# STR91xF

## ARM966E-S™ 16/32-Bit Flash MCU with Ethernet, USB, CAN, AC motor control, 4 timers, ADC, RTC, DMA

- **16/32-bit 96 MHz ARM9E based MCU**
  - ARM966E-S RISC core: Harvard architecture, 5-stage pipeline, Tightly-Coupled Memories (SRAM and Flash)
  - STR91xF implementation of core adds high-speed burst Flash memory interface, instruction prefetch queue, branch cache
  - Up to 96 MIPS directly from Flash memory
  - Single-cycle DSP instructions are supported
  - Binary compatible with 16/32-bit ARM7 code
- **Dual burst Flash memories, 32-bits wide**
  - 256KB/512KB Main Flash, 32KB 2nd Flash
  - Sequential Burst operation up to 96 MHz
  - 100K min erase cycles, 20 yr min retention
- **SRAM, 32-bits wide**
  - 64K or 96K bytes, optional battery backup
- **9 programmable DMA channels**
  - One for Ethernet, eight programmable channels
- **Clock, reset, and supply management**
  - Two supplies required. Core: 1.8 V +/-10%, I/O: 2.7 to 3.6 V
  - Internal oscillator operating with external 4-25 MHz crystal
  - Internal PLL up to 96MHz
  - Real-time clock provides calendar functions, tamper detection, and wake-up functions
  - Reset Supervisor monitors voltage supplies, watchdog timer, wake-up unit, ext. reset
  - Brown-out monitor for early warning interrupt
  - Run, Idle, and Sleep Mode as low as 50 uA
- **Operating temperature -40 to +85°C**
- **Vectored interrupt controller (VIC)**
  - 32 IRQ vectors, 30 intr pins, any can be FIQ
  - Branch cache minimizes interrupt latency
- **8-channel, 10-bit A/D converter (ADC)**
  - 0 to 3.6V range, 0.7 usec conversion



LQFP80 12 x12mm



LQFP128 14 x 14mm

- **11 Communication interfaces**
  - 10/100 Ethernet MAC with DMA and MII port
  - USB Full-speed (12 Mbps) slave device
  - CAN interface (2.0B Active)
  - 3 16550-style UARTs with IrDA protocol
  - 2 Fast $I^2C$™, 400 kHz
  - 2 channels for SPI™, SSI™, or Microwire™
  - 8/16 bit EMI bus on 128 packages
- **Up to 80 I/O pins (muxed with interfaces)**
  - 5 V tolerant, 16 have high sink current (8 mA)
  - Bit-wise manipulation of pins within a port
- **16-bit standard timers (TIM)**
  - 4 timers each with 2 input capture, 2 output compare, PWM and pulse count modes
- **3-Phase induction motor controller (IMC)**
  - 3 pairs of PWM outputs, adjustable centers
  - Emergency stop, dead-time gen, tach input
- **JTAG interface with boundary scan**
  - ARM EmbeddedICE® RT for debugging
  - In-System Programming (ISP) of Flash
- **Embedded trace module (ARM ETM9)**
  - Hi-speed instruction tracing, 9-pin interface

Obsolete Product(s) - Obsolete Product(s)

# Contents


1 **Introduction** . . . . . 6

2 **Functional overview** . . . . . 7

2.1 System-in-a-Package (SiP) . . . . . 7

2.2 Package choice . . . . . 7

2.3 ARM966E-S CPU core . . . . . 7

2.4 Burst Flash memory interface . . . . . 7

2.4.1 Pre-Fetch Queue (PFQ) . . . . . 7

2.4.2 Branch Cache (BC) . . . . . 8

2.4.3 Management of literals . . . . . 8

2.5 SRAM (64K or 96K Bytes) . . . . . 10

2.5.1 Arbitration . . . . . 10

2.5.2 Battery backup . . . . . 10

2.6 DMA data movement . . . . . 10

2.7 Non-volatile memories . . . . . 10

2.7.1 Primary Flash memory . . . . . 11

2.7.2 Secondary Flash memory . . . . . 11

2.8 One-time-programmable (OTP) memory . . . . . 11

2.9 Vectored interrupt controller (VIC) . . . . . 11

2.9.1 FIQ handling . . . . . 12

2.9.2 IRQ handling . . . . . 12

2.9.3 Interrupt sources . . . . . 12

2.10 Clock control unit (CCU) . . . . . 14

2.10.1 Master clock sources . . . . . 14

2.10.2 Reference clock (RCLK) . . . . . 15

2.10.3 AHB clock (HCLK) . . . . . 15

2.10.4 APB clock (PCLK) . . . . . 15

2.10.5 Flash memory interface clock (FMICLK) . . . . . 15

2.10.6 Baud rate clock (BRCLK) . . . . . 16

2.10.7 External memory interface bus clock (BCLK) . . . . . 16

2.10.8 USB interface clock . . . . . 16

2.10.9 Ethernet MAC clock . . . . . 16

2.10.10 External RTC calibration clock . . . . . 16


Obsolete Product(s) - Obsolete Product(s)

2.10.11 Operation example . . . . . 16
2.11 Flexible power management . . . . . 17
2.11.1 Run mode . . . . . 17
2.11.2 Idle mode . . . . . 17
2.11.3 Sleep mode . . . . . 17
2.12 Voltage supplies . . . . . 18
2.12.1 Independent A/D converter supply and reference voltage . . . . . 18
2.12.2 Battery supply . . . . . 18
2.13 System supervisor . . . . . 19
2.13.1 Supply voltage brownout . . . . . 19
2.13.2 Supply voltage dropout . . . . . 19
2.13.3 Watchdog timer . . . . . 19
2.13.4 External RESET_INn pin . . . . . 20
2.13.5 Power-up . . . . . 20
2.13.6 JTAG debug command . . . . . 20
2.13.7 Tamper detection . . . . . 20
2.14 Real-time clock (RTC) . . . . . 20
2.15 JTAG interface . . . . . 21
2.15.1 In-system-programming . . . . . 22
2.15.2 Boundary scan . . . . . 22
2.15.3 CPU debug . . . . . 22
2.15.4 JTAG security bit . . . . . 23
2.16 Embedded trace module (ARM ETM9, v. r2p2) . . . . . 23
2.17 Ethernet MAC interface with DMA . . . . . 24
2.18 USB 2.0 slave device interface with DMA . . . . . 24
2.18.1 Packet buffer interface (PBI) . . . . . 25
2.18.2 DMA . . . . . 25
2.18.3 Suspend mode . . . . . 25
2.19 CAN 2.0B interface . . . . . 25
2.20 UART interfaces with DMA . . . . . 26
2.20.1 DMA . . . . . 26
2.21 I2C interfaces with DMA . . . . . 27
2.21.1 DMA . . . . . 27
2.22 SSP interfaces (SPI, SSI, and Microwire) with DMA . . . . . 27
2.22.1 DMA . . . . . 28

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

2.23 General purpose I/O . . . . . 28
2.24 A/D converter (ADC) . . . . . 29
2.25 Standard timers (TIM) with DMA . . . . . 29
2.25.1 DMA . . . . . 30
2.26 Three-phase induction motor controller (IMC) . . . . . 30
2.27 External memory interface (EMI) . . . . . 30

**3 Related documentation . . . . . 33**

**4 Pin description . . . . . 34**
4.1 Default pin functions . . . . . 36

**5 Memory mapping . . . . . 42**
5.1 Buffered and non-buffered writes . . . . . 42
5.2 System (AHB) and peripheral (APB) buses . . . . . 42
5.3 SRAM . . . . . 42
5.4 Two independent Flash memories . . . . . 43
5.4.1 Default configuration . . . . . 43
5.4.2 Optional configuration . . . . . 43

**6 Electrical characteristics . . . . . 46**
6.1 Absolute maximum ratings . . . . . 46
6.2 Operating conditions . . . . . 47
6.3 LVD electrical characteristics . . . . . 47
6.4 DC electrical characteristics . . . . . 48
6.5 AC electrical characteristics . . . . . 49
6.6 RESET_INn and power-on-reset characteristics . . . . . 50
6.7 Main oscillator electrical characteristics . . . . . 51
6.8 RTC oscillator electrical characteristics . . . . . 51
6.9 PLL electrical characteristics . . . . . 52
6.10 Flash memory characteristics . . . . . 53
6.11 EMC characteristics . . . . . 54
6.11.1 Functional EMS (Electro Magnetic Susceptibility) . . . . . 54
6.11.2 Electro Magnetic Interference (EMI) . . . . . 54
6.11.3 Absolute Maximum Ratings (Electrical Sensitivity) . . . . . 55

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

6.11.4 Electro-Static Discharge (ESD) . . . . . 55
6.11.5 Static and Dynamic Latch-Up . . . . . 55
6.11.6 Designing hardened software to avoid noise problems . . . . . 55
6.11.7 Electrical Sensitivities . . . . . 56
6.12 External memory bus timings . . . . . 56
6.13 ADC electrical characteristics . . . . . 60
6.14 Communication interface electrical characteristics . . . . . 62
6.14.1 10/100 Ethernet MAC electrical characteristics . . . . . 62
6.14.2 USB electrical interface characteristics . . . . . 64
6.14.3 CAN interface electrical characteristics . . . . . 64
6.14.4 I2C electrical characteristics . . . . . 65
6.14.5 SPI electrical characteristics . . . . . 66

**7 Package mechanical data . . . . . 68**
7.1 Thermal characteristics . . . . . 69

**8 Ordering information . . . . . 70**

**9 Revision history . . . . . 72**

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

# 1 Introduction

STR91xF is a series of ARM-powered microcontrollers which combines a 16/32-bit ARM966E-S RISC processor core, dual-bank Flash memory, large SRAM for data or code, and a rich peripheral set to form an ideal embedded controller for a wide variety of applications such as point-of-sale terminals, industrial automation, security and surveillance, vending machines, communication gateways, serial protocol conversion, and medical equipment. The ARM966E-S core can perform single-cycle DSP instructions, good for speech processing, audio algorithms, and low-end imaging.

This datasheet provides STR91xF ordering information, functional overview, mechanical information, and electrical device characteristics.

For complete information on STR91xF memory, registers, and peripherals, please refer to the STR91xF Reference Manual.

For information on programming the STR91xF Flash memory please refer to the STR9 Flash Programming Reference Manual

For information on the ARM966E-S core, please refer to the ARM966E-S Rev. 2 Technical Reference Manual.

**Table 1. Device summary**

| Features | STR910F | | STR911F | | STR912F | |
|---|---|---|---|---|---|---|
| | M32X6 | W32X6 | M42X6 | M44X6 | W42X6 | W44X6 |
| Flash Kbytes | 256+32 | | 256+32 | 512+32 | 256+32 | 512+32 |
| RAM - Kbytes | 64 | | 96 | | 96 | |
| Peripheral functions | CAN, 40 I/Os | CAN, EMI, 80 I/Os | USB, CAN, 40 I/Os | | Ethernet, USB, CAN, EMI, 80 I/Os | |
| Packages | LQFP80 | LQFP128 | LQFP80 | | LQFP128 | |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

# 2 Functional overview

## 2.1 System-in-a-Package (SiP)

The STR91xF is a SiP device, comprised of two stacked die. One die is the ARM966E-S CPU with peripheral interfaces and analog functions, and the other die is the burst Flash. The two die are connected to each other by a custom high-speed 32-bit burst memory interface and a serial JTAG test/programming interface.

## 2.2 Package choice

STR91xF devices are available in 128-pin (14 x 14 mm) and 80-pin (12 x 12 mm) LQFP packages. Refer to the *Table 1 on page 6* and to *Table 31 on page 70* for a list of available peripherals for each of the package choices.

## 2.3 ARM966E-S CPU core

The ARM966E-S core inherently has separate instruction and data memory interfaces (Harvard architecture), allowing the CPU to simultaneously fetch an instruction, and read or write a data item through two Tightly-Coupled Memory (TCM) interfaces as shown in *Figure 1*. The result is streamlined CPU Load and Store operations and a significant reduction in cycle count per instruction. In addition to this, a 5-stage pipeline is used to increase the amount of operational parallelism, giving the most performance out of each clock cycle.

Ten DSP-enhanced instruction extensions are supported by this core, including single-cycle execution of 32x16 Multiply-Accumulate, saturating addition/subtraction, and count leading-zeros.

The ARM966E-S core is binary compatible with 32-bit ARM7 code and 16-bit Thumb® code.

## 2.4 Burst Flash memory interface

A Burst Flash memory interface (*Figure 1*) has been integrated into the Instruction TCM (I-TCM) path of the ARM966E-S core. Also in this path is a 4-instruction Pre-Fetch Queue (PFQ) and a 4-entry Branch Cache (BC), enabling the ARM966E-S core to perform up to 96 MIPS while executing code directly from Flash memory. This architecture provides high performance levels without a costly instruction SRAM, instruction cache, or external SDRAM. Eliminating the instruction cache also means interrupt latency is reduced and code execution becomes more deterministic.

### 2.4.1 Pre-Fetch Queue (PFQ)

As the CPU core accesses sequential instructions through the I-TCM, the PFQ always looks ahead and will pre-fetch instructions, taking advantage any idle bus cycles due to variable length instructions. The PFQ will fetch 32-bits at a time from the Burst Flash memory at a rate of up to 96 MHz.

Obsolete Product(s) - Obsolete Product(s)

### 2.4.2 Branch Cache (BC)

When instruction addresses are not sequential, such as a program branch situation, the PFQ would have to flush and reload which would cause the CPU to stall if no BC were present. Before reloading, the PFQ checks the BC to see if it contains the desired target branch address. The BC contains up to four of the most recently taken branch addresses and the first four instructions associated with each of these branches. This check is extremely fast, checking all four BC entries simultaneously for a branch address match (cache hit). If there is a hit, the BC rapidly supplies the instruction and reduces the CPU stall. This gives the PFQ time to start pre-fetching again while the CPU consumes these four instructions from the BC. The advantage here is that program loops (very common with embedded control applications) run very fast if the address of the loops are contained in the BC.

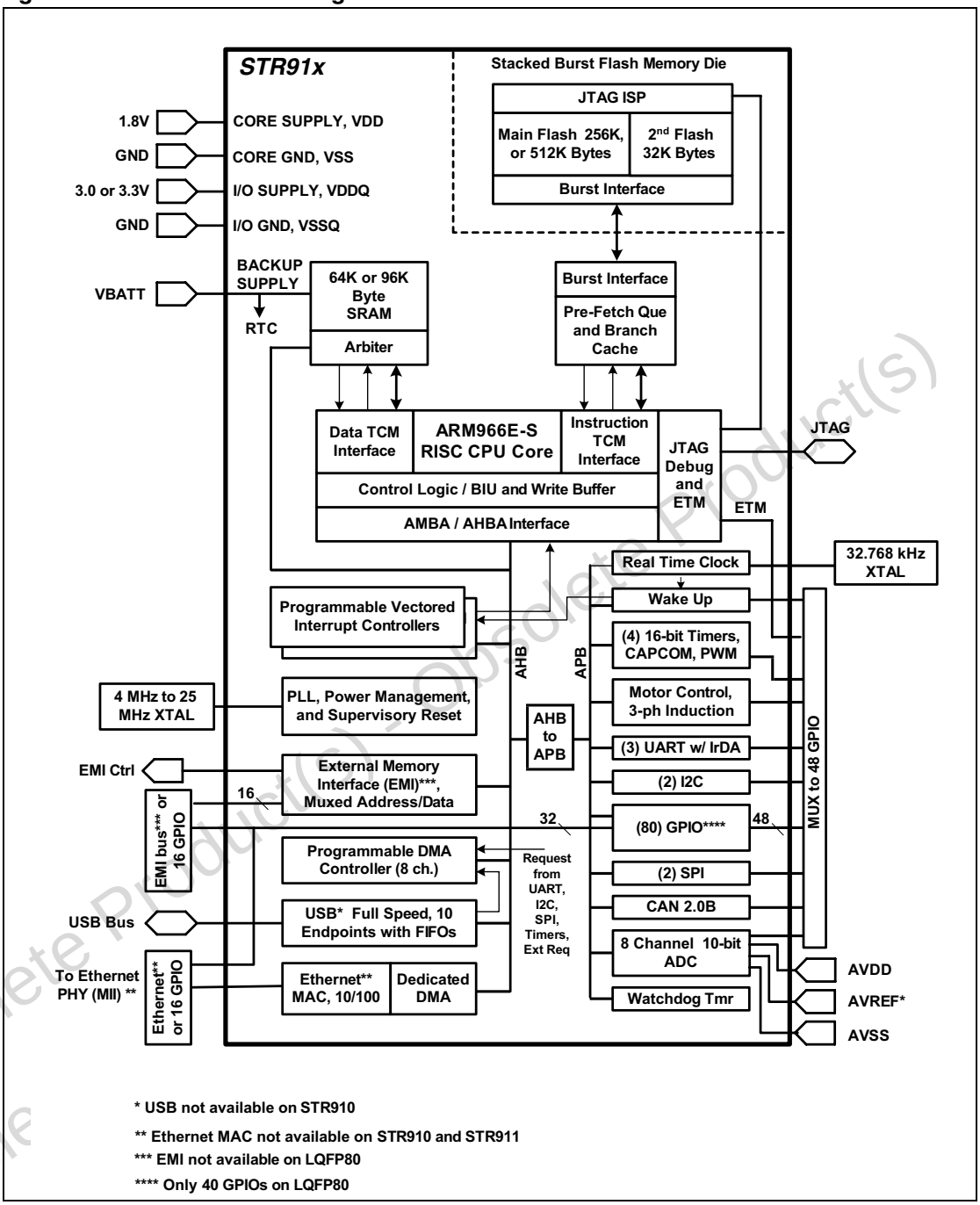
In addition, there is a 5th branch cache entry that is dedicated to the Vectored Interrupt Controller (VIC) to further reduce interrupt latency by eliminating the stall latency typically imposed by fetching the instruction that reads the interrupt vector address from the VIC.

### 2.4.3 Management of literals

Typical ARM architecture and compilers do not place literals (data constants) sequentially in Flash memory with the instructions that use them, but instead the literals are placed at some other address which looks like a program branch from the PFQ's point of view. The STR91xF implementation of the ARM966E-S core has special circuitry to prevent flushing the PFQ when literals are encountered in program flow to keep performance at a maximum.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 1. STR91xF block diagram**




* USB not available on STR910

** Ethernet MAC not available on STR910 and STR911

*** EMI not available on LQFP80

**** Only 40 GPIOs on LQFP80

Obsolete Product(s) - Obsolete Product(s)

## 2.5 SRAM (64K or 96K Bytes)

A 32-bit wide SRAM resides on the CPU's Data TCM (D-TCM) interface, providing single-cycle data accesses. As shown in *Figure 1*, the D-TCM shares SRAM access with the Advanced High-performance Bus (AHB). Sharing is controlled by simple arbitration logic to allow the DMA unit on the AHB to also access to the SRAM.

### 2.5.1 Arbitration

Zero-wait state access occurs for either the D-TCM or the AHB when only one of the two is requesting SRAM. When both request SRAM simultaneously, access is granted on an interleaved basis so neither requestor is starved, granting one 32-bit word transfer to each requestor before relinquishing SRAM to the other. When neither the D-TCM or the AHB are requesting SRAM, the arbiter leaves access granted to the most recent user (if D-TCM was last to use SRAM then the D-TCM will not have to arbitrate to get access next time).

The CPU may execute code from SRAM through the AHB. There are no wait states as long as the D-TCM is not contending for SRAM access and the AHB is not sharing bandwidth with peripheral traffic. The ARM966E-S CPU core has a small pre-fetch queue built into this instruction path through the AHB to look ahead and fetch instructions during idle bus cycles.

### 2.5.2 Battery backup

When a battery is connected to the designated battery backup pin (VBATT), SRAM contents are automatically preserved when the normal operating voltage on VDD pins is lost or sags below threshold. Automatic switchover to SRAM can be disabled by firmware if it is desired that the battery will power only the RTC and not the SRAM during standby.

