AN975
APPLICATION NOTE

UPGRADING FROM ST625X/6XB TO ST625X/6XC

by Microcontroller Application Team

INTRODUCTION

As part of a process of continuous improvement, STMicroelectronics has replaced all ST62T5XB and ST62T6XB devices by ST62T5XC and ST62T6XC.

This opportunity was taken to include new features such as the Low Voltage Detector (LVD) for safe reset, the Oscillator Safeguard (OSG) and a new RC oscillator.

This application note details these new features and draws attention to some precautions that it is mandatory to take when upgrading an application developed with B revision to C revision devices. The first part of the document is related to the silicon itself and the second one to the development tools.

1 SILICON CHANGES

1.1 NEW FEATURES

1.1.1 LOW VOLTAGE DETECTOR (LVD)

ST625X/6XC now features an on-chip Low Voltage Detector which is enabled by setting a bit in the option bytes (refer to the Option Bytes section in the device data sheet).

The LVD allows the device to be used without any external RESET circuitry. In this case, the $\overline{\text{RESET}}$ pin should be left **unconnected**. When the LVD is not activated, an external circuit is mandatory to ensure correct Power On Reset operation. For more details, please refer to the application note AN669.

The LVD generates a static Reset when the supply voltage is below a reference value. This means that it secures the power-up as well as the power-down keeping the ST6 in reset.

The V_{IT-} reference value for a voltage drop is lower than the V_{IT+} reference value for power-on in order to avoid a parasitic reset when the MCU starts running and sinks current on the supply (hysteresis). The LVD Reset circuitry generates a reset when V_{DD} is below:

– V_{IT+} when V_{DD} is rising

– V_{IT-} when V_{DD} is falling

The LVD function is illustrated in Figure 1.

If the LVD is enabled, the MCU can be in only one of two states:

– Over the input threshold voltage, it is running under full software control

– Below the input threshold voltage, it is in static safe reset

In these conditions, secure operation is guaranteed without the need for external reset hardware.

During a Low Voltage Detector Reset, the $\overline{\text{RESET}}$ pin is held low, thus permitting the MCU to reset other devices.

Figure 1. Low Voltage Detector Reset
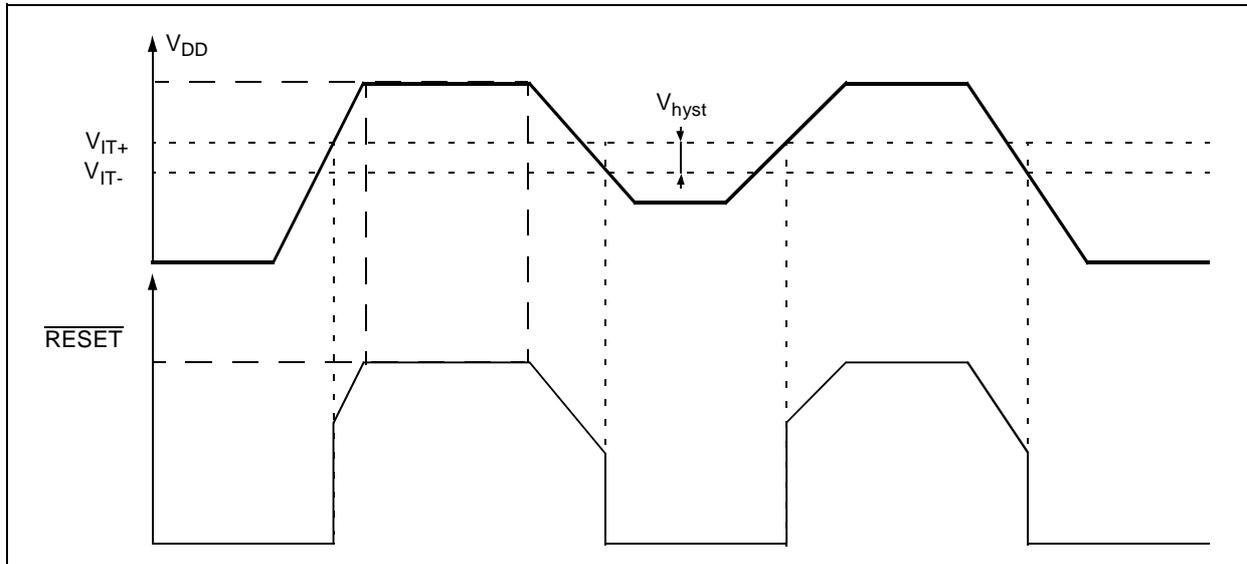
1.1.2 Oscillator Safeguard (OSG)

The Oscillator Safeguard (OSG) feature is a means of dramatically improving the operational integrity of the MCU.The OSG function is enabled by setting the OSG option bit in the Option Byte (refer to the Option Bytes section in the device data sheet).

