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	Active
Core Processor	e200z4d
Core Size	32-Bit Single-Core
Speed	120MHz
Connectivity	CANbus, LINbus, SCI, SPI, UART/USART
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	57
Program Memory Size	768KB (768K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	96K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.63V
Data Converters	A/D 32x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	100-LQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/spc564l54l3bcoqy

Contents

List of tables

The eDMA module provides the following features:

- 16 channels supporting 8-, 16-, and 32-bit value single or block transfers
- Support variable sized queues and circular buffered queue
- Source and destination address registers independently configured to post-increment or stay constant
- Support major and minor loop offset
- Support minor and major loop done signals
- DMA task initiated either by hardware requestor or by software
- Each DMA task can optionally generate an interrupt at completion and retirement of the task
- Signal to indicate closure of last minor loop
- Transfer control descriptors mapped inside the SRAM

The eDMA controller is replicated for each processing channel.

1.5.5 On-chip flash memory with ECC

This device includes programmable, non-volatile flash memory. The non-volatile memory (NVM) can be used for instruction storage or data storage, or both. The flash memory module interfaces with the system bus through a dedicated flash memory array controller. It supports a 64-bit data bus width at the system bus port, and a 128-bit read data interface to flash memory. The module contains four 128-bit prefetch buffers. Prefetch buffer hits allow no-wait responses. Buffer misses incur a 3 wait state response at 120 MHz.

The flash memory module provides the following features

- 1 MB of flash memory in unique multi-partitioned hard macro
- Sectorization: 16 KB + 2 × 48 KB + 16 KB + 2 × 64 KB + 2 × 128 KB + 2 × 256 KB
- EEPROM emulation (in software) within same module but on different partition
- 16 KB test sector and 16 KB shadow block for test, censorship device and user option bits
- Wait states:
  - 3 wait states for frequencies =< 120 MHz
  - 2 wait states for frequencies =< 80 MHz
  - 1 wait state for frequencies =< 60 MHz
- Flash memory line 128-bit wide with 8-bit ECC on 64-bit word (total 144 bits)
- Accessed via a 64-bit wide bus for write and a 128-bit wide array for read operations
- 1-bit error correction, 2-bit error detection

1.5.6 On-chip SRAM with ECC

The SPC56ELx/SPC564Lx SRAM provides a general-purpose single port memory.

ECC handling is done on a 32-bit boundary for data and it is extended to the address to have the highest possible diagnostic coverage including the array internal address decoder.

1.5.25 System Status and Configuration Module (SSCM)

The SSCM on this device features the following:

- System configuration and status
- Debug port status and debug port enable
- Multiple boot code starting locations out of reset through implementation of search for valid Reset Configuration Half Word
- Sets up the MMU to allow user boot code to execute as either Power Architecture code (default) or as VLE code out of flash memory
- Triggering of device self-tests during reset phase of device boot

1.5.26 FlexCAN

The FlexCAN module is a communication controller implementing the CAN protocol according to Bosch Specification version 2.0B. The CAN protocol was designed to be used primarily as a vehicle serial data bus, meeting the specific requirements of this field: real-time processing, reliable operation in the EMI environment of a vehicle, cost-effectiveness and required bandwidth.

The FlexCAN module provides the following features:

- Full implementation of the CAN protocol specification, version 2.0B
  - Standard data and remote frames
  - Extended data and remote frames
  - 0 to 8 bytes data length
  - Programmable bit rate as fast as 1Mbit/s
- 32 message buffers of 0 to 8 bytes data length
- Each message buffer configurable as receive or transmit buffer, all supporting standard and extended messages
- Programmable loop-back mode supporting self-test operation
- 3 programmable mask registers
- Programmable transmit-first scheme: lowest ID or lowest buffer number
- Time stamp based on 16-bit free-running timer
- Global network time, synchronized by a specific message
- Maskable interrupts
- Independent of the transmission medium (an external transceiver is assumed)
- High immunity to EMI
- Short latency time due to an arbitration scheme for high-priority messages
- Transmit features
  - Supports configuration of multiple mailboxes to form message queues of scalable depth
  - Arbitration scheme according to message ID or message buffer number
  - Internal arbitration to guarantee no inner or outer priority inversion
  - Transmit abort procedure and notification

