Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	ARM9®
Core Size	16/32-Bit
Speed	96MHz
Connectivity	CANbus, EBI/EMI, Ethernet, I²C, IrDA, Microwire, SPI, SSI, SSP, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, WDT
Number of I/O	80
Program Memory Size	512KB (512K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	96K x 8
Voltage - Supply (Vcc/Vdd)	1.65V ~ 2V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	128-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/str912fw44x6

2.10 Clock control unit (CCU)

The CCU generates a master clock of frequency f_{MSTR}. From this master clock the CCU also generates individually scaled and gated clock sources to each of the following functional blocks within the STR91xF.

- CPU, f_{CPUCLK}
- Advanced High-performance Bus (AHB), f_{HCLK}
- Advanced Peripheral Bus (APB), f_{PCLK}
- Flash Memory Interface (FMI), f_{FMICLK}
- External Memory Interface (EMI), f_{BCLK}
- UART Baud Rate Generators, f_{BAUD}
- USB, f_{USB}

2.10.1 Master clock sources

The master clock in the CCU (f_{MSTR}) is derived from one of three clock input sources. Under firmware control, the CPU can switch between the three CCU inputs without introducing any glitches on the master clock output. Inputs to the CCU are:

- Main Oscillator (f_{OSC}). The source for the main oscillator input is a 4 to 25 MHz external crystal connected to STR91xF pins X1_CPU and X2_CPU, or an external oscillator device connected to pin X1_CPU.
- PLL (f_{PLL}). The PLL takes the 4 to 25 MHz oscillator clock as input and generates a master clock output up to 96 MHz (programmable). By default, at power-up the master clock is sourced from the main oscillator until the PLL is ready (locked) and then the CPU may switch to the PLL source under firmware control. The CPU can switch back to the main oscillator source at any time and turn off the PLL for low-power operation. The PLL is always turned off in Sleep mode.
- RTC (f_{RTC}). A 32.768 kHz external crystal can be connected to pins X1_RTC and X2_RTC, or an external oscillator connected to pin X1_RTC to constantly run the real-time clock unit. This 32.768 kHz clock source can also be used as an input to the CCU to run the CPU in slow clock mode for reduced power.

As an option, there are a number of peripherals that do not have to receive a clock sourced from the CCU. The USB interface can receive an external clock on pin P2.7, TIM timers TIM0/TIM1 can receive an external clock on pin P2.4, and timers TIM2/TIM3 on pin P2.5.

2.11 Flexible power management

The STR91xF offers configurable and flexible power management control that allows the user to choose the best power option to fit the application. Power consumption can be dynamically managed by firmware and hardware to match the system's requirements. Power management is provided via clock control to the CPU and individual peripherals.

Clocks to the CPU and peripherals can be individually divided and gated off as needed. In addition to individual clock divisors, the CCU master clock source going to the CPU, AHB, APB, EMI, and FMI can be divided dynamically by as much as 1024 for low power operation. Additionally, the CCU may switch its input to the 32.768 kHz RTC clock at any time for low power.

The STR91xF supports the following three global power control modes:

- **Run Mode:** All clocks are on with option to gate individual clocks off via clock mask registers.
- **Idle Mode:** CPU and FMI clocks are off until an interrupt, reset, or wake-up occurs. Pre-configured clock mask registers selectively allow individual peripheral clocks to continue run during Idle Mode.
- **Sleep Mode:** All clocks off except RTC clock. Wake up unit remains powered, PLL is forced off.

A special mode is used when JTAG debug is active which never gates off any clocks even if the CPU enters Idle or Sleep mode.

2.11.1 Run mode

This is the default mode after any reset occurs. Firmware can gate off or scale any individual clock. Also available is a special Interrupt Mode which allows the CPU to automatically run full speed during an interrupt service and return back to the selected CPU clock divisor rate when the interrupt has been serviced. The advantage here is that the CPU can run at a very low frequency to conserve power until a periodic wake-up event or an asynchronous interrupt occurs at which time the CPU runs full speed immediately.

2.11.2 Idle mode

In this mode the CPU suspends code execution and the CPU and FMI clocks are turned off immediately after firmware sets the Idle Bit. Various peripherals continue to run based on the settings of the mask registers that exist just prior to entering Idle Mode. There are 3 ways to exit Idle Mode and return to Run Mode:

- Any reset (external reset pin, watchdog, low-voltage, power-up, JTAG debug command)
- Any interrupt (external, internal peripheral, RTC alarm or interval)
- Input from wake-up unit on GPIO pins

Note: It is possible to remain in Idle Mode for the majority of the time and the RTC can be programmed to periodically wake up to perform a brief task or check status.

2.11.3 Sleep mode

In this mode all clock circuits except the RTC are turned off and main oscillator input pins X1_CPU and X2_CPU are disabled. The RTC clock is required for the CPU to exit Sleep Mode. The entire chip is quiescent (except for RTC and wake-up circuitry). There are three means to exit Sleep Mode and re-start the system:

peripheral clock from the APB, and an 8-bit clock pre-scaler is available. When enabled by firmware as a watchdog, this timer will cause a system reset if firmware fails to periodically reload this timer before the terminal count of 0x0000 occurs, ensuring firmware sanity. The watchdog function is off by default after a reset and must be enabled by firmware.

2.13.4 External RESET_INn pin

This input signal is active-low with hystereses (V_{RHYS}). Other open-drain, active-low system reset signals on the circuit board (such as closure to ground from a push-button) may be connected directly to the RESET_INn pin, but an external pull-up resistor to V_{DDQ} must be present as there is no internal pullup on the RESET_INn pin.

A valid active-low input signal of t_{RINMIN} duration on the RESET_INn pin will cause a system reset within the STR91xF. There is also a RESET_OUTn pin on the STR91xF that can drive other system components on the circuit board. RESET_OUTn is active-low and has the same timing of the Power-On-Reset (POR) shown next, t_{POR}.

2.13.5 Power-up

The LVD circuitry will always generate a global reset when the STR91xF powers up, meaning internal reset is active until V_{DDQ} and V_{DD} are both above the LVD thresholds. This POR condition has a duration of t_{POR}, after which the CPU will fetch its first instruction from address 0x0000.0000 in Flash memory. It is not possible for the CPU to boot from any other source other than Flash memory.

