AN2419
Application note

STR75x hardware development getting started

Introduction

This application note is intended for system designers who require a hardware implementation overview of the development board features such as the power supply, the clock management, the reset control, the boot mode settings and the debug management. It shows how to use the STR75x product family and describes the minimum hardware resources required to develop an STR75x application.

Detailed reference design schematics are also contained in this document with descriptions of the main components, interfaces and modes.

Contents

1 Power supplies

1.1 Introduction

The device has five power pins:

- V_{DD_IO}: power supply for I/Os (3.3V ±0.3V or 5V ±0.5V). Must be kept on, even in STANDBY mode.
- V_{18} (pins V_{18REG} and V_{18} which are internally shorted): Power Supply for Digital, SRAM and Flash: 1.8V ± 0.15V.
- V_{18_BKP}: Backup Power Supply for STANDBY or STOP Mode

Two embedded regulators are available to supply the internal 1.8V digital power:

V_{18} and V_{18_BKP} are normally generated internally by these regulators.

The Main Voltage Regulator (MVREG) supplies V_{18} and V_{18_BKP}. It delivers a power supply of 1.8V ± 0.15V.

The Low Power Voltage Regulator (LPVREG) can supply V_{18_BKP} or V_{18} in STOP or STANDBY mode. It delivers a power supply of around 1.4V.

Two sensitive analog blocks have dedicated power pins:

- V_{DDA_PLL}: Analog Power supply for PLL (must have the same voltage level as V_{DD_IO})
- V_{DDA_ADC}: Analog Power supply for ADC (must have the same voltage level as V_{DD_IO})

1.2 Power supply schemes

The device can be connected in any of the following configurations depending on your application requirements:

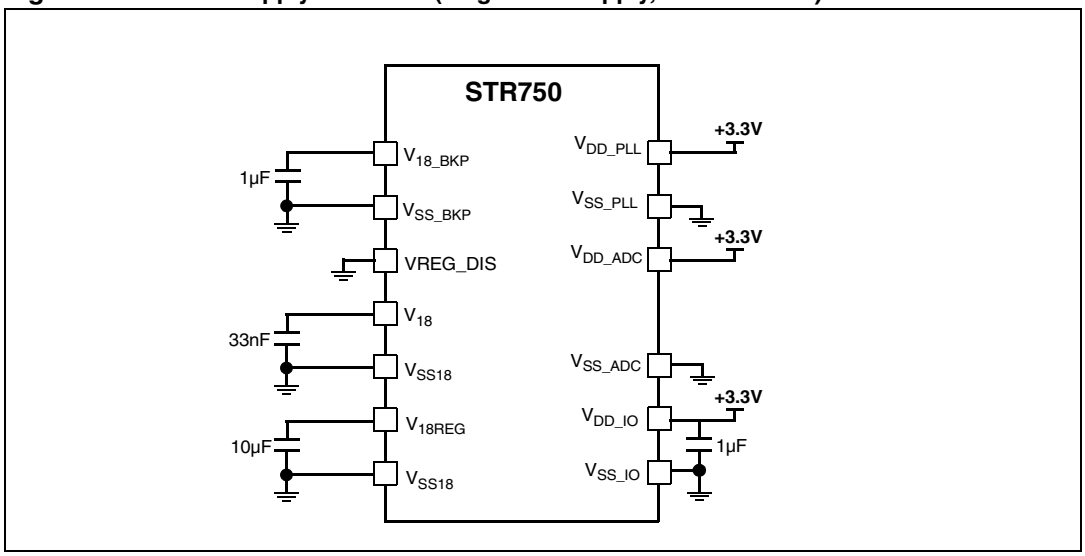
- Power Scheme 1: Single external 3.3V power source
- Power Scheme 2: Dual external 3.3V and 1.8V power sources
- Power Scheme 3: Single external 5.0V power source
- Power Scheme 4: Dual external 5.0V and 1.8V power sources

1.2.1 Power scheme 1: single external 3.3V power source

In this configuration, the internal voltage regulators are switched on by forcing the VREG_DIS pin to low level.

- The V_{18REG} pin must be connected to external stabilization capacitors (min. 10 uF Tantalum, low series resistance).
- The V_{18} pin must be connected to external stabilization capacitors (33nF ceramic).
- The V_{18_BKP} pin must be connected to an external stabilization capacitor of 1µF.
- A decoupling capacitor of 1µF must be added on the V_{DD_IO} pin which is closest to the V_{18REG} pin.

Figure 1. Power supply scheme 1 (single 3.3V supply, VREGDIS=0) in NORMAL mode



1.2.2 Power scheme 2: dual external 3.3V and 1.8V power sources

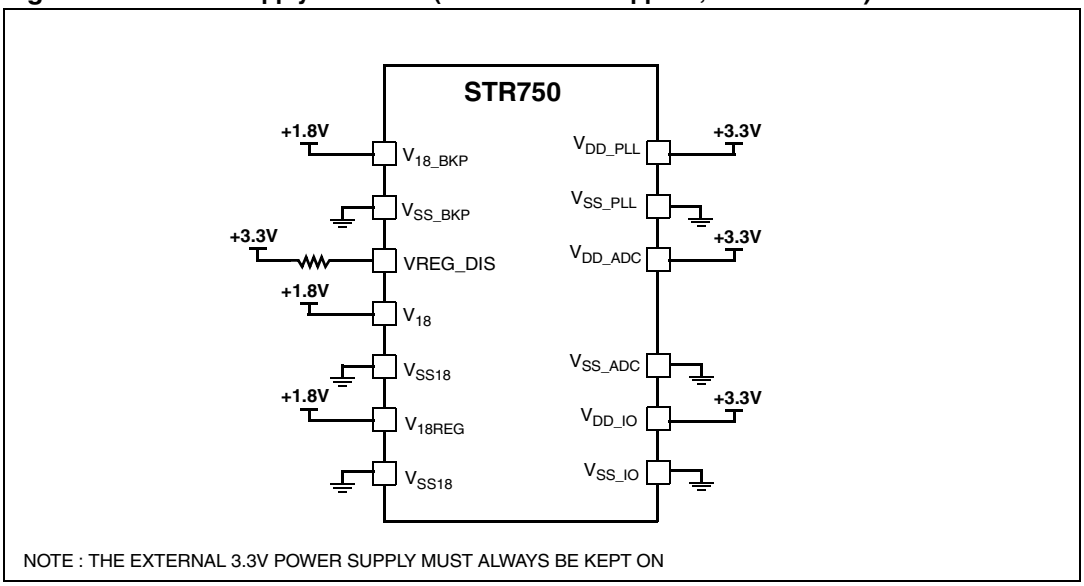
In this configuration, the internal voltage regulators are switched off by forcing the VREG_DIS pin to high level. This scheme has the advantage of saving power consumption when the 1.8V power supply is already available in the application. V_{18} and V_{18_BKP} are provided externally through the V_{18REG}, V_{18} and V_{18_BKP} power pins.