1.5.28 Serial communication interface module (LINFlexD)

The LINFlexD module (LINFlex with DMA support) on this device features the following:

- Supports LIN Master mode, LIN Slave mode and UART mode
- LIN state machine compliant to LIN1.3, 2.0, and 2.1 specifications
- Manages LIN frame transmission and reception without CPU intervention
- LIN features
  - Autonomous LIN frame handling
  - Message buffer to store as many as 8 data bytes
  - Supports messages as long as 64 bytes
  - Detection and flagging of LIN errors (Sync field, delimiter, ID parity, bit framing, checksum and Time-out errors)
  - Classic or extended checksum calculation
  - Configurable break duration of up to 50-bit times
  - Programmable baud rate prescalers (13-bit mantissa, 4-bit fractional)
  - Diagnostic features (Loop back, LIN bus stuck dominant detection)
  - Interrupt driven operation with 16 interrupt sources
- LIN slave mode features
  - Autonomous LIN header handling
  - Autonomous LIN response handling
- UART mode
  - Full-duplex operation
  - Standard non return-to-zero (NRZ) mark/space format
  - Data buffers with 4-byte receive, 4-byte transmit
  - Configurable word length (8-bit, 9-bit, 16-bit, or 17-bit words)
  - Configurable parity scheme: none, odd, even, always 0
  - Speed as fast as 2 Mbit/s
  - Error detection and flagging (Parity, Noise and Framing errors)
  - Interrupt driven operation with four interrupt sources
  - Separate transmitter and receiver CPU interrupt sources
  - 16-bit programmable baud-rate modulus counter and 16-bit fractional
  - Two receiver wake-up methods
- Support for DMA enabled transfers

1.5.29 Deserial Serial Peripheral Interface (DSPI)

The DSPI modules provide a synchronous serial interface for communication between the SPC56ELx/SPC564Lx and external devices.

A DSPI module provides these features:

- Full duplex, synchronous transfers
- Master or slave operation
- Programmable master bit rates
- Programmable clock polarity and phase

The CTU implements the following features:

- Cross triggering between ADC, FlexPWM, eTimer, and external pins
- Double buffered trigger generation unit with as many as 8 independent triggers generated from external triggers
- Maximum operating frequency less than or equal to 120 MHz
- Trigger generation unit configurable in sequential mode or in triggered mode
- Trigger delay unit to compensate the delay of external low pass filter
- Double buffered global trigger unit allowing eTimer synchronization and/or ADC command generation
- Double buffered ADC command list pointers to minimize ADC-trigger unit update
- Double buffered ADC conversion command list with as many as 24 ADC commands
- Each trigger capable of generating consecutive commands
- ADC conversion command allows control of ADC channel from each ADC, single or synchronous sampling, independent result queue selection
- DMA support with safety features

1.5.35 Cyclic Redundancy Checker (CRC) Unit

The CRC module is a configurable multiple data flow unit to compute CRC signatures on data written to its input register.

The CRC unit has the following features:

- 3 sets of registers to allow 3 concurrent contexts with possibly different CRC computations, each with a selectable polynomial and seed
- Computes 16- or 32-bit wide CRC on the fly (single-cycle computation) and stores result in internal register.
  The following standard CRC polynomials are implemented:
  - $x^8 + x^4 + x^3 + x^2 + 1$ [8-bit CRC]
  - $x^{16} + x^{12} + x^5 + 1$ [16-bit CRC-CCITT]
  - $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$ [32-bit CRC-ethernet(32)]
- Key engine to be coupled with communication periphery where CRC application is added to allow implementation of safe communication protocol
- Offloads core from cycle-consuming CRC and helps checking configuration signature for safe start-up or periodic procedures
- CRC unit connected as peripheral bus on internal peripheral bus
- DMA support

1.5.36 Redundancy Control and Checker Unit (RCCU)

The RCCU checks all outputs of the sphere of replication (addresses, data, control signals). It has the following features:

- Duplicated module to guarantee highest possible diagnostic coverage (check of checker)
- Multiple times replicated IPs are used as checkers on the SoR outputs

