Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	ARM9®
Core Size	16/32-Bit
Speed	96MHz
Connectivity	CANbus, EBI/EMI, Ethernet, I²C, IrDA, Microwire, SPI, SSI, SSP, UART/USART, USB
Peripherals	Brown-out Detect/Reset, DMA, Motor Control PWM, POR, PWM, WDT
Number of I/O	80
Program Memory Size	256KB (256K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64K x 8
Voltage - Supply (Vcc/Vdd)	1.65V ~ 2V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	128-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/str912faw32x6

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		

3.4.2 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.

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.

3.13.7 Tamper detection

On 128-pin and 144-ball STR91xFA 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 detects 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.

3.14 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 3.13.7*), and an optional clock calibration output on the JRTCK pin. The time is in 24 hour mode, and time/calendar values are stored in binary-coded decimal format.

The RTC also provides a self-isolation mode that is automatically activated during power down. This feature allows the RTC to continue operation when V_{DDQ} and V_{DD} are absent, as long as an alternate power source, such as a battery, is connected to the VBATT input pin. The current drawn by the RTC unit on the VBATT pin is very low in this standby mode, I_{RTC_STBY}.

3.15 JTAG interface

An IEEE-1149.1 JTAG interface on the STR91xFA provides In-System-Programming (ISP) of all memory, boundary scan testing of pins, and the capability to debug the CPU.

STR91xFA 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 STR91xFA 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 STR91xFA 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 STR91xFA 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 STR91xFA. 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 STR91xFA (CPU die and Flash memory die) are internally daisy-chained on the JTAG bus, see *Figure 3 on page 28*. 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

The STR91xFA 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

3.18 USB 2.0 slave device interface with DMA

The STR91xFA 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:

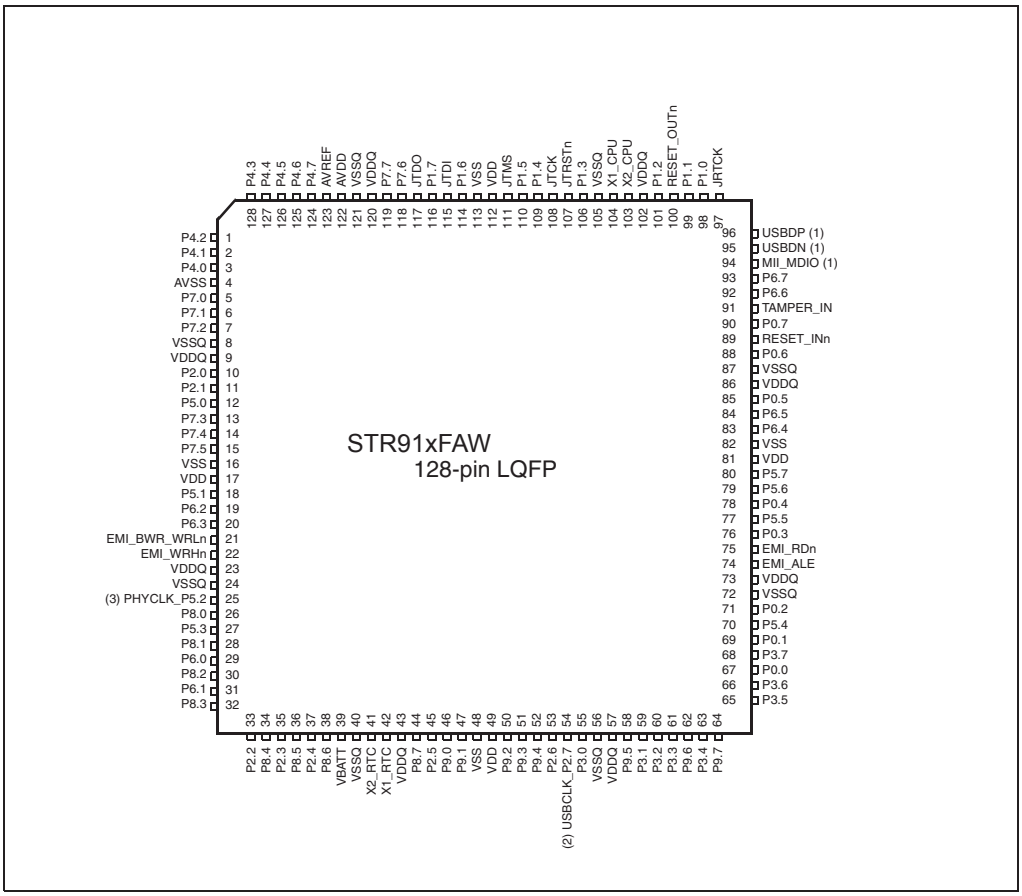
- Supports USB low and full-speed transfers (12 Mbps), certified to comply with the USB 2.0 specification
- Supports isochronous, bulk, control, and interrupt endpoints
- Configurable number of endpoints allowing a mixture of up to 20 single-buffered monodirectional endpoints or up to 10 double-buffered 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

