Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

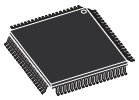
STR91xFAxxx

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

Datasheet - production data

Features

- 16/32-bit 96 MHz ARM9E based MCU
  - ARM966E-S™ RISC core: Harvard architecture, 5-stage pipeline, Tightly-Coupled Memories (SRAM and Flash)
  - STR91xFA 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 supported
  - Binary compatible with ARM7 code
- Dual burst Flash memories, 32-bits wide
  - 256 KB/512 KB/1 MB/2 MB main Flash
  - 32 KB/128 KB secondary Flash
  - Sequential Burst operation up to 96 MHz
  - 100 K min erase cycles, 20 yr min retention
- SRAM, 32-bits wide
  - 64K or 96K bytes, optional battery backup
- 9 programmable DMA channels
- Clock, reset, and supply management
  - Internal oscillator operating with external 4-25 MHz crystal
  - Internal PLL up to 96 MHz
  - Real-time clock provides calendar functions, tamper, and wake-up functions
  - Reset Supervisor monitors supply voltage, watchdog, wake-up unit, external reset
  - Brown-out monitor
  - Run, Idle, and Sleep Mode as low as 50 uA
- Vectored interrupt controller (VIC)
  - 32 IRQ vectors, 30 interrupt pins
  - Branch cache minimizes interrupt latency
- 8-channel, 10-bit A/D converter (ADC)
  - 0 to 3.6 V range, 0.7 usec conversion
- 10 Communication interfaces
  - 10/100 Ethernet MAC with DMA and MII
  - USB Full-speed (12 Mbps) slave device
  - CAN interface (2.0B Active)



LQFP80 12 x12mm

LQFP128 14 x 14mm

LFBGA144 10 x 10 mm

  - 3 16550-style UARTs with IrDA protocol
  - 2 Fast I^2C, 400 kHz
  - 2 channels for SPI, SSI™, or MICROWIRE™
- External Memory Interface (EMI)
  - 8- or 16-bit data, up to 24-bit addressing
  - Static Async modes for LQFP128
  - Additional burst synchronous modes for LFBGA144
- Up to 80 I/O pins (muxed with interfaces)
- 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)
- JTAG interface with boundary scan
- Embedded trace module (ARM® ETM9™)

Table 1. Device summary

Reference	Part number
STR91xFAx32	STR910FAM32, STR910FAW32, STR910FAZ32, STR912FAW32
STR91xFAx42	STR911FAM42, STR911FAW42, STR912FAW42, STR912FAZ42
STR91xFAx44	STR911FAM44 STR911FAW44 STR912FAW44, STR912FAZ44
STR91xFAx46	STR911FAM46, STR911FAW46, STR912FAW46, STR912FAZ46
STR91xFAx47	STR911FAM47, STR911FAW47, STR912FAW47, STR912FAZ47