2.13.6 JTAG debug command

When the STR91xF is in JTAG debug mode, an external device which controls the JTAG interface can command a system reset to the STR91xF over the JTAG channel.

2.13.7 Tamper detection

On 128-pin STR91xF devices only, there is a tamper detect input pin, TAMPER_IN, used to detect and record the time of a tamper event on the end product such as malicious opening of an enclosure, unwanted opening of a panel, etc. The activation mode of the tamper pin is programmable to one of two modes. One is Normally Closed/Tamper Open, the other mode will detect when a signal on the tamper input pin is driven from low-to-high, or high-to-low depending on firmware configuration. Once a tamper event occurs, the RTC time (millisecond resolution) and the date are recorded in the RTC unit. Simultaneously, the SRAM standby voltage source will be cut off to invalidate all SRAM contents. Tamper detection control and status logic are part of the RTC unit.

2.14 Real-time clock (RTC)

The RTC combines the functions of a complete time-of-day clock (millisecond resolution) with an alarm programmable up to one month, a 9999-year calender with leap-year support, periodic interrupt generation from 1 to 512 Hz, tamper detection (described in *Section 2.13.7*), and an optional clock calibration output on the JRTCK pin. The time is in 24 hour mode, and time/calendar values are stored in binary-coded decimal format.

The RTC also provides a self-isolation mode that is automatically activated during power down. This feature allows the RTC to continue operation when V_{DDQ} and V_{DD} are absent, as long as

an alternate power source, such as a battery, is connected to the VBATT input pin. The current drawn by the RTC unit on the VBATT pin is very low in this standby mode, I_{RTC_STBY}.

2.15 JTAG interface

An IEEE-1149.1 JTAG interface on the STR91xF provides In-System-Programming (ISP) of all memory, boundary scan testing of pins, and the capability to debug the CPU.

STR91xF devices are shipped from ST with blank Flash memories. The CPU can only boot from Flash memory (selection of which Flash bank is programmable). Firmware must be initially programmed through JTAG into one of these Flash memories before the STR91xF is used.

Six pins are used on this JTAG serial interface. The five signals JTDI, JTDO, JTMS, JTCK, and JTRSTn are all standard JTAG signals complying with the IEEE-1149.1 specification. The sixth signal, JRTCK (Return TCK), is an output from the STR91xF and it is used to pace the JTCK clock signal coming in from the external JTAG test equipment for debugging. The frequency of the JTCK clock signal coming from the JTAG test equipment must be at least 10 times less than the ARM966E-S CPU core operating frequency (f_{CPUCLK}). To ensure this, the signal JRTCK is output from the STR91xF and is input to the external JTAG test equipment to hold off transitions of JTCK until the CPU core is ready, meaning that the JTAG equipment cannot send the next rising edge of JTCK until the equipment receives a rising edge of JRTCK from the STR91xF. The JTAG test equipment must be able to interpret the signal JRTCK and perform this adaptive clocking function. If it is known that the CPU clock will always be at least ten times faster than the incoming JTCK clock signal, then the JRTCK signal is not needed.

The two die inside the STR91xF (CPU die and Flash memory die) are internally daisy-chained on the JTAG bus, see *Figure 3 on page 22*. The CPU die has two JTAG Test Access Ports (TAPs), one for boundary scan functions and one for ARM CPU debug. The Flash memory die has one TAP for program/erase of non-volatile memory. Because these three TAPs are daisy-chained, only one TAP will converse on the JTAG bus at any given time while the other two TAPs are in BYPASS mode. The TAP positioning order within this JTAG chain is the boundary scan TAP first, followed by the ARM debug TAP, followed by the Flash TAP. All three TAP controllers are reset simultaneously by one of two methods:

- A chip-level global reset, caused only by a Power-On-Reset (POR) or a Low Voltage Detect (LVD).
- A reset command issued by the external JTAG test equipment. This can be the assertion of the JTAG JTRSTn input pin on the STR91xF or a JTAG reset command shifted into the STR91xF serially.

This means that chip-level system resets from watchdog time-out or the assertion of RESET_INn pin do not affect the operation of any JTAG TAP controller. Only global resets effect the TAPs.

- Supports USB low and full-speed transfers (12 Mbps), certified to comply with the USB 2.0 specification
- Supports isochronous, bulk, control, and interrupt endpoints
- Configurable number of endpoints allowing a mixture of up to 20 single-buffered monodirectional endpoints or up to 10 double-buffered bidirectional endpoints
- Dedicated, dual-port 2 Kbyte USB Packet Buffer SRAM. One port of the SRAM is connected by a Packet Buffer Interface (PBI) on the USB side, and the CPU connects to the other SRAM port.
- CRC generation and checking
- NRZI encoding-decoding and bit stuffing
- USB suspend resume operations

2.18.1 Packet buffer interface (PBI)

The PBI manages a set of buffers inside the 2 Kbyte Packet Buffer, both for transmission and reception. The PBI will choose the proper buffer according to requests coming from the USB Serial Interface Engine (SIE) and locate it in the Packet SRAM according to addresses pointed by endpoint registers. The PBI will also auto-increment the address after each exchanged byte until the end of packet, keeping track of the number of exchanged bytes and preventing buffer overrun. Special support is provided by the PBI for isochronous and bulk transfers, implementing double-buffer usage which ensures there is always an available buffer for a USB packet while the CPU uses a different buffer.

2.18.2 DMA

A programmable DMA channel may be assigned by CPU firmware to service the USB interface for fast and direct transfers between the USB bus and SRAM with little CPU involvement. This DMA channel includes the following features:

- Direct USB Packet Buffer SRAM to system SRAM transfers of receive packets, by descriptor chain for bulk or isochronous endpoints.
- Direct system SRAM to USB Packet Buffer SRAM transfers of transmit packets, by descriptor chain for bulk or isochronous endpoints.
- Linked-list descriptor chain support for multiple USB packets

2.18.3 Suspend mode

CPU firmware may place the USB interface in a low-power suspend mode when required, and the USB interface will automatically wake up asynchronously upon detecting activity on the USB pins.