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.
- ***Burst Mode Support (LFBGA package only)*****:** The EMI bus supports synchronized burst read and write bus cycle in multiplexed and non-multiplexed mode. The additional EMI signals in the LFBGA package that support the burst mode are:
  - EMI_BCLK -the bus clock output. The EMI_BCLK has the same frequency or half of that of the HCLK and can be disabled by the user
  - EMI_WAITn - the not ready or wait input signal for synchronous access
  - EMI_BAAn - burst address advance or burst enable signal
  - EMI_WEn - write enable signal
  - EMI_UBn, EMI_LBn - upper byte and lower byte enable signals. These two signals share the same pins as the EMI_WRLn and EMI_WRHn and are user configurable through the EMI register.

  By defining the bus parameters such as burst length, burst type, read and write timings in the EMI control registers, the EMI bus is able to interface to standard burst memory devices. The burst timing specification and waveform will be provided in the next data sheet release

Figure 8. STR91xFAW 128-pin package pinout



1. NU (Not Used) on STR910FAW devices. Pin 95 is not connected, pin 96 must be pulled up by a 1.5Kohm resistor to VDDQ.
2. No USBCLK function on STR910FAW devices.
3. No PHYCLK function on STR910FAW devices.

Table 8. Device pin description

Package			Pin name	Signal type	Default pin function	Default input function	Alternate functions			
LQFP80	LQFP128	LFBGA144					Alternate input 1	Alternate output 1	Alternate output 2	Alternate output 3
-	67	L11	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	K10	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	J11	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	H12	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	H10	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	F11	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
-	88	E11	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	B12	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	B10	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	C10	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	B9	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	C8	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	B7	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	A7	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	F7	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	D6	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	E2	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	E3	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	M1	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	K3	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	L4	P2.4	I/O	GPIO_2.4, GP Input, HiZ	EXTCLK_T0T1E xt 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

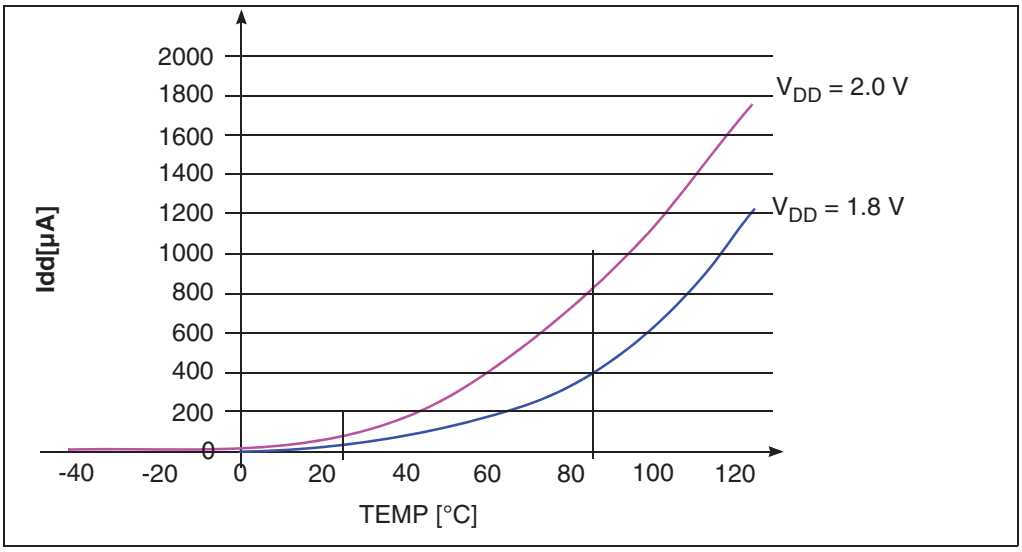
Table 8. Device pin description (continued)