Contents

1 Description 10

2 Device summary 11

3 Functional overview 12

3.1 System-in-a-package (SiP) 12

3.2 Package choice 12

3.3 ARM966E-S CPU core 12

3.4 Burst Flash memory interface 12

3.4.1 Pre-fetch queue (PFQ) 12

3.4.2 Branch cache (BC) 13

3.4.3 Management of literals 13

3.5 SRAM (64 Kbytes or 96 Kbytes) 15

3.5.1 Arbitration 15

3.5.2 Battery backup 15

3.6 DMA data movement 15

3.7 Non-volatile memories 16

3.7.1 Primary Flash memory 16

3.7.2 Secondary Flash memory 16

3.8 One-time-programmable (OTP) memory 17

3.8.1 Product ID and revision level 17

3.9 Vectored interrupt controller (VIC) 18

3.9.1 FIQ handling 18

3.9.2 IRQ handling 18

3.9.3 Interrupt sources 18

3.10 Clock control unit (CCU) 20

3.10.1 Master clock sources 20

3.10.2 Reference clock (RCLK) 21

3.10.3 AHB clock (HCLK) 21

3.10.4 APB clock (PCLK) 21

3.10.5 Flash memory interface clock (FMICLK) 22

3.10.6 UART and SSP clock (BRCLK) 22

3.10.7 External memory interface bus clock (BCLK) 22

3.10.8 USB interface clock . . . 22
3.10.9 Ethernet MAC clock . . . 22
3.10.10 External RTC calibration clock . . . 22
3.10.11 Operation example . . . 23
3.11 Flexible power management . . . 23
3.11.1 Run mode . . . 23
3.11.2 Idle mode . . . 24
3.11.3 Sleep mode . . . 24
3.12 Voltage supplies . . . 24
3.12.1 Independent A/D converter supply and reference voltage . . . 24
3.12.2 Battery supply . . . 25
3.13 System supervisor . . . 25
3.13.1 Supply voltage brownout . . . 25
3.13.2 Supply voltage dropout . . . 26
3.13.3 Watchdog timer . . . 26
3.13.4 External RESET_INn pin . . . 26
3.13.5 Power-up . . . 26
3.13.6 JTAG debug command . . . 26
3.13.7 Tamper detection . . . 27
3.14 Real-time clock (RTC) . . . 27
3.15 JTAG interface . . . 27
3.15.1 In-system-programming . . . 28
3.15.2 Boundary scan . . . 29
3.15.3 CPU debug . . . 29
3.15.4 JTAG security bit . . . 29
3.16 Embedded trace module (ARM ETM9, v. r2p2) . . . 30
3.17 Ethernet MAC interface with DMA . . . 30
3.18 USB 2.0 slave device interface with DMA . . . 31
3.18.1 Packet buffer interface (PBI) . . . 32
3.18.2 DMA . . . 32
3.18.3 Suspend mode . . . 32
3.19 CAN 2.0B interface . . . 32
3.20 UART interfaces with DMA . . . 33
3.20.1 DMA . . . 33
3.21 I2C interfaces . . . 33
3.22 SSP interfaces (SPI, SSI, and MICROWIRE) with DMA . . . 34

3.22.1 DMA . . . 35
3.23 General purpose I/O . . . 35
3.24 A/D converter (ADC) with DMA . . . 35
3.24.1 DMA . . . 36
3.25 Standard timers (TIM) with DMA . . . 36
3.25.1 DMA . . . 36
3.26 Three-phase induction motor controller (IMC) . . . 37
3.27 External memory interface (EMI) . . . 38

4 Related documentation . . . 42

5 Pin description . . . 43
5.1 LFBGA144 ball connections . . . 45
5.2 Default pin functions . . . 46
5.2.1 General notes on pin usage . . . 46

6 Memory mapping . . . 54
6.1 Buffered and non-buffered writes . . . 54
6.2 System (AHB) and peripheral (APB) buses . . . 54
6.3 SRAM . . . 55
6.4 Two independent Flash memories . . . 55
6.4.1 Default configuration . . . 55
6.4.2 Optional configuration . . . 55
6.5 STR91xFA memory map . . . 56

7 Electrical characteristics . . . 58
7.1 Parameter conditions . . . 58
7.1.1 Minimum and maximum values . . . 58
7.1.2 Typical values . . . 58
7.1.3 Typical curves . . . 58
7.1.4 Loading capacitor . . . 58
7.1.5 Pin input voltage . . . 58
7.2 Absolute maximum ratings . . . 59
7.3 Operating conditions . . . 61
7.3.1 Operating conditions at power-up / power-down . . . 61
7.4 RESET_INn and power-on-reset characteristics . . . 62

7.5 LVD electrical characteristics . . . 62
7.5.1 LVD delay timing . . . 63
7.6 Supply current characteristics . . . 64
7.6.1 Typical power consumption for frequencies below 10 MHz . . . 65
7.7 Clock and timing characteristics . . . 66
7.7.1 Main oscillator electrical characteristics . . . 67
7.7.2 X1_CPU external clock source . . . 67
7.7.3 RTC clock generated from a crystal/ceramic resonator . . . 68
7.7.4 PLL electrical characteristics . . . 69
7.8 Memory characteristics . . . 70
7.8.1 SRAM characteristics . . . 70
7.8.2 Flash memory characteristics . . . 70
7.9 EMC characteristics . . . 72
7.9.1 Functional EMS (electro magnetic susceptibility) . . . 72
7.9.2 Electro magnetic interference (EMI) . . . 73
7.9.3 Absolute maximum ratings (electrical sensitivity) . . . 73
7.9.4 Electro-static discharge (ESD) . . . 73
7.9.5 Static latch-up . . . 74
7.9.6 Designing hardened software to avoid noise problems . . . 74
7.9.7 Electrical sensitivity . . . 74
7.10 I/O characteristics . . . 75
7.11 External memory bus timings . . . 76
7.11.1 Asynchronous mode . . . 76
7.11.2 Synchronous mode . . . 81
7.12 Communication interface electrical characteristics . . . 84
7.12.1 10/100 Ethernet MAC electrical characteristics . . . 84
7.12.2 USB electrical interface characteristics . . . 86
7.12.3 CAN interface electrical characteristics . . . 86
7.12.4 I2C electrical characteristics . . . 87
7.12.5 SPI electrical characteristics . . . 88
7.13 ADC electrical characteristics . . . 90