## 2.6 DMA data movement

DMA channels on the Advanced High-performance Bus (AHB) take full advantage of the separate data path provided by the Harvard architecture, moving data rapidly and largely independent of the instruction path. There are two DMA units, one is dedicated to move data between the Ethernet interface and SRAM, the other DMA unit has eight programmable channels with 16 request signals to service other peripherals and interfaces (USB, SSP, I2C, UART, Timers, EMI, and external request pins). Both single word and burst DMA transfers are supported. Memory-to-memory transfers are supported in addition to memory-peripheral transfers. DMA access to SRAM is shared with D-TCM accesses, and arbitration is described in *Section 2.5.1*. Efficient DMA transfers are managed by firmware using linked list descriptor tables. Of the 16 DMA request signals, two are assigned to external inputs. The DMA unit can move data between external devices and resources inside the STR91xF through the EMI bus.

## 2.7 Non-volatile memories

There are two independent 32-bit wide Burst Flash memories enabling true read-while-write operation. The Flash memories are single-voltage erase/program with 20 year minimum data retention and 100K minimum erase cycles. The primary Flash memory is much larger than the secondary Flash.

Both Flash memories are blank when devices are shipped from ST. The CPU can boot only from Flash memory (configurable selection of which Flash bank).

Obsolete Product(s) - Obsolete Product(s)

Flash memories are programmed half-word (16 bits) at a time, but are erased by sector or by full array.

### 2.7.1 Primary Flash memory

Using the STR91xF device configuration software tool and 3rd party Integrated Developer Environments, it is possible to specify that the primary Flash memory is the default memory from which the CPU boots at reset, or otherwise specify that the secondary Flash memory is the default boot memory. This choice of boot memory is non-volatile and stored in a location that can be programmed and changed only by JTAG In-System Programming. See *Section 5: Memory mapping*, for more detail.

The primary Flash memory has equal length 64K byte sectors. Devices with 256 Kbytes of primary Flash have four sectors and 512K devices have eight sectors.

### 2.7.2 Secondary Flash memory

The smaller of the two Flash memories can be used to implement a bootloader, capable of storing code to perform robust In-Application Programming (IAP) of the primary Flash memory. The CPU executes code from the secondary Flash, while updating code in the primary Flash memory. New code for the primary Flash memory can be downloaded over any of the interfaces on the STR91xF (USB, Ethernet, CAN, UART, etc.)

Additionally, the Secondary Flash memory may also be used to store small data sets by emulating EEPROM though firmware, eliminating the need for external EEPROM memories. This raises the data security level because passcodes and other sensitive information can be securely locked inside the STR91xF device.

The secondary Flash memory is 32 Kbytes and has four equal length sectors of 8 Kbytes each.

Both the primary Flash memory and the secondary Flash memory can be programmed with code and/or data using the JTAG In-System Programming (ISP) channel, totally independent of the CPU. This is excellent for iterative code development and for manufacturing.

## 2.8 One-time-programmable (OTP) memory

There are 32 bytes of OTP memory ideally suited for serial numbers, security keys, factory calibration constants, or other permanent data constants. These OTP data bytes can be programmed only one time through either the JTAG interface or by the CPU, and these bytes can never be altered afterwards. As an option, a “lock bit” can be set by the JTAG interface or the CPU which will block any further writing to the this OTP area. The “lock bit” itself is also OTP. If the OTP array is unlocked, it is always possible to go back and write to an OTP byte location that has not been previously written, but it is never possible to change an OTP byte location if any one bit of that particular byte has been written before. The last two OTP bytes are reserved for the STR91xF product ID and revision level. Byte 30 contains the device revision level. For STR91xF devices, the revision is 0x13.

## 2.9 Vectored interrupt controller (VIC)

Interrupt management in the STR91xF is implemented from daisy-chaining two standard ARM VIC units. This combined VIC has 32 prioritized interrupt request channels and generates two

Obsolete Product(s) - Obsolete product(s)

interrupt output signals to the CPU. The output signals are FIQ and IRQ, with FIQ having higher priority.

### 2.9.1 FIQ handling

FIQ (Fast Interrupt reQuest) is the only non-vectored interrupt and the CPU can execute an Interrupt Service Routine (ISR) directly without having to determine/prioritize the interrupt source, minimizing ISR latency. Typically only one interrupt source is assigned to FIQ. An FIQ interrupt has its own set of banked registers to minimize the time to make a context switch. Any of the 32 interrupt request input signals coming into the VIC can be assigned to FIQ.

### 2.9.2 IRQ handling

IRQ is a vectored interrupt and is the logical OR of all 32 interrupt request signals coming into the 32 IRQ channels. Priority of individual vectored interrupt requests is determined by hardware (IRQ channel Intr 0 is highest priority, IRQ channel Intr 31 is lowest). However, CPU firmware may re-assign individual interrupt sources to individual hardware IRQ channels, meaning that firmware can effectively change interrupt priority levels as needed.

When the IRQ signal is activated by an interrupt request, VIC hardware will resolve the IRQ interrupt priority, then the ISR reads the VIC to determine both the interrupt source and the vector address to jump to the service code.

The STR91xF has a feature to reduce ISR response time for IRQ interrupts. Typically, it requires two memory accesses to read the interrupt vector address from the VIC, but the STR91xF reduces this to a single access by adding a 5th entry in the instruction branch cache, dedicated for interrupts. This 5th cache entry always holds the instruction that reads the interrupt vector address from the VIC, eliminating one of the memory accesses typically required in traditional ARM implementations.

### 2.9.3 Interrupt sources

The 32 interrupt request signals coming into the VIC on 32 IRQ channels are from various sources; 5 from a wake-up unit and the remaining 27 come from internal sources on the STR91xF such as on-chip peripherals, see *Table 2*. Optionally, firmware may force an interrupt on any IRQ channel.

One of the 5 interrupt requests generated by the wake-up unit (IRQ25 in *Table 2*) is derived from the logical OR of all 32 inputs to the wake-up unit. Any of these 32 inputs may be used to wake up the CPU and cause an interrupt. These 32 inputs consist of 30 external interrupts on selected and enabled GPIO pins, plus the RTC interrupt, and the USB Resume interrupt.

Each of 4 remaining interrupt requests generated by the wake-up unit (IRQ26 in *Table 2*) are derived from groupings of 8 interrupt sources. One group is from GPIO pins P3.2 to P3.7 plus the RTC interrupt and the USB Resume interrupt; the next group is from pins P5.0 to P5.7; the next group is from pins P6.0 to P6.7; and last the group is from pins P7.0 to P7.7. This allows individual pins to be assigned directly to vectored IRQ interrupts or one pin assigned directly to the non-vectored FIQ interrupt.

Obsolete Product(s) - Obsolete Product(s)

See *Table 2* for recommended interrupt source assignments to physical IRQ interrupt channels. Interrupt source assignments are made by CPU firmware during initialization, thus establishing interrupt priorities.

**Table 2. Recommended IRQ Channel assignments (set by CPU firmware)**

| VIC IRQ Channel | Logic Block | Interrupt Source |
|---|---|---|
| 0 (high priority) | WatchDog | Timeout in WDT mode, Terminal Count in Counter Mode |
| 1 | CPU Firmware | Firmware generated interrupt |
| 2 | CPU Core | Debug Receive Command |
| 3 | CPU Core | Debug Transmit Command |
| 4 | TIM Timer 0 | Logic OR of ICI0_0, ICI0_1, OCI0_0, OCI0_1, Timer overflow |
| 5 | TIM Timer 1 | Logic OR of ICI1_0, ICI1_1, OCI1_0, OCI1_1, Timer overflow |
| 6 | TIM Timer 2 | Logic OR of ICI2_0, ICI2_1, OCI2_0, OCI2_1, Timer overflow |
| 7 | TIM Timer 3 | Logic OR of ICI3_0, ICI3_1, OCI3_0, OCI3_1, Timer overflow |
| 8 | USB | Logic OR of high priority USB interrupts |
| 9 | USB | Logic OR of low priority USB interrupts |
| 10 | CCU | Logic OR of all interrupts from Clock Control Unit |
| 11 | Ethernet MAC | Logic OR of Ethernet MAC interrupts via its own dedicated DMA channel. |
| 12 | DMA | Logic OR of interrupts from each of the 8 individual DMA channels |
| 13 | CAN | Logic OR of all CAN interface interrupt sources |
| 14 | IMC | Logic OR of 8 Induction Motor Control Unit interrupts |
| 15 | ADC | End of AtoD conversion interrupt |
| 16 | UART0 | Logic OR of 5 interrupts from UART channel 0 |
| 17 | UART1 | Logic OR of 5 interrupts from UART channel 1 |
| 18 | UART2 | Logic OR of 5 interrupts from UART channel 2 |
| 19 | I2C0 | Logic OR of transmit, receive, and error interrupts of I2C channel 0 |
| 20 | I2C1 | Logic OR of transmit, receive, and error interrupts of I2C channel 1 |
| 21 | SSP0 | Logic OR of all interrupts from SSP channel 0 |
| 22 | SSP1 | Logic OR of all interrupts from SSP channel 1 |
| 23 | BROWNOUT | LVD warning interrupt |
| 24 | RTC | Logic OR of Alarm, Tamper, or Periodic Timer interrupts |
| 25 | Wake-Up (all) | Logic OR of all 32 inputs of Wake-Up unit (30 pins, RTC, and USB Resume) |
| 26 | Wake-up Group 0 | Logic OR of 8 interrupt sources: RTC, USB Resume, pins P3.2 to P3.7 |
| 27 | Wake-up Group 1 | Logic OR of 8 interrupts from pins P5.0 to P5.7 |
| 28 | Wake-up Group 2 | Logic OR of 8 interrupts from pins P6.0 to P6.7 |
| 29 | Wake-up Group 3 | Logic OR of 8 interrupts from pins P7.0 to P7.7 |
| 30 | USB | USB Bus Resume Wake-up (also input to wake-up unit) |
| 31 (low priority) | PFQ-BC | Special use of interrupts from Prefetch Queue and Branch Cache |

Obsolete Product(s) - Obsolete Product(s)

## 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

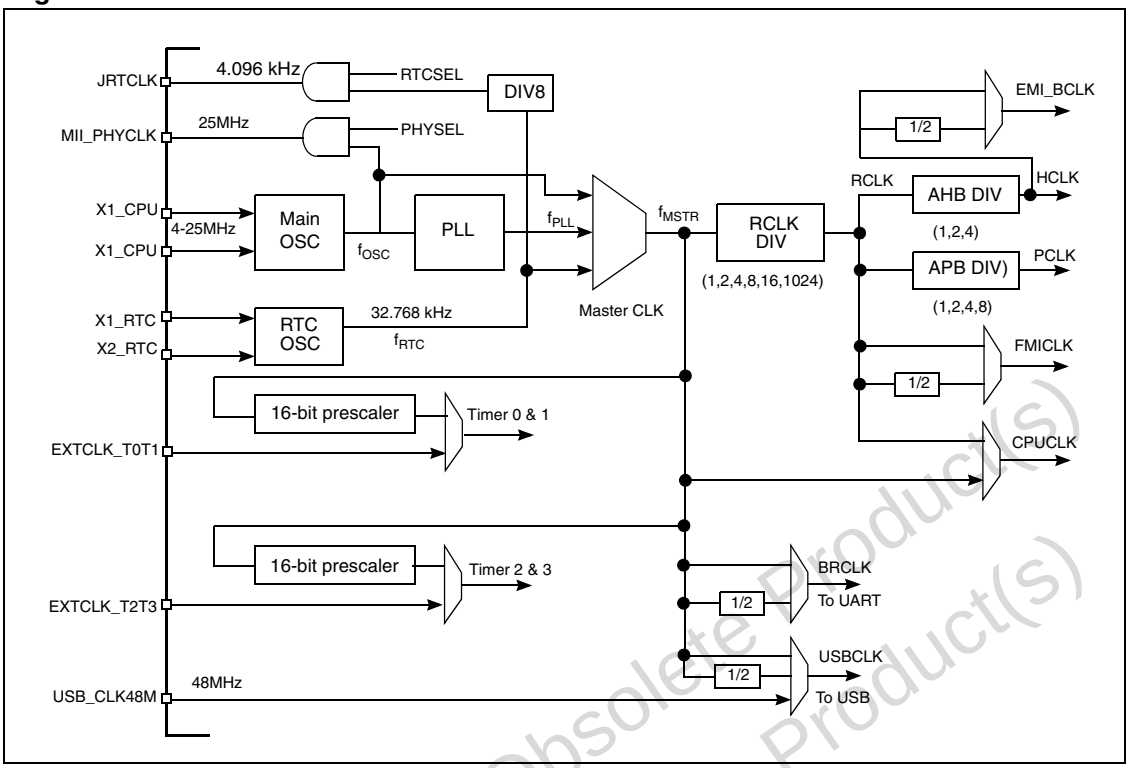
The master clock in the CCU ($f_{MSTR}$) is derived from one of three clock input sources. Under firmware control, the CPU can switch between the three CCU inputs without introducing any glitches on the master clock output. Inputs to the CCU are:

- 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.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 2. Clock control**




## 2.10.2 Reference clock (RCLK)

The main clock ($f_{MSTR}$) can be divided to operate at a slower frequency reference clock (RCLK) for the ARM core and all the peripherals. The RCLK provides the divided clock for the ARM core, and feeds the dividers for the AHB, APB, External Memory Interface, and FMI units.

## 2.10.3 AHB clock (HCLK)

The RCLK can be divided by 1, 2 or 4 to generate the AHB clock. The AHB clock is the bus clock for the AHB bus and all bus transfers are synchronized to this clock. The maximum HCLK frequency is 96 MHz.

## 2.10.4 APB clock (PCLK)

The RCLK can be divided by 1, 2, 4 or 8 to generate the APB clock. The APB clock is the bus clock for the APB bus and all bus transfers are synchronized to this clock. Many of the peripherals that are connected to the AHB bus also use the PCLK as the source for external bus data transfers. The maximum PCLK frequency is 48 MHz.

## 2.10.5 Flash memory interface clock (FMICLK)

The FMICLK clock is an internal clock derived from RCLK, defaulting to RCLK frequency at power up. The clock can be optionally divided by 2. The FMICLK determines the bus bandwidth between the ARM core and the Flash memory. Typically, codes in the Flash memory can be fetched one word per FMICLK clock in burst mode. The maximum FMICLK frequency is 96MHz.

Obsolete Product(s) - Obsolete Product(s)

### 2.10.6 Baud rate clock (BRCLK)

The baud rate clock is an internal clock derived from $f_{MSTR}$ that is used by the three on-chip UART peripherals for baudrate generation. The frequency can be optionally divided by 2.

### 2.10.7 External memory interface bus clock (BCLK)

The BCLK is an internal clock that controls the EMI bus. All EMI bus signals are synchronized to the BCLK. The BCLK is derived from the HCLK and the frequency can be configured to be the same or half that of the HCLK. Refer to *Table 9 on page 50* for the maximum BCLK frequency ($f_{BCLK}$).

### 2.10.8 USB interface clock

Special consideration regarding the USB interface: The clock to the USB interface must operate at 48 MHz and comes from one of three sources, selected under firmware control:

- CCU master clock output of 48 MHz.
- CCU master clock output of 96 MHz. An optional divided-by-two circuit is available to produce 48 MHz for the USB while the CPU system runs at 96MHz.
- STR91xF pin P2.7. An external 48 MHz oscillator connected to pin P2.7 can directly source the USB while the CCU master clock can run at some frequency other than 48 or 96 MHz.

### 2.10.9 Ethernet MAC clock

Special consideration regarding the Ethernet MAC: The external Ethernet PHY interface device requires it's own 25 MHz clock source. This clock can come from one of two sources:

- A 25 MHz clock signal coming from a dedicated output pin (P5.2) of the STR91xF. In this case, the STR91xF must use a 25 MHz signal on its main oscillator input in order to pass this 25 MHz clock back out to the PHY device through pin P5.2. The advantage here is that an inexpensive 25 MHz crystal may be used to source a clock to both the STR91xF and the external PHY device.
- An external 25 MHz oscillator connected directly to the external PHY interface device. In this case, the STR91xF can operate independent of 25 MHz.

### 2.10.10 External RTC calibration clock

The RTC_CLK ($f_{RTC/8}$) can be enabled as an output on the JRTCK pin. The RTC_CLK is used for RTC oscillator calibration. The RTC_CLK is active in Sleep mode and can be used as a system wake up control clock.

### 2.10.11 Operation example

As an example of CCU operation, a 25 MHz crystal can be connected to the main oscillator input on pins X1_CPU and X2_CPU, a 32.768 kHz crystal connected to pins X1_RTC and X2_RTC, and the clock input of an external Ethernet PHY device is connected to STR91xF output pin P5.2. In this case, the CCU can run the CPU at 96 MHz from PLL, the USB interface at 48 MHz, and the Ethernet interface at 25 MHz. The RTC is always running in the background at 32.768 kHz, and the CPU can go to very low power mode dynamically by running from 32.768 kHz and shutting off peripheral clocks and the PLL as needed.

Obsolete Product(s) - Obsolete Product(s)

## 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:

Obsolete Product(s) Obsolete Product(s) Obsolete Product(s)

- Some resets (external reset pin, low-voltage, power-up, JTAG debug command)
- RTC alarm
- Input from wake-up unit

## 2.12 Voltage supplies

The STR91xF requires two separate operating voltage supplies. The CPU and memories operate from a 1.65V to 2.0V on the VDD pins, and the I/O ring operates at 2.7V to 3.6V on the VDDQ pins.

### 2.12.1 Independent A/D converter supply and reference voltage

The ADC unit on 128-pin packages has an isolated analog voltage supply input at pin AVDD to accept a very clean voltage source, independent of the digital voltage supplies. The analog voltage supply range on pin AVDD is the same range as the digital voltage supply on pin VDDQ. Additionally, an isolated analog supply ground connection is provided on pin AVSS only on 128-pin packages for further ADC supply isolation. On 80-pin packages, the analog voltage supply is shared with the ADC reference voltage pin (as described next), and the analog ground is shared with the digital ground at a single point in the STR91xF device on pin AVSS_VSSQ.

A separate external analog reference voltage input for the ADC unit is available on 128-pin packages at the AVREF pin for better accuracy on low voltage inputs, and the voltage on AVREF can range from 1.0V to $V_{DDQ}$. For 80-pin packages, the ADC reference voltage is tied internally to the ADC unit supply voltage at pin AVCC_AVREF, meaning the ADC reference voltage is fixed to the ADC unit supply voltage.

### 2.12.2 Battery supply

An optional stand-by voltage from a battery or other source may be connected to pin VBATT to retain the contents of SRAM in the event of a loss of the $V_{DD}$ supply. The SRAM will automatically switch its supply from the internal $V_{DD}$ source to the VBATT pin when the voltage of $V_{DD}$ drops below that of VBATT.

The VBATT pin also supplies power to the RTC unit, allowing the RTC to function even when the main digital supplies ($V_{DD}$ and $V_{DDQ}$) are switched off. By configuring the RTC register, it is possible to select whether or not to power from VBATT only the RTC unit, or power the RTC unit and the SRAM when the STR91xF device is powered off.

Obsolete Product(s) - Obsolete Product(s)

## 2.13 System supervisor

The STR91xF monitors several system and environmental inputs and will generate a global reset, a system reset, or an interrupt based on the nature of the input and configurable settings. A global reset clears all functions on the STR91xF, a system reset will clear all but the Clock Control Unit (CCU) settings and the system status register. At any time, firmware may reset individual on-chip peripherals. System supervisor inputs include:

- GR: CPU voltage supply ($V_{DD}$) drop out or brown out
- GR: I/O voltage supply ($V_{DDQ}$) drop out or brown out
- GR: Power-Up condition
- SR: Watchdog timer timeout
- SR: External reset pin (RESET_INn)
- SR: JTAG debug reset command

*Note: GR: means the input causes Global Reset, SR: means the input causes System Reset*

The CPU may read a status register after a reset event to determine if the reset was caused by a watchdog timer timeout or a voltage supply drop out. This status register is cleared only by a power up reset.

