# ST62 & ST63 FAMILIES

## 8-BIT MICROCONTROLLERS
## OTP / EPROM PROGRAMMING SPECIFICATION

### 1 INTRODUCTION

This document is the ST62/63 OTP/EPROM microcontrollers programming specification. It should give all informations about serial test and EPROM programming to all involved people who have to implement a programming algorithm.

For further details about device characteristics, please refer to the technical manual and the data book of ST6 family.



Rev. 3.2

October 1998

# Table of Contents

# Table of Contents

# 2 TECHNICAL OVERVIEW

The programming of the on-chip EPROM of the ST6 is done by using the Data ROM Window mechanism for addressing the memory cells. This is done by the way of the serial testmode offered by the ST6 microcontroller family (normally used for testing the devices in the factory).

In this mode, any program independent of the existing ROM/EPROM content can be forced from outside into the ST6 microcontroller and is executed in a similar way as a customer program.

Each instruction is fed sequentially into the ST6 device by a dedicated input line. The ST6 device uses a dedicated line to output the content of the Program Counter, and some registers. These results have to be used to control serial data flow and instructions effects.

For synchronizing the ST6 device with these fed instructions, the clock has to be driven by the test/programming equipment.

## 2.1 EPROM MEMORY DESCRIPTION

### 2.1.1 EPROM management

The EPROM memory is mapped inside the program space of the ST6. It is addressed by the 12-bit Program Counter Register and so the ST6 core can directly address up to 4 Kbytes of program space. Nevertheless, the program space can be extended by the addition of 2 Kbytes ROM banks by the way of the Program ROM Page Register (PRPR) - See ST6 family data sheets for further details -.

For programming, the EPROM has to be defined as Data EPROM. This implies the use of the Data ROM Window (DRWR) to access the EPROM.

The DRWR register can be addressed like a RAM location in the Data Space at the address C9h. This register is used to move the 64-byte read-only data window (from the 40h address to 7Fh address of the Data Space) up and down the EPROM memory of the MCU in steps of 64 bytes.

The effective address of the byte is obtained by the concatenation of the six Least Significant Bits of the data space address and the content of the DRWR register (as six Most Significant Bits of the address)

Using this address mode, the content of the PRPR register has no more effect.

Example: The following sequence will read the data byte stored in the EPROM byte at address A19h and write it in the accumulator.

```
LDI   DRWR, 28h   <-- Content of DRWR
LD    A,    59h   <-- Data Space address Basic address of window
                      40h + offset inside the window.
```

| DRWR (6:0) | | | | | | | DATA SPACE ADDRESS (5:0) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 |

### 2.1.2 Reserved areas

For STMicroelectronics testing purposes, some memory areas are reserved inside the devices. These areas match exactly the mapping of the ROM version given below:

**Table 1. EPROM/OTP Mapping**

| Memory Page | Device Address | Description |
|---|---|---|
| Page 0 | 0000h - 007Fh | Reserved |
| | 0080h - 07FFh | User ROM |
| Page 1 (Static Page) | 0800h - 0F9Fh | User ROM |
| | 0FA0h - 0FEFh | Reserved |
| | 0FF0h - OFF7h | User ROM |
| | 0FF8h - 0FFBh | Reserved |
| | 0FFCh - 0FFFh | User ROM |
| Page 2 | 1000h - 100Fh | Reserved |
| | 1010h - 17FFh | User ROM |
| Page 3 | 1800h - 180Fh | Reserved |
| | 1810h - 1FFFh | User ROM |

## 2.2 ELECTRICAL CHARACTERISTIC

Input and output voltages for normal functions pins follow the data sheet of the device.

### 2.2.1 Absolute maximum ratings

Voltage on $V_{CC}$ . . . . . .- 0.3 to 7.0V

Voltage on $V_{PP}$ . . . . . .- 0.3 to 13.5V

### 2.2.2 Characteristics during programming

The characteristics which are given below, are for a operating temperature of 25°C.

| NAME | Min. | Max | Unit |
|---|---|---|---|
| $V_{CC}$ | 4.75 | 5.25 | V |
| $I_{CC}$ | | 10 | mA |
| $V_{PP}$ | 12.5 | 13.5 | V |
| $I_{PP}$ * | | 10 | mA |

* $I_{PP}$ means $V_{PP}$ Programming current

# 3 SERIAL TEST MODE

In order to check the devices in the production line, some special test modes have been implemented inside the chips. They allow special checks onto EPROM cells like margin checks, gate and drain stress, and most particularly the programming of the EPROM cells.

In this mode the device behaviour is entirely controlled by an external equipment and therefore it is assumed in all this document that such a tool is available for the user/reader. This special tool must be able to handle all the relevant signals used in testmode - See § 3.1 - and also to provide power supplies to the chip.

## 3.1 AFFECTED PINS IN SERIAL TEST MODE

Some pins of the ST6 devices have special functions when used in the serial test mode:

| | | |
|---|---|---|
| $TM/V_{PP}$ | : | general test mode pin / Programming power supply |
| SDOP | : | serial output pin |
| TROMIN | : | serial input pin |
| TM2 | : | test mode select pin |
| OSCIN | : | oscillator input pin |
| OSCOUT | : | oscillator output pin |
| $V_{DD}$ | : | power pin |
| $V_{SS}$ | : | ground pin (common to $V_{DD}$ and $V_{PP}$/TM) |

To enter the serial test mode, certain voltage levels have to be applied to these pins during the reset phase of the device:

- $TM/V_{PP}$ has to be at $V_{DD}$ - 0.5V.
- TM2 has to be forced high during the reset phase only and could be used later on in the test program according to its normal function if needed.

After the reset phase, commands and data have to be forced via the pin TROMIN in serial way and synchronized with the clock input of the ST6. All other pins can be left floating.

If SDOP is identical with an I/O pin, it is recommended to program the related data direction register normally to output mode.

The test mode pins normally are multiplexed with other input, output or I/O pins. For each ST6 device type, this multiplexing is hardwired and may not be affected by customer or ROM/ EPROM content.

The assignments of test mode pins for available ST62 EPROM devices is shown below:

**Table 2. ST62XX Pin Assignments**

| DEVICES | $V_{PP}$/TM | SDOP | TROMIN | TM2 | OSCIN | OSCOUT | RESET | $V_{DD}$ | $V_{SS}$ |
|---|---|---|---|---|---|---|---|---|---|
| 62T00C / T01C / T03C / E01C | 5 | 7 | 8 | 9 | 2 | 3 | 6 | 1 | 16 |
| 62T08C / T09C / T10C / T20C / E20C | 6 | 8 | 9 | 10 | 3 | 4 | 7 | 1 | 20 |
| 62E18C / T18C | 6 | 8 | 9 | 10 | 3 | 4 | 7 | 1 | 20 |
| 62T15C / T25C / E25C | 10 | 12 | 13 | 14 | 3 | 4 | 11 | 1 | 28 |
| 62T28C / E28C | 10 | 12 | 13 | 14 | 3 | 4 | 11 | 1 | 28 |
| 62T30B / E30B | 10 | 12 | 13 | 14 | 3 | 4 | 11 | 1 | 28 |
| 62T32B / E32B | 15 | 18 | 19 | 20 | 6 | 7 | 16 | 14 & 12 | 13 & 11 |
| 62T35B / E35B | 12 | 15 | 16 | 17 | 2 | 3 | 13 | 11 & 9 | 10 & 8 |
| 62T40B / E40B | 18 | 14 | 15 | 16 | 27 | 26 | 25 | 23 | 24 |
| 62T42B / E42B | 16 | 12 | 13 | 14 | 21 | 20 | 19 | 17 | 18 |
| 62T45B / E45B | 11 | 8 | 9 | 10 | 16 | 15 | 14 | 12 | 13 |
| 62T46B / E46B | 20 | 16 | 17 | 18 | 25 | 24 | 23 | 21 | 22 |
| 62T53B / T60B / T63B / E60B / T53C / T60C / T63C / E60C | 3 | 4 | 5 | 1 & 6 (*) | 14 | 15 | 16 | 9 | 10 |
| 62T52B / T62B / E62B / T52C / T62C / E62C | 2 | 3 | 4 | 1 & 5 (*) | 11 | 12 | 13 | 7 | 8 |
| 62T55B / T65B / E65B / T55C / T65C / E65C | 3 | 4 | 5 | 1 & 8 (*) | 20 | 21 | 22 | 11 | 12 |
| 62T80B / E80B | 27 | 31 | 32 | 33 | 29 | 28 | 30 | 40 | 41 |
| 62T85B /E85B | 21 | 25 | 26 | 27 | 23 | 22 | 24 | 30 | 31 |

(*) For these devices, these pins must be connected to $V_{DD}$.