2.19 CAN 2.0B interface

The STR91xF provides a CAN interface complying with CAN protocol version 2.0 parts A and B. An external CAN transceiver device connected to pins CAN_RX and CAN_TX is required for connection to the physical CAN bus.

The CAN interface manages up to 32 Message Objects and Identifier Masks using a Message SRAM and a Message Handler. The Message Handler takes care of low-level CAN bus activity such as acceptance filtering, transfer of messages between the CAN bus and the Message

high-impedance state when not selected. The STR91xF supports SPI multi-Master operation because it provides collision detection.

Each SSP interface on the STR91xF has the following features:

- Full-duplex, three or four-wire synchronous transfers
- Master or Slave operation
- Programmable clock bit rate with prescaler, up to 24MHz for Master mode and 4MHz for Slave mode
- Separate transmit and receive FIFOs, each 16-bits wide and 8 locations deep
- Programmable data frame size from 4 to 16 bits
- Programmable clock and phase polarity
- Specifically for Microwire protocol:
  - Half-duplex transfers using 8-bit control message
- Specifically for SSI protocol:
  - Full-duplex four-wire synchronous transfer
  - Transmit data pin tri-stateable when not transmitting

2.22.1 DMA

A programmable DMA channel may be assigned by CPU firmware to service each SSP channel for fast and direct transfers between the SSP bus and SRAM with little CPU involvement. Both DMA single-transfers and DMA burst-transfers are supported for transmit and receive. Burst transfers require that FIFOs are enabled.

2.23 General purpose I/O

There are up to 80 GPIO pins available on 10 I/O ports for 128-pin devices, and up to 40 GPIO pins on 5 I/O ports for 80-pin devices. Each and every GPIO pin by default (during and just after a reset condition) is in high-impedance input mode, and some GPIO pins are additionally routed to certain peripheral function inputs. CPU firmware may initialize GPIO pins to have alternate input or output functions as listed in *Table 3*. At any time, the logic state of any GPIO pin may be read by firmware as a GPIO input, regardless of its reassigned input or output function.

Bit masking is available on each port, meaning firmware may selectively read or write individual port pins, without disturbing other pins on the same port during a write.

Firmware may designate each GPIO pin to have open-drain or push-pull characteristics.

All GPIO pins are 5V tolerant, meaning in they can drive a voltage level up to V_{DDQ}, and can be safely driven by a voltage up to 5.5V.

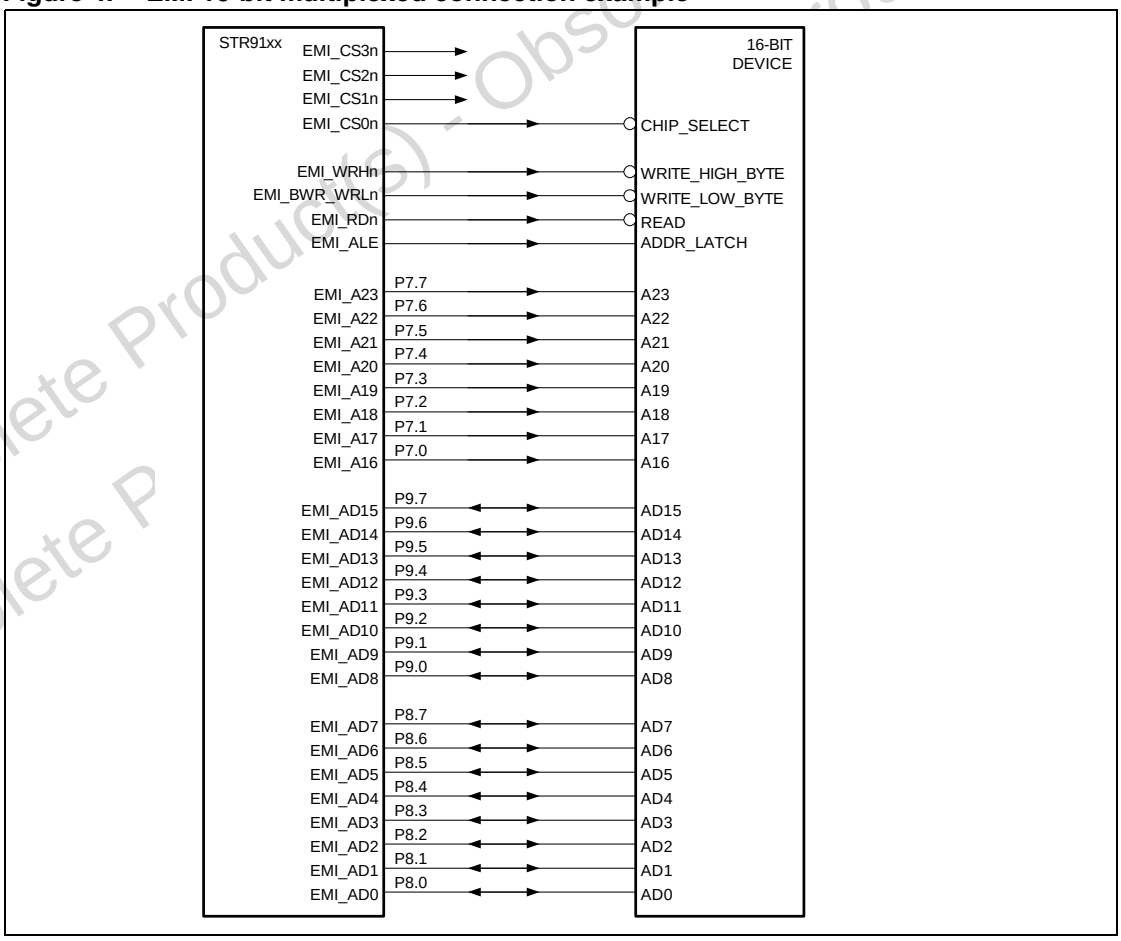
There are no internal pull-up or pull-down resistors on GPIO pins. As such, it is recommended to ground, or pull up to V_{DDQ} with a 100KΩ resistor, all unused GPIO pins to minimize power consumption and noise generation.

respectively. The output signal EMI_RDn is the read strobe for both the low and high data bytes.

