	100nF	RS2	100KΩ SIL10 9R
C23	22MF-25V	RS3	390Ω SIL10 9R
C25	1.0MF	RS5	3.3KΩ SIL8 4R
D1	BYV 10-20 SCHO	RS6	10KΩ SIL10 9R
D2	1N4004	RV1	10KΩ Trimmer
D3,D4,D5	1N4148	SW1,SW2,SW3	SW-PUSH
F1	Not connected	T1,T2	BC547B-NPN-45V
G1	SOLDER BRIDGE	T3	BC557B-PNP-45V
J1	HE10-16DM	T4	BD236-PNP-60V
J3	JACK	TP1,TP3	MW1X1C
J4	2nd Supply conn.	U1,U6	74LS244
L1	2,2?H	U2	4LS125
LD1,LD2,LD3,LD4, LD5	LED-RED-RECT	U3	ST6265x
LD6	LED-RED-5MM	U4	DIL-16 ZIF
P1	SUBD25	U5	74LS04
P2	SUBD9	U7, U10	78L05
R1	47Ω	U8	LM7805
R2,R5,R8,R12	4.7KΩ	U9	MAX232
R3,R4,R7,R14	10KΩ	W1,W2,W10,W11,W12,W13	MW3X1C
R6	560Ω	W4,W5,W6,W7,W8,W15, W16, W17, W18, W19, W20, W21,W22,W23, W24, W25, W26,W27,W28,W29, W30, W31	MW2X1C

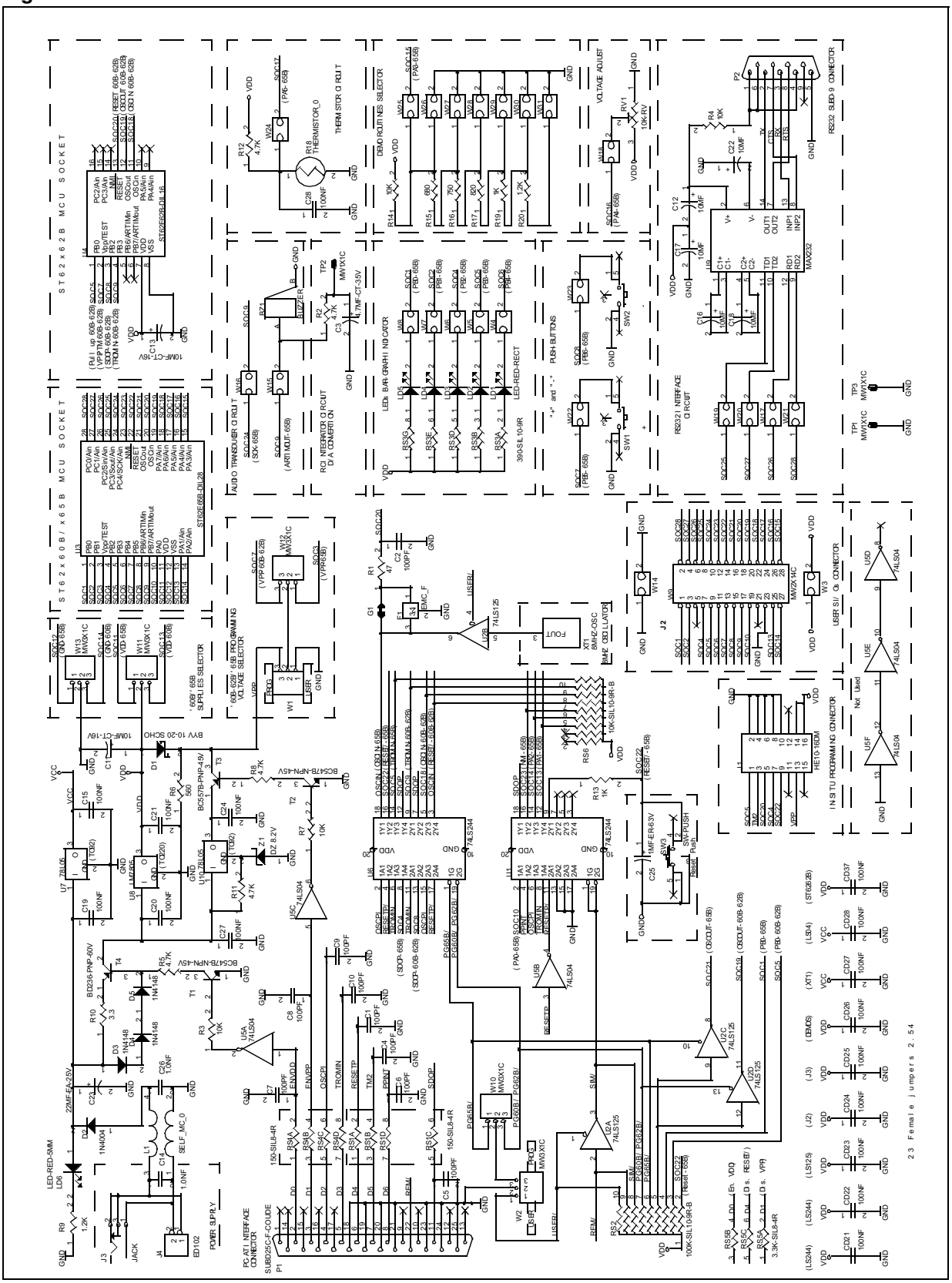
Table 11. Board resources and connections (continued)

Part	Device	Part	Device
R9,R20	1.2KΩ	W9	MW2X14C
R10	3.3Ω	XT1	8MHZ-OSC
R11	Not connected	Z1	DZ 8.2V
R13,R19	1KΩ		

9.2 Starter kit board Schematic

See next page

Figure 29. Starter kit board schematic



10 Revision history

Table 12. Document revision history

Date	Revision	Changes
Feb-1998	1	Initial release.
19-Feb-2013	2	Reformatted the entire document. Updated some part number names.

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY TWO AUTHORIZED ST REPRESENTATIVES, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER’S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com