- The external 3.3V power supply must always be kept on.
- **VREG_DIS pin is tied to high level** which disables the Main Voltage Regulator and the Low Power Voltage Regulator.

Caution: When powered by 5.0V, the USB peripheral cannot operate.

- All digital power pins (V_{18REG}, V_{18} and V_{18_BKP}) **must be externally shorted** to the same 1.8V power supply source.
- In this scheme, **STANDBY Mode is not available**.

Figure 2. Power supply scheme 2 (3.3V and 1.8V supplies, VREGDIS=1)



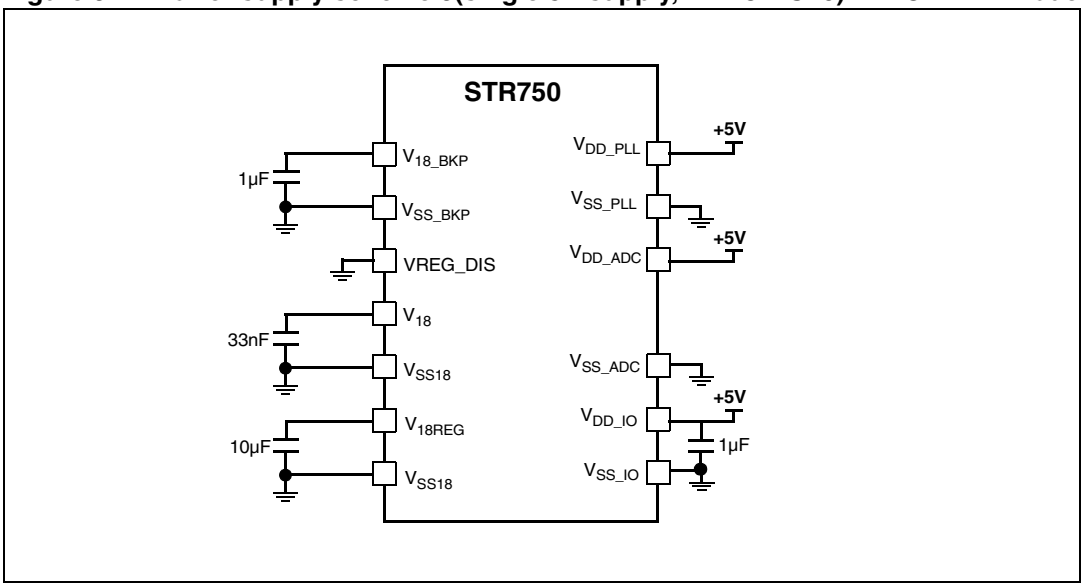
1.2.3 Power scheme 3: single external 5V power source

In this configuration, the internal voltage regulators are switched on by forcing the VREG_DIS pin to low level.

- The V_{18REG} pin must be connected to external stabilization capacitors (min. 10 uF Tantalum, low series resistance).
- The V_{18} pin must be connected to external stabilization capacitors (33nF ceramic).
- The V_{18_BKP} pin must be connected to an external stabilization capacitor of 1µF.
- A decoupling capacitor of 1µF must be added on the V_{DD_IO} pin which is closest to the V_{18REG} pin.

Caution: When powered by 5.0V, the USB peripheral cannot operate.

Figure 3. Power supply scheme 3(single 5V supply, VREGDIS=0) in NORMAL mode



1.2.4 Power scheme 4: dual external 5.0V and 1.8V power sources

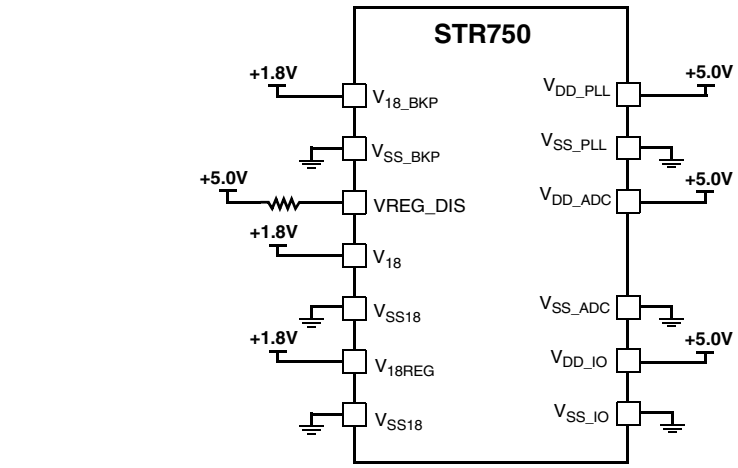
In this configuration, the internal voltage regulators are switched off, by forcing the VREG_DIS pin to high level. This scheme has the advantage of saving power consumption when the 1.8V power supply is already available in the application and providing 5V I/O capability. V_{18} and V_{18_BKP} are provided externally through the V_{18REG}, V_{18} and V_{18_BKP} power pins.

- VREG_DIS pin is tied to high level which disables the Main Voltage Regulator and the Low Power Voltage Regulator.
- All digital power pins (V_{18REG}, V_{18} and V_{18_BKP}) **must be externally shorted** to the same 1.8V power supply source.

In this scheme:

- STANDBY Mode is not available
- USB functionality is not available

Figure 4. Power supply scheme 4 (5.0V and 1.8V supplies, VREGDIS=1)



1.3 Reset and power startup

1.3.1 Power startup specifications

To ensure the MCU starts-up cleanly, the rise time of the V_{DD_IO} power supply must be comprised between 20µs/V and 20ms/V.

In addition, you must provide an external RESET for at least 20µs after the V_{DD_IO} power supply has reached its minimum working value (3.0V). It is recommended to use an external Power-On-Reset circuit monitoring V_{DD_IO} to assert the RESET at power-up .

During V_{DD_IO} power-up (from 0V to 3.3V or 5.0V), all I/Os are guaranteed to be in HiZ state, assuming external RESET is asserted.