- **8-bit multiplexed data mode**: This is a variant of the 16-bit multiplexed mode. Although this mode can provide 24 bits of address and 8 bits of data, it does require an external latch device on Port 8. However, this mode is most efficient when connecting devices that only require 8 bits of address on an 8-bit multiplexed address/data bus, and have simple read, write, and latch inputs as shown in *Figure 5*

  To use all 24 address bits, the following applies: 8 bits of lowest-order data and 8 bits of lowest-order address are multiplexed on port 8. On port 9, 8-bits of mid-order address are multiplexed with 8 bits of data, but these 8 data values are always at logic zero on this port during a write operation, and these 8 data bits are ignored during a read operation. An external latch device (such as a '373 latch) is needed to de-multiplex the mid-order 8 address bits that are generated on port 8. Port 7 outputs the 8 highest-order address signals directly (not multiplexed). The output signal on pin EMI_ALE is used to demultiplex the signals on ports 8 and 9, and the polarity of EMI_ALE is programmable. The output signal on pin EMI_BWR_WRLn is the data write strobe, and the output on pin EMI_RDn is the data read strobe.
- **8-bit non-multiplexed data mode** (*Figure 6*): Eight bits of data are on port 8, while 16 bits of address are output on ports 7 and 9. The output signal on pin EMI_BWR_BWLn is the data write strobe and the output on pin EMI_RDn is the data read strobe.

Figure 4. EMI 16-bit multiplexed connection example



Pkg		Pin Name	Signal Type	Default Pin Function	Default Input Function	Alternate functions			
LQFP80	LQFP128					Alternate Input 1	Alternate Output 1	Alternate Output 2	Alternate Output 3
43	68	P3.7	I/O	GPIO_3.7, GP Input, HiZ	EXINT7, External Intr	SSP1_NSS, SSP slv select in	GPIO_3.7, GP Output	SSP1_NSS, SSP mstr sel out	TIM1_OCMP1, *Out comp/PWM*
4	3	P4.0	I/O	GPIO_4.0, GP Input, HiZ	ADC0, ADC input chnl	TIM0_ICAP1, Input Capture	GPIO_4.0, GP Output	TIM0_OCMP1, *Out comp/PWM*	ETM_PCK0, *ETM Packet*
3	2	P4.1	I/O	GPIO_4.1, GP Input, HiZ	ADC1, ADC input chnl	TIM0_ICAP2, Input Capture	GPIO_4.1, GP Output	TIM0_OCMP2, *Out comp*	ETM_PCK1, *ETM Packet*
2	1	P4.2	I/O	GPIO_4.2, GP Input, HiZ	ADC2, ADC input chnl	TIM1_ICAP1, Input Capture	GPIO_4.2, GP Output	TIM1_OCMP1, *Out comp/PWM*	ETM_PCK2, *ETM Packet*
1	128	P4.3	I/O	GPIO_4.3, GP Input, HiZ	ADC3, ADC input chnl	TIM1_ICAP2, Input Capture	GPIO_4.3, GP Output	TIM1_OCMP2, *Out comp*	ETM_PCK3, *ETM Packet*
80	127	P4.4	I/O	GPIO_4.4, GP Input, HiZ	ADC4, ADC input chnl	TIM2_ICAP1, Input Capture	GPIO_4.4, GP Output	TIM2_OCMP1, *Out comp/PWM*	ETM_PSTAT0, ETM pipe status
79	126	P4.5	I/O	GPIO_4.5, GP Input, HiZ	ADC5, ADC input chnl	TIM2_ICAP2, Input Capture	GPIO_4.5, GP Output	TIM2_OCMP2, *Out comp*	ETM_PSTAT1, ETM pipe status
78	125	P4.6	I/O	GPIO_4.6, GP Input, HiZ	ADC6, ADC input chnl	TIM3_ICAP1, Input Capture	GPIO_4.6, GP Output	TIM3_OCMP1, *Out comp/PWM*	ETM_PSTAT2, ETM pipe status
77	124	P4.7	I/O	GPIO_4.7, GP Input, HiZ	ADC7, ADC input chnl	TIM3_ICAP2, Input Capture	GPIO_4.7, GP Output	TIM3_OCMP2, *Out comp*	ETM_TRSYNC, ETM trace sync
9	12	P5.0	I/0	GPIO_5.0, GP Input, HiZ	EXINT8, External Intr	CAN_RX, CAN rcv data	GPIO_5.0, GP Output	ETM_TRCLK, ETM trace clock	UART0_TX, UART xmit data
12	18	P5.1	I/0	GPIO_5.1, GP Input, HiZ	EXINT9, External Intr	UART0_RxD, UART rcv data	GPIO_5.1, GP Output	CAN_TX, CAN Tx data	UART2_TX, UART xmit data
17	25	PHYCLK _P5.2	I/O	GPIO_5.2, GP Input, HiZ	EXINT10, External Intr	UART2_RxD, UART rcv data	GPIO_5.2, GP Output	MII_PHYCLK, *25Mhz to PHY*	TIM3_OCMP1, *Out comp/PWM*
18	27	P5.3	I/O	GPIO_5.3, GP Input, HiZ	EXINT11, External Intr	ETM_EXTRIG, ETM ext. trigger	GPIO_5.3, GP Output	MII_TX_EN, MAC xmit enbl	TIM2_OCMP1, *Out comp/PWM*
44	70	P5.4	I/O	GPIO_5.4, GP Input, HiZ	EXINT12, External Intr	SSP0_SCLK, SSP slv clk in	GPIO_5.4, GP Output	SSP0_SCLK, SSP mstr clk out	EMI_CS0n, EMI Chip Select
47	77	P5.5	I/O	GPIO_5.5, GP Input, HiZ	EXINT13, External Intr	SSP0_MOSI, SSP slv dat in	GPIO_5.5, GP Output	SSP0_MOSI, SSP mstr dat out	EMI_CS1n, EMI Chip Select
48	79	P5.6	I/O	GPIO_5.6, GP Input, HiZ	EXINT14, External Intr	SSP0_MISO, SSP mstr dat in	GPIO_5.6, GP Output	SSP0_MISO, SSP slv data out	EMI_CS2n, EMI Chip Select
49	80	P5.7	I/O	GPIO_5.7, GP Input, HiZ	EXINT15, External Intr	SSP0_NSS, SSP slv select in	GPIO_5.7, GP Output	SSP0_NSS, SSP mstr sel out	EMI_CS3n, EMI Chip Select
19	29	P6.0	I/O	GPIO_6.0, GP Input, HiZ	EXINT16, External Intr	TIM0_ICAP1, Input Capture	GPIO_6.0, GP Output	TIM0_OCMP1, *Out comp/PWM*	MC_UH, IMC phase U hi
20	31	P6.1	I/O	GPIO_6.1, GP Input, HiZ	EXINT17, External Intr	TIM0_ICAP2, Input Capture	GPIO_6.1, GP Output	TIM0_OCMP2, *Out comp*	MC_UL, IMC phase U lo
13	19	P6.2	I/O	GPIO_6.2, GP Input, HiZ	EXINT18, External Intr	TIM1_ICAP1, Input Capture	GPIO_6.2, GP Output	TIM1_OCMP1, *Out comp/PWM*	MC_VH, IMC phase V hi
14	20	P6.3	I/O	GPIO_6.3, GP Input, HiZ	EXINT19, External Intr	TIM1_ICAP2, Input Capture	GPIO_6.3, GP Output	TIM1_OCMP2, *Out comp*	MC_VL, IMC phase V lo
52	83	P6.4	I/O	GPIO_6.4, GP Input, HiZ	EXINT20, External Intr	TIM2_ICAP1, Input Capture	GPIO_6.4, GP Output	TIM2_OCMP1, *Out comp/PWM*	MC_WH, IMC phase W hi
53	84	P6.5	I/O	GPIO_6.5, GP Input, HiZ	EXINT21, External Intr	TIM2_ICAP2, Input Capture	GPIO_6.5, GP Output	TIM2_OCMP2, *Out comp*	MC_WL, IMC phase W lo
57	92	P6.6	I/O	GPIO_6.6, GP Input, HiZ	EXINT22_TRIG, Ext Intr & Tach	UART0_RxD, UART rcv data	GPIO_6.6, GP Output	TIM3_OCMP1, *Out comp/PWM*	ETM_TRCLK, ETM trace clock
58	93	P6.7	I/O	GPIO_6.7, GP Input, HiZ	EXINT23_STOP, Ext Intr & Estop	ETM_EXTRIG, ETM ext. trigger	GPIO_6.7, GP Output	TIM3_OCMP2, *Out comp*	UART0_TX, UART xmit data