Package			Pin name	Signal type	Default pin function	Default input function	Alternate functions			
LQFP80	LQFP128	LFBGA144					Alternate input 1	Alternate output 1	Alternate output 2	Alternate output 3
12	18	F6	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	K1	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	H2	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	J12	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	H11	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	H9	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	G12	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	H4	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	J3	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	G2	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	G3	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	G8	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	G7	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	E9	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	D12	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
-	5	D1	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	D2	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	B1	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	F1	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	G1	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	E5	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

Table 8. Device pin description (continued)

Package			Pin name	Signal type	Default pin function	Default input function	Alternate functions			
LQFP80	LQFP128	LFBGA144					Alternate input 1	Alternate output 1	Alternate output 2	Alternate output 3
-	118	E6	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	A5	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	L1	P8.0	I/O	GPIO_8.0, GP Input, HiZ	-	-	GPIO_8.0, GP Output	8b) EMI_D0, 16b) EMI_AD0	-
-	28	H3	P8.1	I/O	GPIO_8.1, GP Input, HiZ	-	-	GPIO_8.1, GP Output	8b) EMI_D1, 16b) EMI_AD1	-
-	30	J2	P8.2	I/O	GPIO_8.2, GP Input, HiZ	-	-	GPIO_8.2, GP Output	8b) EMI_D2, 16b) EMI_AD2	-
-	32	K2	P8.3	I/O	GPIO_8.3, GP Input, HiZ	-	-	GPIO_8.3, GP Output	8b) EMI_D3, 16b) EMI_AD3	-
-	34	L3	P8.4	I/O	GPIO_8.4, GP Input, HiZ	-	-	GPIO_8.4, GP Output	8b) EMI_D4, 16b) EMI_AD4	-
-	36	J4	P8.5	I/O	GPIO_8.5, GP Input, HiZ	-	-	GPIO_8.5, GP Output	8b) EMI_D5, 16b) EMI_AD5	-
-	38	M2	P8.6	I/O	GPIO_8.6, GP Input, HiZ	-	-	GPIO_8.6, GP Output	8b) EMI_D6, 16b) EMI_AD6	-
-	44	K5	P8.7	I/O	GPIO_8.7, GP Input, HiZ	-	-	GPIO_8.7, GP Output	8b) EMI_D7, 16b) EMI_AD7	-
-	46	M6	P9.0	I/O	GPIO_9.0, GP Input, HiZ	-	-	GPIO_9.0, GP Output	8b) EMI_A8 16b) EMI_AD8	-
-	47	M7	P9.1	I/O	GPIO_9.1, GP Input, HiZ	-	-	GPIO_9.1, GP Output	8b) EMI_A9, 16b) EMI_AD9	-
-	50	K6	P9.2	I/O	GPIO_9.2, GP Input, HiZ	-	-	GPIO_9.2, GP Output	8b) EMI_A10, 16b)EMI_AD10	-
-	51	J6	P9.3	I/O	GPIO_9.3, GP Input, HiZ	-	-	GPIO_9.3, GP Output	8b) EMI_A11, 16b)EMI_AD11	-
-	52	H6	P9.4	I/O	GPIO_9.4, GP Input, HiZ	-	-	GPIO_9.4, GP Output	8b) EMI_A12, 16b)EMI_AD12	-
-	58	L8	P9.5	I/O	GPIO_9.5, GP Input, HiZ	-	-	GPIO_9.5, GP Output	8b) EMI_A13, 16b)EMI_AD13	-
-	62	M9	P9.6	I/O	GPIO_9.6, GP Input, HiZ	-	-	GPIO_9.6, GP Output	8b) EMI_A14, 16b)EMI_AD14	-
-	64	K9	P9.7	I/O	GPIO_9.7, GP Input, HiZ	-	-	GPIO_9.7, GP Output	8b) EMI_A15, 16b)EMI_AD15	-