If you are using an external 1.8V power supply, the rise time of power supply V_{18} must be comprised between 20µs/V and 20ms/V.

1.3.2 External reset

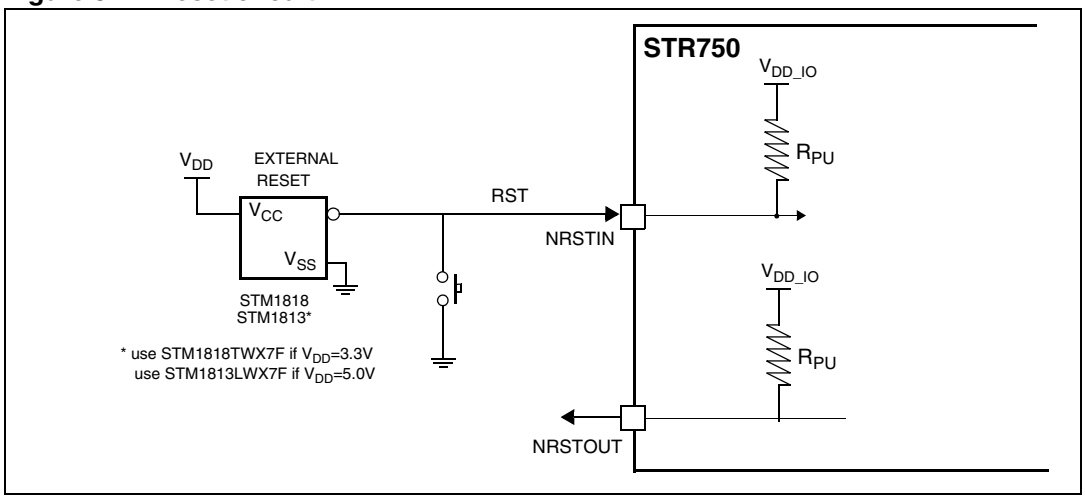
The NRSTIN pin acts as an asynchronous RESET active low.

The NRSTIN pad input is a Schmitt Trigger input pin. A filter is added to ignore all incoming pulses with short duration:

- All negative spikes with a duration less than 150ns are filtered.
- All trains of negative spikes with a ratio of 1/2 are filtered. This means that all spikes with a maximum duration of 150ns with minimum interval between spikes of 75ns are filtered.
- An internal pull up is connected on each of the pins NRSTIN and NRSTOUT.

An external reset circuit can be connected (STM1818) to manage the reset signal: both for power-on reset during startup and management of the reset button (debounce functionality and pulse duration control).

Figure 5. Reset circuit



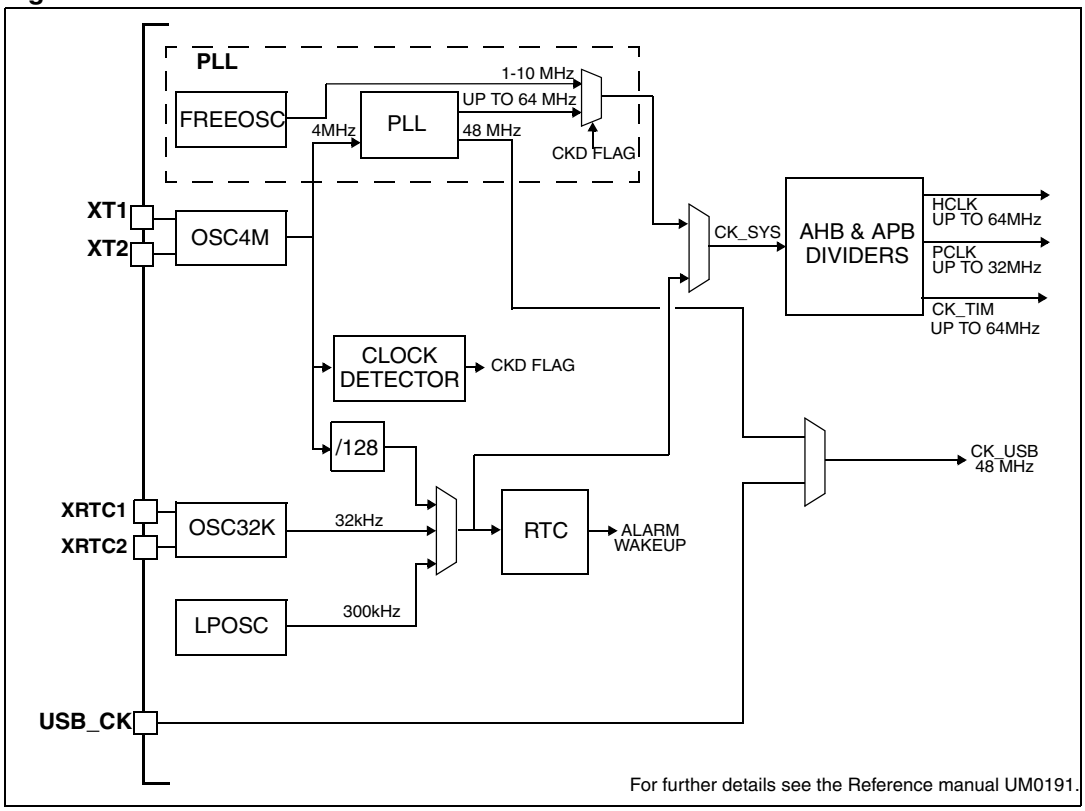
The NRSTOUT pin is an exact image of the system Reset signal (active low) provided to the device which is used to generate the reset of the AHB System and the reset of each APB peripherals. Internal circuitry guarantees a minimum reset pulse duration of 20µs.

Some peripheral registers, like the RTC or the BACKUP registers are not reset by a System Reset but only by an RSM Reset: this means that these registers are only reset at VDD power up. For more details see the *Reference manual UM0191*.

2 Clocks

2.1 Clock overview

Figure 6. Clock overview



Several on-chip oscillators can feed the MCU system clock (CK_SYS) from which the HCLK and PCLK derive:

- FREEOSC: Internal Free Running Oscillator providing a clock between 1 and 10MHz, also used as emergency clock. It consists of the internal VCO of the PLL configured in free running mode
- OSC4M: 4MHz or 8MHz Main Oscillator (Crystal or Ceramic oscillator or external clock)
  - a 4 MHz Crystal/Ceramic oscillator connected to XT1/XT2
  - or an 8 MHz Crystal/Ceramic oscillator to XT1/XT2 followed by a divider by 2
  - or external clock connected to XT1
- OSC32K: 32.768kHz Oscillator (Crystal or Ceramic oscillator) which can drive either the system clock and/or the RTC.
- LPOSC: Internal Low Power RC Oscillator providing a clock around 300kHz which can drive either the system clock and/or the RTC.