Pkg		Pin Name	Signal Type	Default Pin Function	Default Input Function	Alternate functions			
LQFP80	LQFP128					Alternate Input 1	Alternate Output 1	Alternate Output 2	Alternate Output 3
-	21	EMI_BWR_WRLn (used as EMI_LBn in future rev.)	O	EMI byte write strobe (8 bit mode) or low byte write strobe (16 bit mode)	N/A				
-	22	EMI_WRHn (used as EMI_UBn in future rev.)	O	EMI high byte write strobe (16-bit mode)	N/A				
-	74	EMI_ALE	O	EMI address latch enable (mux mode)	N/A				
-	75	EMI_RDn	O	EMI read strobe	N/A				
-	-	EMI_BAAn	O	TBD	N/A				
-	-	EMI_WAITn	I	TBD	N/A				
-	-	EMI_BCLK	O	TBD	N/A				
-	-	EMI_WEn	O	Reserved for future use	N/A				
-	91	TAMPER_IN	I	Tamper detection input	N/A				
-	94	MII_MDIO	I/O	MAC/PHY management data line	N/A				
59	95	USBDN	I/O	USB data (-) bus connect	N/A				
60	96	USBDP	I/O	USB data (+) bus connect	N/A				
56	89	RESET_INn	I	External reset input	N/A				
62	100	RESET_OUTn	O	Global or System reset output	N/A				
65	104	X1_CPU	I	CPU oscillator or crystal input	N/A				
64	103	X2_CPU	O	CPU crystal connection	N/A				
27	42	X1_RTC	I	RTC oscillator or crystal input (32.768 kHz)	N/A				
26	41	X2_RTC	O	RTC crystal connection	N/A				
61	97	JRTCK	O	JTAG return clock or RTC clock	N/A				
67	107	JTRSTn	I	JTAG TAP controller reset	N/A				
68	108	JTCK	I	JTAG clock	N/A				
69	111	JTMS	I	JTAG mode select	N/A				
72	115	JTDI	I	JTAG data in	N/A				
73	117	JTDO	O	JTAG data out	N/A				
-	122	AVDD	V	ADC analog voltage source, 2.7V - 3.6V	N/A				
-	4	AVSS	G	ADC analog ground	N/A				
5	-	AVSS_VSSQ	G	Common ground point for digital I/O & analog ADC	N/A				

When other AHB bus masters (such as a DMA controller) write to SRAM, their access is never buffered. Only the CPU can make use of buffered AHB writes.

5.4 Two independent Flash memories

The STR91xF has two independent Flash memories, the larger primary Flash and the small secondary Flash. It is possible for the CPU to erase/write to one of these Flash memories while simultaneously reading from the other.

One or the other of these two Flash memories may reside at the “boot” address position of 0x0000.0000 at power-up or at reset as shown in *Figure 9*. The default configuration is that the first sector of primary Flash memory is enabled and residing at the boot position, and the secondary Flash memory is disabled. This default condition may be optionally changed as described below.