Figure 15. Sleep mode current vs temperature with LVD on



7.6.1 Typical power consumption for frequencies below 10 MHz

The following conditions apply to *Table 16*:

- Program is executed from Flash. The program consists of an infinite loop.
- A standard crystal source is used.
- The PLL is off.
- All clock dividers are with their default values.

Table 16. Typical current consumption at 25 °C

Symbol	Parameter		Test conditions	Typical current on V_{DD} (1.8 V)	Unit
IDDRUN	Run mode current	All peripherals ON	$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=1 MHz	2.88	mA
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=2 MHz	5.8	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=4 MHz	10.91	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=6 MHz	15.97	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=8 MHz	20.68	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=10 MHz	25.13	
		All peripherals OFF	$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=1 MHz	1.8	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=2 MHz	3.62	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=4 MHz	6.71	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=6 MHz	9.81	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=8 MHz	12.63	
			$f_{MSTR}=f_{OSC}=f_{PCLK}=f_{HCLK}$=10 MHz	15.47	

7.9.5 Static latch-up

Two 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.

7.9.6 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
- 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).

7.9.7 Electrical sensitivity

Table 30. Static latch-up data

Symbol	Parameter	Conditions	Class (1)
LU	Static latch-up class	T_A = +25 °C conforming to JESD78A	II class A

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).

7.11 External memory bus timings

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

Table 32. EMI bus clock period

Symbol	Parameter[1]	Value [2]
t_{BCLK}	EMI bus clock period	1 /(f_{HCLK} x EMI_ratio)

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)