Several configurable dividers provide a high degree of flexibility to the application in the choice of the APB or AHB frequency, while keeping a fixed frequency value for the USB clock (48 MHz).

The Clock Detector (CKD) protects the Microcontroller against OSC4M or external clock failures.

The RTC provides calender, alarm and wake-up functions and can be clocked by any of the oscillators other than FREEOSC.

Caution: The FREEOSC and PLL are reset during the whole assertion of System RESET. After reset release, the PLL is disabled and FREEOSC supplies the system clock (1-10 MHz).

2.2 Main 4MHz or 8MHz oscillator (OSC4M)

XT1 and XT2 pins are used to connect the Main Oscillator source, which can be a resonator (crystal or ceramic) or an external source. Both sources can be used as the input clock to PLL frequency multiplier (PLL).

Crystal or ceramic resonator

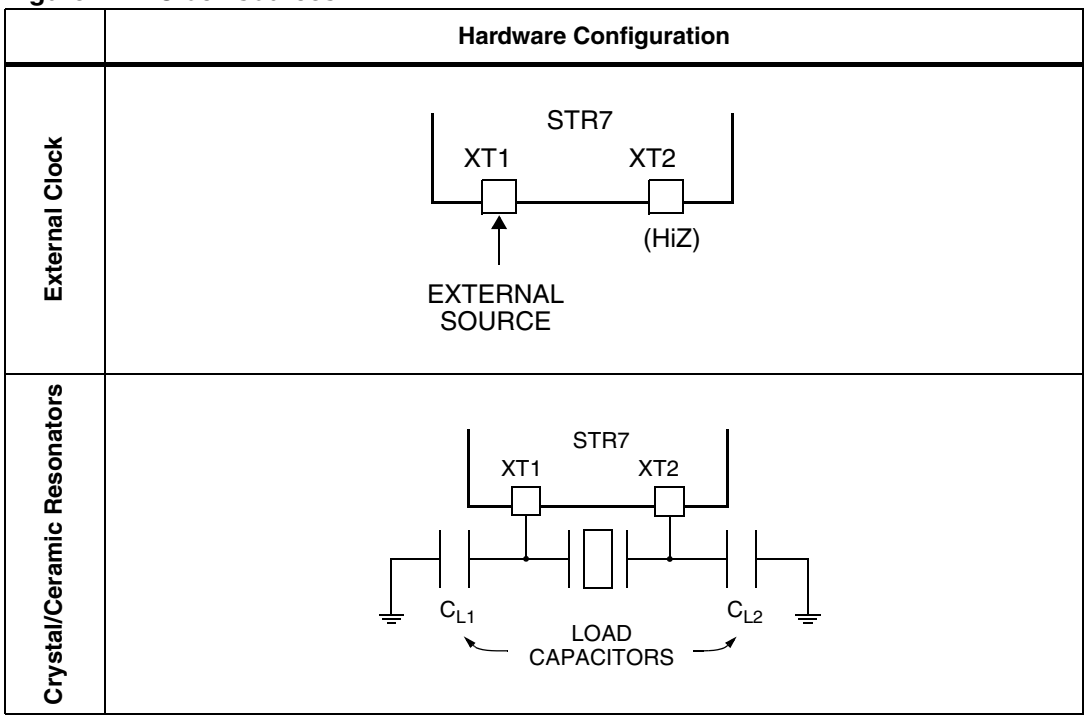
This 4 MHz oscillator (OSC4M) has the advantage of producing a very accurate rate on the main clock. This oscillator can be directly connected to

- a 4 MHz Oscillator
- or an 8 MHz Oscillator followed by an internal divider by 2.

If an 8 MHz Crystal or Ceramic is connected, You must select the divider by 2

The associated hardware configuration are shown in *Figure 7*. Refer to the electrical characteristics section of the datasheet for more details.

Figure 7. Clock sources



The resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and start-up stabilization time. The loading capacitance values must be adjusted according to the selected oscillator.

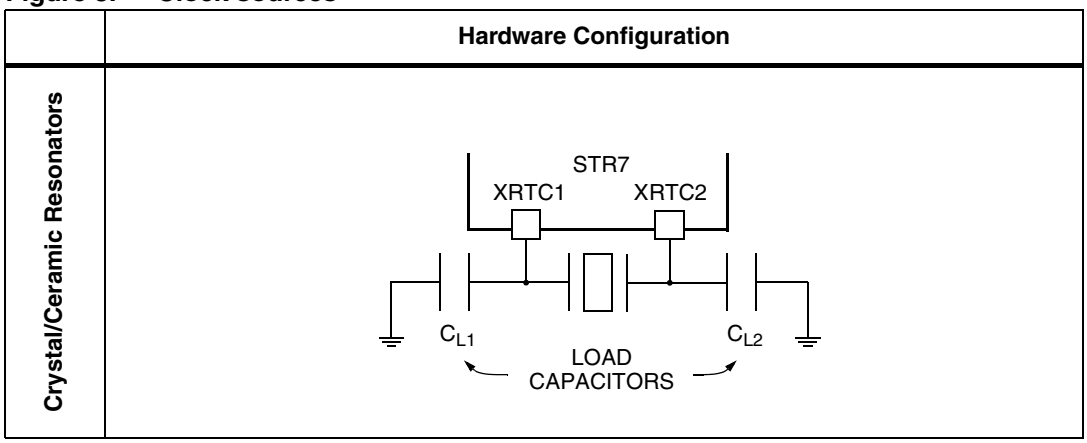
External Source (Bypass Mode)

In this mode, an external clock source must be provided. The external clock signal (square, sinus or triangle) with ~50% duty cycle has to drive the XT1 pin while the XT2 pin should be left hi-Z.