The OSG acts as a filter whose cross-over frequency is device dependent and provides three basic functions:

– Filtering spikes on the oscillator lines which would result in driving the CPU at excessive frequencies

– Management of the Low Frequency Auxiliary Oscillator (LFAO), (useable as low cost internal clock source, backup clock in case of main oscillator failure or for low power consumption)

– Automatically limiting the f_{INT} clock frequency as a function of supply voltage, to ensure correct operation even if the power supply drops.

The OSG function is illustrated in Figure 2. For more details on it, please refer to the Clock, Supply and Reset section in the device data sheet.

Please note that the OSG should be used wherever possible as it provides maximum security for the application. It should be noted however, that it can increase power consumption and reduce the maximum operating frequency to f_{OSG} (refer to the Electrical Characteristics section of the device data sheet).

Caution: Care has to be taken when using the OSG, as the internal frequency is defined between a minimum and a maximum value and may vary depending on both V_{DD} and temperature. For precise timing measurements, it is not recommended to use the OSG.

Figure 2. OSG Filtering Function
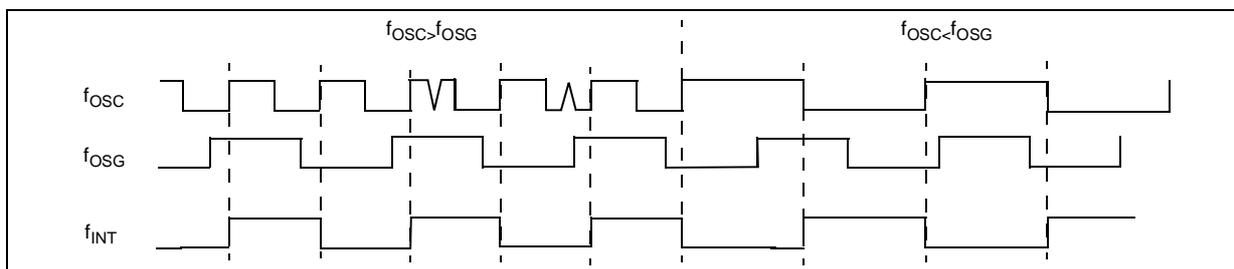
1.1.2.1 Low Frequency Auxiliary Oscillator (LFAO)

The Low Frequency Auxiliary Oscillator has three main purposes. Firstly, it can be used to reduce power consumption in non timing critical routines. Secondly, it offers a fully integrated system clock, without any external components. Lastly, it acts as a backup oscillator in case of main oscillator failure.

This oscillator is available when the OSG option is selected in the option byte. In this case, it automatically starts one of its periods after the first missing edge of the main oscillator, whatever the reason for the failure (main oscillator defective, no clock circuitry provided, main oscillator switched off...). See Figure 3.

Please note that the A/D converter accuracy is decreased, since the internal frequency is below 1.2 MHz.