7.11.1 Asynchronous mode

Non Mux Write

Figure 18. Non-mux write timings

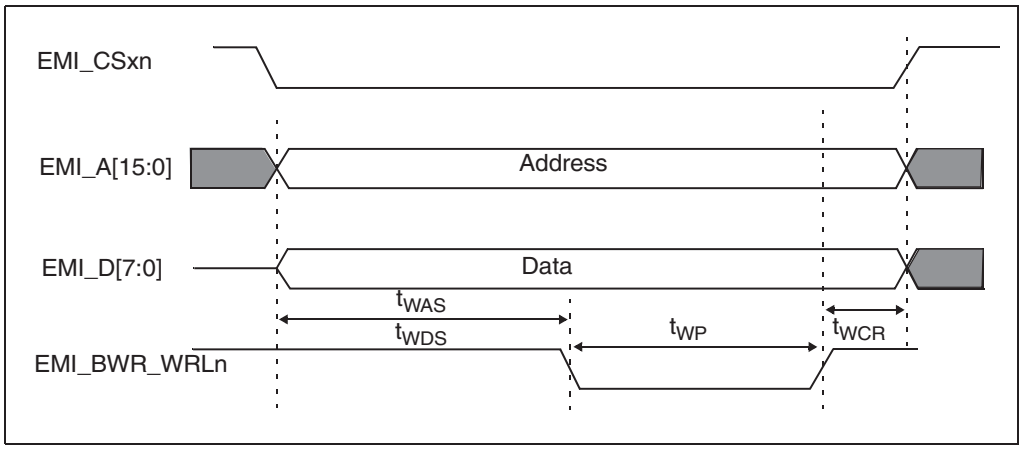


Table 33. EMI non-mux write operation

Symbol	Parameter	Value	
		Min	Max
t_{WCR}	WRn to CSn inactive	(t_{BCLK}/2) - 2 ns	(t_{BCLK}/2) + 2 ns
t_{WAS}	Write address setup time	((WSTWEN + 1/2) x t_{BCLK}) - 2 ns	((WSTWEN + 1/2) x t_{BCLK}) + 1 ns
t_{WDS}	Write data setup time	((WSTWEN + 1/2) x t_{BCLK}) - 5 ns	((WSTWEN + 1/2) x t_{BCLK})
t_{WP}	Write pulse width	(WSTWR-WSTWEN + 1) x t_{BCLK} - 1 ns	(WSTWR-WSTWEN + 1) x t_{BCLK} + 1.5 ns

Mux write

Figure 20. Mux write diagram

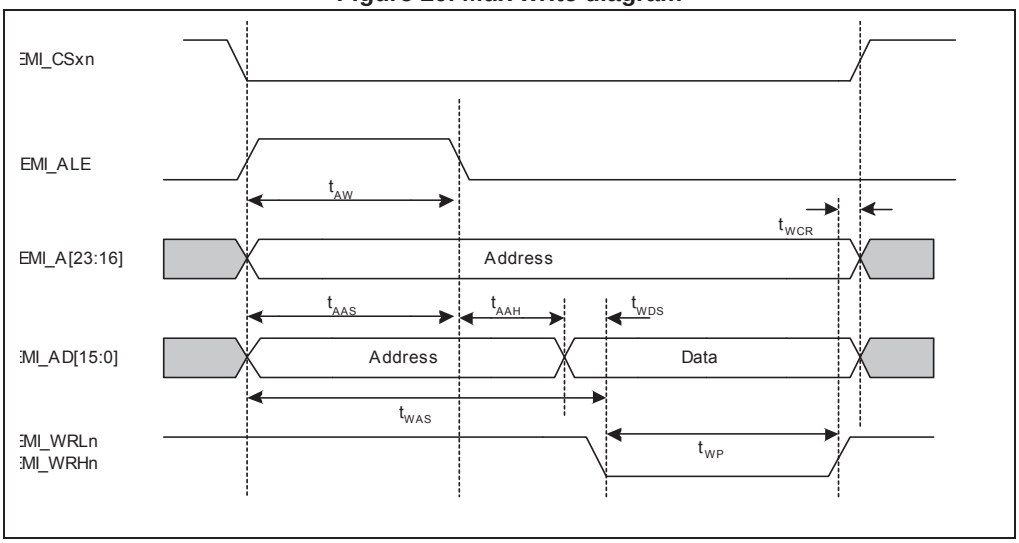


Table 35. Mux write times

Symbol	Parameter	Value	
		Min	Max
t_{WCR}	WRn to CSn inactive	(t_{BCLK}/2) - 2ns	(t_{BCLK}/2) + 2ns
t_{WAS}	Write address setup time	(WSTWEN + 1/2) x t_{BCLK} - 2.5 ns	(WSTWEN + 1/2) x t_{BCLK} + 2 ns
t_{WDS}	Write data setup time	((WSTWEN - ALE_LENGTH) x t_{BCLK}) - 2 ns	((WSTWEN - ALE_LENGTH) x t_{BCLK}) + 1 ns
t_{WP}	Write pulse width	((WSTWR-WSTWEN + 1) x t_{BCLK}) - 1 ns	((WSTWR-WSTWEN + 1) x $t_{BCLK)}$ + 1.5 ns
t_{AW}	ALE pulse width	(ALE_LENGTH x t_{BCLK})- 3.5 ns	(ALE_LENGTH x t_{BCLK})
t_{AAS}	Address to ALE setup time	(ALE_LENGTH x t_{BCLK})- 3.5 ns	(ALE_LENGTH x t_{BCLK})
t_{AAH}	Address to ALE hold time	(t_{BCLK}/2) - 1 ns	(t_{BCLK}/2) + 2 ns