Figure 4. SPC56ELx/SPC564Lx LFBGA257 pinout (top view)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
A	$V_{SS\_HV\_IO}$	$V_{SS\_HV\_IO}$	$V_{DD\_HV\_IO}$	H[2]	H[0]	G[14]	D[3]	C[15]	$V_{DD\_HV\_IO}$	A[12]	H[10]	H[14]	A[10]	B[2]	C[10]	$V_{SS\_HV\_IO}$	$V_{SS\_HV\_IO}$
B	$V_{SS\_HV\_IO}$	$V_{SS\_HV\_IO}$	B[6]	A[14]	F[3]	A[9]	D[4]	D[0]	$V_{SS\_HV\_IO}$	H[12]	E[15]	E[14]	B[3]	F[13]	B[0]	$V_{DD\_HV\_IO}$	$V_{SS\_HV\_IO}$
C	$V_{DD\_HV\_IO}$	NC(1)	$V_{SS\_HV\_IO}$	FCCU_F[1]	D[2]	A[13]	$V_{DD\_HV\_REG\_2}$	$V_{DD\_HV\_REG\_2}$	I[0]	JCOMP	H[11]	I[1]	F[14]	B[1]	$V_{SS\_HV\_IO}$	A[4]	F[12]
D	F[5]	F[4]	A[15]	C[6]	$V_{SS\_LV\_COR}$	$V_{DD\_LV\_COR}$	F[0]	$V_{DD\_HV\_IO}$	$V_{SS\_HV\_IO}$	NC	A[11]	E[13]	F[15]	$V_{DD\_HV\_IO}$	V_{PP_TEST}	D[14]	G[3]
E	MDO0	F[6]	D[1]	NMI										NC	C[14]	G[2]	I[3]
F	H[1]	G[12]	A[7]	A[8]		$V_{DD\_LV\_COR}$	$V_{DD\_LV\_COR}$	$V_{DD\_LV\_COR}$	$V_{DD\_LV\_COR}$	$V_{DD\_LV\_COR}$	$V_{DD\_LV\_COR}$	$V_{DD\_LV\_COR}$		NC	C[13]	I[2]	G[4]
G	H[3]	$V_{DD\_HV\_IO}$	C[5]	A[6]		$V_{DD\_LV\_COR}$	$V_{SS\_LV\_COR}$	$V_{SS\_LV\_COR}$	$V_{SS\_LV\_COR}$	$V_{SS\_LV\_COR}$	$V_{SS\_LV\_COR}$	$V_{DD\_LV\_COR}$		D[12]	H[13]	H[9]	G[6]
H	G[13]	$V_{SS\_HV\_IO}$	C[4]	A[5]		V_{DD_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{DD_LV}		V_{SS_LV}	$V_{DD\_HV\_REG\_1}$	$V_{DD\_HV\_FLA}$	H[6]
J	F[7]	G[15]	$V_{DD\_HV\_REG\_0}$	$V_{DD\_HV\_REG\_0}$		V_{DD_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{DD_LV}		V_{DD_LV}	$V_{DD\_HV\_REG\_1}$	$V_{SS\_HV\_FLA}$	H[15]
K	F[9]	F[8]		C[7]		V_{DD_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{DD_LV}		NC	H[8]	H[7]	A[3]
L	F[10]	F[11]	D[9]	NC		V_{DD_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{SS_LV}	V_{DD_LV}		NC	TCK	H[4]	B[4]
M	$V_{DD\_HV\_OSC}$	$V_{DD\_HV\_IO}$	D[8]	NC		V_{DD_LV}	V_{DD_LV}	V_{DD_LV}	V_{DD_LV}	V_{DD_LV}	V_{DD_LV}	V_{DD_LV}		C[11]	B[5]	TMS	H[5]
N	XTAL	$V_{SS\_HV\_IO}$	D[5]	$V_{SS\_LV\_PLL}$										NC	C[12]	A[2]	G[5]
P	$V_{SS\_HV\_OSC}$	RESET	D[6]	$V_{DD\_LV\_PLL}$	$V_{DD\_LV\_COR}$	$V_{SS\_LV\_COR}$	B[8]	NC	$V_{SS\_HV\_IO}$	$V_{DD\_HV\_IO}$	B[14]	$V_{DD\_LV\_COR}$	$V_{SS\_LV\_COR}$	$V_{DD\_HV\_IO}$	G[10]	G[8]	G[7]
R	EXTAL	FCCU_F[0]	$V_{SS\_HV\_IO}$	D[7]	B[7]	E[6]	$V_{DD\_HV\_ADR0}$	B[10]	$V_{DD\_HV\_ADR1}$	B[13]	B[15]	C[0]	BCTRL	A[1]	$V_{SS\_HV\_IO}$	D[11]	G[9]
T	$V_{SS\_HV\_IO}$	$V_{DD\_HV\_IO}$	NC	C[1]	E[5]	E[7]	$V_{SS\_HV\_ADR0}$	B[11]	$V_{SS\_HV\_ADR1}$	E[9]	E[10]	E[12]	E[0]	A[0]	D[10]	$V_{DD\_HV\_IO}$	$V_{SS\_HV\_IO}$
U	$V_{SS\_HV\_IO}$	$V_{SS\_HV\_IO}$	NC	E[4]	C[2]	E[2]	B[9]	B[12]	$V_{DD\_HV\_ADV}$	$V_{SS\_HV\_ADV}$	E[11]	NC	NC	$V_{DD\_HV\_PMU}$	G[11]	$V_{SS\_HV\_IO}$	$V_{SS\_HV\_IO}$
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