8 Device marking . . . 93
8.1 STR91xFAx32 / STR91xFAx42 / STR91xFAx44 . . . 93
8.2 STR91xFAx46 / STR91xFAx47 . . . 94

9 Package mechanical data 95
9.1 ECOPACK 104
9.2 Thermal characteristics 104

10 Ordering information 105

11 Revision history 106

List of tables

Table 1. Device summary . . . 1
Table 2. Device summary . . . 11
Table 3. Sectoring of primary Flash memory . . . 16
Table 4. Sectoring of secondary Flash memory . . . 17
Table 5. Product ID and revision level values . . . 17
Table 6. VIC IRQ channels . . . 19
Table 7. STR91x LFBGA144 ball connections . . . 45
Table 8. Device pin description . . . 47
Table 9. Absolute maximum ratings . . . 59
Table 10. Current characteristics . . . 60
Table 11. Operating conditions . . . 61
Table 12. Operating conditions at power-up / power-down . . . 61
Table 13. RESET_INn and power-on-reset characteristics . . . 62
Table 14. LVD electrical characteristics . . . 62
Table 15. Supply current characteristics . . . 64
Table 16. Typical current consumption at 25 °C . . . 65
Table 17. Internal clock frequencies . . . 66
Table 18. Main oscillator electrical characteristics . . . 67
Table 19. External clock characteristics . . . 67
Table 20. RTC oscillator electrical characteristics . . . 68
Table 21. RTC crystal electrical characteristics . . . 69
Table 22. PLL electrical characteristics . . . 69
Table 23. SRAM and hardware registers . . . 70
Table 24. Flash memory program/erase characteristics (Flash size ≤ 512 KB) . . . 70
Table 25. Flash memory program/erase characteristics (Flash size = 1 MB / 2 MB) . . . 71
Table 26. Flash memory endurance . . . 71
Table 27. EMS data . . . 72
Table 28. EMI data . . . 73
Table 29. ESD data . . . 73
Table 30. Static latch-up data . . . 74
Table 31. I/O characteristics . . . 75
Table 32. EMI bus clock period . . . 76
Table 33. EMI non-mux write operation . . . 76
Table 34. EMI read operation . . . 77
Table 35. Mux write times . . . 78
Table 36. Mux read times . . . 79
Table 37. Page mode read times . . . 80
Table 38. Sync burst write times . . . 82
Table 39. Sync burst read times . . . 83
Table 40. MII_RX_CLK and MII_TX_CLK timing table . . . 84
Table 41. MDC timing table . . . 84
Table 42. Ethernet MII management timing table . . . 85
Table 43. Ethernet MII transmit timing table . . . 86
Table 44. Ethernet MII receive timing table . . . 86
Table 45. I2C electrical characteristics . . . 87
Table 46. SPI electrical characteristics . . . 88
Table 47. General ADC electrical characteristics . . . 90
Table 48. ADC conversion time (silicon Rev G) . . . 91

Table 49. ADC conversion time (silicon Rev H and higher) 91
Table 50. LQFP80 12 x12 mm low-profile quad flat package mechanical data 96
Table 51. LQFP128 - 128-pin, 14 x 14 mm low-profile quad flat package mechanical data 99
Table 52. LFBGA144 – 144-ball low profile fine pitch ball grid array, 10 x 10 mm, 0.8 mm pitch, package mechanical data 102
Table 53. Thermal characteristics 104
Table 54. Ordering information scheme 105
Table 55. Document revision history 106