Mux read

Figure 21. Mux read diagram

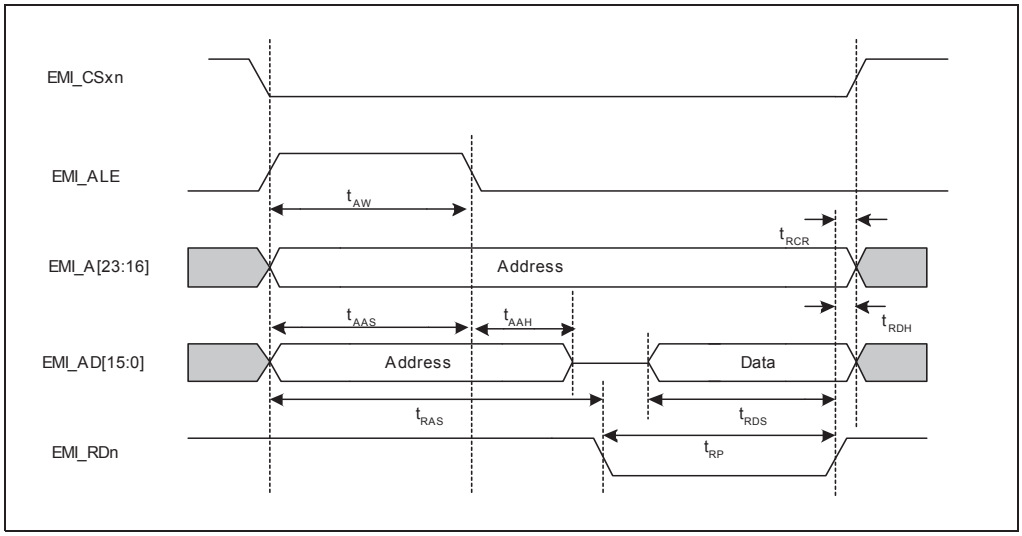
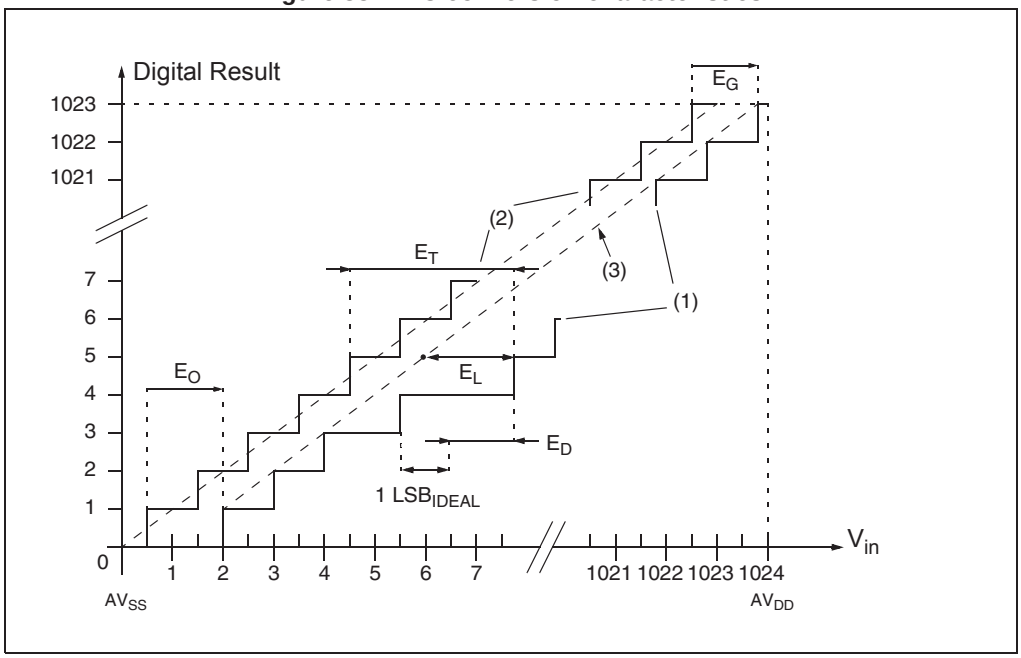