The assignments of testmode pins for available ST63 EPROM devices is shown below:

**Table 3. ST63XX Pin Assignments**

| DEVICES | $V_{PP}$/TM | SDOP | TROMIN | OSCIN | OSCOUT | RESET | $V_{DD}$ | $V_{SS}$ |
|---|---|---|---|---|---|---|---|---|
| 63E85/E87/63E76/E78 | 30 | 39 | 36 | 31 | 32 | 33 | 42 | 21 |
| 63E69 | 27 | 37 | 34 | 28 | 29 | 30 | 40 | 21 |
| 63E73 | 29 | 39 | 36 | 30 | 31 | 32 | 42 | 21 |
| 63E156 | 16 | 1 | 5 | 25 | 26 | 24 | 40 | 20 |
| 63E70H / 71H (PDIP 40) | 27 | 37 | 34 | 28 | 29 | 30 | 40 | 21 |
| 63E70J / 71J (PSDIP 42) | 29 | 39 | 36 | 30 | 31 | 32 | 42 | 21 |

### 3.2 RECOMMENDED INPUT LEVELS IN SERIAL TEST MODES

The test mode pins $V_{PP}$/TM, TM2 and TROMIN and the oscillator input signal must have the input levels shown below, also in the case their normal function levels are specified with different voltage values.

**Table 4. Input levels in serial test mode**

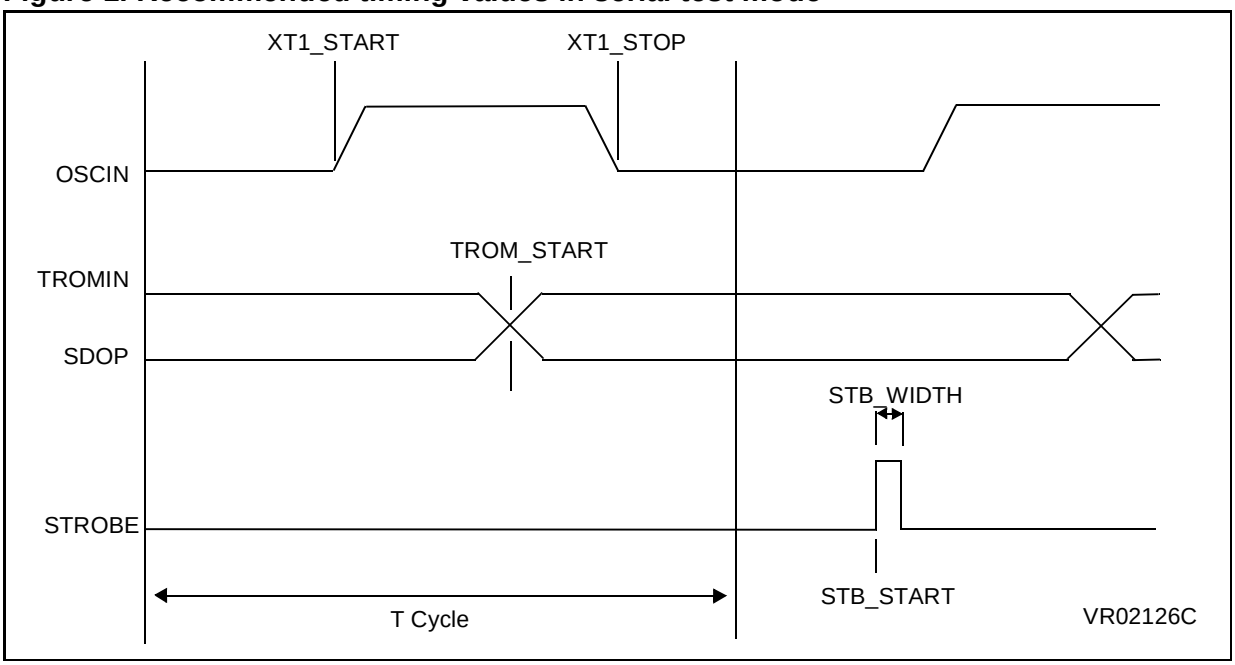
| PIN | LOGIC LEVEL | MIN | NOM | MAX |
|---|---|---|---|---|
| $V_{PP}$/TM | 0<br>1 | GND<br>$0.7V_{DD}$ | GND<br>$V_{DD}$ | GND + 0.4V<br>$V_{DD}$ + 0.5V |
| TM2 | 0<br>1 | GND<br>$0.7V_{DD}$ | GND<br>$V_{DD}$ | GND + 0.4 V<br>$V_{DD}$ + 0.5V |
| TROMIN | 0<br>1 | GND<br>$0.7V_{DD}$ | GND<br>$V_{DD}$ | GND + 0.4V<br>$V_{DD}$ + 0.5V |
| OSC INPUT | 0<br>1 | GND<br>$0.7V_{DD}$ | GND<br>$V_{DD}$ | GND + 0.4V<br>$V_{DD}$ + 0.5V |

Input and output voltages for normal function pins, see data sheet of the device. Above values are valid for temperatures in the range from +15°C to +35°C.

### 3.3 RECOMMENDED TIMING VALUES IN SERIAL TEST MODE

The test mode pins TROMIN and the oscillator input signal must follow the recommended timings shown hereafter:

**Figure 1. Recommended timing values in serial test mode**




**Warning**: The TROMIN signal must mandatory change after the rising edge of the clock signal.

**Table 5. Timing values for serial test mode at frequencies 8MHz and 4MHz**

| | 8MHz | | | 4 MHz | | |
|---|---|---|---|---|---|---|
| | MIN | NOM | MAX | MIN | NOM | MAX |
| **T Cycle** | | 125 ns | | | 250 ns | |
| **XT1_START** | | 40 ns | | | 80 ns | |
| **XT1_STOP** | | 102 ns | | | 200 ns | |
| **TROM_START** | 55 ns | 77 ns | 100 ns | 110 ns | 150 | 200 ns |
| **STB_START** | 10 ns | | 65 ns | 20 ns | | 130 ns |
| **STB_WIDTH** | | 5 ns | | | 10 ns | |

### 3.4 SIGNALS POLARITIES

Depending of the device the different signals used in the serial test mode may have different polarities. They can be active at '1' or at '0'.

For the ST62E2X/3X/4X/8X devices the polarities are the following:

OSCIN active at level 1

OSCOUT active at level 0

TM2 active at level 1

SDOP active at level 1

TROMIN active at level 0

For the ST62E5XB/ E6XB devices, some signals have a different polarity, that is:

SDOP active at level 1

TROMIN active at level 1

For the ST62E5XC/ E6XC devices, some signals have a different polarity, that is:

SDOP active at level 1

TROMIN active at level 0

For the ST63EXX devices, some signals have a different polarity, that is:

SDOP active at level 0

TROMIN active at level 0

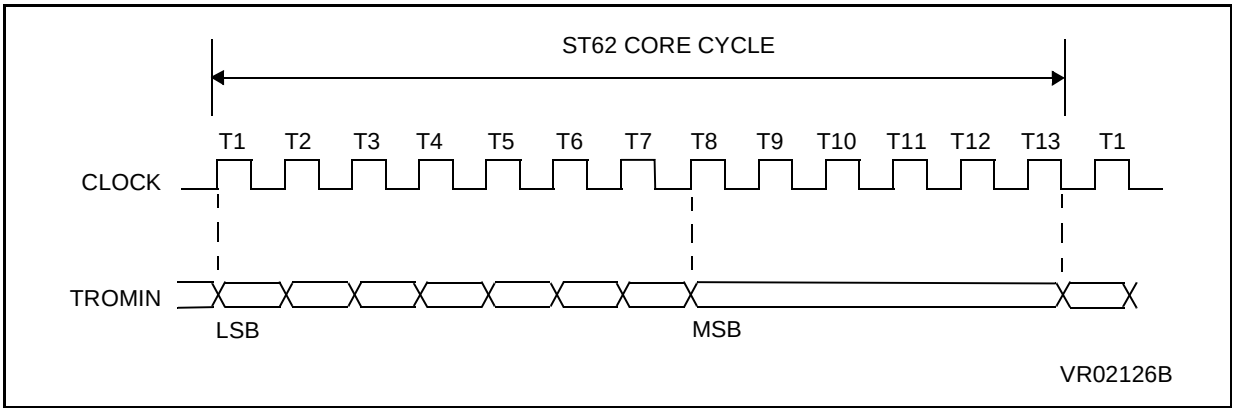
**Important note**: All operand code and data fed via TROMIN have to be inverted regarding the values mentioned inside this documentation.

### 3.5 TESTMODE CYCLES

Each instruction consists of 2, 4 or 5 ST62xx machine cycles. Each of these machine cycles consists of 13 clock pulses T1... T13:

- During T1 the LSB of an opcode, an address or a data has to be forced via TROMIN, and during T8 the MSB.
- During T9 to T13 the state of the MSB can be maintained at TROMIN or reset to 0.