List of figures

Figure 1. STR91xFA block diagram . . . 14
Figure 2. Clock control . . . 21
Figure 3. JTAG chaining inside the STR91xFA . . . 28
Figure 4. EMI 16-bit multiplexed connection example . . . 40
Figure 5. EMI 8-bit multiplexed connection example . . . 40
Figure 6. EMI 8-bit non-multiplexed connection example . . . 41
Figure 7. STR91xFAM 80-pin package pinout . . . 43
Figure 8. STR91xFAW 128-pin package pinout . . . 44
Figure 9. STR91xFA memory map . . . 57
Figure 10. Pin loading conditions . . . 58
Figure 11. Pin input voltage . . . 59
Figure 12. LVD reset delay case 1 . . . 63
Figure 13. LVD reset delay case 2 . . . 63
Figure 14. LVD reset delay case 3 . . . 63
Figure 15. Sleep mode current vs temperature with LVD on . . . 65
Figure 16. Typical application with an external clock source . . . 68
Figure 17. Typical application with a 32.768 kHz crystal . . . 69
Figure 18. Non-mux write timings . . . 76
Figure 19. Non-mux bus read timings . . . 77
Figure 20. Mux write diagram . . . 78
Figure 21. Mux read diagram . . . 79
Figure 22. Page mode read diagram . . . 80
Figure 23. Sync burst write diagram . . . 81
Figure 24. Sync burst read diagram . . . 83
Figure 25. MII_RX_CLK and MII_TX_CLK timing diagram . . . 84
Figure 26. MDC timing diagram . . . 84
Figure 27. Ethernet MII management timing diagram . . . 85
Figure 28. Ethernet MII transmit timing diagram . . . 85
Figure 29. Ethernet MII receive timing diagram . . . 86
Figure 30. SPI slave timing diagram with CPHA = 0 . . . 88
Figure 31. SPI slave timing diagram with CPHA = 1 . . . 89
Figure 32. SPI master timing diagram . . . 89
Figure 33. ADC conversion characteristics . . . 92
Figure 34. Device marking for revision G LQFP80 and LQFP128 packages . . . 93
Figure 35. Device marking for revision G LFBGA144 packages . . . 93
Figure 36. Device marking for revision H LQFP80 and LQFP128 packages . . . 93
Figure 37. Device marking for revision H LFBGA144 packages . . . 93
Figure 38. Device marking for revision A LQFP80 and LQFP128 packages . . . 94
Figure 39. Device marking for revision A LFBGA144 packages . . . 94
Figure 40. LQFP80 12 x 12 mm 80 pin low-profile quad flat package outline . . . 95
Figure 41. LQFP80 - 80 pin, 12 x 12 mm low-profile quad flat package footprint . . . 96
Figure 42. LQFP80 package top view . . . 97
Figure 43. LQFP128 14 x 14 mm 128 pin low-profile quad flat package outline . . . 98
Figure 44. LQFP128 package top view . . . 100
Figure 45. LFBGA144 – 144-ball low profile fine pitch ball grid array, 10 x 10 mm, 0.8 mm pitch, package outline . . . 101
Figure 46. LFBGA144 package top view . . . 103
Figure 47. Recommended PCB design rules (0.80/0.75 mm pitch BGA) . . . 103

1 Description

STR91xFA 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.

2 Device summary

Table 2. Device summary

Part number	Flash KB	RAM KB	Major peripherals	Package
STR910FAM32	**256+32**	64	**CAN, 40 I/Os**	**LQFP80, 12x12 mm**
STR910FAW32	**256+32**	64	**CAN, EMI, 80 I/Os**	**LQFP128, 14x14 mm**
STR910FAZ32	**256+32**	64	**CAN, EMI, 80 I/Os**	LFBGA144 10 x 10 x 1.7
STR911FAM42	**256+32**	96	**USB, CAN, 40 I/Os**	**LQFP80, 12x12mm**
STR911FAM44	**512+32**	96		
STR911FAM46	**1024+128**	96	**USB, CAN, 40 I/Os**	**LQFP80, 12x12mm**
STR911FAM47	**2048+128**	96		
STR911FAW42	**256+32**	96	**USB, CAN, EMI, 80 I/Os**	**LQFP128, 14x14mm**
STR911FAW44	**512+32**	96		
STR911FAW46	**1024+128**	96	**USB, CAN, EMI, 80 I/Os**	**LQFP128, 14x14mm**
STR911FAW47	**2048+128**	96		
STR912FAW32	**256+32**	64	**Ethernet, USB, CAN, EMI, 80 I/Os**	**LQFP128**
STR912FAW42	**256+32**	96	**Ethernet, USB, CAN, EMI, 80 I/Os**	**LQFP128**
STR912FAW44	**512+32**	96		
STR912FAW46	**1024+128**	96	**Ethernet, USB, CAN, EMI, 80 I/Os**	**LQFP128**
STR912FAW47	**2048+128**	96		
STR912FAZ42	**256+32**	96	**Ethernet, USB, CAN, EMI, 80 I/Os**	LFBGA144 10 x 10 x 1.7
STR912FAZ44	**512+32**	96		
STR912FAZ46	**1024+128**	96	**Ethernet, USB, CAN, EMI, 80 I/Os**	LFBGA144 10 x 10 x 1.7
STR912FAZ47	**2048+128**	96		

3 Functional overview

3.1 System-in-a-package (SiP)

The STR91xFA 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.