1. NC = Not connected (the pin is physically not connected to anything on the device).

Table 3, *Table 4*, and *Table 5* provide the pin function summaries for the 100-pin, 144-pin, and 257-pin packages, respectively, listing all the signals multiplexed to each pin.

Table 3. LQFP100 pin function summary

Pin #	Port/function	Peripheral	Output function	Input function
1	NMI	—		
2	A[6]	SIUL	GPIO[6]	GPIO[6]
		DSPI_1	SCK	SCK
		SIUL	—	EIRQ[6]
3	D[1]	SIUL	GPIO[49]	GPIO[49]
		eTimer_1	ETC[2]	ETC[2]
		CTU_0	EXT_TGR	—
		FlexRay	—	CA_RX
4	A[7]	SIUL	GPIO[7]	GPIO[7]
		DSPI_1	SOUT	—
		SIUL	—	EIRQ[7]
5	C[4]	SIUL	GPIO[36]	GPIO[36]
		DSPI_0	CS0	CS0
		FlexPWM_0	X[1]	X[1]
		SSCM	DEBUG[4]	—
		SIUL	—	EIRQ[22]
6	A[8]	SIUL	GPIO[8]	GPIO[8]
		DSPI_1	—	SIN
		SIUL	—	EIRQ[8]
7	C[5]	SIUL	GPIO[37]	GPIO[37]
		DSPI_0	SCK	SCK
		SSCM	DEBUG[5]	—
		FlexPWM_0	—	FAULT[3]
		SIUL	—	EIRQ[23]
8	A[5]	SIUL	GPIO[5]	GPIO[5]
		DSPI_1	CS0	CS0
		eTimer_1	ETC[5]	ETC[5]
		DSPI_0	CS7	—
		SIUL	—	EIRQ[5]
9	C[7]	SIUL	GPIO[39]	GPIO[39]
		FlexPWM_0	A[1]	A[1]
		SSCM	DEBUG[7]	—
		DSPI_0	—	SIN
10	$V_{DD\_HV\_REG\_0}$	—		
11	$V_{SS\_LV\_COR}$	—		

Table 4. LQFP144 pin function summary (continued)