Figure 3. LFAO Oscillator Function
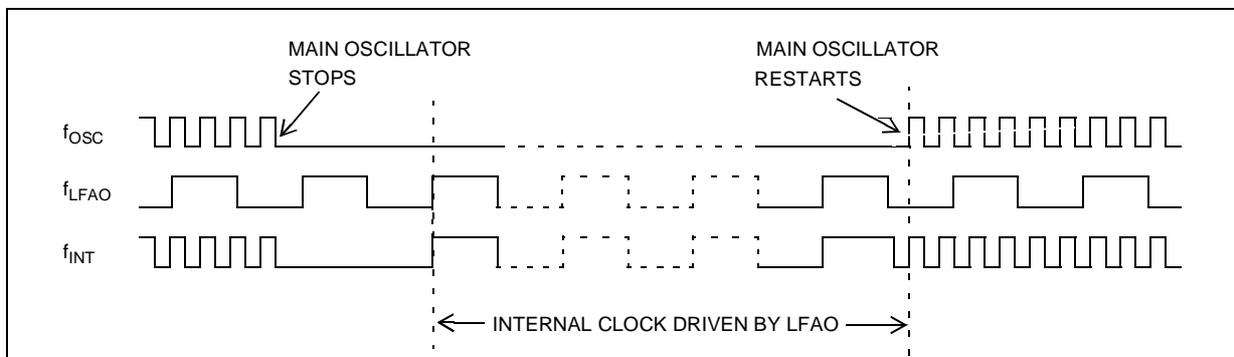
1.2 COMPATIBILITY ISSUES AND RECOMMENDATIONS

1.2.1 EEPROM

ST625X/6XC have been designed to ensure data EEPROM integrity as long as the application designer follows the guidelines described in application note AN671.

In general, EEPROM data corruption occurs whenever the reset signal is not well controlled when the power supply goes either up or down. This is particularly true with a slow power supply ramp-up and/or fall time, since the device may be in a supply voltage range where its functionality is not guaranteed. If the device is not kept in reset state during both the power up or down sequence then the microcontroller core may start writing to the EEPROM.

To avoid EEPROM data corruption, it is therefore highly recommended to disable the EEPROM when it is not used. Note that the EEPROM is automatically enabled on reset (EECTL reset state = 00H).

It is also possible to prevent unwanted EEPROM write accesses by enabling the on-chip LVD which will reset the ST6 when the voltage is out of the device operating range.

Caution: When writing parallel data (E2PAR2 bit set), the EEPROM page must not be changed (no access to the DRBR register is allowed).

1.2.2 I/O ports

This section gives some important recommendations on using I/O ports. These recommendations also apply to B devices but are even more important for C devices.

– Do NOT use single-bit instructions on port Data Registers.

– Configure the I/O ports following the safe I/O state switching sequence.

1.2.2.1 Instructions NOT to be used to access Port Data registers (SET, RES, INC and DEC)

DO NOT USE SINGLE-BIT INSTRUCTIONS (SET, RES, INC and DEC) ON PORT DATA REGISTERS IF ANY PIN OF THE PORT IS CONFIGURED IN INPUT MODE.

These instructions make an implicit read and write back of the entire register. In port input mode, however, the data register reads from the input pins directly, and not from the data register latches. Since data register information in input mode is used to set the characteristics of the input pin (interrupt, pull-up, analog input), these may be unintentionally reprogrammed depending on the state of the input pins.

As a general rule, it is better to only use single bit instructions on data registers when the whole (8-bit) port is in output mode. In the case of inputs or of mixed inputs and outputs, it is advisable to keep a copy of the data register in RAM. Single bit instructions may then be used on the RAM copy, after which the whole copy register can be written to the port data register:

```
set bit, datacopy
ld A, datacopy
ld DRA, A
```

1.2.2.2 Safe I/O State Switching Sequence

Switching the I/O ports from one state to another should be done in a sequence which ensures that no unwanted side-effects can occur. The recommended safe transitions are illustrated in Figure 4. All other transitions are potentially risky and should be avoided when changing the I/O operating mode.

Figure 4. Diagram showing Safe I/O State Transitions
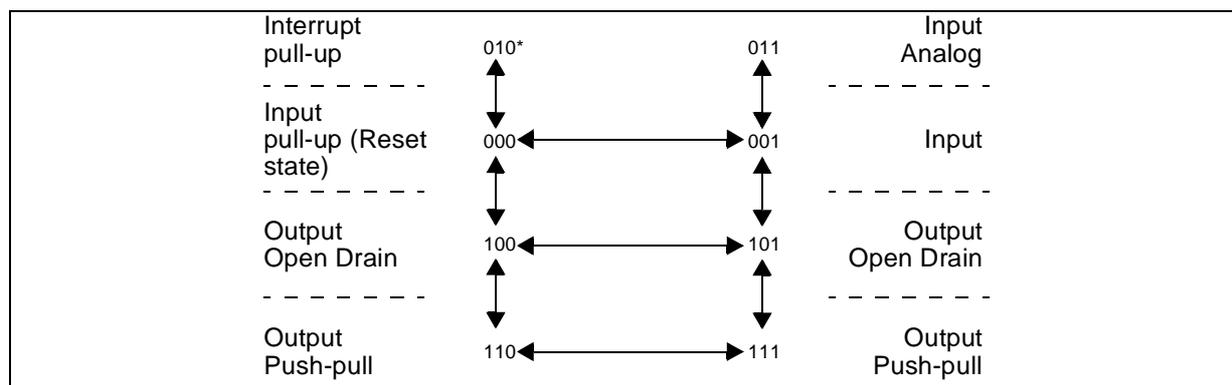
1.2.2.3 Handling Unused Port Bits