**Figure 2. Machine cycle**



### 3.6 RESET PHASE

The serial test mode is activated at the end of the reset phase of the ST6 device, depending on the voltage levels applied on TM2 and $V_{PP}$/TM pins during this phase.

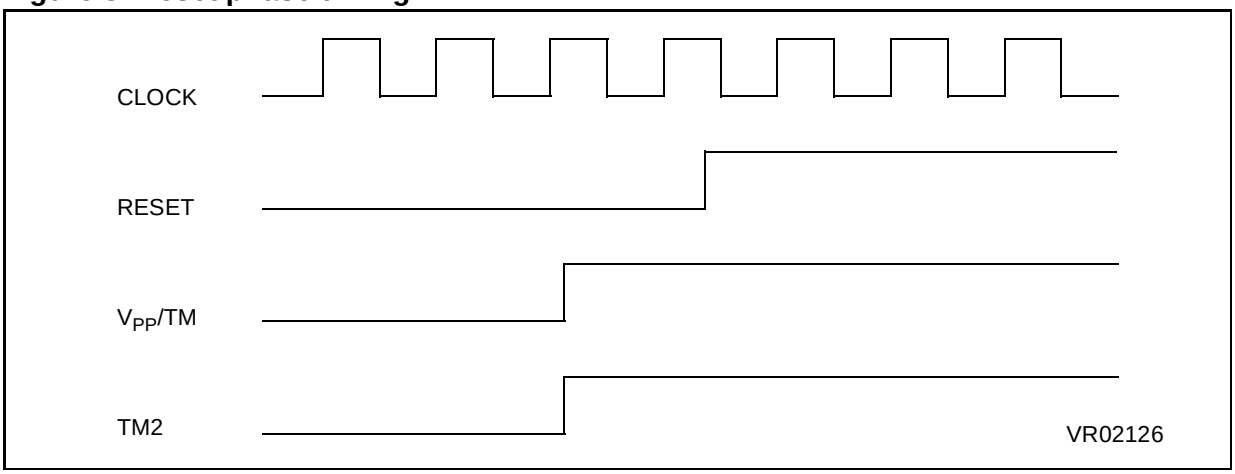
During the whole reset phase (RESET is active low for all ST6):

- TM2 pin has to be forced high.
- $V_{PP}$/TM must be at $V_{DD}$ - 0.5 V
- TROMIN pin has to be kept low.

The RESET pin must be kept low during at least 500 ns to achieve a proper reset of the ST6. To avoid race conditions, the rising edge of the Reset signal must not occur during low phase of the clock input.

As soon as the logic level at pin RESET is set to '1', the internal ST6 build-up counter will start. To finish the ST6 reset phase, it is necessary to apply 2048 ($2^{11}$) clock pulses + 13 clocks pulses for the first internal cycle called CYC I. From this point, commands and data have to be forced via pin TROMIN in a serial way synchronized with the clock input. SDOP pin can be used to watch program counter, serial data flow and ST6 registers.

**Figure 3. Reset phase timing**



### 3.7 SYNCHRONIZATION PHASE

Once the reset phase is completed, it is necessary to check if the ST6 device is synchronized, meaning that the reset phase has been properly executed by the internal circuitry.

For synchronization check, the **NOP** operation can be used just after the first internal cycle CYC I. Opcode of this operation is 00h and therefore it is sufficient to keep the TROMIN pin at low level like during the reset phase. The NOP operation uses two ST6 machine cycles called CYC1 and CYC5. As long as TROMIN is tied low, the ST6 will repeat a CYC1/CYC5 sequence.

After Reset, the Program Counter is set to FFEh. Each NOP operation will increase the Program Counter by one. The PC value is shifted out on SDOP pin, LSBit first, starting with T1, and therefore can be checked and used for synchronization. During CYC5, the current complete value of the PC is shifted out (12 bits during T1 - T12), the last bit has no signification.

A table summarizing the RESET and an extended SYNCHRONIZATION phase is given below:

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| high | high | low | some pulses | low | - | | |
| | | pos.edge | high | low | | | **Start-up Phase** |
| | | high | $2^{11}$ Pulses | low | - | | |
| | | | 13 Pulses | low | - | CYCI | - |
| | | | 13 Pulses | low | | CYC1 | - |
| | | | 13 Pulses | low | FFF | CYC5 | check synchronization |
| | | | | | | | 12 LSB of PC = FFFh |
| | | | 13 Pulses | low | | CYC1 | |
| | | | 13 Pulses | low | 000 | CYC5 | check synchronization |
| | | | 13 Pulses | low | | CYC1 | |
| | | | 13 Pulses | low | 001 | CYC5 | check synchronization |
| | | | 13 Pulses | low | | CYC1 | |
| | | | 13 Pulses | low | 002 | CYC5 | check synchronization |
| | | | 13 Pulses | low | | CYC1 | |

### 3.8 REGISTERS INITIALIZATION

The Watchdog Register must be initialized after the synchronization phase in order to guarantee proper operation for the ST6 device.

The **Watchdog Register** has to be programmed in order to be inactive. If it is a hardware watchdog, it is necessary to refresh it regularly. This can be done in the same time as the DRWR is changed every scope of 64 bytes to program. It is not mandatory with all the devices, but it is recommended to avoid possible problems with devices having a hardware watchdog.

### 3.9 COMMANDS USED IN TEST MODE

Table below shows some commands useful for the programming of the device and allowed in the serial test mode. It is assumed, that the ST6 command set is well known.

**Table 6. The most important commands for the serial test**

| COMMAND | CYCLE1 | CYCLE2 | CYCLE3 | CYCLE4 | CYCLE5 |
|---|---|---|---|---|---|
| ldi adr, dat | 0Dh | n.e. | adr | % | dat |
| ld a, adr | 1Fh | n.e. | adr | % | dat |
| ld adr, a | 9Fh | n.e. | adr | % | dat |

% = don't care

n.e. = not existent for this command

All opcodes, addresses and data are shifted via TROMIN LSB first and MSB last. Other commands, see data sheet.

**Important note**:All input via TROMIN has to be inverted if its polarity is negative!

#### 3.9.1 ldi adr, dat

This command writes an 8-bit data **dat** into a ST6 data space address **adr**. It is used for initialization of the registers and programming the EPROM bytes.

Depending of its use, the data has to be forced at TROMIN at different times.

The command consists of 3 bytes (opcode, destination address, operand) and is executed in four ST6 machine cycles. During the first cycle (CYC1), the opcode (ODh, 8bit) is loaded into the ST6 microprocessor via TROMIN and in the second cycle (CYC3) the address adr (8bit).

- When using this command for register initialization, in the third cycle (CYC4) the ST6 works only internally, no input via TROMIN is required. In the fourth cycle (CYC5), the data has to be forced serially on TROMIN pad. In pulse T10 of this cycle, the data is stored in the register.
- When using this command for programming, the data to be programmed has to be forced via TROMIN during the third cycle (CYC4). In the fourth cycle (CYC5), the TROMIN pin can be kept low.

#### 3.9.2 ld a, adr

This command reads the content of a data space register **adr** to the accumulator **a**. It is used for reading out, verifying and blank checking the EPROM content.

The command consists of two bytes and its execution needs four ST6 machine cycles again. In the first cycle, the opcode (1Fh, 8bit) has to be driven to the CPU via TROMIN, in the second cycle the address adr (8bit). In the remaining two cycles, the ST6 needs no more inputs from outside. During the last cycle, the data space register content is shifted out on SDOP and can be read.

### 3.10 REGISTER INITIALIZATION WITH THE LDI INSTRUCTION

After initialization, the Data ROM Window Register (DRWR) has to be loaded first. The normal address for this register (write only register) is 0C9h.

Programming is done in scopes of 64 bytes (40h - 7Fh). The scopes (blocks) will be selected by the content of the DRWR. The number of scopes depends from the EPROM size!

32 (00h-1Fh) for 2k-EPROM

64 (00h-3Fh) for 4k-EPROM

128 (00h-7Fh) for 8k-EPROM

**Important notes:**

- The 64 bytes of every scope are in the data space at address 40h-7Fh. The DRWR must be loaded with the LDI instruction.
- All opcodes, addresses and data must be shifted serially in TROMIN LSBit first and MSBit last.
- The LDI instruction consists of four machine cycles. Each of this cycle consists of 13 clock pulses T1 - T13. The forcing of opcodes, addresses and data takes always place from T1 - T8, during T9 - T13 the state of the MSBit can be maintained at TROMIN.
- The shifting out of data at pin SDOP always starts at T1 by the LSBit. Polarity of SDOP has to be taken into account.

**Machine cycles description**

Machine Cycle 1 (CYC1)

During the first eight OSCIN pulses of CYC1, the opcode (0Dh) is shifted into the TROMIN pin, at the same time, the result of the previous ALU operation is shifted out on the pin SDOP. During the final five OSCIN cycles the TROMIN pin can be kept low.