Pin #	Port/function	Peripheral	Output function	Input function
15	C[7]	SIUL	GPIO[39]	GPIO[39]
		FlexPWM_0	A[1]	A[1]
		SSCM	DEBUG[7]	—
		DSPI_0	—	SIN
16	$V_{DD\_HV\_REG\_0}$	—		
17	$V_{SS\_LV\_COR}$	—		
18	$V_{DD\_LV\_COR}$	—		
19	F[7]	SIUL	GPIO[87]	GPIO[87]
		NPC	MCKO	—
20	F[8]	SIUL	GPIO[88]	GPIO[88]
		NPC	MSEO[1]	—
21	$V_{DD\_HV\_IO}$	—		
22	$V_{SS\_HV\_IO}$	—		
23	F[9]	SIUL	GPIO[89]	GPIO[89]
		NPC	MSEO[0]	—
24	F[10]	SIUL	GPIO[90]	GPIO[90]
		NPC	EVTO	—
25	F[11]	SIUL	GPIO[91]	GPIO[91]
		NPC	—	EVTI
26	D[9]	SIUL	GPIO[57]	GPIO[57]
		FlexPWM_0	X[0]	X[0]
		LINFlexD_1	TXD	—
27	$V_{DD\_HV\_OSC}$	—		
28	$V_{SS\_HV\_OSC}$	—		
29	XTAL	—		
30	EXTAL	—		
31	RESET	—		
32	D[8]	SIUL	GPIO[56]	GPIO[56]
		DSPI_1	CS2	—
		eTimer_1	ETC[4]	ETC[4]
		DSPI_0	CS5	—
		FlexPWM_0	—	FAULT[3]
33	D[5]	SIUL	GPIO[53]	GPIO[53]
		DSPI_0	CS3	—
		FlexPWM_0	—	FAULT[2]

Table 4. LQFP144 pin function summary (continued)

Pin #	Port/function	Peripheral	Output function	Input function
95	$V_{DD\_HV\_REG\_1}$	—		
96	$V_{SS\_HV\_FLA}$	—		
97	$V_{DD\_HV\_FLA}$	—		
98	G[6]	SIUL	GPIO[102]	GPIO[102]
		FlexPWM_0	A[3]	A[3]
99	D[12]	SIUL	GPIO[60]	GPIO[60]
		FlexPWM_0	X[1]	X[1]
		LINFlexD_1	—	RXD
100	G[4]	SIUL	GPIO[100]	GPIO[100]
		FlexPWM_0	B[2]	B[2]
		eTimer_0	—	ETC[5]
101	C[13]	SIUL	GPIO[45]	GPIO[45]
		eTimer_1	ETC[1]	ETC[1]
		CTU_0	—	EXT_IN
		FlexPWM_0	—	EXT_SYNC
102	G[2]	SIUL	GPIO[98]	GPIO[98]
		FlexPWM_0	X[2]	X[2]
		DSPI_1	CS1	—
103	C[14]	SIUL	GPIO[46]	GPIO[46]
		eTimer_1	ETC[2]	ETC[2]
		CTU_0	EXT_TGR	—
104	G[3]	SIUL	GPIO[99]	GPIO[99]
		FlexPWM_0	A[2]	A[2]
		eTimer_0	—	ETC[4]
105	D[14]	SIUL	GPIO[62]	GPIO[62]
		FlexPWM_0	B[1]	B[1]
		eTimer_0	—	ETC[3]
106	F[12]	SIUL	GPIO[92]	GPIO[92]
		eTimer_1	ETC[3]	ETC[3]
		SIUL	—	EIRQ[30]
107	V_{PP_TEST}(1)	—		

Table 5. LFBGA257 pin function summary (continued)

Pin #	Port/function	Peripheral	Output function	Input function
U15	G[11]	SIUL	GPIO[107]	GPIO[107]
		FlexRay	DBG3	—
		FlexPWM_0	—	FAULT[3]
U16	$V_{SS\_HV\_IO\_RING}$	—		
U17	$V_{SS\_HV\_IO\_RING}$	—		