Table 36. Mux read times

Symbol	Parameter	Value	
		Min	Max
t_{RCR}	Read to CSn inactive	0	1.5 ns
t_{RAS}	Read address setup time	((WSTOEN) x t_{BCLK})- 4 ns	((WSTOEN) x t_{BCLK})
t_{RDS}	Read data setup time	12 ns	-
t_{RDH}	Read data hold time	0	
t_{RP}	Read pulse width	((WSTRD-WSTOEN+1) x t_{BCLK}) - 0.5 ns	((WSTRD-WSTOEN+1) x t_{BCLK}) + 2.5 ns
t_{AW}	ALE pulse width	(ALE_LENGTH x t_{BCLK}) - 3.5 ns	(ALE_LENGTH x t_{BCLK})
t_{AAS}	Address to ALE setup time	(ALE_LENGTH x t_{BCLK}) - 3.5 ns	(ALE_LENGTH x t_{BCLK})
t_{AAH}	Address to ALE hold time	(t_{BCLK}/2)- 1 ns	(t_{BCLK}/2) + 2 ns

Figure 33. ADC conversion characteristics



1. Legend:
   (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.

Equation 1

$$1LSB_{IDEAL} = \frac{V_{DDA} - V_{SSA}}{1024}$$

8.2 STR91xFAx46 / STR91xFAx47

Figure 38. Device marking for revision A LQFP80 and LQFP128 packages

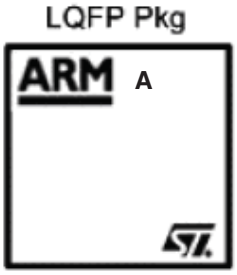
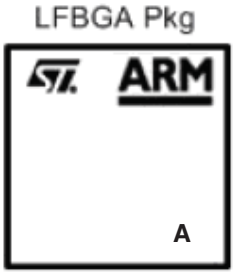


Figure 39. Device marking for revision A LFBGA144 packages



11 Revision history

Table 55. Document revision history

Date	Revision	Changes
09-May-2007	1	Initial release
26-Nov-2007	2	Updated Standby current in *Table 15: Supply current characteristics on page 64* Added *Section 7.1: Parameter conditions on page 58* Added *Section 7.7.2: X1_CPU external clock source on page 67* Updated *Section 7.11: External memory bus timings on page 76* Added *Figure 14: LVD reset delay case 3 on page 63* Added *Table 48* and *Table 49* in ADC characteristics section Added min/max values for E, D, E1, D1 in *Figure 43 on page 98*
14-May-2008	3	Added 1MB and 2M devices, creating merged datasheet from seperate STR91xFAx32, 42, 44, 46 and 47 devices. Added STR912FAW32 to *Table 1: Device summary on page 1* Added paragraph on voltage supply shutdown in *Section 3.12 on page 24* Removed DMA feature for I2C in *Section 3.21 on page 33* Updated Sleep mode current in *Table 10: Current characteristics on page 60* Added *Table 16: Typical current consumption at 25 °C on page 65* Updated operating conditions for V_{DD} and f_{CPUCLK} in *Section 7.3 on page 61* and *Section 7.7: Clock and timing characteristics on page 66* Changed SPI master t_{SU} and $t_{H\ to}$ TBD in *Table 46: SPI electrical characteristics on page 88*
17-Jul-2008	4	Updated *Section 3.10.6: UART and SSP clock (BRCLK) on page 22* Updated *Table 11: Operating conditions on page 61* Updated $I_{SLEEP(IDDQ)}$ in *Table 15: Supply current characteristics on page 64* Updated *Table 17: Internal clock frequencies on page 66* Updated *Table 31: I/O characteristics on page 75*
22-Dec-2008	5	Updated *Section 7.7.3 on page 68*. Small text changes.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.