### 2.13.1 Supply voltage brownout

Each operating voltage source ($V_{DD}$ and $V_{DDQ}$) is monitored separately by the Low Voltage Detect (LVD) circuitry. The LVD will generate an early warning interrupt to the CPU when voltage sags on either $V_{DD}$ or $V_{DDQ}$ voltage inputs. This is an advantage for battery powered applications because the system can perform an orderly shutdown before the batteries become too weak. The voltage trip point to cause a brown out interrupt is typically 0.25V above the LVD dropout thresholds that cause a reset.

CPU firmware may prevent all brown-out interrupts by writing to interrupt mask registers at run-time.

### 2.13.2 Supply voltage dropout

LVD circuitry will always cause a global reset if the CPU's $V_{DD}$ source drops below it's fixed threshold of 1.4V.

However, the LVD trigger threshold to cause a global reset for the I/O ring's $V_{DDQ}$ source is set to one of two different levels, depending if $V_{DDQ}$ will be operated in the range of 2.7V to 3.3V, or 3.0V to 3.6V. If $V_{DDQ}$ operation is at 2.7V to 3.3V, the LVD dropout trigger threshold is 2.4V. If $V_{DDQ}$ operation is 3.0V and 3.6V, the LVD threshold is 2.7V. The choice of trigger level is made by STR91xF device configuration software from STMicroelectronics or IDE from 3rd parties, and is programmed into the STR91xF device along with other configurable items through the JTAG interface when the Flash memory is programmed.

CPU firmware may prevent some LVD resets if desired by writing a control register at run-time. Firmware may also disable the LVD completely for lowest-power operation when an external LVD device is being used.

### 2.13.3 Watchdog timer

The STR91xF has a 16-bit down-counter (not one of the four TIM timers) that can be used as a watchdog timer or as a general purpose free-running timer/counter. The clock source is the

Obsolete Product(s) - Obsolete Product(s)

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

Obsolete Product(s) - Obsolete Product(s)

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.

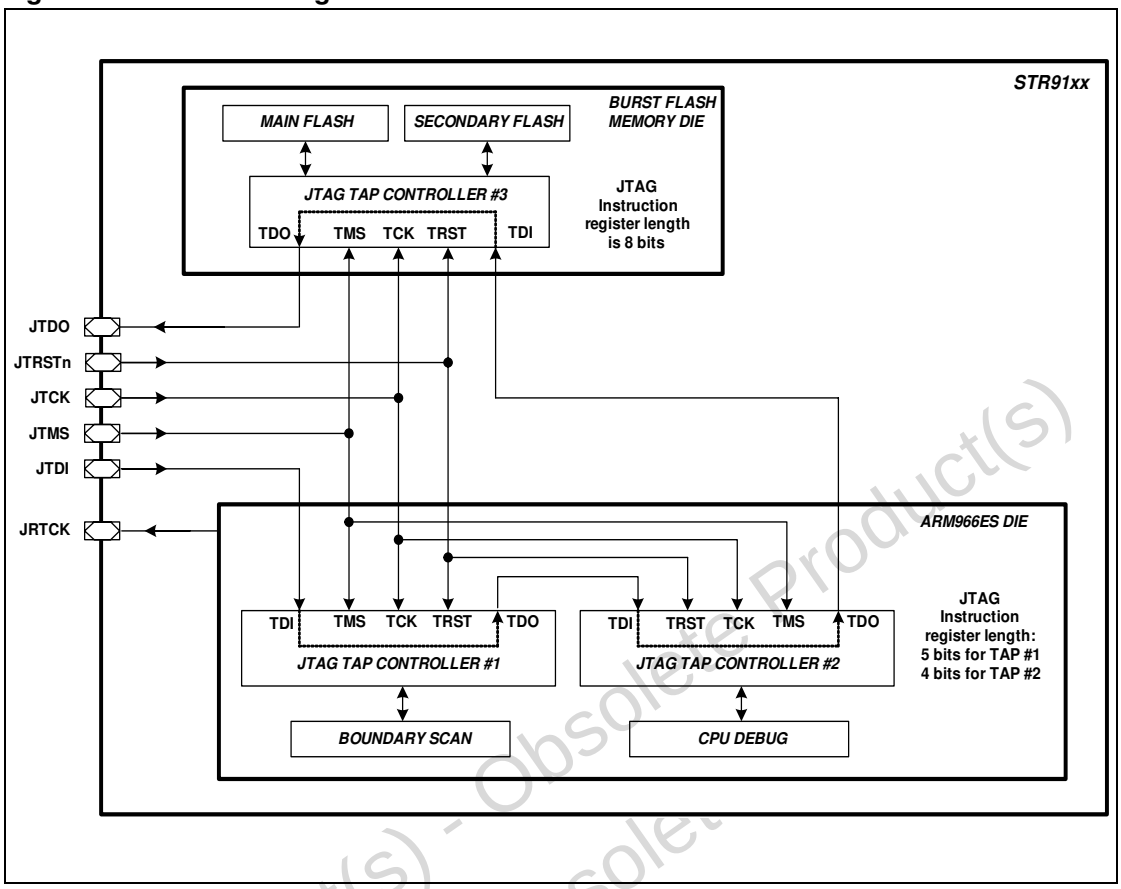
The two die inside the STR91xF (CPU die and Flash memory die) are internally daisy-chained on the JTAG bus, see *Figure 3 on page 22*. The CPU die has two JTAG Test Access Ports (TAPs), one for boundary scan functions and one for ARM CPU debug. The Flash memory die has one TAP for program/erase of non-volatile 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.

Obsolete Product(s) - Obsolete Product(s)

**Figure 3. JTAG chaining inside the STR91xF**




### 2.15.1 In-system-programming

The JTAG interface is used to program or erase all memory areas of the STR91xF device. The pin RESET_INn must be asserted during ISP to prevent the CPU from fetching invalid instructions while the Flash memories are being programmed.

Note that the 32 bytes of OTP memory locations cannot be erased by any means once programmed by JTAG ISP or the CPU.

### 2.15.2 Boundary scan

Standard JTAG boundary scan testing compliant with IEEE-1149.1 is available on the majority of pins of the STR91xF for circuit board test during manufacture of the end product. STR91xF pins that are not serviced by boundary scan are the following:

- JTAG pins JTCK, JTMS, JTDI, JTDO, JTRSTn, JRTCK
- Oscillator input pins X1_CPU, X2_CPU, X1_RTC, X2_RTC
- Tamper detect input pin TAMPER_IN (128-pin packages only)

### 2.15.3 CPU debug

The ARM966E-S CPU core has standard ARM EmbeddedICE-RT logic, allowing the STR91xF to be debugged through the JTAG interface. This provides advanced debugging features making it easier to develop application firmware, operating systems, and the hardware itself.

Obsolete Product(s) - Obsolete Product(s)

Debugging requires that an external host computer, running debug software, is connected to the STR91xF target system via hardware which converts the stream of debug data and commands from the host system's protocol (USB, Ethernet, etc.) to the JTAG EmbeddedICE-RT protocol on the STR91xF. These protocol converters are commercially available and operate with debugging software tools.

The CPU may be forced into a Debug State by a breakpoint (code fetch), a watchpoint (data access), or an external debug request over the JTAG channel, at which time the CPU core and memory system are effectively stopped and isolated from the rest of the system. This is known as Halt Mode and allows the internal state of the CPU core, memory, and peripherals to be examined and manipulated. Typical debug functions are supported such as run, halt, and single-step. The EmbeddedICE-RT logic supports two hardware compare units. Each can be configured to be either a watchpoint or a breakpoint. Breakpoints can also be data-dependent.

Debugging (with some limitations) may also occur through the JTAG interface while the CPU is running full speed, known as Monitor Mode. In this case, a breakpoint or watchpoint will not force a Debug State and halt the CPU, but instead will cause an exception which can be tracked by the external host computer running monitor software. Data can be sent and received over the JTAG channel without affecting normal instruction execution. Time critical code, such as Interrupt Service Routines may be debugged real-time using Monitor Mode.

### 2.15.4 JTAG security bit

This is a non-volatile bit (Flash memory based), which when set will not allow the JTAG debugger or JTAG programmer to read the Flash memory contents.

Using JTAG ISP, this bit is typically programmed during manufacture of the end product to prevent unwanted future access to firmware intellectual property. The JTAG Security Bit can be cleared only by a JTAG "Full Chip Erase" command, making the STR91xF device blank and ready for programming again. The CPU can read the status of the JTAG Security Bit, but it may not change the bit value.

## 2.16 Embedded trace module (ARM ETM9, v. r2p2)

The ETM9 interface provides greater visibility of instruction and data flow happening inside the CPU core by streaming compressed data at a very high rate from the STR91xF though a small number of ETM9 pins to an external Trace Port Analyzer (TPA) device. The TPA is connected to a host computer using USB, Ethernet, or other high-speed channel. Real-time instruction flow and data activity can be recorded and later formatted and displayed on the host computer running debugger software, and this software is typically integrated with the debug software used for EmbeddedICE-RT functions such as single-step, breakpoints, etc. Tracing may be triggered and filtered by many sources, such as instruction address comparators, data watchpoints, context ID comparators, and counters. State sequencing of up to three triggers is also provided. TPA hardware is commercially available and operates with debugging software tools.

The ETM9 interface is nine pins total, four of which are data lines, and all pins can be used for GPIO after tracing is no longer needed. The ETM9 interface is used in conjunction with the JTAG interface for trace configuration. When tracing begins, the ETM9 engine compresses the data by various means before broadcasting data at high speed to the TPA over the four data lines. The most common ETM9 compression technique is to only output address information when the CPU branches to a location that cannot be inferred from the source code. This means

Obsolete Product(s) - Obsolete Product(s)

the host computer must have a static image of the code being executed for decompressing the ETM9 data. Because of this, self-modified code cannot be traced.

## 2.17 Ethernet MAC interface with DMA

STR91xF devices in 128-pin packages provide an IEEE-802.3-2002 compliant Media Access Controller (MAC) for Ethernet LAN communications through an industry standard Medium Independent Interface (MII). The STR91xF requires an external Ethernet physical interface device (PHY) to connect to the physical LAN bus (twisted-pair, fiber, etc.). The PHY is connected to the STR91xF MII port using as many as 18 signals (see pins which have signal names MII_* in *Table 3*).

The MAC corresponds to the OSI Data Link layer and the PHY corresponds to the OSI Physical layer. The STR91xF MAC is responsible for:

- Data encapsulation, including frame assembly before transmission, and frame parsing/ error detection during and after reception.
- Media access control, including initiation of frame transmission and recover from transmission failure.

The STR91xF MAC includes the following features:

- Supports 10 and 100 Mbps rates
- Tagged MAC frame support (VLAN support)
- Half duplex (CSMA/CD) and full duplex operation
- MAC control sublayer (control frames) support
- 32-bit CRC generation and removal
- Several address filtering modes for physical and multicast address (multicast and group addresses)
- 32-bit status code for each transmitted or received frame
- Internal FIFOs to buffer transmit and receive frames. Transmit FIFO depth is 4 words (32 bits each), and the receive FIFO is 16 words deep.

A 32-bit burst DMA channel residing on the AHB is dedicated to the Ethernet MAC for high-speed data transfers, side-stepping the CPU for minimal CPU impact during transfers. This DMA channel includes the following features:

- Direct SRAM to MAC transfers of transmit frames with the related status, by descriptor chain
- Direct MAC to SRAM transfers of receive frames with the related status, by descriptor chain
- Open and Closed descriptor chain management

## 2.18 USB 2.0 slave device interface with DMA

The STR91xF provides a USB slave controller that implements both the OSI Physical and Data Link layers for direct bus connection by an external USB host on pins USBDP and USBPN. The USB interface detects token packets, handles data transmission and reception, and processes handshake packets as required by the USB 2.0 standard.

The USB slave interface includes the following features:

Obsolete Product(s) - Obsolete Product(s)

- 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

Obsolete Product(s) - Obsolete Product(s)

SRAM, handling of transmission requests, and interrupt generation. The CPU has access to the Message SRAM via the Message Handler using a set of 38 control registers.

The follow features are supported by the CAN interface:

- Bitrates up to 1 Mbps
- Disable Automatic Retransmission mode for Time Triggered CAN applications
- 32 Message Objects
- Each Message Object has its own Identifier Mask
- Programmable FIFO mode
- Programmable loopback mode for self-test operation

The CAN interface is not supported by DMA.

## 2.20 UART interfaces with DMA

The STR91xF supports three independent UART serial interfaces, designated UART0, UART1, and UART2. Each interface is very similar to the industry-standard 16C550 UART device. All three UART channels support IrDA encoding/decoding, requiring only an external LED transceiver to pins UARTx_RX and UARTx_Tx for communication. One UART channel (UART0) supports full modem control signals.

UART interfaces include the following features:

- Maximum baud rate of 1.5 Mbps
- Separate FIFOs for transmit and receive, each 16 deep, each FIFO can be disabled by firmware if desired
- Programmable FIFO trigger levels between 1/8 and 7/8
- Programmable baud rate generator based on CCU master clock, or CCU master clock divided by two
- Programmable serial data lengths of 5, 6, 7, or 8 bits with start bit and 1 or 2 stop bits
- Programmable selection of even, odd, or no-parity bit generation and detection
- False start-bit detection
- Line break generation and detection
- Support of IrDA SIR ENDEC functions for data rates of up to 115.2K bps
- IrDA bit duration selection of 3/16 or low-power (1.14 to 2.23 µsec)
- Channel UART0 supports modem control functions CTS, DCD, DSR, RTS, DTR, and RI

For your reference, only two standard 16550 UART features are not supported, 1.5 stop bits and independent receive clock.

### 2.20.1 DMA

A programmable DMA channel may be assigned by CPU firmware to service channels UART0 and UART1 for fast and direct transfers between the UART 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 UART FIFOs are enabled.

Obsolete Product(s) - Obsolete Product(s)

## 2.21 I²C interfaces with DMA

The STR91xF supports two independent I2C serial interfaces, designated I2C0, and I2C1. Each interface allows direct connection to an I2C bus as either a bus master or bus slave device (firmware configurable). I2C is a two-wire communication channel, having a bi-directional data signal and a single-directional clock signal based on open-drain line drivers, requiring external pull-up resistors.

Byte-wide data is transferred between a Master device and a Slave device on two wires. More than one bus Master is allowed, but only one Master may control the bus at any given time. Data is not lost when another Master requests the use of a busy bus because I2C supports collision detection and arbitration. More than one Slave device may be present on the bus, each having a unique address. The bus Master initiates all data movement and generates the clock that permits the transfer. Once a transfer is initiated by the Master, any device that is addressed is considered a Slave. Automatic clock synchronization allows I2C devices with different bit rates to communicate on the same physical bus. A single device can play the role of Master or Slave, or a single device can be a Slave only. A Master or Slave device has the ability to suspend data transfers if the device needs more time to transmit or receive data.

Each I2C interface on the STR91xF has the following features:

- Programmable clock supports various rates up to I2C Standard rate (100 KHz) or Fast rate (400 KHz).
- Serial I/O Engine (SIOE) takes care of serial/parallel conversion; bus arbitration; clock generation and synchronization; and handshaking
- Multi-master capability
- 7-bit or 10-bit addressing

### 2.21.1 DMA

A programmable DMA channel may be assigned by CPU firmware to service each I2C channel for fast and direct transfers between the I2C bus and SRAM with little CPU involvement. Both DMA single-transfers and DMA burst-transfers are supported for transmit and receive.

## 2.22 SSP interfaces (SPI, SSI, and Microwire) with DMA

The STR91xF supports two independent Synchronous Serial Port (SSP) interfaces, designated SSP0, and SSP1. Primary use of each interface is for supporting the industry standard Serial Peripheral Interface (SPI) protocol, but also supporting the similar Synchronous Serial Interface (SSI) and Microwire communication protocols.

SPI is a three or four wire synchronous serial communication channel, capable of full-duplex operation. In three-wire configuration, there is a clock signal, and two data signals (one data signal from Master to Slave, the other from Slave to Master). In four-wire configuration, an additional Slave Select signal is output from Master and received by Slave.

The SPI clock signal is a gated clock generated from the Master and regulates the flow of data bits. The Master may transmit at a variety of baud rates, up to 24 MHz

In multi-Slave operation, no more than one Slave device can transmit data at any given time. Slave selection is accomplished when a Slave's "Slave Select" input is permanently grounded or asserted active-low by a Master device. Slave devices that are not selected do not interfere with SPI activities. Slave devices ignore the clock signals and keep their data output pins in

Obsolete Product(s) - Obsolete Product(s)

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.

Obsolete Product(s) - Obsolete Product(s)

## 2.24 A/D converter (ADC)

The STR91xF provides an eight-channel, 10-bit successive approximation analog-to-digital converter. The ADC input pins are multiplexed with other functions on Port 4 as shown in *Table 3*. Following are the major ADC features:

- Fast conversion time, as low as 0.7 usec
- Accuracy. Integral and differential non-linearity are typically within 4 conversion counts.
- 0 to 3.6V input range. External reference voltage input pin (AVREF) available on 128-pin packages for better accuracy on low-voltage inputs. The voltage on AVREF can range from 1.0V to $V_{DDQ}$.
- CPU Firmware may convert one ADC input channel at a time, or it has the option to set the ADC to automatically scan and convert all eight ADC input channels sequentially before signalling an end-of-conversion
- Automatic continuous conversion mode is available for any number of designated ADC input channels
- Analog watchdog mode provides automatic monitoring of any ADC input, comparing it against two programmable voltage threshold values. The ADC unit will set a flag or it will interrupt the CPU if the input voltage rises above the higher threshold, or drops below the lower threshold.
- The ADC unit goes to stand-by mode (very low-current consumption) after any reset event. CPU firmware may also command the ADC unit to stand-by mode at any time.

## 2.25 Standard timers (TIM) with DMA

The STR91xF has four independent, free-running 16-bit timer/counter modules designated TIM0, TIM1, TIM2, and TIM3. Each general purpose timer/counter can be configured by firmware for a variety of tasks including; pulse width and frequency measurement (input capture), generation of waveforms (output compare and PWM), event counting, delay timing, and up/down counting.

Each of the four timer units have the following features:

- 16-bit free running timer/counter
- Internal timer/counter clock source from a programmable 8-bit prescale of the CCU PCLK clock output
- Optional external timer/counter clock source from pin P2.4 shared by TIM0/TIM1, and pin P2.5 shared by TIM2/TIM3. Frequency of these external clocks must be at least 4 times less the frequency of the internal CCU PCLK clock output. The Master clock ($f_{MSTR}$) with a 16-bit prescaler can also be selected as an external clock source
- Two dedicated 16-bit Input Capture registers for measuring up to two input signals. Input Capture has programmable selection of input signal edge detection
- Two dedicated 16-bit Output Compare registers for generation up to two output signals
- PWM output generation with 16-bit resolution of both pulse width and frequency
- One pulse generation in response to an external event
- A dedicated interrupt to the CPU with five interrupt flags

Obsolete Product(s) - Obsolete Product(s)

### 2.25.1 DMA

A programmable DMA channel may be assigned by CPU firmware to service each timer/ counter module TIM0 and TIM1 for fast and direct transfers.

## 2.26 Three-phase induction motor controller (IMC)

The STR91xF provides an integrated controller for variable speed motor control applications.

Six PWM outputs are generated on high current drive pins P6.0 to P6.5 for controlling a three-phase AC induction motor drive circuit assembly. Rotor speed feedback is provided by capturing a tachometer input signal on pin P6.6, and an asynchronous hardware emergency stop input is available on pin P6.7 to stop the motor immediately if needed, independently of firmware.

The IMC unit has the following features:

- Three PWM outputs generated using a 10-bit PWM counter, one for each phase U, V, W. Complimentary PWM outputs are also generated for each phase.
- Choice of classic or zero-centered PWM generation modes
- 10-bit PWM counter clock is supplied through a programmable 8-bit prescaler of the APB clock.
- Programmable 6-bit dead-time generator to add delay to each of the three complimentary PWM outputs
- 8-bit repetition counter
- Automatic rotor speed measurement with 16-bit resolution. Schmitt trigger tachometer input with programmable edge detection
- Hardware asynchronous emergency stop input
- A dedicated interrupt to CPU with eight flags

## 2.27 External memory interface (EMI)

STR91xF devices in 128-pin packages offer an external memory bus for connecting external parallel peripherals and memories. The EMI bus resides on ports 7, 8, and 9 and operates with either an 8 or 16-bit data path. The configuration of 8 or 16 bit mode is specified by CPU firmware writing to configuration registers at run-time. If the application does not use the EMI bus, then these port pins may be used for general purpose I/O as shown in *Table 3*.