5.4.1 Default configuration

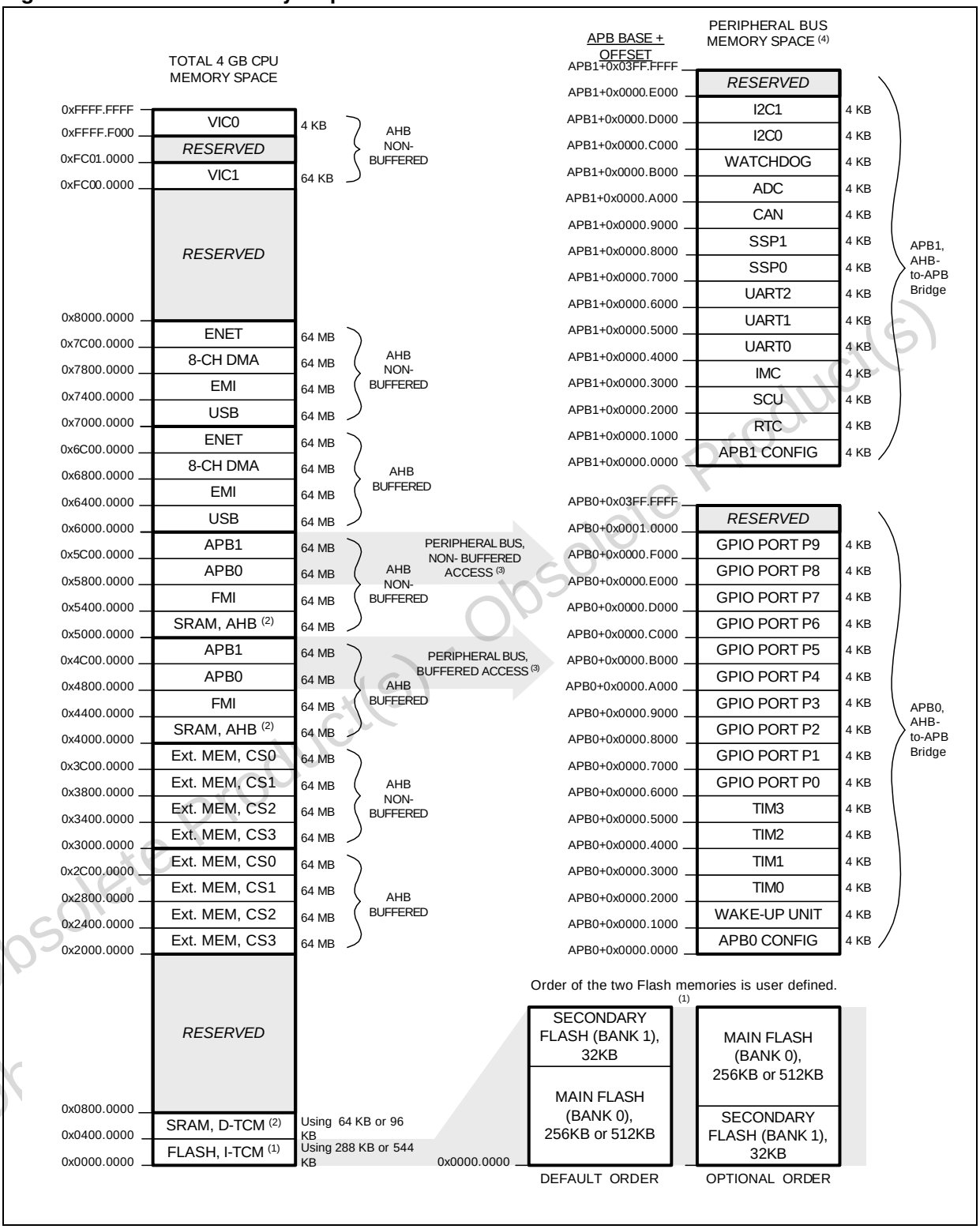
When the primary Flash resides at boot position, typical CPU initialization firmware would set the start address and size of the main Flash memory, and go on to enable the secondary Flash, define it’s start address and size. Most commonly, firmware would place the secondary Flash start address at the location just after the end of the primary Flash memory. In this case, the primary Flash is used for code storage, and the smaller secondary flash can be used for data storage (EEPROM emulation).

5.4.2 Optional configuration

Using the STR91xF device configuration software tool, or IDE from 3rd party, one can specify that the smaller secondary Flash memory is at the boot location at reset and the primary Flash is disabled. The selection of which Flash memory is at the boot location is programmed in a non-volatile Flash-based configuration bit during JTAG ISP. The boot selection choice will remain as the default until the bit is erased and re-written by the JTAG interface. The CPU cannot change this choice for boot Flash, only the JTAG interface has access.

In this case where the secondary Flash defaults to the boot location upon reset, CPU firmware would typically initialize the Flash memories the following way. The secondary Flash start address and size is specified, then the primary Flash is enabled and its start address and size is specified. The primary Flash start address would typically be located just after the final address location of the secondary Flash. This configuration is particularly well-suited for In-Application-Programming (IAP). The CPU would boot from the secondary Flash memory, initialize the system, then check the contents of the primary Flash memory (by checksum or other means). If the contents of primary Flash is OK, then CPU execution continues from either Flash memory. If the main Flash contents are incorrect, the CPU, while executing code from the secondary Flash, can download new data from any STR91xF communication channel and program into primary Flash memory. Application code then starts after the new contents of primary Flash are verified.

Figure 9. STR91xF memory map



6.7 Main oscillator electrical characteristics

V_{DDQ} = 2.7 - 3.6V, V_{DD} = 1.65 - 2V, T_A = -40 / 85 °C unless otherwise specified.

Table 11. Main oscillator electrical characteristics

Symbol	Parameter	Test Conditions	Value			Unit
			Min	Typ	Max	
$t_{STUP(OSC)}$	Oscillator Start-up Time	Stable V_{DDQ}			3	mS

6.8 RTC oscillator electrical characteristics

V_{DDQ} = 2.7 - 3.6V, V_{DD} = 1.65 - 2V, T_A = -40 / 85 °C unless otherwise specified.