Machine Cycle 3 (CYC3)

During the first eight OSCIN pulses of machine cycle CYC3, the address of the Data ROM Window Register is shifted into pin TROMIN. During the same time, the least significant eight bits of the PC are shifted out, the value must be one greater than the PC from the preceding machine cycle. This should be used to verify that the process is still synchronized. During the final five OSCIN cycles, the TROMIN pin can be kept low.

Machine Cycle 4 (CYC4)

During the next thirteen OSCIN pulses, the TROMIN pin is pulled to VIL. In most cases, no useful data is shifted out of the SDOP pin.

Machine Cycle 5 (CYC5)

During the first eight OSCIN pulses of machine cycle CYC5, the content of the Data ROM Window Register is shifted into pin TROMIN. (This is the scope number).

**Latest now, the programming voltage $V_{PP}$ must be applied to the $V_{PP}$/TM pin.**

During the last five pulses TROMIN is pulled low.

Table below summarizes these actions:

**Table 7. Initialization phase**

| CYCLE | TROMIN | INFORMATION AT SDOP | USED FOR |
|---|---|---|---|
| CYC1 | 0Dh | Result of previous operation | |
| CYC3 | Register address (0C9h for DRWR) | 8 first LSBits of program counter (PC) | Synchronization check |
| CYC4 | Don't care | | Not used |
| CYC5 | Data to be written into register (Scope number for DRWR) | Data (register contents for readable registers only) | |

### 3.11 PROGRAMMING WITH THE LDI INSTRUCTION

After the initialization of the Data ROM Window Register with the scope number, now the 64 bytes pointed by the DRWR can be programmed by using the load immediate instruction (LDI).

**Important notes:**

- Be aware that the actual programming is done during CYC4 / clock 13 and that the low phase of this clock must be extended to time of 1 ms by $V_{PP}$ = 12.5 Volts.
- It is not necessary to write the bytes of the EPROM memory in any particular order.
- The LDI instruction is normally used for programming, however any instruction which access the address range of 40h-7Fh while $V_{PP}$ is > $V_{DD}$ + 0.5 Volts will initiate a programming cycle.
- Take care that instructions other than the LDI may produce unexpected results.

**Machine cycle description**

Machine Cycle 1

During the first eight OSCIN pulses of CYC1, the opcode (0Dh) is shifted into the TROMIN pin.

During the final five OSCIN pulses the TROMIN has to be kept at low level. The data shifted out on the SDOP pin is delayed by one instruction.

There is no limit established on the number of write attempts, but if data has not been correctly retained after the fifth attempt, programming should be abandoned and an error reported.

Machine Cycle 3

During the first eight OSCIN cycles of machine cycle 3, the address (pointer to the Data ROM) is shifted into the TROMIN pin, at the same time the low order eight bits of the PC are shifted out on the SDOP pin.

The PC address shifted out of the SDOP pin may be used to verify that the processor is properly synchronized. If the PC is not the expected value the process should be abandoned and restarted.

During the final five OSCIN cycles the TROMIN pin has to be kept at low level.

Machine Cycle 4

During the next eight OSCIN cycles, data to be programmed are shifted into the TROMIN pin. At the same time, the contents of the Data ROM byte is shifted out of the SDOP pin. If an erased part is being programmed then the byte should be equal to 00h, if not, programming should normally be suspended.

The part might be erased with longer exposure to Ultraviolet flux or discarded, as appreciated.

During the final five OSCIN cycles the TROMIN pin has to be kept at low level. No useful data is shifted from the SDOP pin during these OSCIN cycles.

Machine Cycle 5

During this cycle the TROMIN signal can be kept at its current value (MSBit of previous cycle) as it has no effect. No useful data is shifted from the SDOP pin during these OSCIN cycles.

After programming the first 64 bytes the Data ROM Window Register has to be loaded with the next scope value then the next 64 bytes can be programmed. The voltage on the $V_{PP}$/TM pin can be kept to $V_{PP}$ (12.5/13 Volts) for all the programming time even during the new initialization of the Data ROM Window Register.

Table below summarizes these actions:

**Table 8. EPROM byte programming**

| CYCLE | TROMIN | INFORMATION AT SDOP | USED FOR |
|---|---|---|---|
| CYC1 | 0Dh | Result of previous operation | |
| CYC3 | Data ROM (40h for the first) | 8 first LSBits of PC | Synchronization check |
| CYC4 | Data to program into EPROM cell | Content of EPROM cell before programming | Blank check |
| CYC5 | Don’t care | | |

# 4 EPROM HANDLING

## 4.1 EPROM PROGRAMMING OVERVIEW

The complete sequence for programming the device can be split in different phases:

– Reset phase - See § 3.6 -
This performs the entry in the serial mode test.

– Synchronization phase - See § 3.7 -
This checks the device synchronization.

– Registers initialization - See § 3.8 -
This guarantees proper reset conditions for the device.

– DRWR initialization - See § 3.10 -
This sets the data window on the first scope of 64 bytes.

– Programming 64 bytes inside current data window. - See § 3.11 -
This is the actual programming sequence.

**Notes:**

- To program the entire device, the DRWR must be programmed with consecutive values once the 64 bytes, pointed by the window, have been programmed.
- Each defective byte has to be rewrite up to 5 times before aborting the programming sequence.
- It is recommended to apply one security programming count after the successful programming for the EPROM byte. This will enforce the cell margin.

A simplify flow chart below, describes the algorithm to apply for programming the entire device. The example is given for a 2 Kbytes device.

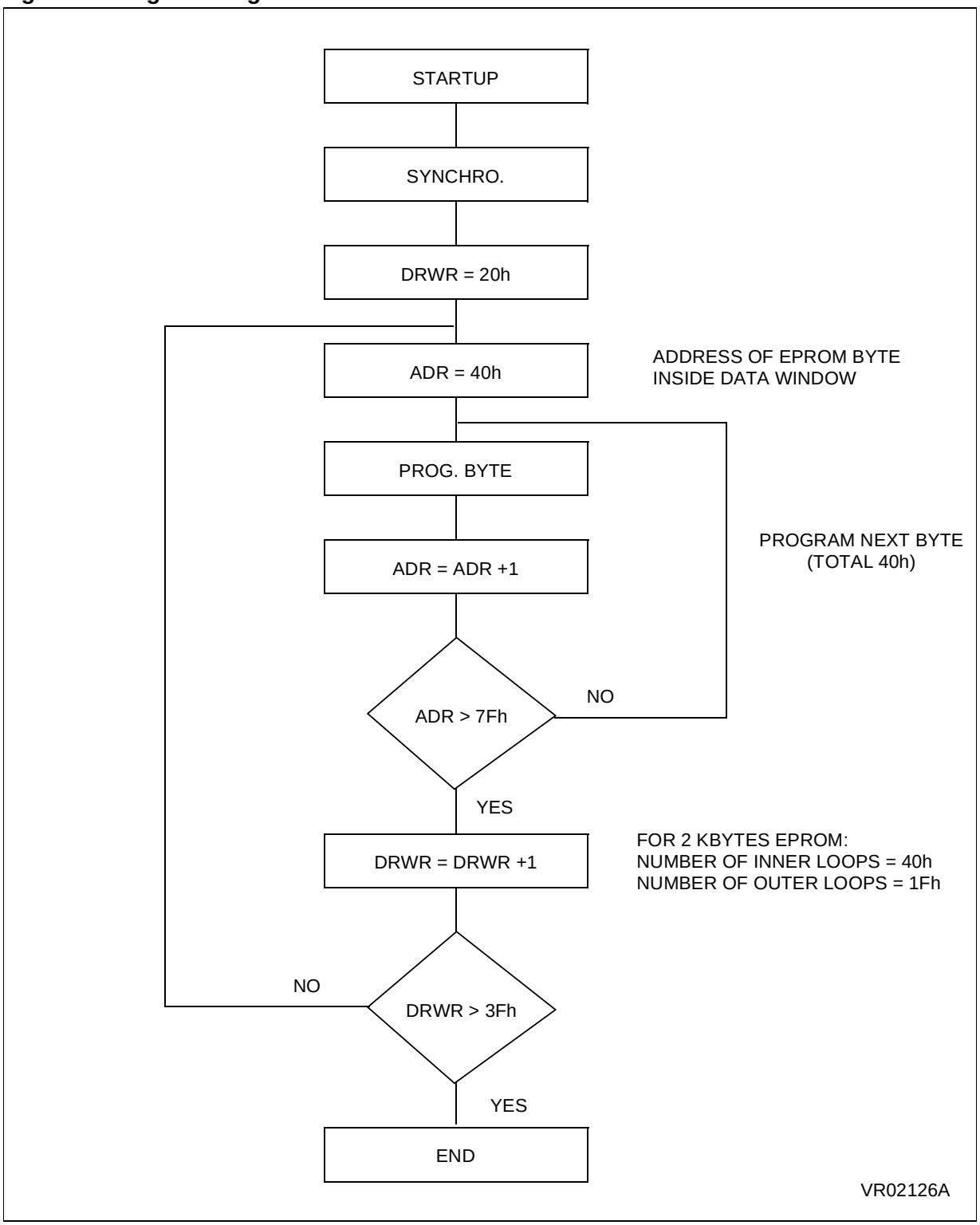
This concerns only the main part of the sequence: the EPROM programming.