3.2 Package choice

STR91xFA devices are available in 128-pin (14 x 14 mm) and 80-pin (12 x 12 mm) LQFP and LFBGA144 (10 x 10 mm) packages. Refer to *Table 2: Device summary on page 11* for a list of available peripherals for each of the package choices.

3.3 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.

3.4 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 an 8-instruction Pre-Fetch Queue (PFQ) and a 15-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.

3.4.1 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.

3.4.2 Branch cache (BC)

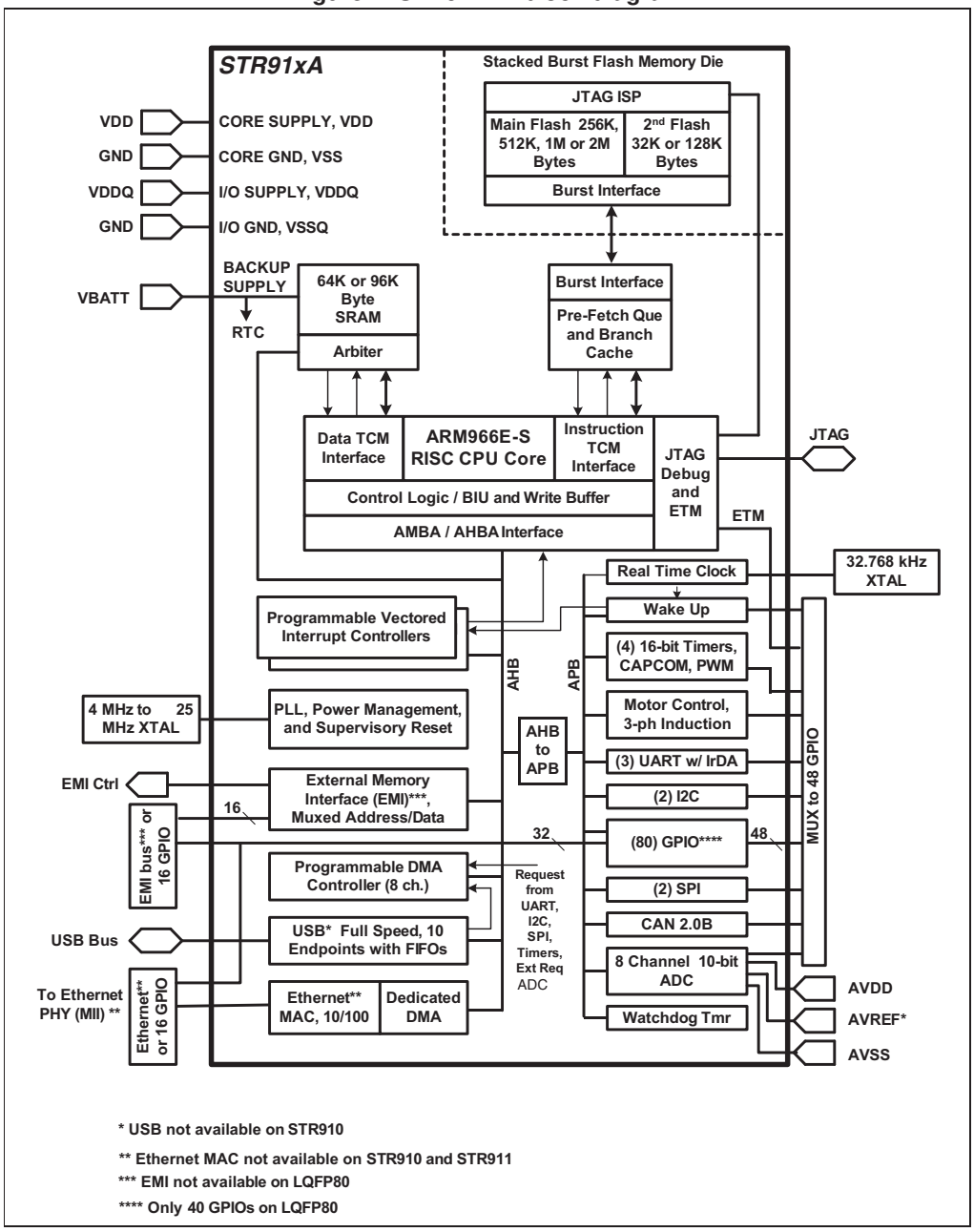
When instruction addresses are not sequential, such as a program branch situation, the PFQ would have to flush and reload which would cause the CPU to stall if no BC were present. Before reloading, the PFQ checks the BC to see if it contains the desired target branch address. The BC contains up to fifteen of the most recently taken branch addresses and the first eight instructions associated with each of these branches. This check is extremely fast, checking all fifteen 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 eight 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 16th 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.