The EMI has the following features:

- Supports static asynchronous memory access cycles, including page mode for non-mux operation.
- Four configurable memory regions, each with a chip select output (EMI_CS0n ... EMI_CS3n)
- Programmable wait states per memory region for both write and read operations
- **16-bit multiplexed data mode** (*Figure 4*): 16 bits of data and 16 bits of low-order address are multiplexed together on ports 8 and 9, while port 7 contains eight more high-order address signals. 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 signals on pins EMI_BWR_WRLn and EMI_WRHn are the write strobes for the low and high data bytes

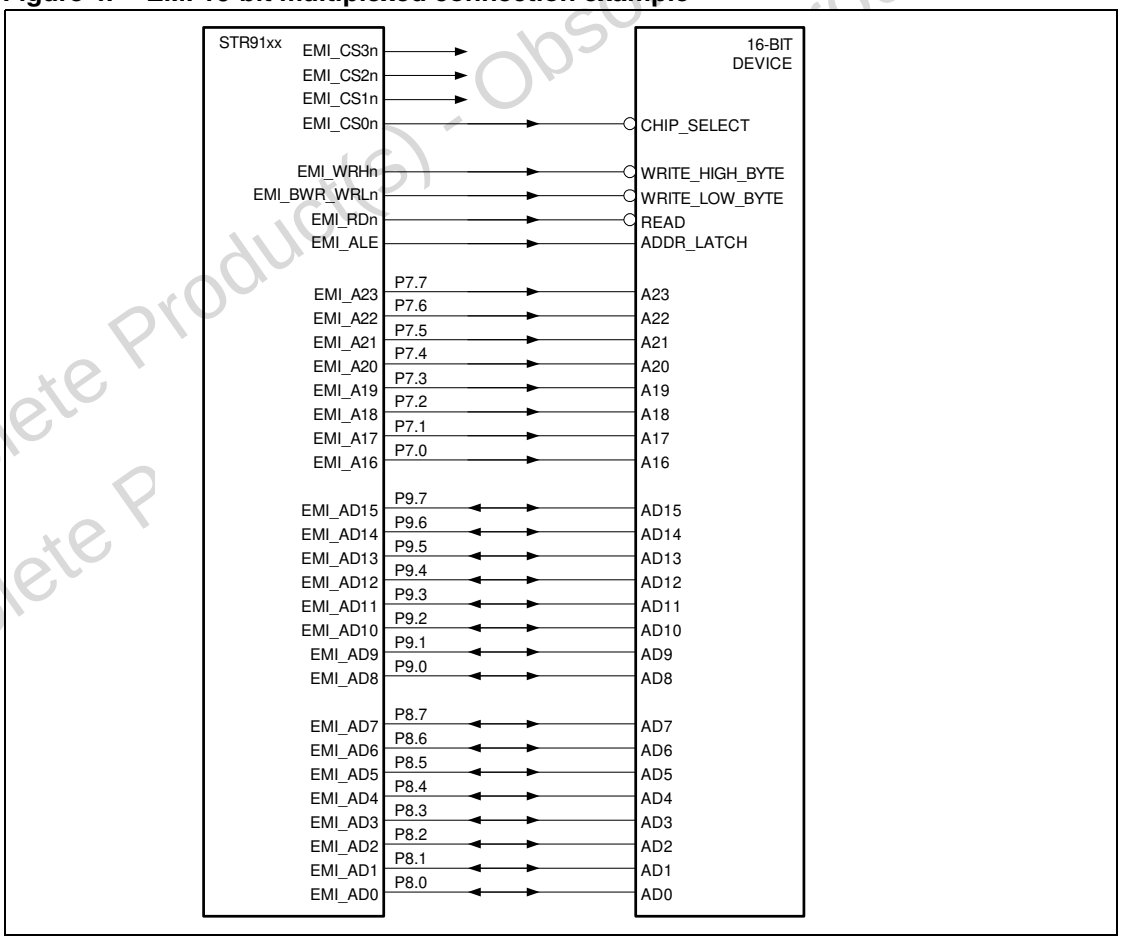
Obsolete Product(s) - Obsolete Product(s)

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**




Obsolete Product(s) - Obsolete Product(s)

**Figure 5. EMI 8-bit multiplexed connection example**

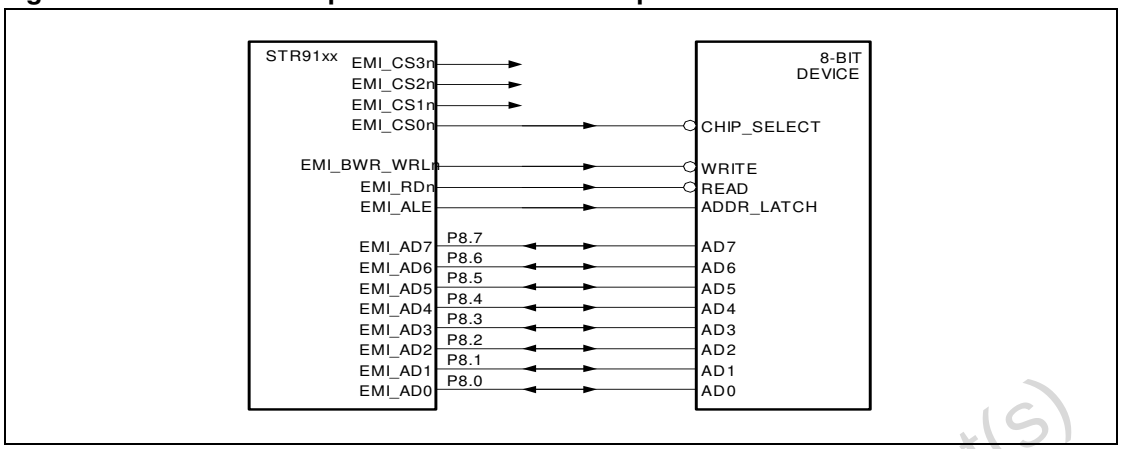
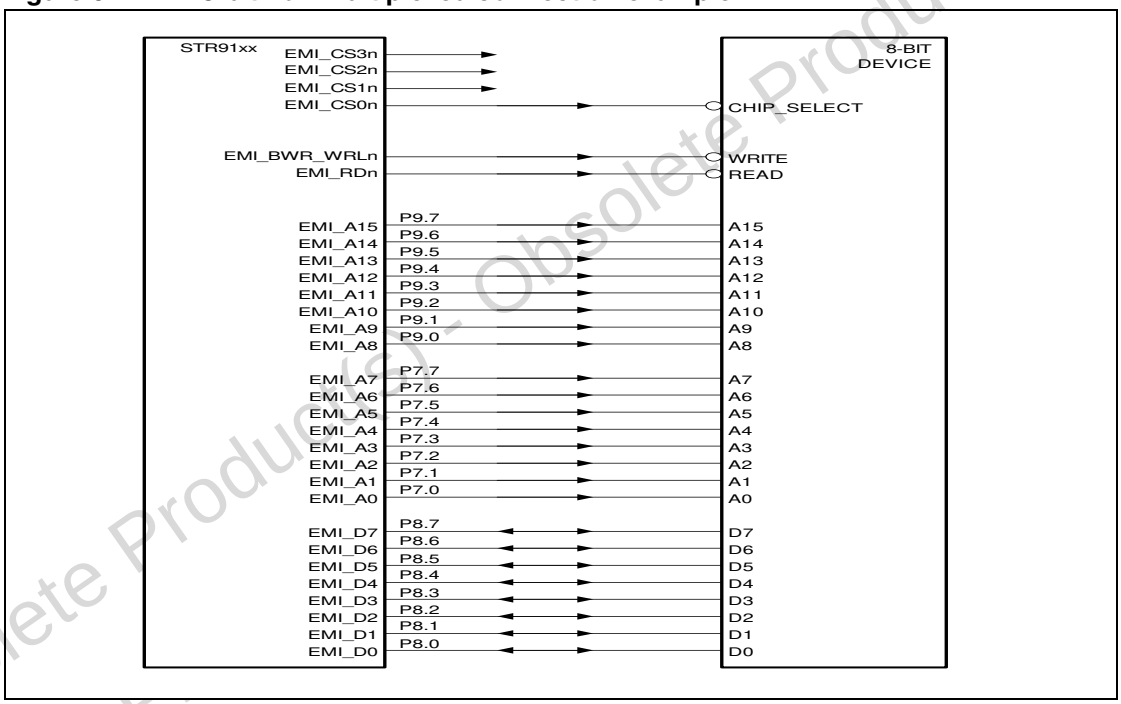



**Figure 6. EMI 8-bit non-multiplexed connection example**




Obsolete Product(s) - Obsolete Product(s)

# 3 Related documentation

**Available from www.arm.com:**

ARM966E-S Rev 2 Technical Reference Manual

**Available from www.st.com:**

STR91xF Reference Manual

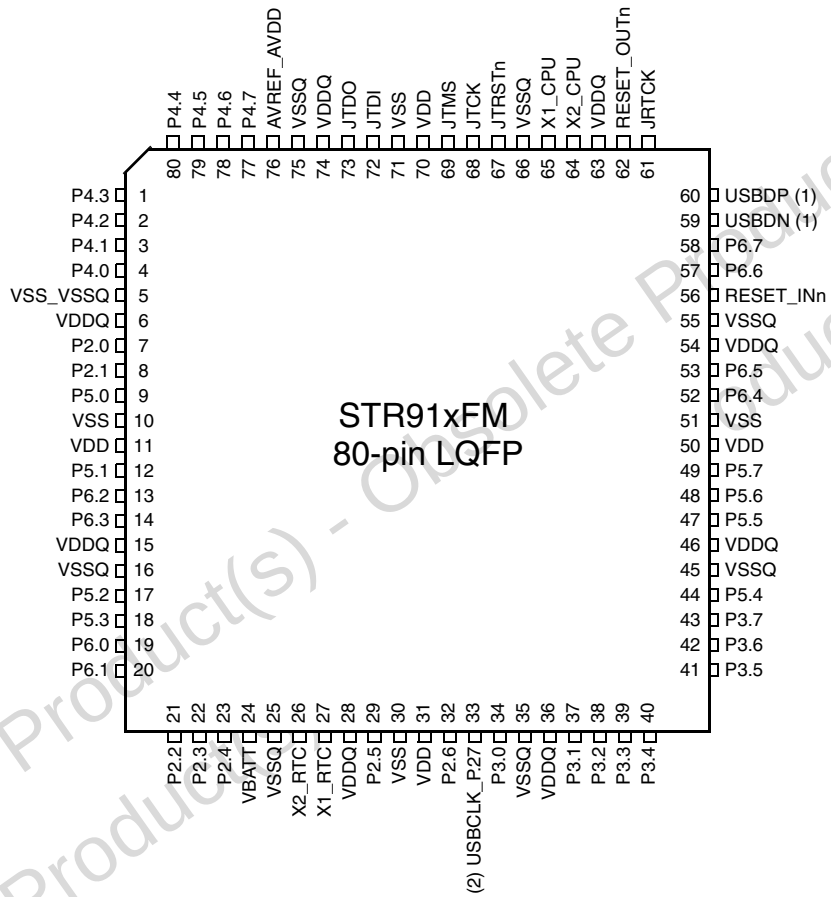
STR9 Flash Programming Manual (PM0020)

The above is a selected list only, a full list STR91xF application notes can be viewed at *http://www.st.com*.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

# 4 Pin description

**Figure 7. STR91xFM 80-pin package pinout**

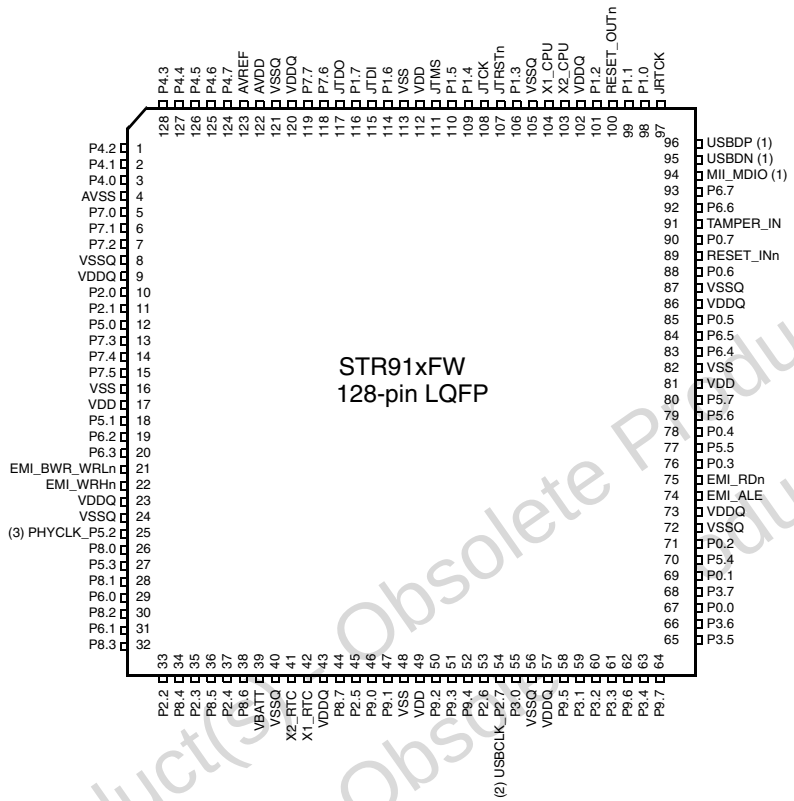



Obsolete Product(s) - Obsolete Product(s)

1) NU (Not Used) on STR910FM devices. Pin 59 is not connected, pin 60 must be pulled up by a 1.5Kohm resistor to VDDQ.

2) No USBCLK function on STR910FM devices.

**Figure 8. STR91xFW 128-pin package pinout**




1) NU (Not Used) on STR910FW devices. Pin 95 is not connected, pin 96 must be pulled up by a 1.5Kohm resistor to VDDQ.

2) No USBCLK function on STR910FW devices.

3) No PHYCLK function on STR910FW devices.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## 4.1 Default pin functions

During and just after reset, all pins on ports 0-9 default to high-impedance input mode until CPU firmware assigns other functions to the pins. This initial input mode routes all pins on ports 0-9 to be read as GPIO inputs as shown in the “Default Pin Function” column of *Table 3*. Simultaneously, certain port pin signals are also routed to other functional inputs as shown in the “Default Input Function” column of *Table 3*, and these pin input functions will remain until CPU firmware makes other assignments. At any time, even after the CPU assigns pins to alternate functions, the CPU may always read the state of any pin on ports 0-9 as a GPIO input. CPU firmware may assign alternate functions to port pins as shown in columns “Alternate Input 1” or “Alternate Output 1, 2, 3” of *Table 3* by writing to control registers at run-time.

**Notes for *Table 3*:**

*Notes: 1 STMicroelectronics advises to ground, or pull up to $V_{DDQ}$ using a 100 KΩ resistor, all unused pins on port 0 - 9 to reduce noise susceptibility, noise generation, and minimize power consumption. There are no internal or programmable pull-up resistors on ports 0-9.*

*2 All pins on ports 0 - 9 are 5V tolerant*

*3 Pins on ports 0,1,2,4,5,7,8,9 have 4 mA drive and 4mA sink. Ports 3 and 6 have 8 mA drive and 8 mA sink.*

*4 For 8-bit non-muxed EMI operation: Port 8 is eight bits of data, ports 7 and 9 are 16 bits of address.*

*5 For 16-bit muxed EMI operation: Ports 8 and 9 are 16 bits of muxed address and data bits, port 7 is up to eight additional bits of high-order address*

*6 Signal polarity is programmable for interrupt request inputs, EMI_ALE, timer input capture inputs and output compare/PWM outputs, motor control tach and emergency stop inputs, and motor control phase outputs.*

*7 HiZ = High Impedance, V = Voltage Source, G = Ground, I/O = Input/Output*

*8 STR910F devices do not support USB. On these devices USBDP and USBDN signals are "Not Used" (USBDN is not connected, USBDP must be pulled up by a 1.5K ohm resistor to VDDQ), and all functions named “USB" are not available.*

*9 STR910F 128-pin devices do not support Ethernet. On these devices PHYCLK and all functions named “MII*" are not available.*

**Table 3. Device pin description**

| 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 |
| - | 67 | P0.0 | I/O | GPIO_0.0, GP Input, HiZ | MII_TX_CLK, PHY Xmit clock | I2C0_CLKIN, *I2C clock in* | GPIO_0.0, GP Output | I2C0_CLKOUT, *I2C clock out* | ETM_PCK0, *ETM Packet* |
| - | 69 | P0.1 | I/O | GPIO_0.1, GP Input, HiZ | - | I2C0_DIN, *I2C data in* | GPIO_0.1, GP Output | I2C0_DOUT, I2C data out | ETM_PCK1, *ETM Packet* |
| - | 71 | P0.2 | I/O | GPIO_0.2, GP Input, HiZ | MII_RXD0, PHY Rx data0 | I2C1_CLKIN, *I2C clock in* | GPIO_0.2, GP Output | I2C1_CLKOUT, *I2C clock out* | ETM_PCK2, *ETM Packet* |
| - | 76 | P0.3 | I/O | GPIO_0.3, GP Input, HiZ | MII_RXD1, PHY Rx data | I2C1_DIN, I2C data in | GPIO_0.3, GP Output | I2C1_DOUT, I2C data out | ETM_PCK3, *ETM Packet* |
| - | 78 | P0.4 | I/O | GPIO_0.4, GP Input, HiZ | MII_RXD2, PHY Rx data | TIM0_ICAP1, Input Capture | GPIO_0.4, GP Output | EMI_CS0n, EMI Chip Select | ETM_PSTAT0, ETM pipe status |
| - | 85 | P0.5 | I/O | GPIO_0.5, GP Input, HiZ | MII_RXD3, PHY Rx data | TIM0_ICAP2, Input Capture | GPIO_0.5, GP Output | EMI_CS1n, EMI Chip Select | ETM_PSTAT1, ETM pipe status |