– The Data ROM Window Register is first initialized with the basic address of the first scope of 64 bytes of EPROM. As the example device is 2 Kbytes device, it means that the EPROM is located from 800h up to FFFh, so the first window is located at the address 20h.

– Then these 64 bytes are programmed one after the others. In fact inside this flow chart, is not represented the possible retry programming (a byte can be programmed up to 5 times before to be declared defective) and the security programming count.

– Once whole window has been programmed, the DRWR has to be reprogrammed with the next one. This outer loop has to be repeated till the end of EPROM space.

**Figure 4. Programming Flowchart**



### 4.1.1 ST63 EPROM Programming

The programming of the EPROM is done in the same way as for the other members of the ST6 family. With the Data Rom Window Register (C8h), we are able to reach addresses up to 16K (4000h). The extension of the memory is done with the additional bit 7 of register CAh.

## 4.2 EPROM PROGRAMMING SEQUENCE

The following table gives a complete programming sequence for the ST6 devices. It must be adapted for the programming of the pin SDOP as output. This pin location depends on the device type. For the following sequence the example used is the ST62E20 where the pin SDOP is the bit 7 of port B, so that the Data Direction Register address is 0C5h and the value to program is 80h.

The symbol - inside this table means that the value read has no real signification.

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| | | | | | | | **Start-up Phase** |
| high | high | low | some pulses | low | - | | |
| | | pos. edge | high | low | | | |
| | | high | $2^{11}$ Pulses | low | - | | Build up counter |
| | | | 13 Pulses | low | - | CYCI | First internal cycle |
| | | | 13 Pulses | low | - | CYC1 | |
| | | | | | | | **Synchronization** |
| | | | 12 Pulses | low | FFFh | CYC5 | Check synchronization<br>12 LSB of PC = FFFh |
| | | | 1 Pulse | low | - | CYC5 | |
| | | | | | | | **Watchdog Register** |
| | | | 8 Pulses | 0Dh | | CYC1 | Check IOR writing result |
| | | | 5 Pulses | low | - | CYC1 | LDI instruction into TROMIN to restart watchdog |
| | | | 8 Pulses | 0D8h | 00h | CYC3 | Check LSBits of PC and shift watchdog register address |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | FEh | FEh | CYC5 | Restart value |
| | | | 5 Pulses | low | - | CYC5 | |

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| | | | | | | | **Data ROM Window Register** |
| | | | 8 Pulses | 0Dh | | CYC1 | LDI instruction into TROMIN (LSB first) to set first the Data ROM Window Register, normally to zero |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | (DRWR) address (C9h) | 03h | CYC3 | Check Synchr. on SDOP. Shift address of Data ROM. Window Register into TROMIN |
| | | | 5 Pulses | | - | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | (DRWR) content | (DRWR) content | CYC5 | Check function of TROMIN SDOP & internal shift register |
| | | | | (00) | (00) | | Apply initial content of DRWR (normally zero) to TROMIN and verify at SDOP |
| | 12.5V | | 5 Pulses | low | - | CYC5 | Switch $V_{PP}$ to 12.5V |
| | | | | | | | **Programming First Byte** |
| | | | 8 Pulses | 0Dh | - | CYC1 | Shift opcode of LDI instruction into TROMIN. Program first EPROM byte of a 64 bytes block |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | window address | 06h | CYC3 | Check synchr. on SDOP<br>Shift address of first Data EPROM byte into TROMIN (normally 40). LSB first. |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 8 Pulses | (EDAT) | (00) | CYC4 | Blank check on SDOP.<br>Shift data to be programmed. |
| | | | 5 Pulses | low | - | CYC4 | Extend low phase of 5th pulse to about 5 msec for EPROM byte writing. |
| | | | 8 Pulses | low | EDAT | CYC5 | Check whether shifting into TROMIN was done correctly. Not yet a verify: check shift register only. |
| | $V_{DD}$ -0.5V | | 5 Pulses | low | - | CYC5 | Switch $V_{PP}$ to $V_{DD}$ -0.5V |

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| | | | | | | | **Verifying First Byte** |
| | | | 8 Pulses | 1Fh | - | CYC1 | Shift opcode of LD instruction |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | window address | 09h | CYC3 | Shift address of first Data ROM byte into TROMIN (normally 40h) |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | low | EDAT | CYC5 | Retrieve EPROM content on SDOP. The read value has to be compared with the one programmed before at the same location |
| | 12.5V | | 5 Pulses | low | - | CYC5 | Switch $V_{PP}$ to 12.5V |
| | | | | | | | **Programming Second Byte** |
| | | | 8 Pulses | 0Dh | - | CYC1 | Shift opcode of LDI instruction into TROMIN. Program first EPROM byte of a 64 bytes block |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | window address | 0Bh | CYC3 | Check synchr. on SDOP<br>Shift address of first Data EPROM byte into TROMIN (normally 40). LSB first. |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 8 Pulses | (EDAT) | (00) | CYC4 | Blank check on SDOP.<br>Shift data to be programmed. |
| | | | 5 Pulses | low | - | CYC4 | Extend low phase of 5th pulse to about 5 msec for EPROM byte writing. |
| | | | 8 Pulses | low | EDAT | CYC5 | Check whether shifting into TROMIN was done correctly. Not yet a verify: check shift register only. |
| | | | 5 Pulses | low | - | CYC5 | |
| After each 64 bytes block set bank register to new value. | | | | | | | |

### 4.3 EPROM VERIFYING PROCEDURE

Once the entire EPROM space programming is completed, it is necessary to perform a verify on the entire space in order to be quite sure of a correct programming.

The complete sequence for verifying the device is similar to the programming at least in the first phases:

– Reset phase - See § 3.6 -
This performs the entry in the serial mode test.

– Synchronization phase - See § 3.7 -
This checks the device synchronization.

– Registers initialization - See § 3.8 -
This guarantees proper reset conditions for the device.

– DRWR initialization - See § 3.10 -
This sets the data window on the first scope of 64 bytes.

The difference with programming concerns two points described below:

- The voltage on the pin $V_{PP}$ / TM has to be maintained at $V_{DD}$ - 0.5V during all the operation.
- The access to the EPROM content is done with the instruction LD A, ADR (- See § 3.9.2 -) instead of the LDI instruction. During the last cycle of this instruction, the EPROM byte content will be shifted out, LSBit first, and therefore can be checked with the programmed value.

The table below describes a complete sequence for verifying a device.

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| high | high | low | some pulses | low | - | | **Start-up Phase** |
| | | pos.edge | high | low | | | |
| | | high | $2^{11}$ Pulses | low | - | | Build up counter |
| | | | 13 Pulses | low | - | CYCI | First internal cycle |
| | | | 13 Pulses | low | | CYC1 | |
| | | | | | | | **Synchronization** |
| | | | 12 Pulses | low | FFFh | CYC5 | Check synchronization 12 LSB of PC = FFFh |
| | | | 1 Pulse | low | - | CYC5 | |
| | | LIKE IN THE PROGRAMMING SEQUENCE | | ( ( ( | | | **REGISTER Initialization Watchdog Register** |
| | | | | | | | **Data ROM Window Register** |
| | | | 8 Pulses | 0Dh | | CYC1 | LDI instruction into TROMIN |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | (DRWR) address | 03h | CYC3 | Check Synchr. on SDOP Shift address of Data ROM Window Register into TROMIN |
| | | | 5 Pulses | low | - | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | (DRWR) content | (DRWR) content | CYC5 | Check function of TROMIN SDOP & internal shift register |
| | | | | (00) | (00) | | Apply initial content of DRWR (normally zero) to TROMIN and verify at SDOP |
| | | | 5 Pulses | low | - | CYC5 | |
| | | | | | | | **Verifying First Byte** |
| | | | 8 Pulses | 1Fh | - | CYC1 | Shift opcode of LD instruction |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | window address | 06h | CYC3 | Shift address of first Data ROM byte into TROMIN (normally 40h) |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | low | EDAT | CYC5 | Retrieve EPROM content on SDOP. The read value has to be compared with the one programmed before at the same location |
| | | | 5 Pulses | low | - | CYC5 | |
| | | If verification is correct, continue with next byte inside the window. After each 64 bytes, set bank register to new value. | | | | | |

### 4.4 EPROM BLANK CHECK PROCEDURE

To perform a blank check on the entire device, the same procedure as for verifying the EPROM has to be used. The only difference consists in the EPROM content comparison that must be done against the blank value of the ST6 device.