2.3 Low power 32.768 kHz oscillator (OSC32K)

XRTC1 and XRTC2 pins are used to connect the 32k Oscillator source, which can be a resonator (crystal or ceramic).The OSC32k clock can be used as a low power system clock

Figure 8. Clock sources



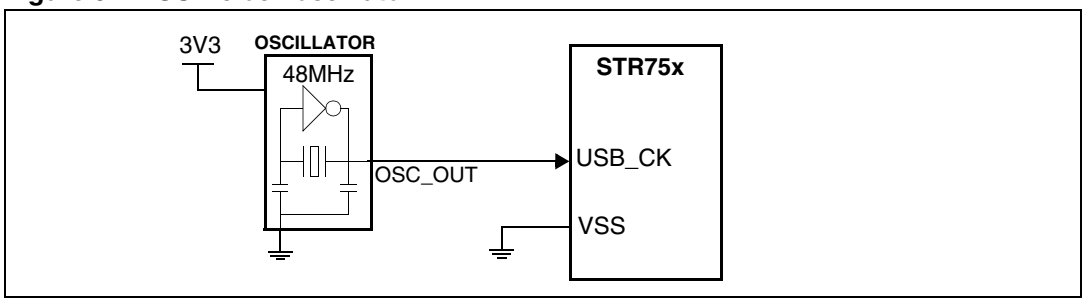
The resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and start-up stabilization time. The loading capacitance values must be adjusted according to the selected oscillator.

2.4 USB clock

STR75x series microcontrollers contain a USB 2.0 Full Speed device module interface that operates at a precise frequency of 48 MHz. This clock is usually generated by the internal PLL using one single external oscillator for both the system and USB module to save board space and cost. However, if the chosen system clock is not compatible with the 48 MHz clock generation (frequency below 48 MHz), the USB clock can also provided by an external oscillator connected to the USB clock pin USBCLK.

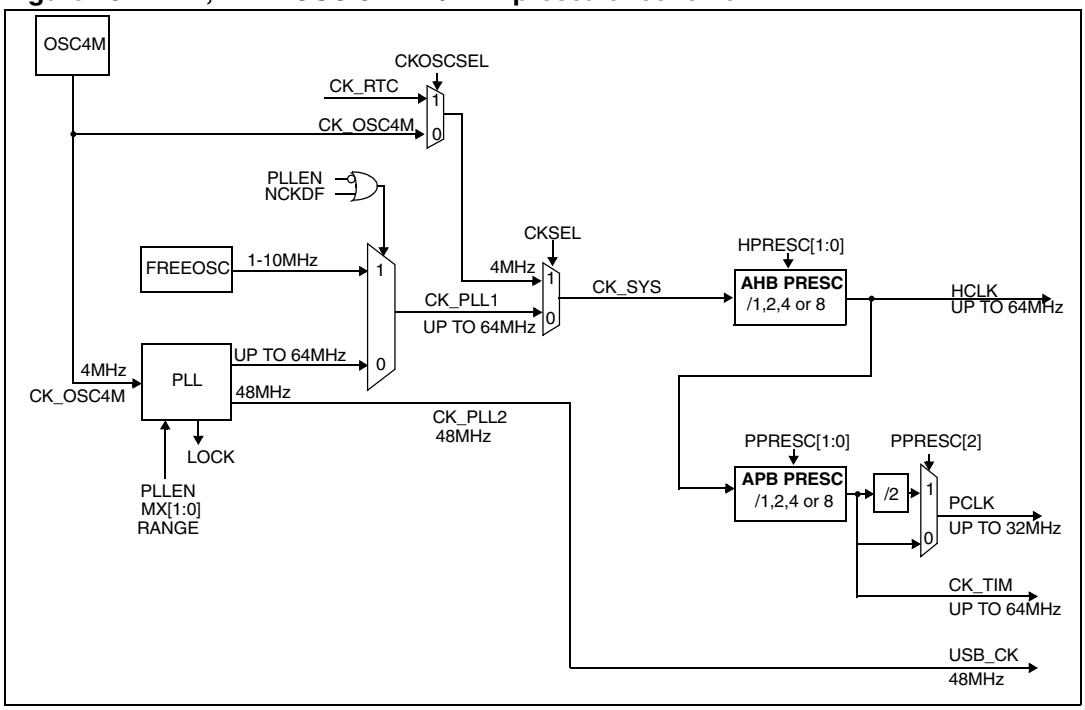
The following diagram shows the basic implementation of the USB external clock.

Figure 9. USB clock oscillator



2.5 PLL, FREEOSC, and AHB/APB prescalers

Figure 10. PLL, FREEOSC & AHB/APB prescaler scheme



The CPU can execute:

- the SRAM up to 64 MHz at zero wait state.
- the Flash up to 60 MHz speed in burst mode (zero wait state for consecutive accesses, 1 wait state for any non-consecutive access)
- the Flash up to 32 MHz at zero wait state

The PLL provides a Frequency Multiplier starting from a single input clock (OSC4M source) and providing the 2 following independent output clocks:

- CK_PLL1 output with 4 programmable multiplication factors (up to 64 MHz) used for generating CK_SYS
- CK_PLL2 output with fixed 48 MHz frequency when input clock is 4 MHz used for generating CK_USB

FREEOSC provides a Free Running Oscillator for the system clock: this clock is selected when:

- The PLL is disabled: FREEOSC acts as an oscillator source
- The clock failure flag is active: FREEOSC acts as an emergency clock source.

For security, hardware prevents the software from making certain unrecoverable errors:

- Software cannot switch CK_SYS to CK_PLL1 (multiplied clock output) until the PLL is locked.
- CK_USB is gated until the software enables the PLL2EN bit in the MRCC_CLKCTL register.

The AHB and APB prescalers allow you to choose the AHB and APB frequencies from a wide range of possibilities:

- HCLK (AHB Clock) can be generated from CK_SYS divided by 1, 2, 4 or 8
- PCLK (APB Clock) can be generated from HCLK divided by 1, 2, 4, 8 or 16

The clock provided to the Timers (CK_TIM) can have twice or the same frequency as PCLK. This allows the timers to count at high frequency (up to 64MHz)

Table 1 gives some typical clock configurations:

Table 1. Typical prescaler uses

f_{OSC4M} (MHz)	PLL Factor	f_{CK_SYS} (MHz)	HPRESC [1:0]	f_{HCLK} (MHz)	PPRESC [1:0]	f_{CK_TIM} (MHz)	PPRESC2	f_{PCLK} (MHz)	f_{CK_USB} (MHz)
4	x16	64	00	64	00	64	1	32	48
					01	32		16	
	x15	60	00	60	00	60		30	
	x14	56	00	56	00	56		28	
	x12	48	00	48	00	48		24	
	x16	64	01	32	00	32		16	

The software must respect the configuration constraints of the PLL.