3.4.3 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 STR91xFA 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.

Figure 1. STR91xFA block diagram



3.5 SRAM (64 Kbytes or 96 Kbytes)

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 the SRAM.

3.5.1 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.

3.5.2 Battery backup

When a battery is connected to the designated battery backup pin (VBATT), SRAM contents are automatically preserved when the operating voltage on the main digital supplies (VDD and VDDQ are lost or sag below the LVD 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.

3.6 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 14 request signals to service other peripherals and interfaces (USB, SSP, ADC, 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 3.5.1*. 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 STR91xFA through the EMI bus.

3.7 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).

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

3.7.1 Primary Flash memory

Using the STR91xFA 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 6: Memory mapping*, for more detail.

The primary Flash memory has equal length 64K byte sectors. See *Table 3* for number of sectors per device type.

Table 3. Sectoring of primary Flash memory

Size of primary Flash	256 Kbytes	512 Kbytes	1 Mbyte	2 Mbytes
Number of sectors	4	8	16	32
Size of each sector	64 Kbytes		64 Kbytes	

3.7.2 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 STR91xFA (USB, Ethernet, CAN, UART, etc.)

Additionally, the secondary Flash memory may also be used to store small data sets by emulating EEPROM through 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 STR91xFA device.

The secondary Flash memory is sectored as shown in *Table 4* according to device type.

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.

Table 4. Sectoring of secondary Flash memory

Size of secondary Flash	32 Kbytes	128 Kbytes
Number of sectors	4	8
Size of each sector	8 Kbytes	16 Kbytes

3.8 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 (bytes 31 and 30) are reserved for the STR91xFA product ID and revision level.

3.8.1 Product ID and revision level

OTP bytes 31 and 30 are programmed at ST factory before shipment and may be read by firmware to determine the STR91xFA product type and silicon revision so it can optionally take action based on the silicon on which it is running. In Rev H devices and 1MB/2MB Rev A devices, byte 31 contains the major family identifier of "9" (for STR9) in the high-nibble location and the minor family identifier in the low nibble location, which can be used to determine the size of primary flash memory. In all devices, byte 30 contains the silicon revision level indicator. See *Table 5* for values related to the revisions of STR9 production devices and size of primary Flash memory. See *Section 8* for details of external identification of silicon revisions.

Table 5. Product ID and revision level values

Production salestype	Silicon revision	Size of primary Flash	OTP byte 31	OTP byte 30
STR91xFAxxxxx	Rev G	256K or 512K	91h	20h
STR91xFAxxxxx	Rev H	256K	90h	21h
STR91xFAxxxxx	Rev H	512K	91h	21h
STR91xFAx46xx	Rev A	1024K	92h	21h
STR91xFAx47xx	Rev A	2048K	93h	21h

3.9 Vectored interrupt controller (VIC)

Interrupt management in the STR91xFA is implemented from daisy-chaining two standard ARM VIC units. This combined VIC has 32 prioritized interrupt request channels and generates two interrupt output signals to the CPU. The output signals are FIQ and IRQ, with FIQ having higher priority.

3.9.1 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.

3.9.2 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, inside the same VIC (primary or secondary VIC), CPU firmware may re-assign individual interrupt sources to individual hardware IRQ channels, meaning that firmware can effectively change interrupt priority levels as needed within the same VIC (from priority 0 to priority 16).

Note: VIC0 (primary VIC) interrupts always have higher priority than VIC1 (secondary VIC) interrupts

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 STR91xFA 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 STR91xFA reduces this to a single access by adding a 16th entry in the instruction branch cache, dedicated for interrupts. This 16th 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.