On ports that have less than 8 external pins connected:

– Leave the bits related to the unbonded pins in reset state and do not change their configuration.

– Do not use instructions that act on a whole port register (INC, DEC, or read operations). Unavailable bits must be masked by software (AND instruction). Thus, when a read operation performed on an incomplete port is followed by a comparison, use a mask.

1.2.3 8-bit Timer

On C revision devices, in order to use the 8-bit timer in gated mode, the PC1/TIM1 pin has to be configured in input mode either with or without pull-up through the Port C DDR, OR and DR registers.

On B revision devices, it was possible to define PC1/TIM1 pin as output in this mode. But on C revision devices, if this pin is configured as output, the Schmitt trigger input is disabled and the timer will therefore not work properly.

1.2.4 SPI

When the SPI is used in master mode, the SCK pin (PC4) and Sout pin (PC3) work in output mode and the Sin pin (PC2) in input mode. On ST625X/6XC, the Port C pins used for the SPI have to be configured in input mode either with or without pull-up. Their reset configuration doesn't have to be changed.

To enable the SPI in master mode, the SPCLK bit in the SPI mode control register has to be set. By setting this bit, the SCK pin (PC4) is automatically configured in output push-pull mode and the clock is output at the frequency configured in the SPIDIV register. The output of the SPI data register Sout pin (PC3) is also automatically configured in output push-pull by setting bit 0 in the MISCR register.

The following is a software example of how to use the SPI in master mode:

```
.org 080h (or CSEG AT 080H depending on the toolchain)
reset: clr X
       clr A
       reti                    ; Exit NMI mode

       ldi WDGR, watchtim      ; Load the watchdog
       ldi OSCR, 00h           ; Set the oscillator ratio to 1
       ldi DDRC, 00h           ; Port C in reset state
       ldi ORC, 00h
       ldi DRC, 00h

       ldi MISCR, 001h         ; Enable Sout
                               ; (PC3 is configured in output push-pull)
       ldi SPIDIV, 048h        ; Clock ratio = 1 / 9 bits frame
       ldi SPIMOD, 03Eh        ; CPOL = 0 / Filters enabled / SPCLK = 1 / CPHA = 1
                               ; SPSTRT = 1 / SPIN = 1 / SPIE = 0
                               ; (PC4 is configured as output push-pull)
       ldi SPIDR, 0AAh         ; Load the data to be send

       set 7, SPIMOD           ; Enable the transmission (SPIRUN = 1)
loop:  jrs 7, SPIMOD, loop     ; Poll the bit SPIRUN for end of transmission
       .....
```

1.2.5 ADC

When an I/O pin is used as an analog input, A/D conversion accuracy will be impaired if negative current injections ($V_{INJ} < V_{SS}$) occur from adjacent I/O pins with analog input capability. ST625X/6XC devices are more sensitive to this phenomenon than ST625X/6XB devices. In the case of a loss of accuracy, we recommend to (refer to Figure 5.):

– Use another I/O port located further away from the analog pin, preferably not multiplexed on the A/D converter

– Increase the input resistance $R_{IN\ J}$ (to reduce the current injections) and reduce R_{ADC} (to preserve conversion accuracy).