1. V_{PP_TEST} should always be tied to ground (V_{SS}) for normal operations.

2.2 Supply pins

Table 6. Supply pins

Supply		Pin #		
Symbol	Description	100 pkg	144 pkg	257 pkg
VREG control and power supply pins				
BCTRL	Voltage regulator external NPN ballast base control pin	47	69	R13
$V_{DD\_LV\_COR}$	Core logic supply	48	70	VDD_LV(1)
$V_{SS\_LV\_COR}$	Core regulator ground	49	71	VSS_LV(2)
$V_{DD\_HV\_PMU}$	Voltage regulator supply	50	72	U14
ADC_0/ADC_1 reference voltage and ADC supply				
$V_{DD\_HV\_ADR0}$	ADC_0 high reference voltage	33	50	R7
$V_{SS\_HV\_ADR0}$	ADC_0 low reference voltage	34	51	T7
$V_{DD\_HV\_ADR1}$	ADC_1 high reference voltage	39	56	R9
$V_{SS\_HV\_ADR1}$	ADC_1 low reference voltage	40	57	T9
$V_{DD\_HV\_ADV}$	ADC voltage supply for ADC_0 and ADC_1	41	58	U9
$V_{SS\_HV\_ADV}$	ADC ground for ADC_0 and ADC_1	42	59	U10
Power supply pins (3.3 V)				
$V_{DD\_HV\_IO}$	3.3 V Input/Output supply voltage	—	6	VDD_HV(3)
$V_{SS\_HV\_IO}$	3.3 V Input/Output ground	—	7	VSS_HV(4)
$V_{DD\_HV\_REG\_0}$	VDD_HV_REG_0	10	16	J3
$V_{DD\_HV\_IO}$	3.3 V Input/Output supply voltage	13	21	VDD_HV(3)
$V_{SS\_HV\_IO}$	3.3 V Input/Output ground	14	22	VSS_HV(4)
$V_{DD\_HV\_OSC}$	Crystal oscillator amplifier supply voltage	16	27	M1
$V_{SS\_HV\_OSC}$	Crystal oscillator amplifier ground	17	28	P1
$V_{SS\_HV\_IO}$	3.3 V Input/Output ground	62	90	VSS_HV(4)
$V_{DD\_HV\_IO}$	3.3 V Input/Output supply voltage	63	91	VDD_HV(3)

Table 8. Pin muxing (continued)

Port name	PCR	Peripheral	Alternate output function	Output mux sel	Input functions	Input mux select	Weak pull config during reset	Pad speed(1)		Pin #		
								SRC = 1	SRC = 0	100 pkg	144 pkg	257 pkg
A[3]	PCR[3]	SIUL	GPIO[3]	ALT0	GPIO[3]	—	Pull down	M	S	64	92	K17
		eTimer_0	ETC[3]	ALT1	ETC[3]	PSMI[38]; PADSEL=0						
		DSPI_2	CS0	ALT2	CS0	PSMI[3]; PADSEL=0						
		FlexPWM_0	B[3]	ALT3	B[3]	PSMI[27]; PADSEL=0						
		MC_RGM	—	—	ABS[2]	—						
		SIUL	—	—	EIRQ[3]	—						
A[4]	PCR[4]	SIUL	GPIO[4]	ALT0	GPIO[4]	—	Pull down	M	S	75	108	C16
		eTimer_1	ETC[0]	ALT1	ETC[0]	PSMI[9]; PADSEL=0						
		DSPI_2	CS1	ALT2	—	—						
		eTimer_0	ETC[4]	ALT3	ETC[4]	PSMI[7]; PADSEL=0						
		MC_RGM	—	—	FAB	—						
		SIUL	—	—	EIRQ[4]	—						
A[5]	PCR[5]	SIUL	GPIO[5]	ALT0	GPIO[5]	—	—	M	S	8	14	H4
		DSPI_1	CS0	ALT1	CS0	—						
		eTimer_1	ETC[5]	ALT2	ETC[5]	PSMI[14]; PADSEL=0						
		DSPI_0	CS7	ALT3	—	—						
		SIUL	—	—	EIRQ[5]	—						

Table 8. Pin muxing (continued)