Obsolete Product(s) - Obsolete Product(s)

| 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 |
| - | 88 | P0.6 | I/O | GPIO_0.6, GP Input, HiZ | MII_RX_CLK, PHY Rx clock | TIM2_ICAP1, Input Capture | GPIO_0.6, GP Output | EMI_CS2n, EMI Chip Select | ETM_PSTAT2, ETM pipe status |
| - | 90 | P0.7 | I/O | GPIO_0.7, GP Input, HiZ | MII_RX_DV, PHY data valid | TIM2_ICAP2, Input Capture | GPIO_0.7, GP Output | EMI_CS3n, EMI Chip Select | ETM_TRSYNC, ETM trace sync |
| | | | | | | | | | |
| - | 98 | P1.0 | I/O | GPIO_1.0, GP Input, HiZ | MII_RX_ER, PHY rcv error | ETM_EXTRIG, ETM ext. trigger | GPIO_1.0, GP Output | UART1_TX, UART xmit data | SSP1_SCLK, SSP mstr clk out |
| - | 99 | P1.1 | I/O | GPIO_1.1, GP Input, HiZ | - | UART1_RX, UART rcv data | GPIO_1.1, GP Output | MII_TXD0, MAC Tx data | SSP1_MOSI, SSP mstr dat out |
| - | 101 | P1.2 | I/O | GPIO_1.2, GP Input, HiZ | - | SSP1_MISO, SSP mstr data in | GPIO_1.2, GP Output | MII_TXD1, MAC Tx data | UART0_TX, UART xmit data |
| - | 106 | P1.3 | I/O | GPIO_1.3, GP Input, HiZ | - | UART2_RX, UART rcv data | GPIO_1.3, GP Output | MII_TXD2, MAC Tx data | SSP1_NSS, SSP mstr sel out |
| - | 109 | P1.4 | I/O | GPIO_1.4, GP Input, HiZ | - | I2C0_CLKIN, *I2C clock in* | GPIO_1.4, GP Output | MII_TXD3, MAC Tx data | I2C0_CLKOUT, *I2C clock out* |
| - | 110 | P1.5 | I/O | GPIO_1.5, GP Input, HiZ | MII_COL, PHY collision | CAN_RX, CAN rcv data | GPIO_1.5, GP Output | UART2_TX, UART xmit data | ETM_TRCLK, ETM trace clock |
| - | 114 | P1.6 | I/O | GPIO_1.6, GP Input, HiZ | MII_CRS, PHY carrier sns | I2C0_DIN, *I2C data in* | GPIO_1.6, GP Output | CAN_TX, CAN Tx data | I2C0_DOUT, I2C data out |
| - | 116 | P1.7 | I/O | GPIO_1.7, GP Input, HiZ | - | ETM_EXTRIG, ETM ext. trigger | GPIO_1.7, GP Output | MII_MDC, MAC mgt dat ck | ETM_TRCLK, ETM trace clock |
| | | | | | | | | | |
| 7 | 10 | P2.0 | I/O | GPIO_2.0, GP Input, HiZ | UART0_CTS, Clear To Send | I2C0_CLKIN, *I2C clock in* | GPIO_2.0, GP Output | I2C0_CLKOUT, *I2C clock out* | ETM_PCK0, *ETM Packet* |
| 8 | 11 | P2.1 | I/O | GPIO_2.1, GP Input, HiZ | UART0_DSR, Data Set Ready | I2C0_DIN, *I2C data in* | GPIO_2.1, GP Output | I2C0_DOUT, I2C data out | ETM_PCK1, *ETM Packet* |
| 21 | 33 | P2.2 | I/O | GPIO_2.2, GP Input, HiZ | UART0_DCD, Dat Carrier Det | I2C1_CLKIN, *I2C clock in* | GPIO_2.2, GP Output | I2C1_CLKOUT, *I2C clock out* | ETM_PCK2, *ETM Packet* |
| 22 | 35 | P2.3 | I/O | GPIO_2.3, GP Input, HiZ | UART0_RI, Ring Indicator | I2C1_DIN, I2C data in | GPIO_2.3, GP Output | I2C1_DOUT, I2C data out | ETM_PCK3, *ETM Packet* |
| 23 | 37 | P2.4 | I/O | GPIO_2.4, GP Input, HiZ | EXTCLK_T0T1*Ext clk timer0/1* | SSP0_SCLK, SSP slv clk in | GPIO_2.4, GP Output | SSP0_SCLK, SSP mstr clk out | ETM_PSTAT0, ETM pipe status |
| 29 | 45 | P2.5 | I/O | GPIO_2.5, GP Input, HiZ | EXTCLK_T2T3*Ext clk timer2/3* | SSP0_MOSI, SSP slv dat in | GPIO_2.5, GP Output | SSP0_MOSI, SSP mstr dat out | ETM_PSTAT1, ETM pipe status |
| 32 | 53 | P2.6 | I/O | GPIO_2.6, GP Input, HiZ | - | SSP0_MISO, SSP mstr data in | GPIO_2.6, GP Output | SSP0_MISO, SSP slv data out | ETM_PSTAT2, ETM pipe status |
| 33 | 54 | USBCLK _P2.7 | I/O | GPIO_2.7, GP Input, HiZ | USB_CLK48M, *48MHz to USB* | SSP0_NSS, SSP slv sel in | GPIO_2.7, GP Output | SSP0_NSS, SSP mstr sel out | ETM_TRSYNC, ETM trace sync |
| | | | | | | | | | |
| 34 | 55 | P3.0 | I/O | GPIO_3.0, GP Input, HiZ | DMA_RQST0, Ext DMA requst | UART0_RxD, UART rcv data | GPIO_3.0, GP Output | UART2_TX, UART xmit data | TIM0_OCMP1, *Out comp/PWM* |
| 37 | 59 | P3.1 | I/O | GPIO_3.1, GP Input, HiZ | DMA_RQST1, Ext DMA requst | UART2_RxD, UART rcv data | GPIO_3.1, GP Output | UART0_TX, UART xmit data | TIM1_OCMP1, *Out comp/PWM* |
| 38 | 60 | P3.2 | I/O | GPIO_3.2, GP Input, HiZ | EXINT2, External Intr | UART1_RxD, UART rcv data | GPIO_3.2, GP Output | CAN_TX, CAN Tx data | UART0_DTR, Data Trmnl Rdy |
| 39 | 61 | P3.3 | I/O | GPIO_3.3, GP Input, HiZ | EXINT3, External Intr | CAN_RX, CAN rcv data | GPIO_3.3, GP Output | UART1_TX, UART xmit data | UART0_RTS, Ready To Send |
| 40 | 63 | P3.4 | I/O | GPIO_3.4, GP Input, HiZ | EXINT4, External Intr | SSP1_SCLK, SSP slv clk in | GPIO_3.4, GP Output | SSP1_SCLK, SSP mstr clk out | UART0_TX, UART xmit data |
| 41 | 65 | P3.5 | I/O | GPIO_3.5, GP Input, HiZ | EXINT5, External Intr | SSP1_MISO, SSP mstr data in | GPIO_3.5, GP Output | SSP1_MISO, SSP slv data out | UART2_TX, UART xmit data |
| 42 | 66 | P3.6 | I/O | GPIO_3.6, GP Input, HiZ | EXINT6, External Intr | SSP1_MOSI, SSP slv dat in | GPIO_3.6, GP Output | SSP1_MOSI, SSP mstr dat out | CAN_TX, CAN Tx data |

Obsolete Product(s) - Obsolete Product(s)

| 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 |

Obsolete Product(s) - Obsolete Product(s)

| 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 |
| - | 5 | P7.0 | I/O | GPIO_7.0, GP Input, HiZ | EXINT24, External Intr | TIM0_ICAP1, Input Capture | GPIO_7.0, GP Output | 8b) EMI_A0, 16b) EMI_A16 | ETM_PCK0, *ETM Packet* |
| - | 6 | P7.1 | I/O | GPIO_7.1, GP Input, HiZ | EXINT25, External Intr | TIM0_ICAP2, Input Capture | GPIO_7.1, GP Output | 8b) EMI_A1, 16b) EMI_A17 | ETM_PCK1, *ETM Packet* |
| - | 7 | P7.2 | I/O | GPIO_7.2, GP Input, HiZ | EXINT26, External Intr | TIM2_ICAP1, Input Capture | GPIO_7.2, GP Output | 8b) EMI_A2, 16b) EMI_A18 | ETM_PCK2, *ETM Packet* |
| - | 13 | P7.3 | I/O | GPIO_7.3, GP Input, HiZ | EXINT27, External Intr | TIM2_ICAP2, Input Capture | GPIO_7.3, GP Output | 8b) EMI_A3, 16b) EMI_A19 | ETM_PCK3, *ETM Packet* |
| - | 14 | P7.4 | I/O | GPIO_7.4, GP Input, HiZ | EXINT28, External Intr | UART0_RxD, UART rcv data | GPIO_7.4, GP Output | 8b) EMI_A4, 16b) EMI_A20 | EMI_CS3n, EMI Chip Select |
| - | 15 | P7.5 | I/O | GPIO_7.5, GP Input, HiZ | EXINT29, External Intr | ETM_EXTRIG, ETM ext. trigger | GPIO_7.5, GP Output | 8b) EMI_A5, 16b) EMI_A21 | EMI_CS2n, EMI Chip Select |
| - | 118 | P7.6 | I/O | GPIO_7.6, GP Input, HiZ | EXINT30, External Intr | TIM3_ICAP1, Input Capture | GPIO_7.6, GP Output | 8b) EMI_A6, 16b) EMI_A22 | EMI_CS1n, EMI Chip Select |
| - | 119 | P7.7 | I/O | GPIO_7.7, GP Input, HiZ | EXINT31, External Intr | TIM3_ICAP2, Input Capture | GPIO_7.7, GP Output | EMI_CS0n, EMI chip select | 16b) EMI_A23, 8b) EMI_A7 |
| | | | | | | | | | |
| - | 26 | P8.0 | I/O | GPIO_8.0, GP Input, HiZ | - | - | GPIO_8.0, GP Output | 8b) EMI_D0, 16b) EMI_AD0 | - |
| - | 28 | P8.1 | I/O | GPIO_8.1, GP Input, HiZ | - | - | GPIO_8.1, GP Output | 8b) EMI_D1, 16b) EMI_AD1 | - |
| - | 30 | P8.2 | I/O | GPIO_8.2, GP Input, HiZ | - | - | GPIO_8.2, GP Output | 8b) EMI_D2, 16b) EMI_AD2 | - |
| - | 32 | P8.3 | I/O | GPIO_8.3, GP Input, HiZ | - | - | GPIO_8.3, GP Output | 8b) EMI_D3, 16b) EMI_AD3 | - |
| - | 34 | P8.4 | I/O | GPIO_8.4, GP Input, HiZ | - | - | GPIO_8.4, GP Output | 8b) EMI_D4, 16b) EMI_AD4 | - |
| - | 36 | P8.5 | I/O | GPIO_8.5, GP Input, HiZ | - | - | GPIO_8.5, GP Output | 8b) EMI_D5, 16b) EMI_AD5 | - |
| - | 38 | P8.6 | I/O | GPIO_8.6, GP Input, HiZ | - | - | GPIO_8.6, GP Output | 8b) EMI_D6, 16b) EMI_AD6 | - |
| - | 44 | P8.7 | I/O | GPIO_8.7, GP Input, HiZ | - | - | GPIO_8.7, GP Output | 8b) EMI_D7, 16b) EMI_AD7 | - |
| | | | | | | | | | |
| - | 46 | P9.0 | I/O | GPIO_9.0, GP Input, HiZ | - | - | GPIO_9.0, GP Output | 8b) EMI_A8 16b) EMI_AD8 | - |
| - | 47 | P9.1 | I/O | GPIO_9.1, GP Input, HiZ | - | - | GPIO_9.1, GP Output | 8b) EMI_A9, 16b) EMI_AD9 | - |
| - | 50 | P9.2 | I/O | GPIO_9.2, GP Input, HiZ | - | - | GPIO_9.2, GP Output | 8b) EMI_A10, 16b)EMI_AD10 | - |
| - | 51 | P9.3 | I/O | GPIO_9.3, GP Input, HiZ | - | - | GPIO_9.3, GP Output | 8b) EMI_A11, 16b)EMI_AD11 | - |
| - | 52 | P9.4 | I/O | GPIO_9.4, GP Input, HiZ | - | - | GPIO_9.4, GP Output | 8b) EMI_A12, 16b)EMI_AD12 | - |
| - | 58 | P9.5 | I/O | GPIO_9.5, GP Input, HiZ | - | - | GPIO_9.5, GP Output | 8b) EMI_A13, 16b)EMI_AD13 | - |
| - | 62 | P9.6 | I/O | GPIO_9.6, GP Input, HiZ | - | - | GPIO_9.6, GP Output | 8b) EMI_A14, 16b)EMI_AD14 | - |
| - | 64 | P9.7 | I/O | GPIO_9.7, GP Input, HiZ | - | - | GPIO_9.7, GP Output | 8b) EMI_A15, 16b)EMI_AD15 | - |

Obsolete Product(s) - Obsolete Product(s)

| 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 | | | | |

Obsolete Product(s) - Obsolete Product(s)

| 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 |
| - | 123 | AVREF | V | ADC reference voltage input | N/A | | | | |
| 76 | - | AVREF _AVDD | V | Combined ADC ref voltage and ADC analog voltage source, 2.7V - 3.6V | N/A | | | | |
| 24 | 39 | VBATT | V | Standby voltage input for RTC and SRAM backup | N/A | | | | |
| 6 | 9 | VDDQ | V | V Source for I/O and USB. 2.7V to 3.6V | N/A | | | | |
| 15 | 23 | VDDQ | V | | | | | | |
| 36 | 57 | VDDQ | V | | | | | | |
| 46 | 73 | VDDQ | V | | | | | | |
| 54 | 86 | VDDQ | V | | | | | | |
| 28 | 43 | VDDQ | V | | | | | | |
| 63 | 102 | VDDQ | V | | | | | | |
| 74 | 120 | VDDQ | V | | | | | | |
| - | - | VDDQ | V | | | | | | |
| - | 8 | VSSQ | G | Digital Ground for !/O and USB | N/A | | | | |
| 16 | 24 | VSSQ | G | | | | | | |
| 35 | 56 | VSSQ | G | | | | | | |
| - | - | VSSQ | G | | | | | | |
| 45 | 72 | VSSQ | G | | | | | | |
| 55 | 87 | VSSQ | G | | | | | | |
| 25 | 40 | VSSQ | G | | | | | | |
| 66 | 105 | VSSQ | G | | | | | | |
| 75 | 121 | VSSQ | G | | | | | | |
| 11 | 17 | VDD | V | V Source for CPU. 1.65V - 2.0V | N/A | | | | |
| 31 | 49 | VDD | V | | | | | | |
| 50 | 81 | VDD | V | | | | | | |
| 70 | 112 | VDD | V | | | | | | |
| 10 | 16 | VSS | G | Digital Ground for CPU | N/A | | | | |
| 30 | 48 | VSS | G | | | | | | |
| 51 | 82 | VSS | G | | | | | | |
| 71 | 113 | VSS | G | | | | | | |
| - | - | PLLVDDQ | V | V Source for PLL 2.7 to 3.6 V | N/A | | | | |
| - | - | PLLVSSQ | G | Digital Ground for PLL | | | | | |

Obsolete Product(s) - Obsolete Product(s)

# 5 Memory mapping

The ARM966E-S CPU addresses a single linear address space of 4 giga-bytes ($2^{32}$) from address 0x0000.0000 to 0xFFFF.FFFF as shown in *Figure 9*. Upon reset the CPU boots from address 0x0000.0000, which is chip-select zero at address zero in the Flash Memory Interface (FMI).

The Instruction TCM and Data TCM enable high-speed CPU operation without incurring any performance or power penalties associated with accessing the system buses (AHB and APB). I-TCM and D-TCM address ranges are shown at the bottom of the memory map in *Figure 9*.

## 5.1 Buffered and non-buffered writes

The CPU makes use of write buffers on the AHB and the D-TCM to decouple the CPU from any wait states associated with a write operation. The user may choose to use write with buffers on the AHB by setting bit 3 in control register CP15 and selecting the appropriate AHB address range when writing. By default at reset, buffered writes are disabled (bit 3 of CP15 is clear) and all AHB writes are non-buffered until enabled. *Figure 9* shows that most addressable items on the AHB are aliased at two address ranges, one for buffered writes and another for non-buffered writes. A buffered write will allow the CPU to continue program execution while the write-back is performed through a FIFO to the final destination on the AHB. If the FIFO is full, the CPU is stalled until FIFO space is available. A non-buffered write will impose an immediate delay to the CPU, but results in a direct write to the final AHB destination, ensuring data coherency. Read operations from AHB locations are always direct and never buffered.

## 5.2 System (AHB) and peripheral (APB) buses

The CPU will access SRAM, higher-speed peripherals (USB, Ethernet, Programmable DMA), and the external bus (EMI) on the AHB at their respective base addresses indicated in *Figure 9*. Lower-speed peripherals reside on the APB and are accessed using two separate AHB-to-APB bridge units (APB0 and APB1). These bridge units are essentially address windows connecting the AHB to the APB. To access an individual APB peripheral, the CPU will place an address on the AHB bus equal to the base address of the appropriate bridge unit APB0 or APB1, plus the offset of the particular peripheral, plus the offset of the individual data location within the peripheral. *Figure 9* shows the base addresses of bridge units APB0 and APB1, and also the base address of each APB peripheral. Please consult the STR91xF Reference manual for the address of data locations within each individual peripheral.

## 5.3 SRAM

The SRAM is aliased at three separate address ranges as shown in *Figure 9*. When the CPU accesses SRAM starting at 0x0400.0000, the SRAM appears on the D-TCM. When CPU access starts at 0x4000.0000, SRAM appears in the buffered AHB range. Beginning at CPU address 0x5000.0000, SRAM is in non-buffered AHB range. The SRAM size must be specified by CPU intitialization firmware writing to a control register after any reset condition. Default SRAM size is 32K bytes, with option to set to 64K bytes on STR91xFx32 devices, and to 96K bytes on STR91xFx44 devices.

Obsolete Product(s) - Obsolete Product(s)

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.

Obsolete Product(s) - Obsolete Product(s)

**Notes for *Figure 9: STR91xF memory map on page 45*:**

*Notes: 1 Either of the two Flash memories may be placed at CPU boot address 0x0000.0000. By default, the primary Flash memory is in boot position starting at CPU address 0x0000.0000 and the secondary Flash memory may be placed at a higher address following the end of the primary Flash memory. This default option may be changed using the STR91xx device configuration software, placing the secondary Flash memory at CPU boot location 0x0000.0000, and then the primary Flash memory may be placed at a higher address.*

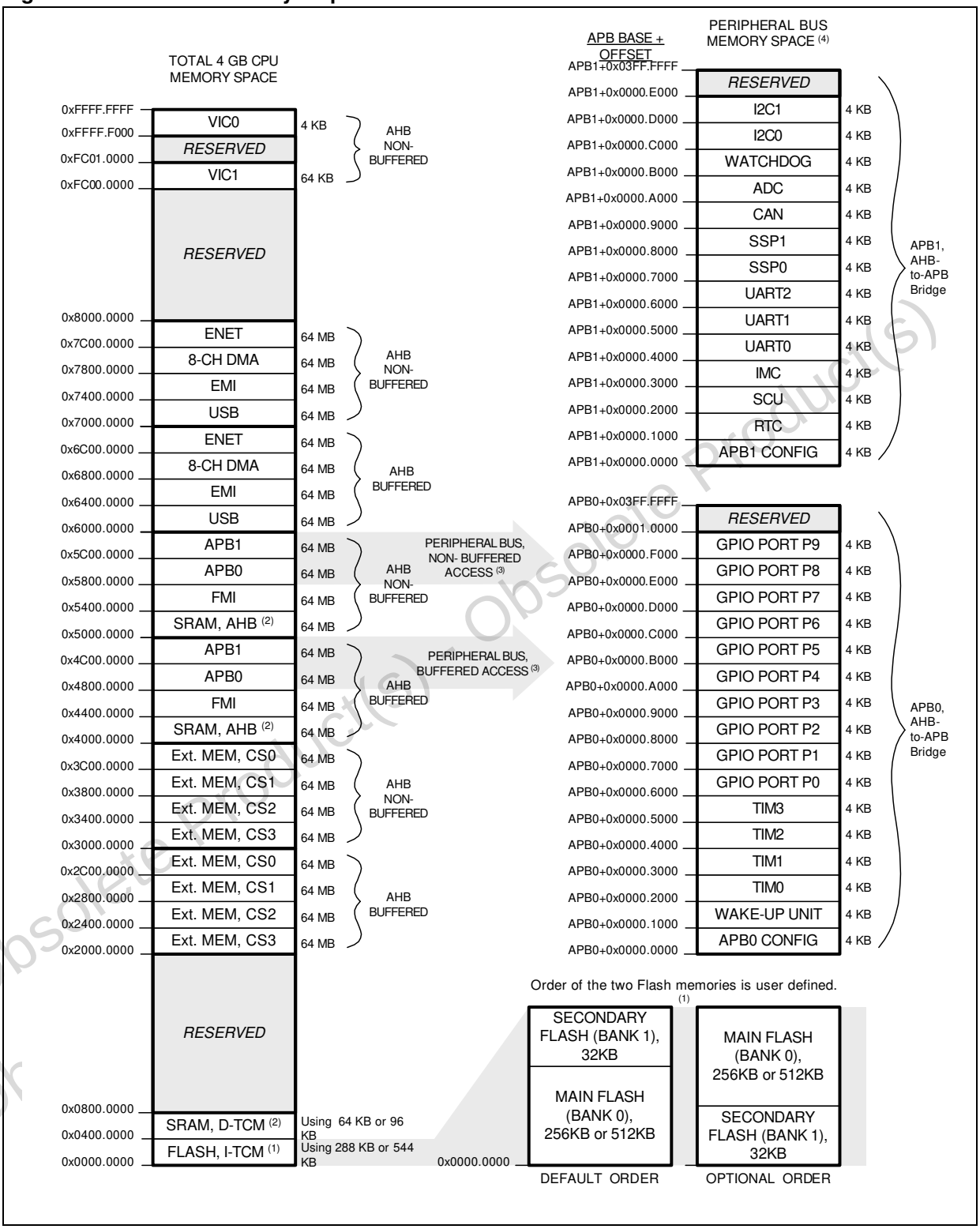
*2 The local SRAM (64KB or 96KB) is aliased in three address windows. A) At 0x0400.0000 the SRAM is accessible through the CPU's D-TCM, at 0x4000.0000 the SRAM is accessible through the CPU's AHB in buffered accesses, and at 0x5000.0000 the SRAM is accessible through the CPU's AHB in non-buffered accesses. An AHB bus master other than the CPU can access SRAM in all three aliased windows, but these accesses are always non-buffered. The CPU is the only AHB master that can performed buffered writes.*

*3 APB peripherals reside in two AHB-to-APB peripheral bridge address windows, APB0 and APB1. These peripherals are accessible with buffered AHB access if the CPU addresses them in the address range of 0x4800.0000 to 0x4FFF.FFFF, and non-buffered access in the address range of 0x5800.0000 to 0x5FFF.FFFF.*

*4 Individual peripherals on the APB are accessed at the listed address offset plus the base address of the appropriate AHB-to-APB bridge.*

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 9. STR91xF memory map**




Obsolete Product(s) - Obsolete Product(s)

# 6 Electrical characteristics

## 6.1 Absolute maximum ratings

This product contains devices to protect the inputs against damage due to high static voltages. However, it is advisable to take normal precautions to avoid application of any voltage higher than the specified maximum rated voltages. It is also recommended to ground any unused input pin to reduce power consumption and minimize noise.