Figure 5. Leakage from Digital Inputs
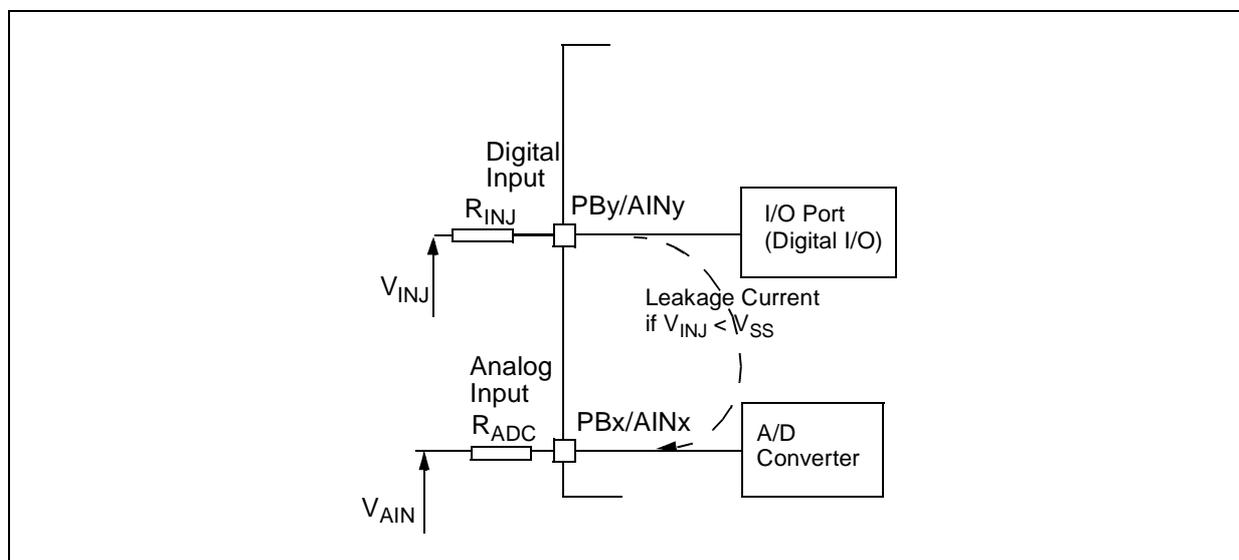
1.2.6 OSCR Register

On C revision devices, bit 3 of the OSCR register must be set to 0 in order to achieve low consumption. On B revision devices, it was mandatory to set it to 1.

1.2.7 TROMIN input signal

The PB3/TROMIN input signal on ST625X/6XC devices for serial programming has been inverted.

This has to be taken into account if the user has implemented his own programming algorithm and does not use STMicroelectronics or third party programming tools.

2 DEVELOPMENT TOOLS

The information given in this section applies only to ST development tools. For information on Third Party tools, please contact your supplier.

2.1 STARTER KIT

Some communication problems have been encountered when programming ST625X/6XC devices with the old ST626X starter kit (MB080 board). If, during the programming phase, the message "Cannot communicate with the device" is displayed, first check the parallel cable and jumper settings. If everything is correct, we then recommend to replace the U2 chip (74HC125) by a 74LS125 which has better thresholds.

To program ST625X/6XC devices we also advise to use the latest version of the ST626X starter kit (MB196 board) (which can be easily distinguished from the old one as it has 2 sockets for programming DIP16/20/28 devices). The order code of the new starter kit is ST626XC-KIT.

2.2 HDS2 EMULATOR SERIES

The ST626X-EMU2 emulator can be used to emulate ST625X/6XC devices without any modification. However, new features such as the LVD, OSG, LFAO are only emulated on the new ST62GP-EMU2 emulator.

2.3 EPROM PROGRAMMING BOARD

In order to program ST625X/6XC devices, the ST62E6XB EPROM Programming Board firmware has been updated. The revision level is identified by a sticker on the main IC on the board. The marking must be 310-32 or higher.

If the board has an earlier revision level, contact your local distributor or ST sales office to order an upgrade or a new EPB (Order code ST626XC-EPB).

2.4 GANG PROGRAMMER

Like the EPB, the firmware of the gang programmer has been updated. The marking on the chip must be at least 623-30.

If the programmer has an earlier revision level, contact your local distributor or ST sales office to order an upgrade or a new Gang Programmer.

2.5 WINDOWS EPROMER

To program ST625X/6XC devices, it is mandatory to use the Windows Epromer V3.2 or above.

The latest version available can be downloaded from our web site at the address:
http://mcu.st.com.