**Important notes:**

- The blank value of a byte is 00h.
- Blank checking must be done with $V_{PP}$ / TM equal to $V_{DD}$-0.5V Volts.

## 5 EEPROM HANDLING

### 5.1 EEPROM DESCRIPTION

The EEPROM of the ST6 family is addressed using the banks of 64 bytes located between addresses 00h and 3Fh. The selection of the bank is made by programming the Data RAM Bank Register (DRBR) located inside the data space. This register is used to select the desired 64 bytes EEPROM bank of the Data Space. The number of the bank has to be loaded in the DRBR and the instruction has to point to the selected location as it if was in bank 0 (from addresses 00h to 3Fh). In this register only one bit must be set at a time, otherwise two or more pages are enabled in parallel, producing errors. - See devices data sheet for further information on DRBR -

The EEPROM is controlled by the EEPROM Control Register (EECTL). Only two bits of this register have to be used for EEPROM writing:

bit 0 (Enable) This bit MUST be set to one in order to write to any EEPROM location.

bit 1 (Busy) This bit is automatically set by the internal EEPROM logic when a write operation is on going. So it has to be polled in order to detect the completion of the operation.

The writing of a byte is made simply by executing a load instruction, after verifying that no other write operation is in progress (using bit 1 of EECTL). No hardware mechanism prevents to make a double programming attempt and it is the program's responsibility to avoid this situation.

**NOTE:**

Depending of the devices, the registers addresses and the EEPROM bytes count are different. Please, refer to the device data sheet to get them.

### 5.2 EEPROM WRITING IN SERIAL MODE

The serial mode can be used to program, in a similar way as for EPROM, the EEPROM of the ST6 devices.

The write operation has to be done like in normal mode, that is, the usual program for writing inside the EEPROM has to be forced serially via TROMIN after the testmode start up and synchronization phases.

EEPROM writing is performed as followed:

– Reset phase - See § 3.6 -
This performs the entry in the serial mode test.

– Synchronization phase - See § 3.7 -
This checks the device synchronization.

– Registers initialization - See § 3.8 -
This guarantees proper reset conditions for the device.

– EEPROM writing operation validated
This enables the writing operation on EEPROM.

– DRBR initialization
This sets the data window on the first EEPROM bank of 64 bytes.

– EEPROM byte writing
This can be split in two different steps:

- Actual write on EEPROM location performed by loading the EEPROM location with the value to write, using instruction **LD EEP_ADR, EEP_DAT**
- Polling the bit busy of EECTL in order to detect the end of operation. this can be done by the instruction **LD A, EECTL**. During the last cycle of this instruction, the register content will be shifted out, LSBit first, and therefore the bit busy (bit 1) can be checked.

These 2 steps have to be repeated for each byte inside the selected bank, then the DRBR has to be reprogrammed with the new bank number in order to select a new scope of 64 bytes.

**NOTE:**

As there is an internal charge pump for providing the high voltage necessary to program the EEPROM cells, there is no need to apply a programming voltage on the device when accessing the EEPROM.

The table below describes a complete sequence for writing the EEPROM of a ST6 device.

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| high | high | low | some pulses | low | - | | **Start-up Phase** |
| | | pos.edge | high | low | | | |
| | | high | $2^{11}$ Pulses | low | - | | Build up counter |
| | | | 13 Pulses | low | - | CYCI | First internal cycle |
| | | | 13 Pulses | low | | CYC1 | |
| | | | | | | | **Synchronization** |
| | | | 12 Pulses | low | FFFh | CYC5 | Check synchronization<br>12 LSB of PC = FFFh |
| | | | 1 Pulse | low | - | CYC5 | |
| | | LIKE IN THE | | ( | | | **REGISTER Initialization** |
| | | PROGRAMMING | | ( | | | **Watchdog Register** |
| | | SEQUENCE | | ( | | | |
| | | | | | | | **Enable EEPROM Writing** |
| | | | 8 Pulses | 0Dh | | CYC1 | Shift opcode of LDI instruction |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | EECTL address | 03h | CYC3 | Shift EECTL address and check synchr. on SDOP |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | 01h | 01h | CYC5 | Set Enable bit (0) |
| | | | 5 Pulses | low | | CYC5 | |
| | | | | | | | **Data RAM Bank Register** |
| | | | 8 Pulses | 0Dh | 01h | CYC1 | Check EECTL writing and shift LDI into TROMIN |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | DRBR | 06h | CYC3 | Check synchr. on SDOP<br>Shift address of Data RAM Bank Register into TROMIN (normally 40h) |
| | | | 5 Pulses | low | - | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | (DRBR) content (01) | (DRBR) (01) | CYC5 | Apply initial content of DRBR (bank number) to TROMIN and verify at SDOP |
| | | | 5 Pulses | low | - | CYC5 | |
| | | | | | | | **Writing First Byte** |

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| | | | 8 Pulses | 0Dh | 01h | CYC1 | Shift opcode of LDI instruction<br>Program first EEPROM byte |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | EEP-ROM address | 06h | CYC3 | Shift address of first Data EEPROM byte into TROMIN (normally 00h) |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | EEDAT | EEDAT | CYC5 | Shift EEPROM data to write |
| | | | 5 Pulses | low | 00h | CYC5 | |
| | | | | | | | **Check Busy Byte** |
| | | | 8 Pulses | 1Fh | - | CYC1 | Shift opcode of LD instruction |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | EECTL address | 09h | CYC3 | Check LSBits of PC and shift EECTL address |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | - |
| | | | 8 Pulses | low | (EECTL) content | CYC5 | Retrieve EECTL content on SDOP. The bit busy (1) has to be checked. While it is set to 1, this instruction has to be looped. |
| | | | 5 Pulses | low | 00h | CYC5 | |
| When Busy is reset, continue with next byte inside the current window.<br>After each 64 bytes, set Data RAM bank register to new value. | | | | | | | |

### 5.3 EEPROM VERIFYING PROCEDURE

Once the entire EEPROM space writing is completed, it is necessary to perform a verify on the entire space in order to be quite sure of a correct writing.

The complete sequence for verifying the device is similar to the writing at least in the first phases:

– Reset phase - See § 3.6 -
This performs the entry in the serial test mode.

– Synchronization phase - See § 3.7 -
This checks the device synchronization.

– Registers initialization - See § 3.8 -
This guarantees proper reset conditions for the device.

– DRBR initialization
This sets the data window on the first EEPROM bank of 64 bytes.

The difference with the writing operation concerns two points described below:

- The bit 0 into EECTL register does not need to be programmed to 1 as no write will be performed on EEPROM.
- The access to the EEPROM content is done with the instruction LD A, ADR (-See § 3.9.2-) During the last cycle of this instruction, the EPROM byte content will be shifted out, LSBit first, and therefore can be checked against the written value.

The table below describes a complete sequence for verifying a device.

| TM2 | $V_{PP}$ | RESET | CLOCK (OSCIN) | TROMIN | SDOP | CYCLE | WHAT TO DO |
|---|---|---|---|---|---|---|---|
| high | high | low | some pulses | low | - | | **Start-up Phase** |
| | | pos.edge | high | low | | | |
| | | high | $2^{11}$ Pulses | low | - | | Build up counter |
| | | | 13 Pulses | low | - | CYCI | First internal cycle |
| | | | 13 Pulses | low | | CYC1 | |
| | | | | | | | **Synchronization** |
| | | | 12 Pulses | low | FFFh | CYC5 | Check synchronization 12 LSB of PC = FFFh |
| | | | 1 Pulse | low | - | CYC5 | |
| | | | LIKE IN THE PROGRAMMING SEQUENCE | ( ( ( | | | **REGISTER Initialization Watchdog Register** |
| | | | | | | | **Data RAM Bank Register** |
| | | | 8 Pulses | 0Dh | | CYC1 | LDI instruction into TROMIN |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | DRBR | 00h | CYC3 | Check synchr. on SDOP Shift address of Data RAM Bank Register into TROMIN |
| | | | 5 Pulses | low | - | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | (DRBR) content | (DRBR) | CYC5 | Apply initial content of DRBR (bank number) to TROMIN and verify at SDOP |
| | | | 5 Pulses | low | - | CYC5 | |
| | | | | | | | **Verifying First Byte** |
| | | | 8 Pulses | 1Fh | - | CYC1 | Shift opcode of LD instruction |
| | | | 5 Pulses | low | - | CYC1 | |
| | | | 8 Pulses | EEP-ROM address | 03h | CYC3 | Shift address of first Data EEPROM byte into TROMIN (normally 00h) |
| | | | 5 Pulses | low | | CYC3 | |
| | | | 13 Pulses | low | - | CYC4 | |
| | | | 8 Pulses | low | EDAT | CYC5 | Retrieve EEPROM content on SDOP. The read value has to be compared with the one written before at the same location |
| | | | 5 Pulses | low | 00h | CYC5 | |
| | | | If verify is correct, continue with next byte inside the window. After each 64 bytes, set bank register to new value. | | | | |

# 6 OPTION BYTE HANDLING

The ROM mask options that can be selected by the user in the ROM devices can be emulated by the EPROM / OTP devices by an EPROM code byte that is programmable in the serial test mode.