**Table 4. Absolute maximum ratings**

| Symbol | Parameter | Value | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| $V_{DD}$ | Voltage on VDD pin with respect to ground $V_{SS}$ | -0.3 | 2.4 | V |
| $V_{DDQ}$ | Voltage on VDDQ pin with respect to ground $V_{SS}$ | -0.3 | 4.0 | V |
| $V_{BATT}$ | Voltage on VBATT pin with respect to ground $V_{SS}$ | -0.3 | 4.0 | V |
| $AV_{DD}$ | Voltage on AVDD pin with respect to ground $V_{SS}$ (128-pinpackage) | -0.3 | 4.0 | V |
| $AV_{REF}$ | Voltage on AVREF pin with respect to ground $V_{SS}$ (128-pin package) | -0.3 | 4.0 | V |
| $AV_{REF_AVDD}$ | Voltage on AVREF_AVDD pin with respect to Ground $V_{SS}$ (80-pin package) | -0.3 | 4.0 | V |
| $V_{IN}$ | Voltage on 5V tolerant pins with respect to ground $V_{SS}$ | -0.3 | 5.5 | V |
| | Voltage on any other pin with respect to ground $V_{SS}$ | -0.3 | 4.0 | V |
| $I_{OV}$ | Input current on any pin during overload condition | -10 | +10 | mA |
| $I_{TDV}$ | Absolute sum of all input currents during overload condition | | \|200\| | mA |
| $T_{ST}$ | Storage Temperature | -55 | +150 | °C |
| $T_J$ | Junction Temperature | | +125 | °C |
| ESD | ESD Susceptibility (Human Body Model) | 2000 | | V |

*Note: Stresses exceeding above listed recommended "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. During overload conditions ($V_{IN}>V_{DDQ}$ or $V_{IN}<V_{SSQ}$) the voltage on pins with respect to ground ($V_{SSQ}$) must not exceed the recommended values.*

Obsolete Product(s) - Obsolete Product(s)

## 6.2 Operating conditions

**Table 5. Operating conditions**

| Symbol | Parameter | Test Conditions | Value | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| $V_{DD}$ | Digital CPU supply voltage | | 1.65 | 2.0 | V |
| $V_{DDQ}$ | Digital I/O supply voltage | | 2.7 | 3.6 | V |
| $V_{BATT}$(1) | SRAM backup and RTC supply voltage | | 2.5 | 3.6 | V |
| $AV_{DD}$ | Analog ADC supply voltage (128-pin package) | | 2.7 | 3.6 | V |
| $AV_{REF}$ | Analog ADC reference voltage (128-pin package) | | 1.0 | 3.6 | V |
| $AV_{REF_AVDD}$ | Combined analog ADC reference and ADC supply voltage (80-pin package) | | 2.7 | 3.6 | V |
| $T_A$ | Ambient temperature under bias | | -40 | +85 | C |

*Notes: 1 The $V_{BATT}$ pin should be connected to $V_{DDQ}$ if no battery is installed*

## 6.3 LVD electrical characteristics

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

**Table 6. LVD Electrical Characteristics**

| Symbol | Parameter | Test Conditions | Value | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min | Typ | Max | |
| $V_{DD_LVD}$ | $V_{DD}$ LVD Threshold | | 1.35 | 1.4 | 1.45 | V |
| $V_{DDQ_LVD}$ | $V_{DDQ}$ LVD Threshold | (1) | 2.35 | 2.4 | 2.45 | V |
| | | (2) | 2.65 | 2.7 | 2.75 | |
| $V_{DD_BRN}$ | $V_{DD}$ Brown Out Warning Threshold | | 1.6 | 1.65 | 1.7 | V |
| $V_{DDQ_BRN}$ | $V_{DDQ}$ Brown Out Warning Threshold | (1) | 2.6 | 2.65 | 2.7 | V |
| | | (2) | 2.9 | 2.95 | 3.0 | |

*Notes: 1 For VDDQ I/O voltage operating at 2.7 - 3.3V.*

*2 For VDDQ I/O voltage operating at 3.0 - 3.6V.*

*3 Selection of $V_{DDQ}$ operation range is made using configuration software from ST, or IDE from 3rd parties. The default condition is $V_{DDQ}$=2.7V - 3.3V.*

Obsolete Product(s) - Obsolete Product(s)

## 6.4 DC electrical characteristics

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

**Table 7. DC Electrical Characteristics**

| Symbol | Parameter | Test Conditions | Value | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min | Typ | Max | |
| $V_{IH}$ | Input High Level | General inputs | 2.0 | | (1) | V |
| | | RESET and TCK inputs | $0.8V_{DDQ}$ | | | |
| $V_{IL}$ | Input Low Level | General inputs | | | 0.8 | |
| | | RESET and TCK inputs | | | $0.2V_{DDQ}$ | |
| $V_{HYS}$ | Input Hysteresis Schmitt Trigger | General inputs | 0.4 | | | V |
| $V_{OH}$ | Output High Level High current pins | I/O ports 3 and 6: Push-Pull, $I_{OH}$ = 8mA | $V_{DDQ}$-0.7 | | | V |
| | Output High Level Standard current pins | I/O ports 0,1,2,4,5,7,8,9: Push-Pull, $I_{OH}$ = 4mA | $V_{DDQ}$-0.7 | | | |
| $V_{OL}$ | Output Low Level High current pins | I/O ports 3 and 6: Push-Pull, $I_{OL}$ = 8mA | | | 0.4 | V |
| | Output Low Level Standard current pins | I/O ports 0,1,2,4,5,7,8,9: Push-Pull, $I_{OL}$ = 4mA | | | 0.4 | |

*Notes: 1 Input pins are 5V tolerant, max input voltage is 5.5V*

Obsolete Product(s) - Obsolete Product(s)

## 6.5 AC electrical characteristics

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

**Table 8. AC electrical characteristics**

| Symbol | Parameter | Test Conditions | | Value | | | Unit |
|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | |
| $I_{DDRUN}$ | Run Mode Current | All peripherals on | CPU_CLK = 96MHz [1] [5] | | 1.7 | 2.3 | mA/ MHz |
| | | All peripherals off | | | 1.3 | 1.6 | |
| $I_{IDLE}$ | Idle Mode Current | All peripherals on [2] [5] | | | 1.14 | 1.7 | mA/ MHz |
| | | All peripherals off [3] [5] | | | 0.45 | 0.75 | mA/ MHz |
| $I_{SLEEP}$ | Sleep Mode Current | LVD On [4] [5] | | | 55 | 825 | µA |
| | | LVD Off [4] [5] | | | 50 | 820 | µA |
| $I_{RTC_STBY}$ | RTC Standby Current | Measured on VBATT pin | | | 0.3 | 0.9 | µA |
| $I_{SRAM_STBY}$ | SRAM Standby Current | Measured on VBATT pin | | | 5 | 85 | µA |

*Notes: 1 ARM core and peripherals active with all clocks on. Power can be conserved by turning off clocks to peripherals which are not required.*

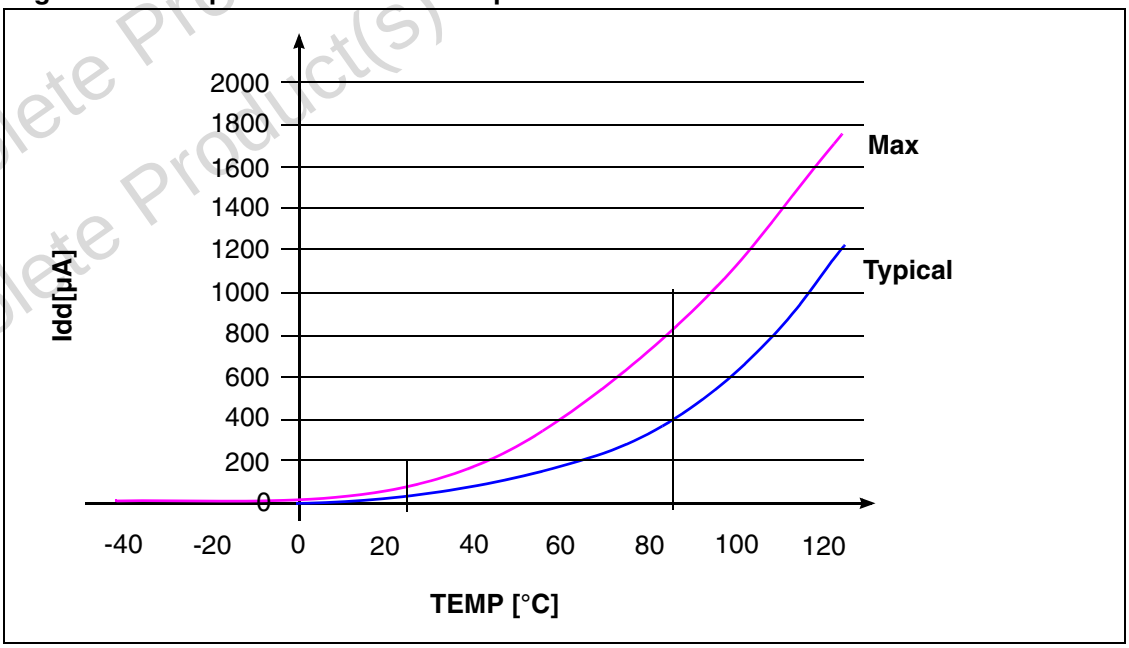
*2 ARM core stopped and all peripheral clocks active.*

*3 ARM core stopped and all peripheral clocks stopped.*

*4 ARM core and all peripheral clocks stopped (with exception of RTC).*

*5 Current measured on the $V_{DD}$ pins. $V_{DDQ}$ current is not included.*

**Figure 10. Sleep Mode current vs temperature**




Obsolete Product(s) - Obsolete Product(s)

**Table 9. AC electrical characteristics**

| Symbol | Parameter | Test Conditions | Value | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min | Typ | Max | |
| $f_{MSTR}$ | CCU Master Clk Output | | 32.768 | | 96,000 | kHz |
| $f_{CPUCLK}$ | CPU Core Frequency | Executing from SRAM | | | 96 | MHz |
| | | Executing from Flash | | | 96 | MHz |
| $f_{PCLK}$ | Peripheral Clock for APB | | | | 48 | MHz |
| $f_{HCLK}$ | Peripheral Clock for AHB | | | | 96 | MHz |
| $f_{OSC}$ | Clock Input | | 4 | | 25 | MHz |
| $f_{FMICLK}$ | FMI Flash Bus clock (internal clock) | | | | 96 | MHz |
| $f_{BCLK}$ | External Memory Bus clock (internal clock) | | | | 66 | MHz |
| $f_{RTC}$ | RTC Clock | | 32.768 | | | kHz |
| $f_{EMAC}$ | EMAC PHY Clock | | 25 | | | MHz |
| $f_{USB}$ | USB Clock | | 48 | | | MHz |

## 6.6 RESET_INn and power-on-reset characteristics

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

**Table 10. RESET_INn and Power-On-Reset Characteristics**

| Symbol | Parameter | Test Conditions | Value | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min | Typ | Max | |
| $t_{RINMIN}$ | RESET_INn Valid Active Low | | 100 | | | ns |
| $t_{POR}$ | Power-On-Reset Condition duration | $V_{DDQ}$,$V_{DD}$ ramp time is less than 10ms | 10 | | | ms |
| $t_{RSO}$ | RESET_OUT Duration (Watchdog reset) | | one PCLK | | | ns |

Obsolete Product(s) - Obsolete Product(s)

## 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 |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## 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 |
| $\Delta t_{JITTER}$ | PLL Jitter (peak to peak) | | | 0.1 | 0.2 | ns |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## 6.10 Flash memory characteristics

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

**Table 15. Flash memory program/erase characteristics**

| Parameter | | Test Conditions | Value | | | Unit |
|---|---|---|---|---|---|---|
| | | | Typ[2] | Typ after 100K W/E cycles[2] | Max | |
| Bank erase | Primary Bank (512 Kbytes)[1] | | 8 | 9 | 11.5 | s |
| | Primary Bank (256 Kbytes)[1] | | 4 | 4.5 | 6 | s |
| | Secondary Bank (32 Kbytes) | | 700 | 750 | 950 | ms |
| Sector erase | Of Primary Bank (64 Kbytes) | | 1300 | 1400 | 1800 | ms |
| | Of Secondary Bank (8 Kbytes) | | 300 | 320 | 450 | ms |
| Bank program | Primary Bank (512 Kbytes)[1] | | 3700 | 4700 | 5100 | ms |
| | Primary Bank (256 Kbytes)[1] | | 1900 | 2000 | 2550 | ms |
| | Secondary Bank (32 Kbytes) | | 250 | 260 | 320 | ms |
| Sector program | Of Primary Bank (64 Kbytes) | | 500 | 520 | 640 | ms |
| | Of Secondary Bank (8 Kbytes) | | 60 | 62 | 80 | ms |
| Word program | | | 8 | 9 | 11 | µs |
| Sector erase timeout | | | | | | µs |

*Notes: 1 STR91xFx44 devices have 512 Kbytes primary Flash, STR91xFx32 devices have 256 Kbytes primary Flash*

2 $V_{DD}$ = 1.8V, $V_{DDQ}$ = 3.3V, $T_A$ = 25°C.

3 Flash read access for synchronous addresses is 96 MHz maximum.

4 Flash read access for asynchronous accesses requires 2 wait states when FMI clock is above 66 MHz. See STR91xF Flash Programming Manual for more information.

**Table 16. Flash memory endurance**

| Parameter | Test Conditions | Value | | | Unit |
|---|---|---|---|---|---|
| | | Min | Typ | Max | |
| Program/erase cycles | Per word | 100K | | | cycles |
| Data retention | | 20 | | | years |

Obsolete Product(s) - Obsolete Product(s)

## 6.11 EMC characteristics

Susceptibility tests are performed on a sample basis during product characterization.

### 6.11.1 Functional EMS (Electro Magnetic Susceptibility)

Based on a simple running application on the product (toggling 2 LEDs through I/O ports), the product is stressed by two electro magnetic events until a failure occurs (indicated by the LEDs).

- **ESD**: Electro-Static Discharge (positive and negative) is applied on all pins of the device until a functional disturbance occurs. This test conforms with the IEC 1000-4-2 standard.
- **FTB**: A Burst of Fast Transient voltage (positive and negative) is applied to $V_{DD}$, $V_{DDQ}$ and $V_{SS}$ through a 100pF capacitor, until a functional disturbance occurs. This test conforms with the IEC 1000-4-4 standard.

A device reset allows normal operations to be resumed.

**Table 17. EMS data**

| Symbol | Parameter | Conditions | Neg. | Pos. | Unit |
|---|---|---|---|---|---|
| $V_{FESD}$ | Voltage limits to be applied on any I/O pin to induce a functional disturbance | $V_{DD}$=1.8V, $V_{DDQ}$=3.3V, $T_A$=+25°C, $f_{OSC}/f_{CPUCLK}$ =4 MHz/96MHz PLL | -1(1) | >2(1) | kV |
| $V_{FFTB}$ | Fast transient voltage burst limits to be applied through 100pF on $V_{DD}$ and $V_{DDQ}$ pins to induce a functional disturbance | $V_{DD}$=1.8V, $V_{DDQ}$=3.3V, $T_A$=+25°C, $f_{OSC}/f_{CPUCLK}$ =4 MHz/96 MHz PLL conforms to IEC 1000-4-4 | -4(1) | 4(1) | |

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

### 6.11.2 Electro Magnetic Interference (EMI)

Based on a simple application running on the product (toggling 2 LEDs through the I/O ports), the product is monitored in terms of emission. This emission test is in line with the norm SAE J 1752/3 which specifies the board and the loading of each pin.

**Table 18. EMI data**

| Symbol | Parameter | Conditions | Monitored Frequency Band | Max vs. [$f_{OSC}/f_{CPUCLK}$] | Unit |
|---|---|---|---|---|---|
| | | | | 4 MHz/ 96 MHz | |
| $S_{EMI}$ | Peak level | $V_{DDQ}$=3.3V, $V_{DD}$=1.8V, $T_A$=+25°C, LQFP128 package conforming to SAE J 1752/3 | 0.1MHz to 30 MHz | 10 | dBµV |
| | | | 30 MHz to 130 MHz | 10 | |
| | | | 130 MHz to 1GHz | 22 | |
| | | | SAE EMI Level | 4 | - |

**Notes**:

1. Data based on characterization results, not tested in production.
2. BGA and LQFP devices have similar EMI characteristics.

Obsolete Product(s) - Obsolete Product(s)

### 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

Obsolete Product(s) - Obsolete Product(s)

- Critical Data corruption (control registers...)

Prequalification trials:

Most of the common failures (unexpected reset and program counter corruption) can be reproduced by manually forcing a low state on the RESET pin or the Oscillator pins for 1 second.

To complete these trials, ESD stress can be applied directly on the device, over the range of specification values. When unexpected behavior is detected, the software can be hardened to prevent unrecoverable errors occurring (see application note AN1015).

### 6.11.7 Electrical Sensitivities

| Symbol | Parameter | Conditions | Class [1)] |
|---|---|---|---|
| LU | Static latch-up class | $T_A$=+25°C | A |
| DLU | Dynamic latch-up class | $V_{DDQ}$=3.3V, $V_{DD}$=1.8V, $f_{OSC}/f_{CPUCLK}$=4 MHz/96 MHz | A |

**Notes:**

1. Class description: A Class is an STMicroelectronics internal specification. All its limits are higher than the JEDEC specifications, that means when a device belongs to Class A it exceeds the JEDEC standard. B Class strictly covers all the JEDEC criteria (international standard).

## 6.12 External memory bus timings

$V_{DDQ}$ = 2.7 - 3.6V, $V_{DD}$ = 1.65 - 2V, $T_A$ = -40 / 85 °C, $C_L$= 30 pF unless otherwise specified.