Port name	PCR	Peripheral	Alternate output function	Output mux sel	Input functions	Input mux select	Weak pull config during reset	Pad speed(1)		Pin #		
								SRC = 1	SRC = 0	100 pkg	144 pkg	257 pkg
C[10]	PCR[42]	SIUL	GPIO[42]	ALT0	GPIO[42]	—	—	M	S	78	111	A15
		DSPI_2	CS2	ALT1	—	—						
		FlexPWM_0	A[3]	ALT3	A[3]	PSMI[23]; PADSEL=1						
		FlexPWM_0	—	—	FAULT[1]	PSMI[17]; PADSEL=0						
C[11]	PCR[43]	SIUL	GPIO[43]	ALT0	GPIO[43]	—	—	M	S	55	80	M14
		eTimer_0	ETC[4]	ALT1	ETC[4]	PSMI[7]; PADSEL=1						
		DSPI_2	CS2	ALT2	—	—						
C[12]	PCR[44]	SIUL	GPIO[44]	ALT0	GPIO[44]	—	—	M	S	56	82	N15
		eTimer_0	ETC[5]	ALT1	ETC[5]	PSMI[8]; PADSEL=0						
		DSPI_2	CS3	ALT2	—	—						
C[13]	PCR[45]	SIUL	GPIO[45]	ALT0	GPIO[45]	—	—	M	S	71	101	F15
		eTimer_1	ETC[1]	ALT1	ETC[1]	PSMI[10]; PADSEL=0						
		CTU_0	—	—	EXT_IN	PSMI[0]; PADSEL=0						
		FlexPWM_0	—	—	EXT_SYNC	PSMI[15]; PADSEL=0						
C[14]	PCR[46]	SIUL	GPIO[46]	ALT0	GPIO[46]	—	—	M	S	72	103	E15
		eTimer_1	ETC[2]	ALT1	ETC[2]	PSMI[11]; PADSEL=1						
		CTU_0	EXT_TGR	ALT2	—	—						

Table 8. Pin muxing (continued)

Port name	PCR	Peripheral	Alternate output function	Output mux sel	Input functions	Input mux select	Weak pull config during reset	Pad speed(1)		Pin #		
								SRC = 1	SRC = 0	100 pkg	144 pkg	257 pkg
D[2]	PCR[50]	SIUL	GPIO[50]	ALT0	GPIO[50]	—	—	M	S	—	140	C5
		eTimer_1	ETC[3]	ALT2	ETC[3]	PSMI[12]; PADSEL=1						
		FlexPWM_0	X[3]	ALT3	X[3]	PSMI[30]; PADSEL=0						
		FlexRay	—	—	CB_RX	—						
D[3]	PCR[51]	SIUL	GPIO[51]	ALT0	GPIO[51]	—	—	SYM	S	89	128	A7
		FlexRay	CB_TX	ALT1	—	—						
		eTimer_1	ETC[4]	ALT2	ETC[4]	PSMI[13]; PADSEL=1						
		FlexPWM_0	A[3]	ALT3	A[3]	PSMI[23]; PADSEL=2						
D[4]	PCR[52]	SIUL	GPIO[52]	ALT0	GPIO[52]	—	—	SYM	S	90	129	B7
		FlexRay	CB_TR_EN	ALT1	—	—						
		eTimer_1	ETC[5]	ALT2	ETC[5]	PSMI[14]; PADSEL=2						
		FlexPWM_0	B[3]	ALT3	B[3]	PSMI[27]; PADSEL=2						
D[5]	PCR[53]	SIUL	GPIO[53]	ALT0	GPIO[53]	—	—	M	S	22	33	N3
		DSPI_0	CS3	ALT1	—	—						
		FlexPWM_0	—	—	FAULT[2]	PSMI[18]; PADSEL=0						

Table 8. Pin muxing (continued)

Port name	PCR	Peripheral	Alternate output function	Output mux sel	Input functions	Input mux select	Weak pull config during reset	Pad speed(1)		Pin #		
								SRC = 1	SRC = 0	100 pkg	144 pkg	257 pkg
H[11]	PCR[123]	SIUL	GPIO[123]	ALT0	GPIO[123]	—	—	M	S	—	—	C11
		FlexPWM_1	A[2]	ALT1	A[2]	—						
H[12]	PCR[124]	SIUL	GPIO[124]	ALT0	GPIO[124]	—	—	M	S	—	—	B10
		FlexPWM_1	B[2]	ALT1	B[2]	—						
H[13]	PCR[125]	SIUL	GPIO[125]	ALT0	GPIO[125]	—	—	M	S	—	—	G15
		FlexPWM_1	X[3]	ALT1	X[3]	—						
		eTimer_2	ETC[3]	ALT2	ETC[3]	PSMI[42]; PADSEL=0						
H[14]	PCR[126]	SIUL	GPIO[126]	ALT0	GPIO[126]	—	—	M	S	—	—	A12
		FlexPWM_1	A[3]	ALT1	A[3]	—						
		eTimer_2	ETC[4]	ALT2	ETC[4]	—						
H[15]	PCR[127]	SIUL	GPIO[127]	ALT0	GPIO[127]	—	—	M	S	—	—	J17
		FlexPWM_1	B[3]	ALT1	B[3]	—						
		eTimer_2	ETC[5]	ALT2	ETC[5]	—						
Port I												
I[0]	PCR[128]	SIUL	GPIO[128]	ALT0	GPIO[128]	—	—	M	S	—	—	C9
		eTimer_2	ETC[0]	ALT1	ETC[0]	PSMI[39]; PADSEL=1						
		DSPI_0	CS4	ALT2	—	—						
		FlexPWM_1	—	—	FAULT[0]	—						