## 6.1 OPTION BYTE DESCRIPTION

There exists two types of Option Byte over the ST6 family: a DYNAMIC one and a STATIC one. The static one is based on a latch, with two EPROM cells permanently read while the dynamic one consists on one EPROM cell read at reset.

In static type, one of these two EPROM cells is the master one, the other is the slave one. The value of each option bit is -well- defined when one of the two EPROM cells is programmed while the other one is still erased. So, for the programming of one option byte (8 bits), we must program the both corresponding addresses (of this option byte) with the complementary content. The content of the master cell agrees to the value of the datasheet and the content of the slave one to its complement.

The access of each option bit is performed through some bit lines of the EPROM matrix. The addresses of bus EPADR[0-11] which reach the option byte are described in the table below.

**Table 9. ST62 Option Byte addresses**

| DEVICES | OPTION BYTE | Master / Slave | OPTIONS | ADDRESS |
|---|---|---|---|---|
| ST62T0X, T1X, T2X<br>ST62E0X, E1X, E2X | No Option Byte available* | | | |
| ST62T0XC / E0XC<br>ST62T1XC / E1XC<br>ST62T2XC / E2XC | A | Master | [0:7] | F80h |
| | $\overline{A}$ | Slave | [0:7] | B80h |
| | B | Master | [8:15] | 780h |
| | $\overline{B}$ | Slave | [8:15] | 380h |
| ST62T28C / E28C | A | Master | [0:7] | 5Fh |
| | $\overline{A}$ | Slave | [0:7] | 57h |
| | B | Master | [8:15] | 4Fh |
| | $\overline{B}$ | Slave | [8:15] | 47h |
| ST62T3XB / E3XB | A | Master | [0:7] | 1700h |
| | $\overline{A}$ | Slave | [0:7] | 700h |
| ST62T4XB / E4XB | A | Master | [0:7] | 1700h |
| | $\overline{A}$ | Slave | [0:7] | 700h |
| ST62T5XB / E5XB<br>ST62T6XB / E6XB | A | Master<br>(dynamic type) | [0:7] | B80h |
| ST62T5XC / E5XC<br>ST62T6XC / E6XC | A | Master | [0:7] | 5Fh |
| | $\overline{A}$ | Slave | [0:7] | 57h |
| | B | Master | [8:15] | 4Fh |
| | $\overline{B}$ | Slave | [8:15] | 47h |
| ST62T8XB / E8XB | A | Master | [0:7] | 1700h |
| | $\overline{A}$ | Slave | [0:7] | 700h |

*product termination in progress

The option byte can be accessed in serial mode once the bit 7 of the Interrupt Option Register (address 0C8h) has been set. This bit (called MDIS) selects a special EPROM matrix inside the chip where is located the option byte. It is only possible to set this bit in test mode, so user mode is not affected.

For further information on options bytes, see data sheet of devices. For any operations (programming, verifying, blank check) performed on the option byte the same procedure and the same algorithm as for the normal EPROM have to be used.

Only two points are slightly different:

- The IOR must be programmed with 80h in the registers initialization phase (instead of 00h). Any access to this register must be done with $V_{PP}$ / TM = $V_{DD}$ - 0.5V.
- The programming time (low phase of 13th pulse in CYC4 of actual programming instruction) has to be extended to 10 milliseconds.

### 6.2 OPTION BYTE MODES

Here after are described the different option byte modes and the registers value to be set in order to access to them:

**Table 10. Option byte modes**

| MDIS<br>IOR[7] | $V_{PP}$ | Mode |
|---|---|---|
| 1 | 5V | Read Option Byte<br>EPROM cells |
| 1 | 13V | Program Option Byte<br>EPROM cells |

### 6.3 READ OPTION BYTE MEMORY CELLS

The reading of the option byte value is performed as a read of an EPROM matrix byte. Before reading the byte, enter in matrice disable mode (MDIS=1) by setting the bit IOR[7].

Select the appropriate Data ROM Window in order to access to the correct option byte (refer to Table 10). Then, (for the ST62E2X), read (LD A, adr) from the Data ROM Window the content of option byte at the needed address 0380h or 0B80h for both option bytes (0780h and 0F80h contains the complementary values.)

### 6.4 PROGRAM OPTION BYTE (TEST MODE)

The programming of the option byte is performed as a programming of a matrix EPROM byte. First, apply 13 Volts on pad $V_{PP}$. Before programming the byte, enter in matrix disable mode (MDIS=1) by setting bit IOR[7].

Select the appropriate Data ROM Window in order to access to the correct option byte (refer to Table 10). Then, execute a write instruction (LDI adr, #data) into the Data ROM Window. In order to program the option byte, the “write” instruction must be performed exactly as a programming sequence (refer to chapter 7 for programming sequence).

### 6.5 ST63 EPROM OPTION BYTE PROGRAMMING

The programming procedure of this byte is the same as the one used for ST62 (With the management of the MDIS bit 7 of Register C9h) and by setting bit 7 of Register CAh.

**Table 11. ST63 Option Byte addresses**

| DEVICE | Master / Slave | OPTIONS | ADDRESS |
|---|---|---|---|
| ST63E71/73 | M | [0:7] | 1700h |
| ST63E76/78 | M | [0:7] | 1700h |
| ST63E85/87 | M | [0:7] | 4B80h |
| ST63E156 | M | [0:7] | 4B80h |

## 7 READ-OUT PROTECTION

### 7.1 HARDWARE PROTECTION

Hardware read-out protection can be achieved on all ST62 devices thanks to option byte.

The read-out protection is effective as soon as the relevant bit of the option byte is programmed (see datasheet for further information). Once programmed, this bit disables the access to the program memory in serial testmode. Thus, it is no more possible to read out the device content. More precisely, the whole program memory space will then be read at 0FFh.

The whole option byte content can be programmed in single pass. However, it is recommended to separate the two operations, since setting the hardware read-out protection prevents the read of option byte.

If the protection bit isn't set, the corresponding bit of the Slave address (if it exists) must be set to 1 in order to keep the complementary content between the Master and the Slave.

### 7.2 SOFTWARE PROTECTION

When there is no option byte in devices, the hardware read-out protection is not possible. Then 2 bytes are dedicated to implement a software read-out protection. This concerns all ST62 devices without option byte and of which the production has been stopped in 1997.

The programming tool must be able to handle this feature in the following manner:

– Program these two bytes to protect the devices if requested by customer.

– Refuse to read-out the devices if these bytes have been previously programmed.

This will ensure compatibility with the STMicroelectronics programmers.

The relevant addresses are 0FFAh and 0FEFh. Both addresses are in the reserved areas, so that their programming will not corrupt the device behaviour inside customer application.

These two bytes allow the management of both EPROM and OTP device as followed:

– If 0FFAh content is 00h, the device is unprotected and may be read.

– If 0FFAh low nibble content is equal to 9h, byte 0FEFh has to be tested. (Case of OTP device).

– If 0FEFh content is 00h the device is unprotected.

– In all other cases, the device has to be considered as protected.

These two bytes have to be checked or programmed by the test equipment in order to implement this SOFTWARE protection. As they are normal EPROM bytes, the programming and the verifying phases must respect the EPROM algorithm described in previous chapters.

Regarding the above description, it appears that in order to protect the devices, the address 0FFAh or the address 0FEFh (in case of OTP) has to be programmed with 0FFh.

# 8 APPENDIX

## APPENDIX 1: ENVIRONMENT

### Working environment

These recommendations list the major points defined in the CECC00015 specifications named “Protection of electrostatic sensitive devices”. Refer to this document for detailed information.

### Equipment

The equipment, tools and instruments used during the programming flow must be connected to ground.

### Working station

The working station must be antistatic. This station is to include a table with a top in conductive material, or covered with an antistatic mat (surface resistivity <0.5 MΩ/cm²), connected to ground via a conductor cable in series with a 1 MΩ resistance. It is to include also an antistatic ground carpet grounded via a conductive cable (0.9 to 1.5 MΩ).

The usage of a grounded antistatic armlet connected to the mat or to a grounded equipment is mandatory. Antistatic gloves, or finger coats, are also required.

Any plastic object on the station must be of an antistatic material.