**Table 20. EMI Bus Clock Period**

| Symbol | Parameter | Value |
|---|---|---|
| $t_{BCLK}$ | EMI Bus Clock period | 1 /($f_{HCLK}$ x EMI_ratio) |

*Notes: 1 The internal EMI Bus clock signal is available externally only on LFBGA144 packages (ball M8), and not available on LQFP packages.*

*2 EMI_ratio =1/ 2 by default (can be programmed to be 1 by setting the proper bits in the SCU_CLKCNTR register)*

**Table 21. EMI read operation**

| Symbol | Parameter | Value | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| $t_{RCR}$ | Read to CSn inactive | -1 | +1 | ns |
| $t_{RP}$ | Read Pulse Width | (WSTRD-WSTOEN+1) x $t_{BCLK}$ - 1 | (WSTRD-WSTOEN+1) x $t_{BCLK}$ + 1 | ns |
| $t_{RDS}$ | Read Data Setup Time | 4 | | ns |
| $t_{RDH}$ | Read Data Hold Time | 0 | | ns |
| $t_{RAS}$ | Read Address Setup Time | (WSTOEN) x $t_{BCLK}$- 1 | (WSTOEN) x $t_{BCLK}$ + 1 | ns |
| $t_{RAH}$ | Read Address Hold Time | 0 | | ns |
| $t_{AW}$ | ALE pulse width | (ALE_LENGTH) x $t_{BCLK}$ - 1 | (ALE_LENGTH) x $t_{BCLK}$ + 1 | ns |

Obsolete Product(s) - Obsolete Product(s)

| Symbol | Parameter | Value | | Unit |
|---|---|---|---|---|
| | | Min | Max | |
| $t_{AAH}$ | Address to ALE hold time | $t_{BCLK/2}$ - 1 | $t_{BCLK/2}$ + 1 | ns |
| $t_{AAS}$ | Address to ALE setup time | (ALE_LENGTH) x $t_{BCLK}$ - 1 | | ns |

*Notes: 1 ALE_LENGTH = 1 by default (can be programmed to be 2 by setting the bits In the SCU_SCR0 register)*

*2 WSTRD = 1Fh by default (RD wait state time = WSTRD x $t_{BCLK}$, WSTRD can be programmed in the EMI_RCRx Register)*

*3 WSTOEN = 1 by default (RD assertion delay from chip select. WSTOEN can be programmed in the EMI_OECRx Register)*

**Figure 11. Non-mux bus (8-bit) read timings**

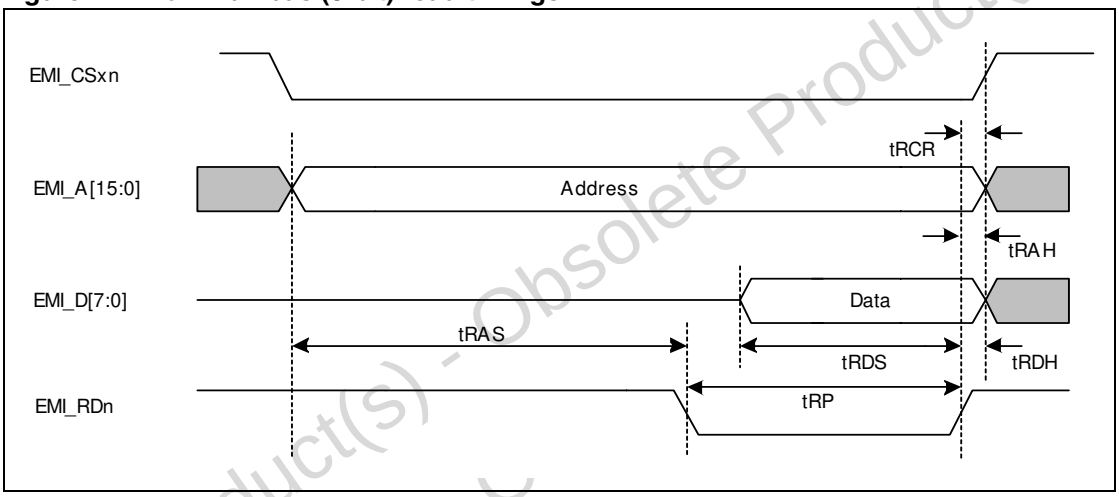
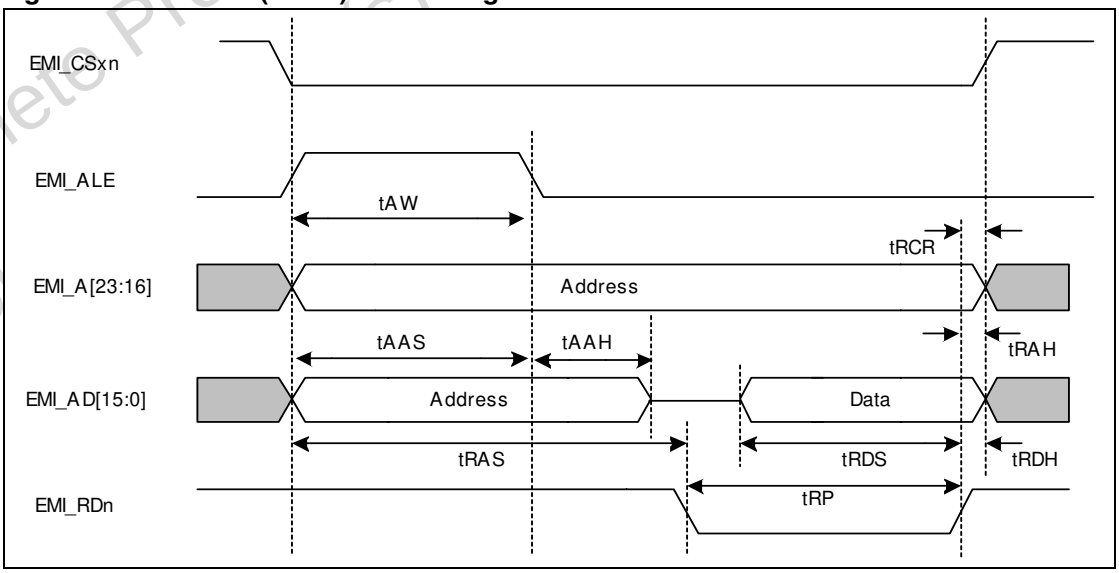



**Figure 12. Mux bus (16-bit) read timings**




Obsolete Product(s) - Obsolete Product(s)

**Table 22. EMI write operation**

| Symbol | Parameter | Test Conditions | Value | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| $t_{WCR}$ | WRn to CSn inactive | | $(t_{BCLK}/2)$ - 1 | $(t_{BCLK}/2)$ +1 | ns |
| $t_{WP}$ | Write Pulse Width | | (WSTWR-WSTWEN + 1) x $t_{BCLK}$ - 1 | (WSTWR-WSTWEN + 1) x $t_{BCLK}$ + 1 | ns |
| $t_{WDS}$ | Write Data Setup Time (non-mux mode) | | (WSTWEN + 1/2) x $t_{BCLK}$ | | ns |
| | Write Data Setup Time (mux mode ) | ALE length=1 WSTWEN>2 | (WSTWEN - 1.5) x $t_{BCLK}$ | | |
| | | ALE length=2 WSTWEN>3 | (WSTWEN - 2.5) x $t_{BCLK}$ | | |
| $t_{WDH}$ | Write Data Hold Time | | $(t_{BCLK}/2)$ - 1 | $(t_{BCLK}/2)$ + 1 | ns |
| $t_{WAS}$ | Write Address Setup Time | | (WSTWEN + 1/2) x $t_{BCLK}$ -1 [4] | (WSTWEN + 1/2) x $t_{BCLK}$ +1 [4] | ns |
| $t_{WAH}$ | Write Address Hold Time | | $t_{BCLK}/2$ | | ns |
| $t_{AW}$ | ALE pulse width | | (ALE_LENGTH x $t_{BCLK}$ )-1 | (ALE_LENGTH x $t_{BCLK}$ ) +1 | ns |
| $t_{AAH}$ | Address to ALE hold time | | $(t_{BCLK}/2)$ -1 | $(t_{BCLK}/2)$ + 1 | ns |
| $t_{AAS}$ | Address to ALE setup time | | (ALE_LENGTH x $t_{BCLK}$)- 1 | | ns |

*Notes: 1 ALE_LENGTH = 1 by default (can be programmed to be 2 by setting the bits In the SCU_SCR0 register)*

*2 WSTWR =1Fh by default (WR wait state time = WSTWR x $t_{BCLK}$, WSTWR can be programmed in the EMI_WCRx Register)*

*3 WSTWEN= 0 by default (WR assertion delay from chip select. WSTWEN can be programmed in the EMI_WECRx Register)*

*4 When the CPU executes a 16-bit write to a x8 EMI bus, the second write cycle's address setup time is defined as $t_{WAS}$=(WSTWEN - ½) x $t_{BCLK}$*

Obsolete Product(s) - Obsolete Product(s)

**Figure 13. Non-Mux Bus (8-bit) write timings**

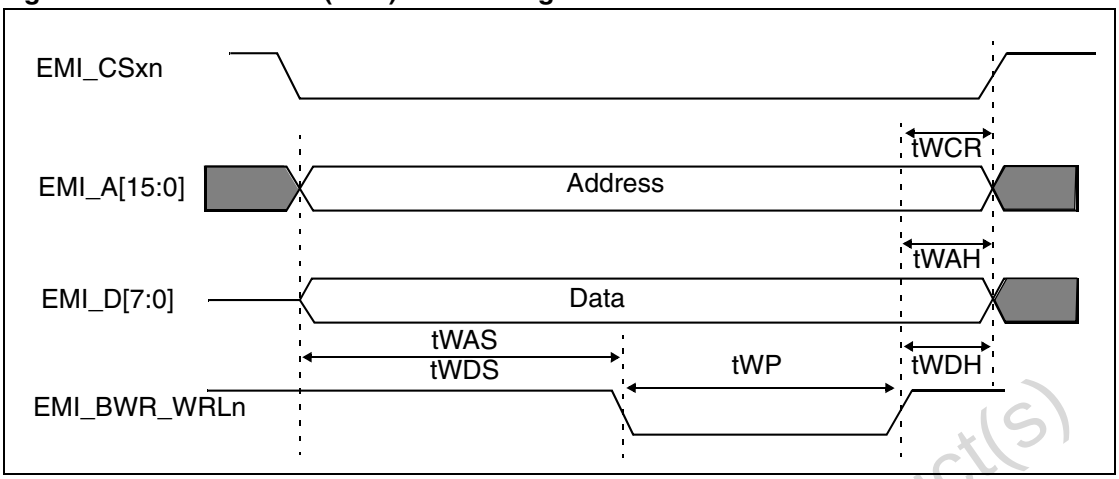
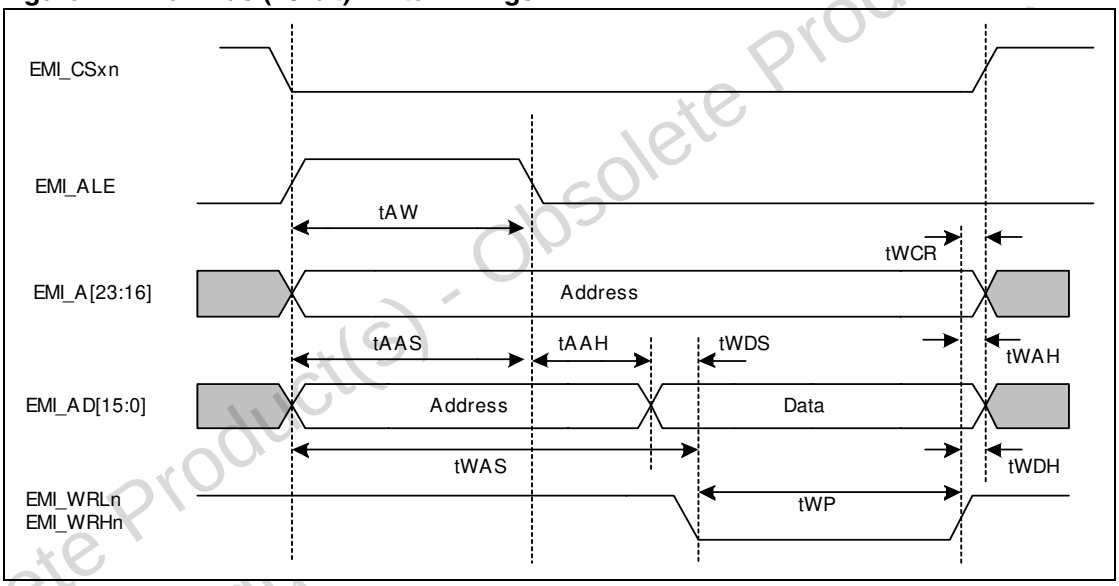



**Figure 14. Mux Bus (16-bit) Write Timings**




Obsolete Product(s) - Obsolete Product(s)

## 6.13 ADC electrical characteristics

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

**Table 23. ADC Electrical Characteristics**

| Symbol | Parameter | Test Conditions | Value | | | Unit |
|---|---|---|---|---|---|---|
| | | | Min | Typ | Max | |
| $V_{AIN}$ | Input Voltage Range | | 0 | | $AV_{REF}$ | V |
| RES | Resolution | | | | 10 | Bits |
| $N_{CH}$ | Number of Input Channels | | | | 8 | N |
| $f_{ADC}$ | ADC Clock Frequency | | | | 25 | MHz |
| | POR bit set to Standby mode | | | | 500 | ms |
| | Conversion Time | $f_{ADC}$ = 25 MHz | | 0.7 | | µs |
| | Throughput Rate | $f_{ADC}$ = 25 MHz | | 1400 | | ksps |
| $C_{IN}$ | Input Capacitance | | | 5 | | pF |
| $E_D$ | Differential Non-Linearity | [1] [2] | | 1 | 3 | LSB[3] |
| $E_L$ | Integral Non-Linearity | [1] | | 3 | 6 | LSB |
| $E_O$ | Offset Error | [1] | | 3 | 6 | LSB |
| $E_G$ | Gain Error | [1] | | 0.5 | 2 | LSB |
| $E_T$ | Absolute Error | [1] | | 4 | 6 | LSB |
| $I_{ADC}$ | Power Consumption | | | 4.6 | 6 | mA |

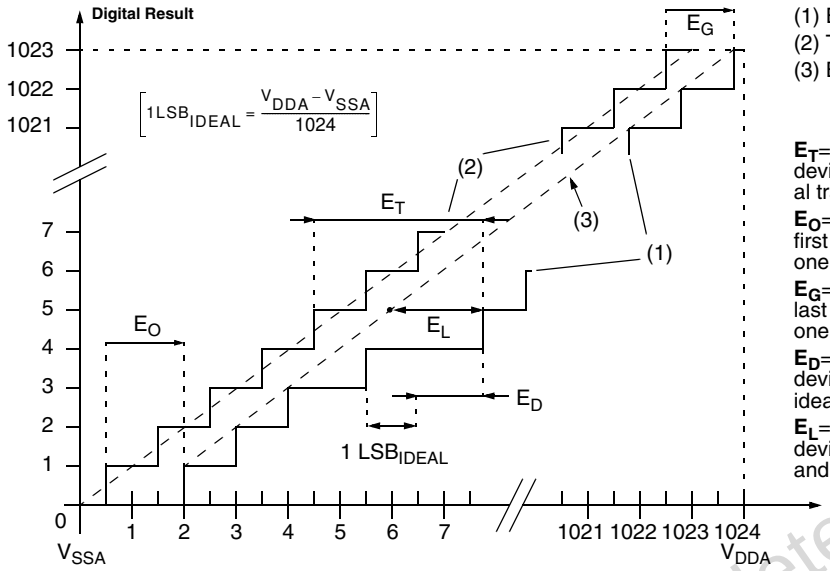
*Notes: 1 Conditions: $A_{VSS}$ = 0 V, $AV_{DD}$ = 3.3 V $f_{ADC}$ = 25 MHz.*

*2 The A/D is monotonic, there are no missing codes.*

*3 1 LSB = ($V_{DDA}$ - $V_{SSA}$)/1024*

Obsolete Product(s) - Obsolete Product(s)

**Figure 15. ADC conversion characteristics**




(1) Example of an actual transfer curve
(2) The ideal transfer curve
(3) End point correlation line

**$E_T$**=Total Unadjusted Error: maximum deviation between the actual and the ideal transfer curves.

**$E_O$**=Offset Error: deviation between the first actual transition and the first ideal one.

**$E_G$**=Gain Error: deviation between the last ideal transition and the last actual one.

**$E_D$**=Differential Linearity Error: maximum deviation between actual steps and the ideal one.

**$E_L$**=Integral Linearity Error: maximum deviation between any actual transition and the end point correlation line.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## 6.14 Communication interface electrical characteristics

### 6.14.1 10/100 Ethernet MAC electrical characteristics

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

#### Ethernet MII Interface Timings

**Figure 16. MII_RX_CLK and MII_TX_CLK timing diagram**

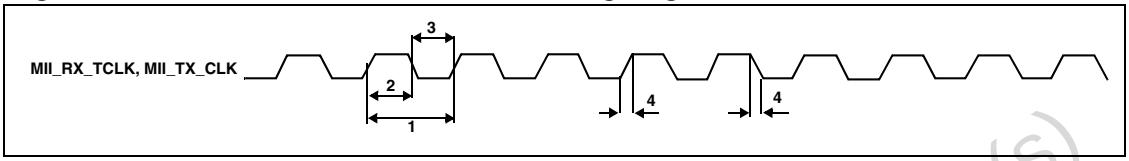



**Table 24. MII_RX_CLK and MII_TX_CLK timing table**

| Symbol | Parameter | Symbol | Value | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| 1 | Cycle time | $t_c$(CLK) | 40 | | ns |
| 2 | Pulse duration HIGH | $t_{HIGH}$(CLK) | 40% | 60% | |
| 3 | Pulse duration LOW | $t_{LOW}$(CLK) | 40% | 60% | |
| 4 | Transition time | $t_t$(CLK) | | 1 | ns |

**Figure 17. MDC timing diagram**

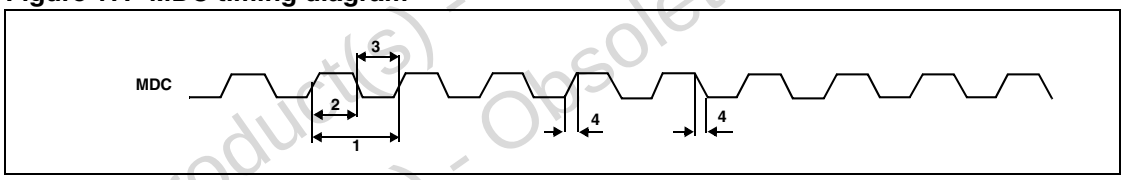
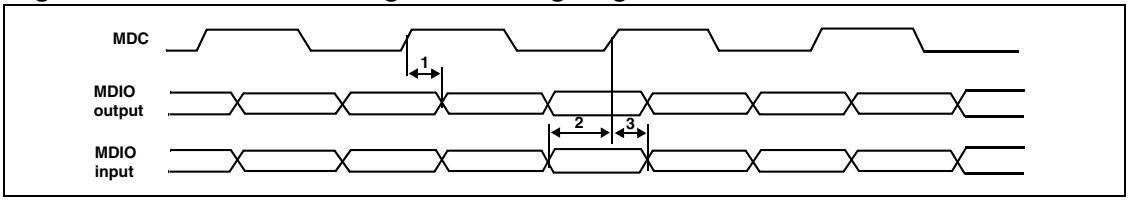



**Table 25. MDC timing table**

| Symbol | Parameter | Symbol | Value | | Unit |
|---|---|---|---|---|---|
| | | | Min | Max | |
| 1 | Cycle time | $t_c$(MDC) | 266 | | ns |
| 2 | Pulse duration HIGH | $t_{HIGH}$(MDC) | 40% | 60% | |
| 3 | Pulse duration LOW | $t_{LOW}$(MDC) | 40% | 60% | |
| 4 | Transition time | $t_t$(MDC) | | 1 | ns |

#### Ethernet MII management timings

**Figure 18. Ethernet MII management timing diagram**




Obsolete Product(s) - Obsolete Product(s)