3.9.3 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 STR91xFA such as on-chip peripherals, see *Table 6*. 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 6*) 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 6*) 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.

Table 6. VIC IRQ channels

IRQ channel hardware priority	VIC input channel	Logic block	Interrupt source
0 (high priority)	VIC0.0	Watchdog	Timeout in WDT mode, Terminal Count in Counter Mode
1	VIC0.1	CPU Firmware	Firmware generated interrupt
2	VIC0.2	CPU Core	Debug Receive Command
3	VIC0.3	CPU Core	Debug Transmit Command
4	VIC0.4	TIM Timer 0	Logic OR of ICI0_0, ICI0_1, OCI0_0, OCI0_1, Timer overflow
5	VIC0.5	TIM Timer 1	Logic OR of ICI1_0, ICI1_1, OCI1_0, OCI1_1, Timer overflow
6	VIC0.6	TIM Timer 2	Logic OR of ICI2_0, ICI2_1, OCI2_0, OCI2_1, Timer overflow
7	VIC0.7	TIM Timer 3	Logic OR of ICI3_0, ICI3_1, OCI3_0, OCI3_1, Timer overflow
8	VIC0.8	USB	Logic OR of high priority USB interrupts
9	VIC0.9	USB	Logic OR of low priority USB interrupts
10	VIC0.10	CCU	Logic OR of all interrupts from Clock Control Unit
11	VIC0.11	Ethernet MAC	Logic OR of Ethernet MAC interrupts via its own dedicated DMA channel.
12	VIC0.12	DMA	Logic OR of interrupts from each of the 8 individual DMA channels
13	VIC0.13	CAN	Logic OR of all CAN interface interrupt sources
14	VIC0.14	IMC	Logic OR of 8 Induction Motor Control Unit interrupts
15	VIC0.15	ADC	End of AtoD conversion interrupt
16	VIC1.0	UART0	Logic OR of 5 interrupts from UART channel 0
17	VIC1.1	UART1	Logic OR of 5 interrupts from UART channel 1
18	VIC1.2	UART2	Logic OR of 5 interrupts from UART channel 2
19	VIC1.3	I2C0	Logic OR of transmit, receive, and error interrupts of I2C channel 0
20	VIC1.4	I2C1	Logic OR of transmit, receive, and error interrupts of I2C channel 1
21	VIC1.5	SSP0	Logic OR of all interrupts from SSP channel 0
22	VIC1.6	SSP1	Logic OR of all interrupts from SSP channel 1
23	VIC1.7	BROWNOUT	LVD warning interrupt
24	VIC1.8	RTC	Logic OR of Alarm, Tamper, or Periodic Timer interrupts

Table 6. VIC IRQ channels (continued)

IRQ channel hardware priority	VIC input channel	Logic block	Interrupt source
25	VIC1.9	Wake-Up (all)	Logic OR of all 32 inputs of Wake-Up unit (30 pins, RTC, and USB Resume)
26	VIC1.10	Wake-up Group 0	Logic OR of 8 interrupt sources: RTC, USB Resume, pins P3.2 to P3.7
27	VIC1.11	Wake-up Group 1	Logic OR of 8 interrupts from pins P5.0 to P5.7
28	VIC1.12	Wake-up Group 2	Logic OR of 8 interrupts from pins P6.0 to P6.7
29	VIC1.13	Wake-up Group 3	Logic OR of 8 interrupts from pins P7.0 to P7.7
30	VIC1.14	USB	USB Bus Resume Wake-up (also input to wake-up unit)
31 (low priority)	VIC1.15	PFQ-BC	Special use of interrupts from Prefetch Queue and Branch Cache

3.10 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 STR91xFA.