Table 12. RTC oscillator electrical characteristics

Symbol	Parameter	Test Conditions	Value			Unit
			Min	Typ	Max	
$g_{M(RTC)}$	Oscillator Start _voltage		LVD [1)]			V
$t_{STUP(RTC)}$	Oscillator Start-up Time	Stable V_{DDQ}			1	S

Notes: 1 Min oscillator start voltage is the same as low voltage detect level (2.4V or 2.7V) for VDDQ

Table 13. RTC crystal electrical characteristics

Symbol	Parameter	Test Conditions	Value			Unit
			Min	Typ	Max	
f_O	Resonant frequency			32.768		kHz
R_S	Series resistance				40	kΩ
C_L	Load capacitance			8		pF

6.9 PLL electrical characteristics

V_{DDQ} = 2.7 - 3.6V, V_{DD} = 1.65 - 2V, T_A = -40 / 85 °C unless otherwise specified.

Table 14. PLL Electrical Characteristics

Symbol	Parameter	Test Conditions	Value			Unit
			Min	Typ	Max	
f_{PLL}	PLL Output Clock		6.25		96	MHz
f_{OSC}	Clock Input		4		25	MHz
t_{LOCK}	PLL lock time			300	1500	µs
Δt_{JITTER}	PLL Jitter (peak to peak)			0.1	0.2	ns

6.11.3 Absolute Maximum Ratings (Electrical Sensitivity)

Based on three different tests (ESD, LU and DLU) using specific measurement methods, the product is stressed in order to determine its performance in terms of electrical sensitivity. For more details, refer to the application note AN1181.

6.11.4 Electro-Static Discharge (ESD)

Electro-Static Discharges (3 positive then 3 negative pulses separated by 1 second) are applied to the pins of each sample according to each pin combination. The sample size depends on the number of supply pins in the device (3 parts*(n+1) supply pin). Two models can be simulated: Human Body Model and Charge Device Model. This test conforms to the JESD22-A114A/A115A standard.

Table 19. ESD Absolute Maximum ratings

Symbol	Ratings	Conditions	Maximum value [1]	Unit
$V_{ESD(HBM)}$	Electro-static discharge voltage (Human Body Model)	T_A=+25°C	+/-2000	V
$V_{ESD(CDM)}$	Electro-static discharge voltage (Charge Device Model)		1000	

Notes:

1. Data based on characterization results, not tested in production.

6.11.5 Static and Dynamic Latch-Up

- **LU**: 3 complementary static tests are required on 10 parts to assess the latch-up performance. A supply overvoltage (applied to each power supply pin) and a current injection (applied to each input, output and configurable I/O pin) are performed on each sample. This test conforms to the EIA/JESD 78 IC latch-up standard. For more details, refer to the application note AN1181.
- **DLU**: Electro-Static Discharges (one positive then one negative test) are applied to each pin of 3 samples when the micro is running to assess the latch-up performance in dynamic mode. Power supplies are set to the typical values, the oscillator is connected as near as possible to the pins of the micro and the component is put in reset mode. This test conforms to the IEC1000-4-2 and SAEJ1752/3 standards. For more details, refer to the application note AN1181.

6.11.6 Designing hardened software to avoid noise problems

EMC characterization and optimization are performed at component level with a typical application environment and simplified MCU software. It should be noted that good EMC performance is highly dependent on the user application and the software in particular.

Therefore it is recommended that the user applies EMC software optimization and prequalification tests in relation with the EMC level requested for his application.

Software recommendations:

The software flowchart must include the management of runaway conditions such as:

- Corrupted program counter
- Unexpected reset

Table 26. Ethernet MII management timing table

Symbol	Parameter	Symbol	Value		Unit
			Min	Max	
1	MDIO delay from rising edge of MDC	t_c(MDIO)		2.83	ns
2	MDIO setup time to rising edge of MDC	T_{su} (MDIO)	2.70		ns
3	MDIO hold time from rising edge of MDC	T_h (MDIO)	-2.03		ns

Ethernet MII transmit timings

Figure 19. Ethernet MII transmit timing diagram

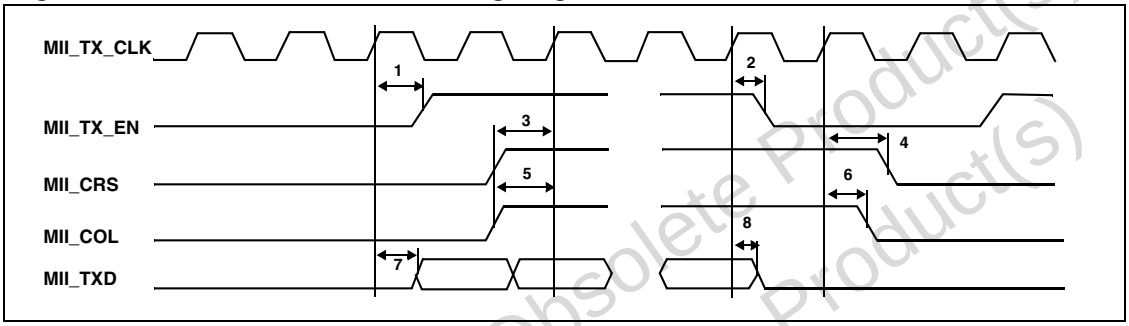


Table 27. Ethernet MII transmit timing table

Symbol	Parameter	Symbol	Value		Unit
			Min	Max	
1	MII_TX_CLK high to MII_TX_EN valid	t_{VAL}(MII_TX_EN)		4.20	ns
2	MII_TX_CLK high to MII_TX_EN invalid	T_{inval}(MII_TX_EN)		4.86	ns
3	MII_CRS valid to MII_TX_CLK high	T_{su}(MII_CRS)	0.61		ns
4	MII_TX_CLK high to MII_CRS invalid	T_h(MII_CRS)	0.00		ns
5	MII_COL valid to MII_TX_CLK high	T_{su}(MII_COL)	0.81		ns
6	MII_TX_CLK high to MII_COL invalid	T_h(MII_COL)	0.00		ns
7	MII_TX_CLK high to MII_TXD valid	t_{VAL}(MII_TXD)		5.02	ns
8	MII_TXCLK high to MII_TXD invalid	T_{inval}(MII_TXD		5.02	ns