## APPENDIX 2: LIST OF EXISTING DEVICES

Hereafter is given the list of the existing devices of the ST6 family that you can order with the request form attached to this document. This list is not exhaustive but allows to get components able to emulate any other existing type.

**Table 12. List of ST62 devices with option byte**

| MASTER DEVICE | DEVICE | PACKAGE | PART-NUMBER | SAMPLES AVAILABILITY |
|---|---|---|---|---|
| ST62T01C | ST62T00C | PDIP16 | ST62T00CB6 | 5 supplied samples |
| | ST62T00C | PSO16 | ST62T00CM6 | |
| | ST62T01C | PDIP16 | ST62T01CB6 | |
| | ST62T01C | PSO16 | ST62T01CM6 | |
| | ST62T03C | PDIP16 | ST62T03CB6 | |
| | ST62T03C | PSO16 | ST62T03CM6 | |
| ST62T20C | ST62T08C | PDIP20 | ST62T08CB6 | 5 supplied samples |
| | ST62T08C | PSO20 | ST62T08CM6 | |
| | ST62T09C | PDIP20 | ST62T09CB6 | |
| | ST62T09C | PSO20 | ST62T09CM6 | |
| | ST62T10C | PDIP20 | ST62T10CB6 | |
| | ST62T10C | PSO20 | ST62T10CM6 | |
| | ST62T20C | PDIP20 | ST62T20CB6 | |
| | ST62T20C | PSO20 | ST62T20CM6 | 5 supplied samples |
| ST62T25C | ST62T15C | PDIP28 | ST62T15CB6 | |
| | ST62T15C | PSO28 | ST62T15CM6 | |
| | ST62T25C | PDIP28 | ST62T25CB6 | |
| | ST62T25C | PSO28 | ST62T25CM6 | |
| ST62T18C | ST62T18C | PDIP20 | ST62T18CB6 | 5 supplied samples |
| | ST62T18C | PSO20 | ST62T18CM6 | |
| ST62T28C | ST62T28C | PDIP28 | ST62T28CB6 | 5 supplied samples |
| | ST62T28C | PSO28 | ST62T28CM6 | |
| ST62T30B | ST62T30B | PDIP28 | ST62T30BB6 | 5 supplied samples |
| | ST62T30B | PSO28 | ST62T30BM6 | |
| ST62T32B | ST62T32B | PSDIP42 | ST62T32BB6 | upon specific request |
| | ST62T32B | PQFP52 | ST62T32BQ6 | |
| ST62T35B | ST62T35B | PQFP52 | ST62T35BQ6 | upon specific request |
| ST62T40B | ST62T40B | PQFP80 | ST62T40BQ6 | 5 supplied samples |
| ST62T42B | ST62T42B | PQFP64 | ST62T42BQ6 | upon specific request |
| ST62T45B | ST62T45B | PQFP52 | ST62T45BQ6 | upon specific request |
| ST62T46B | ST62T46B | SDIP56 | ST62T46BB6 | upon specific request |

| MASTER DEVICE | DEVICE | PACKAGE | PART-NUMBER | SAMPLES AVAILABILITY |
|---|---|---|---|---|
| ST62T60B* | ST62T53B | PDIP20 | ST62T53BB6 | upon specific request |
| | ST62T53B | PSO20 | ST62T53BM6 | |
| | ST62T60B | PDIP20 | ST62T60BB6 | |
| | ST62T60B | PSO20 | ST62T60BM6 | |
| | ST62T63B | PDIP20 | ST62T63BB6 | |
| | ST62T63B | PSO20 | ST62T63BM6 | |
| ST62T62B* | ST62T52B | PSO16 | ST62T52BM6 | upon specific request |
| | ST62T62B | PSO16 | ST62T62BM6 | |
| ST62T65B* | ST62T55B | PDIP28 | ST62T55BB6 | upon specific request |
| | ST62T55B | PSO28 | ST62T55BM6 | |
| | ST62T65B | PDIP28 | ST62T65BB6 | |
| | ST62T65B | PSO28 | ST62T65BM6 | |
| ST62T60C | ST62T53C | PDIP20 | ST62T53CB6 | 5 supplied samples |
| | ST62T53C | PSO20 | ST62T53CM6 | |
| | ST62T60C | PDIP20 | ST62T60CB6 | |
| | ST62T60C | PSO20 | ST62T60CM6 | |
| | ST62T63C | PDIP20 | ST62T63CB6 | |
| | ST62T63C | PSO20 | ST62T63CM6 | |
| ST62T62C | ST62T52C | PDIP16 | ST62T52CB6 | 5 supplied samples |
| | ST62T52C | PSO16 | ST62T52CM6 | |
| | ST62T62C | PDIP16 | ST62T62CB6 | |
| | ST62T62C | PSO16 | ST62T62CM6 | |
| ST62T65C | ST62T55C | PDIP28 | ST62T55CB6 | 5 supplied samples |
| | ST62T55C | PSO28 | ST62T55CM6 | |
| | ST62T65C | PDIP28 | ST62T65CB6 | |
| | ST62T65C | PSO28 | ST62T65CM6 | |
| ST62T80B | ST62T80B | PQFP100 | ST62T80BQ6 | 5 supplied samples |
| ST62T85B | ST62T85B | PQFP80 | ST62T85BQ6 | upon specific request |

* termination in progress, replaced by C version

**Table 13. List of ST62 devices without option byte**

| MASTER DEVICE | DEVICE | PACKAGE | PART-NUMBER | SAMPLES AVAILABILITY |
|---|---|---|---|---|
| ST62T01B6/HWD* | ST62T00/xxx | PDIP16 | ST62T00B6/xxx | upon specific request |
| | ST62T00/xxx | PSO16 | ST62T00M6/xxx | |
| | ST62T01/xxx | PDIP16 | ST62T01B6/xxx | |
| | ST62T01/xxx | PSO16 | ST62T01M6/xxx | |
| | ST62T03/xxx | PDIP16 | ST62T03B6/xxx | |
| | ST62T03/xxx | PSO16 | ST62T03M6/xxx | |
| ST62T20B6/HWD* | ST62T08/xxx | PDIP20 | ST62T08B6/xxx | |
| | ST62T08/xxx | PSO20 | ST62T08M6/xxx | |
| | ST62T09/xxx | PDIP20 | ST62T09B6/xxx | |
| | ST62T09/xxx | PSO20 | ST62T09M6/xxx | |
| | ST62T10/xxx | PDIP20 | ST62T10B6/xxx | |
| | ST62T10/xxx | PSO20 | ST62T10M6/xxx | |
| | ST62T20/xxx | PDIP20 | ST62T20B6/xxx | |
| | ST62T20/xxx | PSO20 | ST62T20M6/xxx | |
| ST62T25B6/HWD* | ST62T15/xxx | PDIP28 | ST62T15B6/xxx | upon specific request |
| | ST62T15/xxx | PSO28 | ST62T15M6/xxx | |
| | ST62T25/xxx | PDIP28 | ST62T25B6/xxx | |
| | ST62T25/xxx | PSO28 | ST62T25M6/xxx | |

* termination in progress, replaced by C version.

xxx: HWD or SWD version

**Table 14. List of ST63 devices**

| MASTER DEVICE | DEVICE | PACKAGE | PART-NUMBER |
|---|---|---|---|
| ST63T73 | ST63T73 | PSDIP42 | ST63T73J5B1 |
| ST63T78 | ST63T78 | PSDIP42 | ST63T78B1/xxx |
| ST63T85 | ST63T85 | PSDIP42 | ST63T85B1/xxx |
| ST63T87 | ST63T87 | PSDIP42 | ST63T87B1/xxx |
| ST63T88 | ST63T88 | PSDIP42 | ST63T88B1 |
| ST63T89 | ST63T89 | PSDIP42 | ST63T89B1 |
| ST63T156 | ST63T156 | PDIP40 | ST63T156B1/xxx |

**REQUEST FOR ST6 PROGRAMMING ALGORITHM**

Fax to STMicroelectronics at + 33 4 76 58 56 26

or mail to STMicroelectronics

Avenue des Martyrs

BP 217

38019 GRENOBLE CEDEX FRANCE

Attn: Development Tools - F. AUCLAIR

YES, I wish to automatically receive the new version of the ST6 Programming Algorithm when it is released.

Please send me 5 samples of ST6 devices in addition to the devices I have already received.

**ST6 Master Device Name**: . . . . . .

(See the list of existing devices in Appendix 2 of ST62EXX Programming Specification)

USER REGISTRATION DATA

Name: . . . . . .

Company: . . . . . .

Address: . . . . . .

. . . . . .

. . . . . .

City: . . . . . .State: . . . . . .

ZIP code: . . . . . .Country: . . . . . .

Telephone number: . . . . . .

Fax number: . . . . . .

**Notes:**