**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**

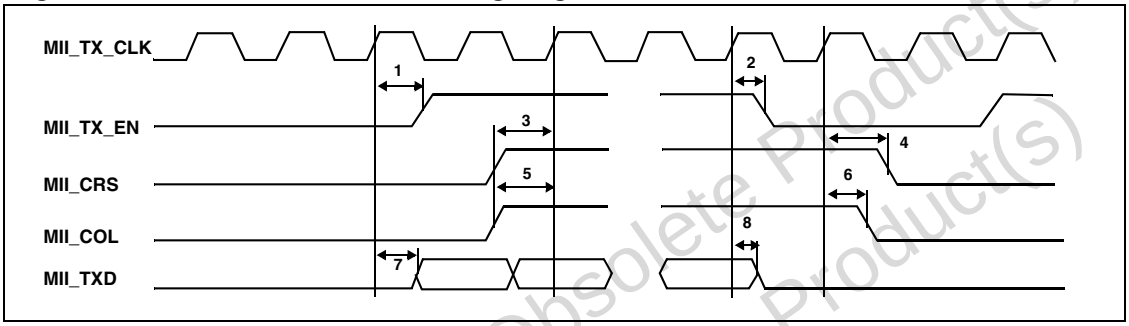



**Table 27. Ethernet MII transmit timing table**

| 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 |

Obsolete Product(s) - Obsolete Product(s)

### Ethernet MII Receive timings

**Figure 20. Ethernet MII receive timing diagram**

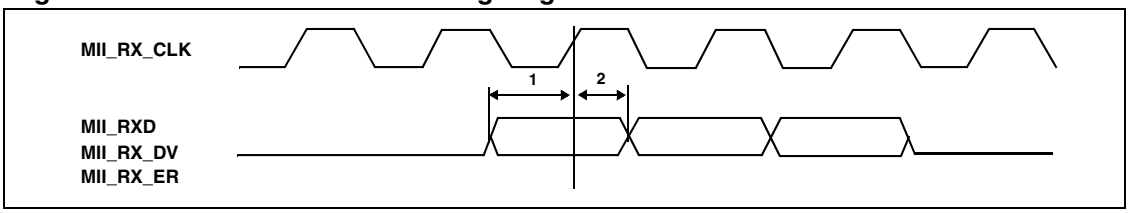



**Figure 21. Ethernet MII receive timing table**

| 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

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

### 6.14.4 $I^2C$ electrical characteristics

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

**Table 28. $I^2C$ Electrical Characteristics**

| Symbol | Parameter | Standard $I^2C$ | | Fast $I^2C$ | | Unit |
|---|---|---|---|---|---|---|
| | | Min | Max | Min | Max | |
| $t_{BUF}$ | Bus free time between a STOP and START condition | 4.7 | | 1.3 | | ms |
| $t_{HD:STA}$ | Hold time START condition. After this period, the first clock pulse is generated | 4.0 | | 0.6 | | µs |
| $t_{LOW}$ | LOW period of the SCL clock | 4.7 | | 1.3 | | µs |
| $t_{HIGH}$ | HIGH period of the SCL clock | 4.0 | | 0.6 | | µs |
| $t_{SU:STA}$ | Set-up time for a repeated START condition | 4.7 | | 0.6 | | µs |
| $t_{HD:DAT}$ | Data hold time | 0 | | 0 | | ns |
| $t_{SU:DAT}$ | Data set-up time | 250 | | 100 | | ns |
| $t_R$ | Rise time of both SDA and SCL signals | | 1000 | $20+0.1C_b$ | 300 | ns |
| $t_F$ | Fall time of both SDA and SCL signals | | 300 | $20+0.1C_b$ | 300 | ns |
| $t_{SU:STO}$ | Set-up time for STOP condition | 4.0 | | 0.6 | | µs |
| $C_b$ | Capacitive load for each bus line | | 400 | | 400 | pF |

*Notes: 1 The device must internally provide a hold time of at least 300 ns for the SDA signal in order to bridge the undefined region of the falling edge of SCL*

*2 The maximum hold time of the START condition has only to be met if the interface does not stretch the low period of SCL signal*

*3 $C_b$ = total capacitance of one bus line in pF*

Obsolete Product(s) Obsolete Product(s) Obsolete Product(s)

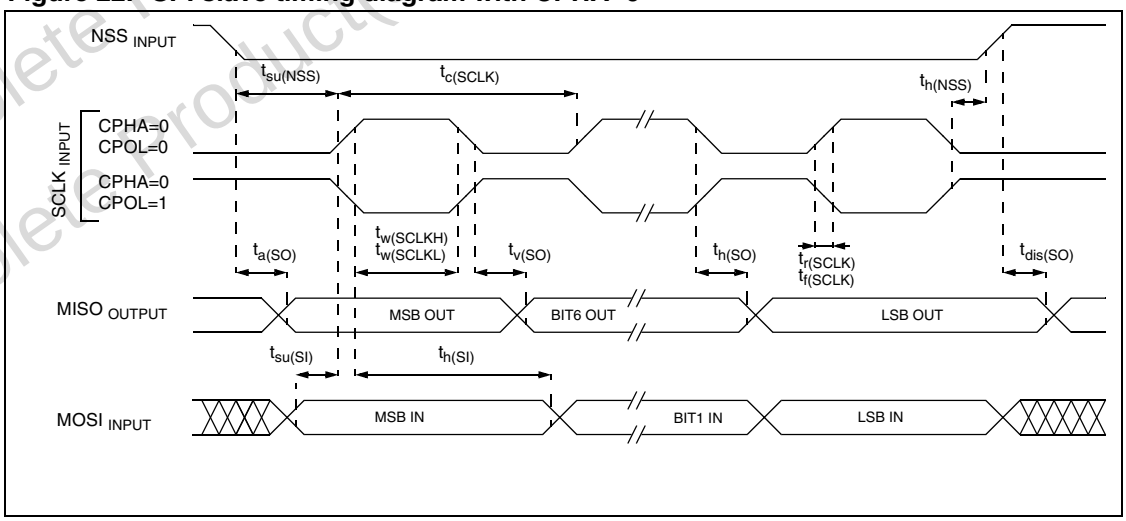
### 6.14.5 SPI electrical characteristics

$V_{DDQ}$ = 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 | |
| $f_{SCLK}$<br>$1/t_{c(SCLK)}$ | SPI clock frequency | Master | | 24 | MHz |
| | | Slave | | 4 | |
| $t_{r(SCLK)}$<br>$t_{f(SCLK)}$ | 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 | | |
| $t_{w(SCLKH)}$<br>$t_{w(SCLKL)}$ | SCLK high and low time | Master<br>Slave | 1 | | |
| $t_{su(MI)}$<br>$t_{su(SI)}$ | Data input setup time | Master<br>Slave | 5 | | |
| $t_{h(MI)}$<br>$t_{h(SI)}$ | Data input hold time | Master<br>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**




Obsolete Product(s) - Obsolete Product(s)

**Figure 23. SPI slave timing diagram with CPHA=1**

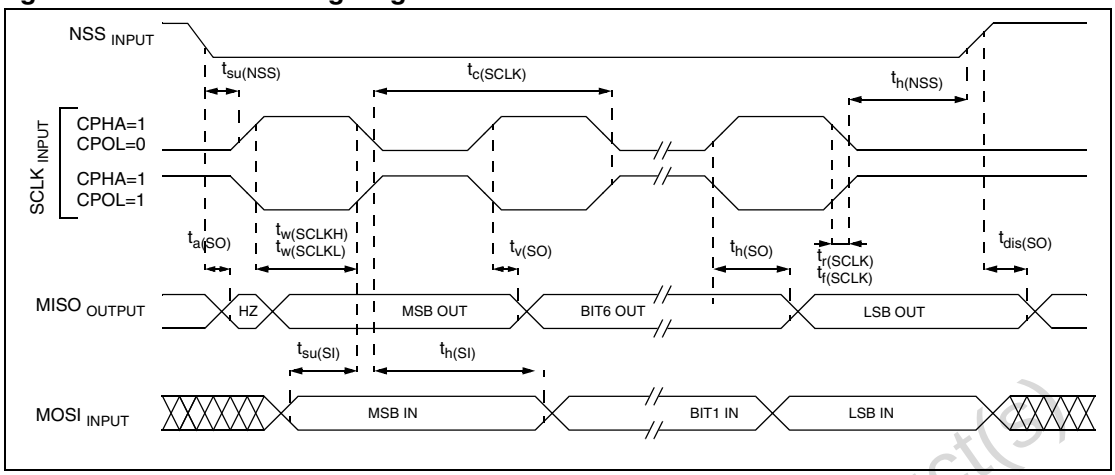
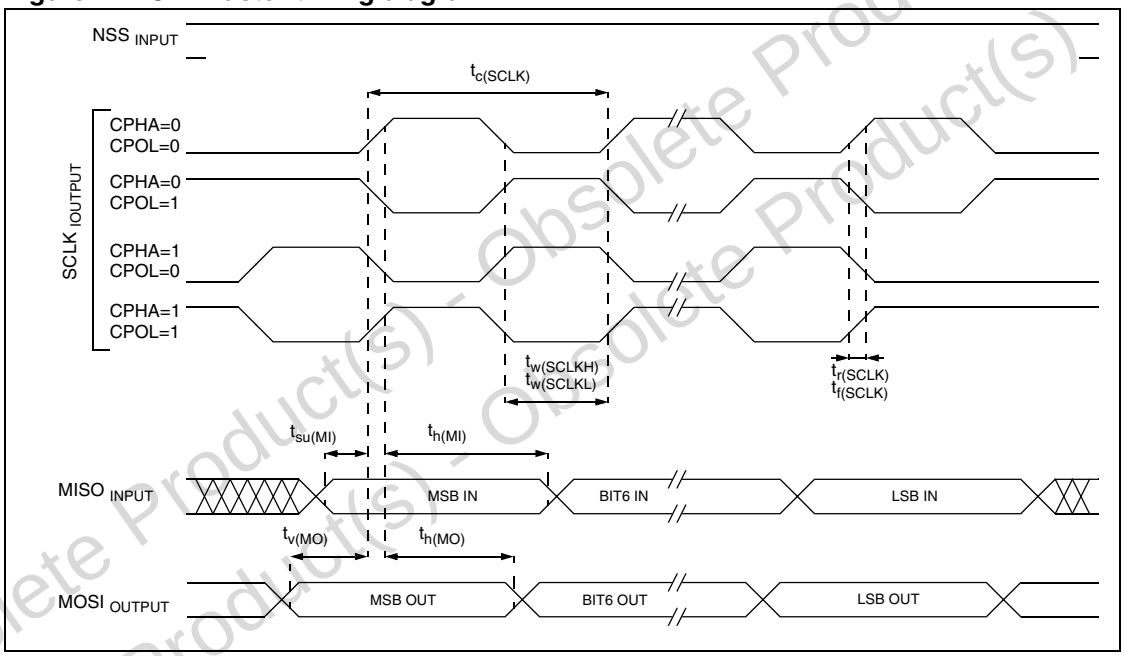


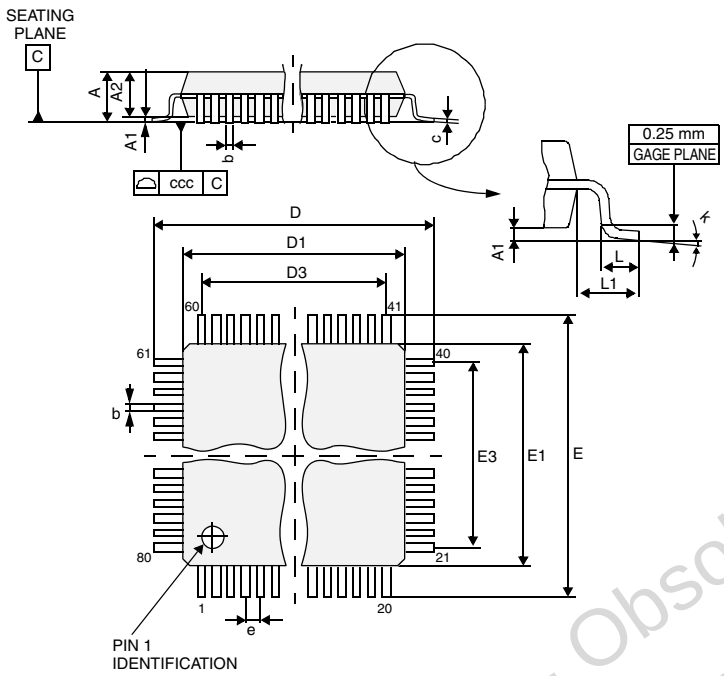

**Figure 24. SPI master timing diagram**




Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

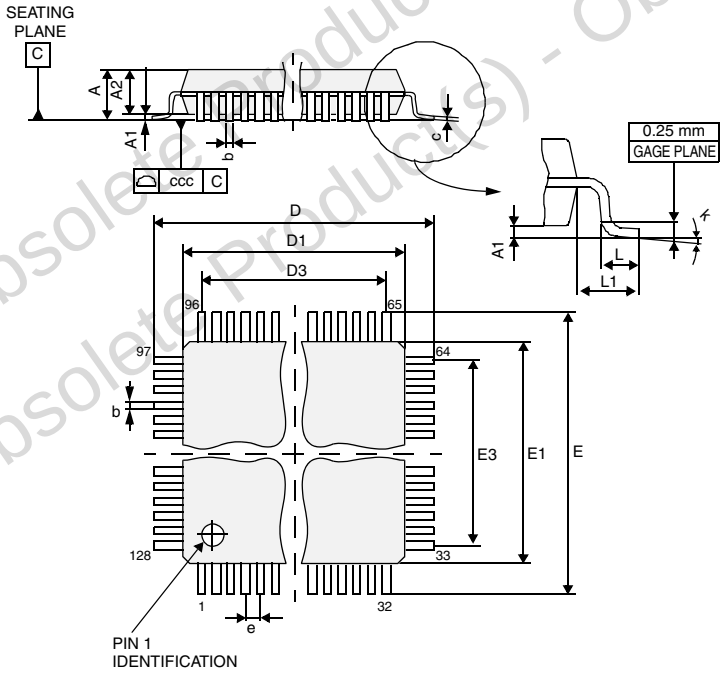
# 7 Package mechanical data

**Figure 25. 80-Pin Low Profile Quad Flat Package**




| Dim. | mm | | | inches | | |
|---|---|---|---|---|---|---|
| | Min | Typ | Max | Min | Typ | Max |
| A | | | 1.60 | | | 0.063 |
| A1 | 0.05 | | 0.15 | 0.002 | | 0.006 |
| A2 | 1.35 | 1.40 | 1.45 | 0.053 | 0.055 | 0.057 |
| b | 0.17 | 0.22 | 0.27 | 0.007 | 0.009 | 0.011 |
| c | 0.09 | | 0.20 | 0.004 | | 0.008 |
| D | | 14.00 | | | 0.551 | |
| D1 | | 12.00 | | | 0.472 | |
| D2 | | 9.50 | | | 0.374 | |
| E | | 14.00 | | | 0.551 | |
| E1 | | 12.00 | | | 0.472 | |
| E2 | | 9.50 | | | 0.374 | |
| e | | 0.50 | | | 0.020 | |
| L | 0.45 | 0.60 | 0.75 | 0.018 | 0.024 | 0.030 |
| L1 | | 1.00 | | | 0.039 | |
| k | 0d | | 7d | 0d | | 7d |
| ddd | | | 0.08 | | | 0.003 |
| | Number of Pins | | | | | |
| N | 80 | | | | | |

**128-Pin Low Profile Quad Flat Package**




| Dim. | mm | | | inches | | |
|---|---|---|---|---|---|---|
| | Min | Typ | Max | Min | Typ | Max |
| A | | | 1.60 | | | 0.063 |
| A1 | 0.05 | | 0.15 | 0.002 | | 0.006 |
| A2 | 1.35 | 1.40 | 1.45 | 0.053 | 0.055 | 0.057 |
| b | 0.13 | 0.18 | 0.23 | 0.005 | 0.007 | 0.009 |
| c | 0.09 | | 0.20 | 0.004 | | 0.008 |
| D | 15.80 | 16.00 | 16.20 | 0.622 | 0.630 | 0.638 |
| D1 | 13.80 | 14.00 | 14.20 | 0.543 | 0.551 | 0.559 |
| D3 | | 12.40 | | | 0.488 | |
| E | 15.80 | 16.00 | 16.20 | 0.622 | 0.630 | 0.638 |
| E1 | 13.80 | 14.00 | 14.20 | 0.543 | 0.551 | 0.559 |
| E3 | | 12.40 | | | 0.488 | |
| e | | 0.40 | | | 0.016 | |
| L | 0.45 | 0.60 | 0.75 | 0.018 | 0.024 | 0.030 |
| L1 | | 1.00 | | | 0.039 | |
| k | 0d | 3.5d | 7d | 0d | 3.5d | 7d |
| ccc | | | 0.08 | | | 0.003 |
| | Number of Pins | | | | | |
| N | 128 | | | | | |

Obsolete Product(s) - Obsolete Product(s) - Obsolete Product(s)

## 7.1 Thermal characteristics

The average chip-junction temperature, $T_J$ must never exceed 125° C.

The average chip-junction temperature, $T_J$, in degrees Celsius, may be calculated using the following equation:

$$T_J = T_A + (P_D \times \Theta_{JA}) (1)$$

Where:

- $T_A$ is the Ambient Temperature in °C,
- $\Theta_{JA}$ is the Package Junction-to-Ambient Thermal Resistance, in ° C/W,
- $P_D$ is the sum of $P_{INT}$ and $P_{I/O}$ ($P_D = P_{INT} + P_{I/O}$),
- $P_{INT}$ is the product of $I_{DD}$ and $V_{DD}$, expressed in Watts. This is the Chip Internal Power.

$P_{I/O}$ represents the Power Dissipation on Input and Output Pins;

Most of the time for the applications $P_{I/O} < P_{INT}$ and may be neglected. On the other hand, $P_{I/O}$ may be significant if the device is configured to drive continuously external modules and/or memories. The worst case $P_{INT}$ of the STR91xF is 500mW ($I_{DD} \times V_{DD}$, or 250mA x 2.0V).

An approximate relationship between $P_D$ and $T_J$ (if $P_{I/O}$ is neglected) is given by:

$$P_D = K / (T_J + 273°C) (2)$$

Therefore (solving equations 1 and 2):

$$K = P_D \times (T_A + 273°C) + \Theta_{JA} \times P_D^2 (3)$$

where:

- K is a constant for the particular part, which may be determined from equation (3) by measuring $P_D$ (at equilibrium) for a known $T_A$. Using this value of K, the values of $P_D$ and $T_J$ may be obtained by solving equations (1) and (2) iteratively for any value of $T_A$.

**Table 30. Thermal characteristics**

| Symbol | Parameter | Value | Unit |
|---|---|---|---|
| $\Theta_{JA}$ | **Thermal Resistance Junction-Ambient** LQFP 80 - 12 x 12 mm / 0.5 mm pitch | 41.5 | °C/W |
| $\Theta_{JA}$ | **Thermal Resistance Junction-Ambient** LQFP128 - 14 x 14 mm / 0.4 mm pitch | 38 | °C/W |

Obsolete Product(s) - Obsolete Product(s)

# 8 Ordering information

**Table 31. Ordering information**

| Part Number | Flash KB | RAM KB | Major Peripherals | Package[1)] |
|---|---|---|---|---|
| STR910FM32X6 | 256+32 | 64 | CAN, 40 I/Os | LQFP80, 12x12 mm |
| STR910FW32X6 | 256+32 | 64 | CAN, EMI, 80 I/Os | LQFP128, 14x14 mm |
| STR911FM42X6 | 256+32 | 96 | USB, CAN, 40 I/Os | LQFP80, 12x12mm |
| STR911FM44X6 | 512+32 | 96 | | |
| STR912FW42X6 | 256+32 | 96 | Ethernet, USB, CAN, EMI, 80 I/Os | LQFP128 |
| STR912FW44X6 | 512+32 | 96 | | |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**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.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

# 9 Revision history

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

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

Obsolete Product(s) - Obsolete Product(s)

**Please Read Carefully:**

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

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

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

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

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

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

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

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

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

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

© 2007 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

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

**www.st.com**