- 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}

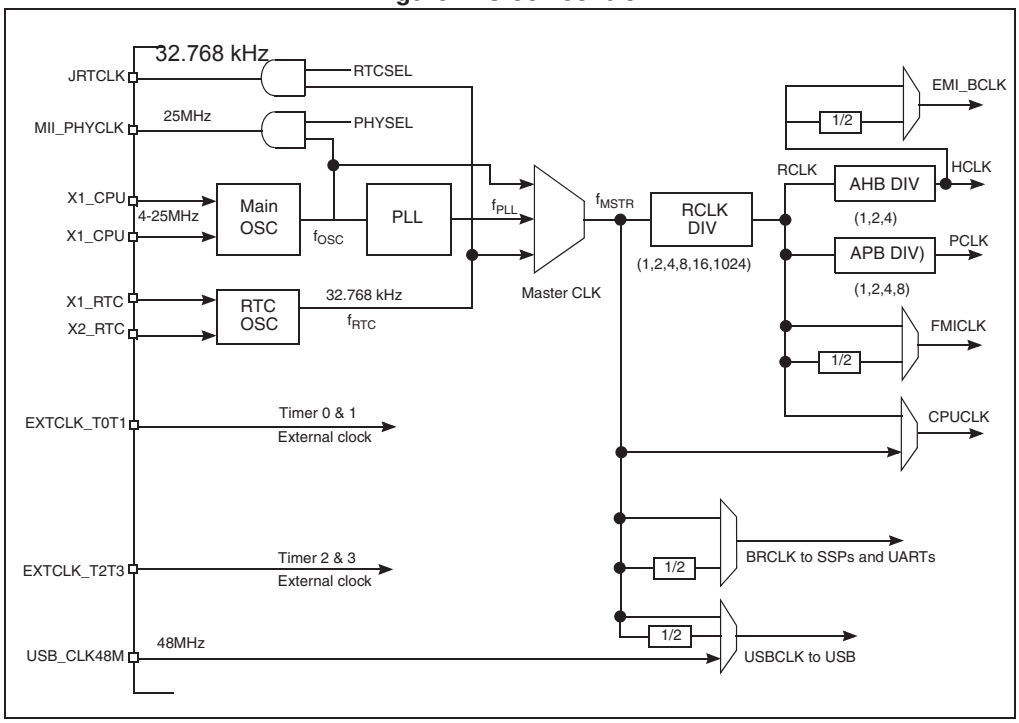
3.10.1 Master clock sources

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 STR91xFA 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.

Figure 2. Clock control



3.10.2 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.

3.10.3 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.

3.10.4 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.

3.10.5 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 96 MHz.

3.10.6 UART and SSP clock (BRCLK)

BRCLK is an internal clock derived from f_{MSTR} that is used to drive the two SSP peripherals and to generate the Baud rate for the three on-chip UART peripherals. The frequency can be optionally divided by 2.

3.10.7 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 17 on page 66* for the maximum BCLK frequency (f_{BCLK}). The BCLK clock is available on the LFBGA package as an output pin.

3.10.8 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.
- STR91xFA 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.

3.10.9 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 STR91xFA. In this case, the STR91xFA 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 STR91xFA and the external PHY device.
- An external 25 MHz oscillator connected directly to the external PHY interface device. In this case, the STR91xFA can operate independent of 25 MHz.

3.10.10 External RTC calibration clock

The RTC_CLK 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.

3.10.11 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 STR91xFA 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.

3.11 Flexible power management

The STR91xFA 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 STR91xFA 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.

3.11.1 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.

3.11.2 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.

3.11.3 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:

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

3.12 Voltage supplies

The STR91xFA 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.

In Standby mode, both VDD and VDDQ must be shut down. Otherwise the specified maximum power consumption for Standby mode (I_{RTC_STBY} and I_{SRAM_STBY}) may be exceeded. Leakage may occur if only one of the voltage supplies is off.

3.12.1 Independent A/D converter supply and reference voltage

The ADC unit on 128-pin and 144-ball packages has an isolated analog voltage supply input at pin AVDD to accept a very clean voltage source, independent of the digital voltage supplies. Additionally, an isolated analog supply ground connection is provided on pin AVSS only on 128-pin and 144-ball 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 STR91xFA device on pin AVSS_VSSQ.

A separate external analog reference voltage input for the ADC unit is available on 128-pin and 144-ball packages at the AVREF pin for better accuracy on low voltage inputs. For 80-pin packages, the ADC reference voltage is tied internally to the ADC unit supply voltage at pin AVREF_AVDD, meaning the ADC reference voltage is fixed to the ADC unit supply voltage.

See *Table 11: Operating conditions*, for restrictions to the relative voltage levels of VDDQ, AVDD, AVREF, and AVREF_AVDD.

3.12.2 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 main digital supplies (V_{DD} and $V_{DDQ)}$. 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 the LVD threshold. In order to use the battery supply, the LVD must be enabled.

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 STR91xFA device is powered off.

3.13 System supervisor

The STR91xFA 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 STR91xFA, 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.

3.13.1 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.