2.6 Clock-out capability: MCO (Main Clock Output)

The Main Clock Output (MCO) capability allows you to output a clock on the external MCO pin. The configuration registers of corresponding GPIO port must programmed in alternate function mode. You can select one of 4 clock signals as MCO clock.

- CK_PLL2
- HCLK
- PCLK
- CK_OSC4M

The selection is controlled by the MCOS[1:0] bits of CLKCTL register. A dedicated prescaler (divide by 1 or 2, selected by MCOP bit) can be applied to this clock before outputting it to the MCO pin. Care must be taken when switching MCO clock selection, the alternate function should be disabled to avoid any glitches on the MCO pin.

2.7 Clock detector (CKD)

A CKD (Clock Detector) is implemented to:

- detect if no clock is present on OSC4M (broken or disconnected resonator) and prevent the software from selecting it.
- automatically feeds the MCU with the FREEOSC used as emergency clock if no clock is detected.
- generate an interrupt if enabled, allowing the MCU to perform some rescue operations

For more detail see the *Reference Manual (UM0191).*

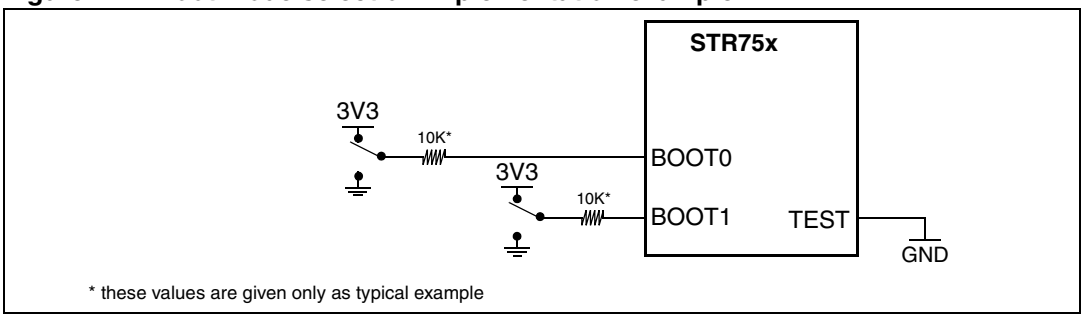
3 Boot configuration

In the STR750, 4 different boot modes can be selected by means of the BOOT[1:0] pins as shown in *Table 2*.

Table 2. Boot modes

BOOT Mode Selection Pins		Boot Mode	Aliasing	Note
BOOT1	BOOT0			
0	0	Embedded Flash	Embedded FLASH sector B0F0 mapped at 0h	All FLASH sectors accessible except SystemMemory sector
1	0	Embedded SRAM	Embedded SRAM mapped at 0h	
0	1	SystemMemory	SystemMemory mapped at 0h	-
1	1	External SMI Serial Flash memory	SMI Bank 0 mapped at 0h	-

Figure 11. Boot mode selection implementation example



This aliases the physical memory associated with each boot mode to Block 000 (boot memory).The value of the BOOT pin is latched on the 4th rising edge of CK_SYS after Reset.

It is up to the user to manage the BOOT1 and BOOT0 pins at reset release to select the required boot mode. Note that the user should also manage these pins when exiting Standby mode as the BOOT pins are resampled.

Even when aliased in the boot memory space, the related memory (FLASH, SRAM or SMI) is still accessible at its original memory space.

After this start-up delay has elapsed, the ARM CPU will start code execution from the boot memory space, located at the bottom of the memory space starting from 0x0000_0000h.

The application can read the status of the boot pins that was latched at start-up and change the memory aliasing on-the-fly by modifying the SW_BOOT bits in the CFG_GLCONF register.

Caution: The TEST pin of the STR75x must always be forced to ground (ST reserved test pin)

3.1 Embedded boot loader mode

Embedded Boot Loader Mode is used to re-program the FLASH using one of the serial interfaces (typically a UART). This program, called, *ICP boot loader,* is located in the SystemMemory and is programmed by ST during production.

Refer to the *STR7 Family Flash Programming Reference Manual* for details.

3.2 External memory (SMI) boot mode

When SMI boot mode is selected the Serial Memory Interface is automatically configured as follows:

- Chip Select Polarity = low
- SMI bank 0 is selected and the associated I/O alternate functions are enabled.
- Boot Space (0000_0000h to 00FF_FFFFh -16MB) is aliased to SMI bank 0.
- The SMI is configured as NORMAL READ MODE (reset value)
- The SMI_PRESCALER is set to “2” (reset value)

Programming considerations when booting from SMI

After RESET, the PLL is disabled and both CK_SYS and HCLK are clocked by the internal FREEOSC oscillator (1-10MHz). Consequently, the SMI clock output is also between 0.5 and 5MHz (SMI_PRESCALER reset state is 2). To use a higher frequency, software has to configure the clock and PLLs.

Care is needed if the program performing the PLL and clock configuration is executed directly from serial memory. The software must ensure that a proper clock frequency is provided to the serial memory when changing the SMI_PRESCALER and switching the system clock. That is why the SMI_PRESCALER must be changed first before switching the system clock to the PLL output clock.

For example, to use the SMI in NORMAL READ MODE with a 60 MHz HCLK frequency, set the SMI_PRESCALER to 4 before switching HCLK to 60 MHz. The SMI clock frequency is then 60 MHz / 4 = 15 MHz.

It is possible to obtain the highest SMI frequency using the FAST READ MODE, if the serial memory supports this mode. For instance, f_{HCLK} can be set to 48 MHz and the SMI_PRESCALER can be be loaded to "1" to address a high speed Serial Memory at 48 MHz (for read only).

Note: Make sure that the SMI clock frequency (resulting from of your AHB clock and SMI_PRESCALER settings) does not exceed the maximum allowed value. The maximum frequency of the SMI is limited by the I/O speed.