Ethernet MII Receive timings

Figure 20. Ethernet MII receive timing diagram

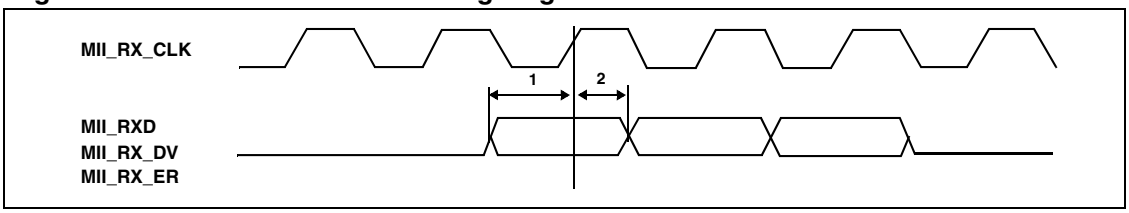


Figure 21. Ethernet MII receive timing table

Symbol	Parameter	Symbol	Value		Unit
			Min	Max	
1	MII_RXD valid to MII_RX_CLK high	T_{su}(MII_RXD)	0.81		ns
2	MII_RX_CLK high to MII_RXD invalid	T_h(MII_RXD)	0.00		ns

6.14.2 USB electrical interface characteristics

USB 2.0 Compliant in Full Speed Mode

6.14.3 CAN interface electrical characteristics

Conforms to CAN 2.0B protocol specification

6.14.5 SPI electrical characteristics

V_{DDQ} = 2.7 - 3.6V, V_{DD} = 1.65 - 2V, T_A = -40 / 85 °C unless otherwise specified.

Table 29. SPI electrical characteristics

Symbol	Parameter	Test Conditions	Value		Unit
			Typ	Max	
f_{SCLK} $1/t_{c(SCLK)}$	SPI clock frequency	Master		24	MHz
		Slave		4	
$t_{r(SCLK)}$ $t_{f(SCLK)}$	SPI clock rise and fall times	50pF load	0.1		V/ns
$t_{su(SS)}$	SS setup time	Slave	1		t_{PCLK}
$t_{h(SS)}$	SS hold time	Slave	1		
$t_{w(SCLKH)}$ $t_{w(SCLKL)}$	SCLK high and low time	Master Slave	1		
$t_{su(MI)}$ $t_{su(SI)}$	Data input setup time	Master Slave	5		
$t_{h(MI)}$ $t_{h(SI)}$	Data input hold time	Master Slave	6		
$t_{a(SO)}$	Data output access time	Slave		6	
$t_{dis(SO)}$	Data output disable time	Slave		6	
$t_{v(SO)}$	Data output valid time	Slave (after enable edge)		6	
$t_{h(SO)}$	Data output hold time		0		
$t_{v(MO)}$	Data output valid time	Master (before capture edge)	0.25		
$t_{h(MO)}$	Data output hold time		0.25		

Figure 22. SPI slave timing diagram with CPHA=0

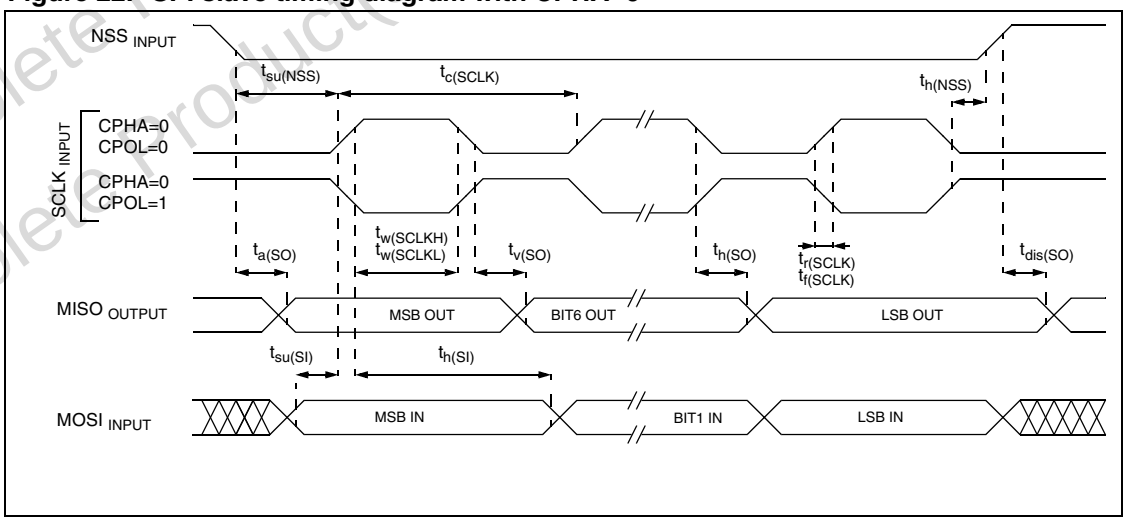


Table 32. Ordering information scheme

Example: STR9 1 2 F W 4 4 X 6 T

Family

ARM9 Microcontroller Family

Series

1 = STR9 Series 1

Feature set

0 = CAN, UART, IrDA, I2C, SSP

1 = USB, CAN, UART, IrDA, I2C, SSP

2 = USB, CAN, UART, IrDA, I2C, SSP, ETHERNET

Memory type

F = Flash

No. of pins

M = 80

W = 128

SRAM size

3 = 64K

4 = 96K

Program Memory Size

2 = 256K

4 = 512K

Package

X = plastic LQFP

Temperature Range

6 = -40 to 85°C

Shipping Option

T = Tape & Reel Packing

For a list of available options (e.g. speed, package) or for further information on any aspect of this device, please contact the ST Sales Office nearest to you.

9 Revision history

Date	Revision	Changes
12-Apr-2006	1	Initial release
28-June-2006	2	Added LFBGA144 package Updated electrical characteristics section
04-Sep-2006	3	Changed number of GPIOs in 80 pin packge to 40 Changed EMI_RDYn pin name to EMI_WAITn Added RTC clock to description of JRTCK in *Table 3* UART max baud rate changed to 1.5 Mbps in *Section 2.20 on page 26* Modified *Figure 2: Clock control on page 15*
01-Feb-2007	4	Removed LFBGA144 package, (transferred to separate STR91xFA datasheet).