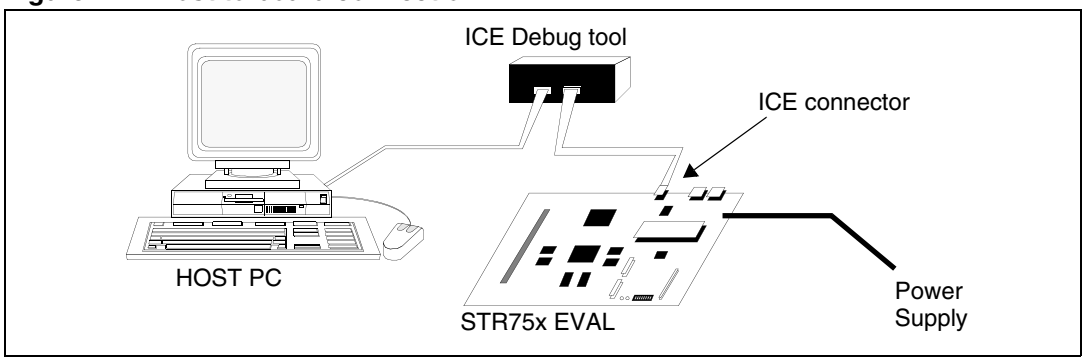
Take care to load the SMI_PRESCALER and the FAST_READ mode with the same write transaction. This can be done by a program which is executed from the serial Flash. In this case, the SMI will change the clock and the READ MODE only at the end of this access. For more details refer to the *Reference Manual*.

4 Debug management

The Host/Target interface is the hardware equipment that connects the Host to the application board. This interface is made of three components: a hardware debug tool, such as Micro-ICE from ARM, a JTAG connector and a cable connecting the host to the debug tool.

Figure 12 shows the connection of the host to the STR75x board.

Figure 12. Host to board connection



4.1 ICE debug tool

ICE Debug tool is a host interface that connects a PC to an STR75x development board featuring a debug interface as shown in *Figure 12*. The Embedded ICE is an intelligent host interface that provides fast access to host services, access to on-chip emulation and debug facilities. When the STR75x board is configured as a stand-alone system, the ICE Debug tool can be used to download programs.

The STR75x development kit supports the ARM RealView ICE Micro Edition. The Micro-ICE is plugged in to the host via a USB cable.

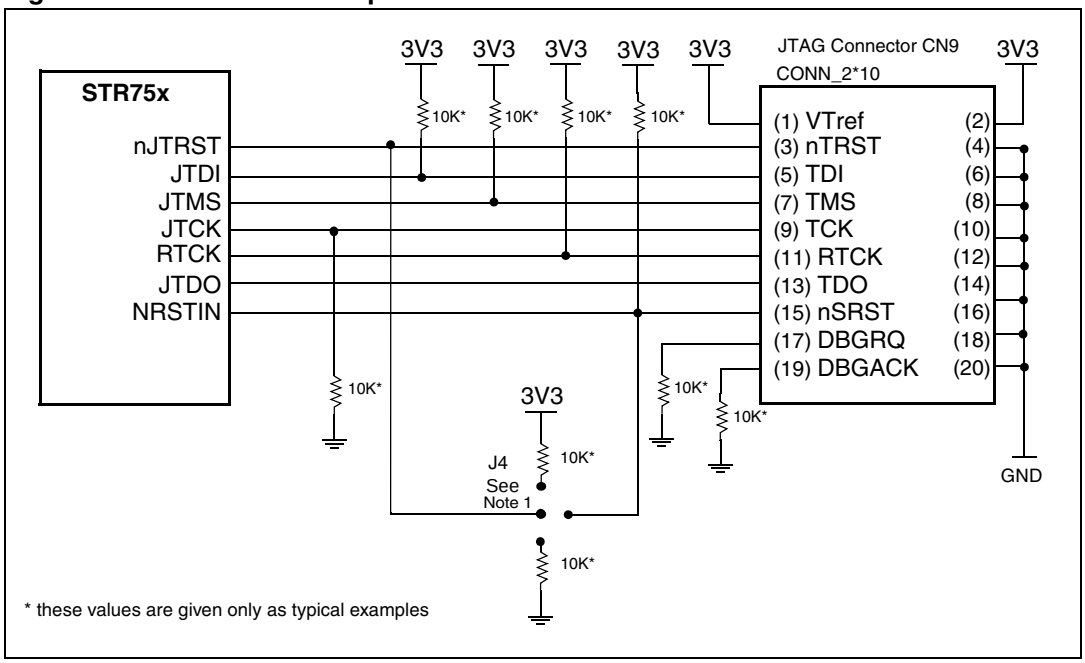
4.2 JTAG / ICE connector

The ICE connector enables JTAG hardware debugging equipment, such as RealView-ICE, to be connected to the STR75x board. It is possible to both drive and sense the system-reset line, and to then send a JTAG reset to the core through the ICE connector. *Figure 13* shows the ARM ICE connector pin-out.

The STR75x has a user debug interface. This interface contains a five-pin serial interface conforming to JTAG, IEEE standard 1149.1-1993, “Standard Test Access Port-Scan Boundary Architecture”. JTAG allows the ICE device to be plugged to the board and used to debug the software running on the STR75x.

JTAG emulation allows the core to be started and stopped under the control of the connected debugger software. The user can then display and modify registers and memory contents, and set break and watch points.

Figure 13. Ice connector implementation



Note: 1 The JTAG reset pin (nJTRST) is a dedicated pin:

When JTAG communication is used on the application board, nJTRST must be connected to a pull-up. However, it is mandatory to apply one negative pulse on nJTRST pin after power-up otherwise the MCU may enter unexpected test modes. For instance, nJTSRT can be connected to the system reset which is pulled-up.

When JTAG communication is not used on the application board, nJTRST can be connected to the system reset or it can be also indefinitely grounded.

Caution: When the internal Flash is readout protected, the Flash will not execute if nJRST is not continuously reset because it is considered as an intrusion. Consequently, in this case, it is mandatory that the nJRST pin sees a continuous low level after reset when the user application is running.

Table 3. JTAG connector pins

Std Name	STR75x	Description	Function
nTRST	JTRST	Test Reset (from JTAG equipment)	This active LOW open-collector is used to reset the JTAG port and the associated debug circuitry. It is asserted at power-up by each module, and can be driven by the JTAG equipment.
TDI	JTDI	Test data in (from JTAG equipment)	TDI goes down the stack of modules to the motherboard and then back up the stack, labelled TDO, connecting to each component in the scan chain.
TMS	JTMS	Test mode select (from JTAG equipment)	TMS controls transitions in the tap controller state machine. TMS connects to all JTAG components in the scan chain as the signal flows down the module stack.
TCK	JTCK	Test clock (from JTAG equipment)	TCK synchronizes all JTAG transactions. TCK connects to all JTAG components in the scan chain. Series termination resistors are used to reduce reflections and maintain good signal integrity. TCK flows down the stack of modules and connects to each JTAG component. However, if there is a device in the scan chain that synchronizes TCK to some other clock, then all down-stream devices are connected to the RTCK signal on that component.
RTCK	RTCK	Return TCK (to JTAG equipment)	Using a mechanism called adaptive clocking, the RTCK signal is returned by the core to the JTAG equipment, and the clock is not advanced until the core had captured the data. In adaptive clocking mode, the debugging equipment waits for an edge on RTCK before changing TCK.
TDO	JTDO	Test data out (to JTAG equipment)	TDO is the return path of the data input signal TDI.
nSRST	NRSTIN	System reset (bidirectional)	nSRST is an active LOW open-collector signal that can be driven by the JTAG equipment to reset the target board. Some JTAG equipment senses this line to determine when a board has been reset by the user. When the signal is driven LOW by the reset controller on the core module, the motherboard resets the whole system by driving nSYSRST low.

For more details on the JTAG port refer to the IEEE standard 1149.1-1993, “Standard Test Access Port-Scan Boundary Architecture” specification.

5 Reference design

5.1 Main

This reference design is based on the STR750, a highly integrated microcontroller, running at 64MHz that combines the popular ARM7TDMI-S™ 32-bit RISC CPU with 256 Kbytes of embedded Flash, 16Kbytes of high speed SRAM.

5.1.1 Clock

Two clock sources are used for the microcontroller.

- X1-32 kHz crystal for embedded RTC
- X2-4 MHz crystal for STR750Fx microcontroller

5.1.2 Reset

The reset signal on this STR750-REF schematic is active low.
The reset sources include:

- Power On Reset from STM1818 (U2)
- Reset button (PB1)
- Debugging tools via the connector CN1

5.1.3 Boot mode

The STR750 is able to boot from:

- Embedded User Flash
- Embedded SRAM for debugging
- System memory with boot loader for ISP
- External SPI Interface Flash

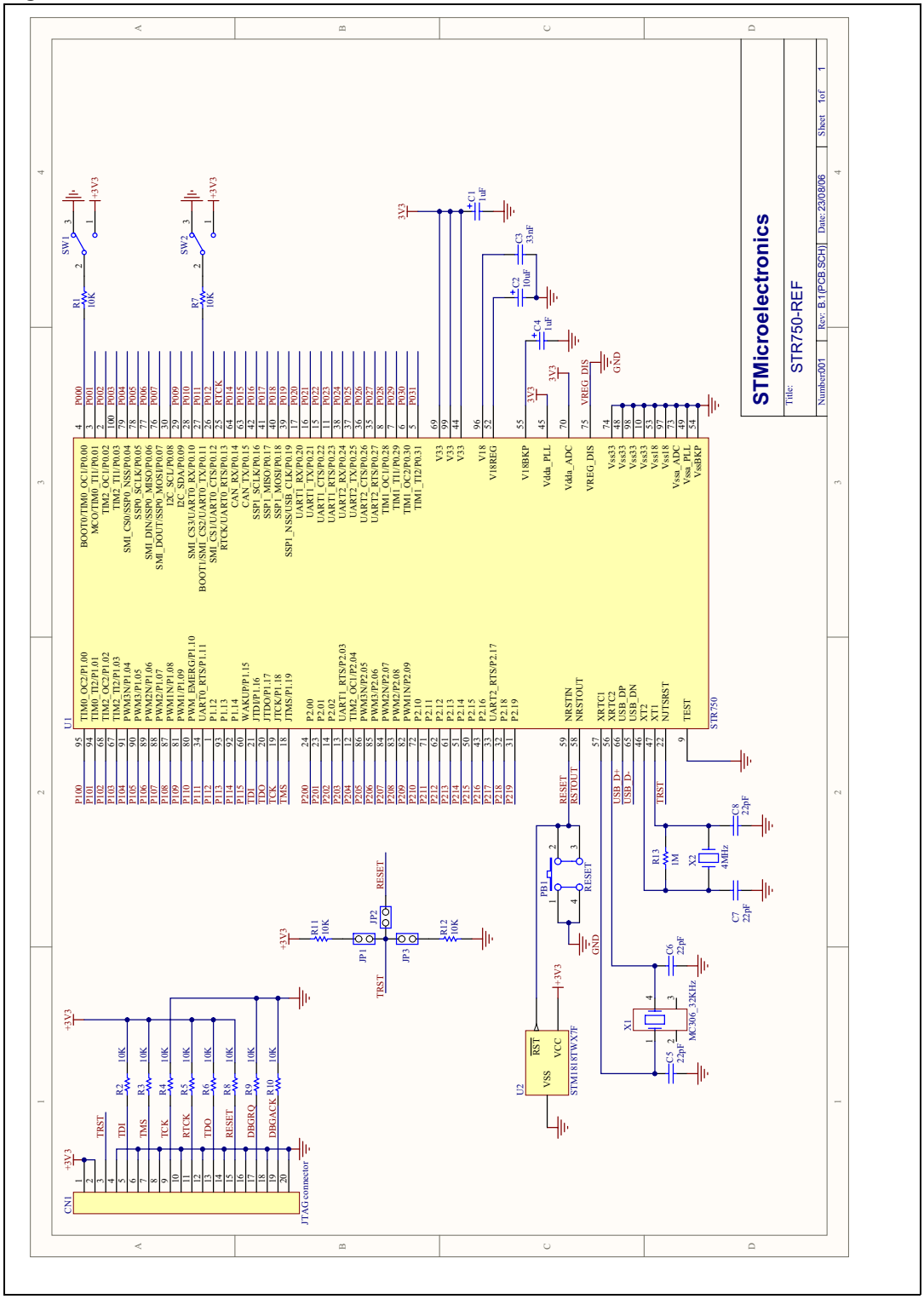
The boot option is configured by setting switches SW1 (Boot 0) and SW2 (Boot 1).

5.2 JTAG interface

Refer to the section *Section 4: Debug management on page 16*.

6 Schematics

Figure 14. STR750F microcontroller connections



7 Revision history

Table 4. Document revision history

Date	Revision	Changes
20-Nov-2006	1	Initial release.
09-Jul-2007	2	References to VREF_ADC pin removed from document *Table 1: Typical prescaler uses on page 13* enhanced