Table 22. Current consumption characteristics (continued)

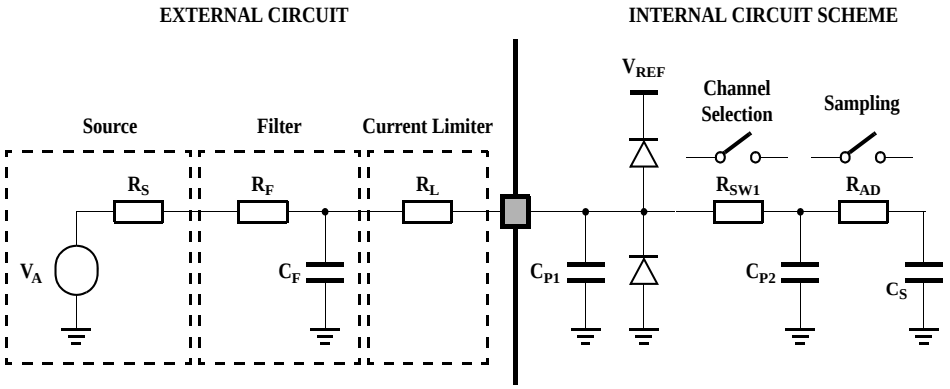
Symbol		Parameter	Conditions[1]	Min	Typ	Max	Unit
$I_{DD\_LV\_TYP}$ + $I_{DD\_LV\_PLL}$[2]	T	Operating current	1.2V supplies Tj=105°C $V_{DD\_LV\_COR}$ = 1.2V LSM mode	—	—	275	mA
			1.2V supplies Tj=125°C $V_{DD\_LV\_COR}$ = 1.2V LSM mode	—	—	299	mA
$I_{DD\_LV\_TYP}$ + $I_{DD\_LV\_PLL}$[2]	T	Operating current	1.2V supplies Tj=105°C $V_{DD\_LV\_COR}$ = 1.2V DPM Mode	—	—	189	mA
			1.2V supplies Tj=125°C $V_{DD\_LV\_COR}$ = 1.2V DPM Mode	—	—	214	mA
			1.2V supplies Tj=150°C $V_{DD\_LV\_COR}$ = 1.2V DPM Mode	—	—	235	mA
$I_{DD\_LV\_STOP}$	T	Operating current in V_{DD} STOP mode	T_J = 25 °C $V_{DD\_LV\_COR}$ = 1.32 V	—	—	20	mA
	T		T_J = 55 °C $V_{DD\_LV\_COR}$ = 1.32 V	—	—	57	
	P		T_J = 150 °C $V_{DD\_LV\_COR}$ = 1.32 V	—	—	105	
$I_{DD\_LV\_HALT}$	T	Operating current in V_{DD} HALT mode	T_J = 25 °C $V_{DD\_LV\_COR}$ = 1.32 V	—	—	25	mA
	T		T_J = 55 °C $V_{DD\_LV\_COR}$ = 1.32 V	—	—	64	
	P		T_J = 150 °C $V_{DD\_LV\_COR}$ = 1.32 V	—	—	115	
$I_{DD\_HV\_ADC}$[3], [4]	T	Operating current	T_J = 150 °C 120 MHz ADC operating at 60 MHz $V_{DD\_HV\_ADC}$ = 3.6 V	—	—	10	mA

Equation 4:

$$V_A \bullet \frac{R_S + R_F}{R_{EQ}} < \frac{1}{2} LSB$$

Equation 4 generates a constraint for external network design, in particular on resistive path. Internal switch resistances (R_{SW} and R_{AD}) can be neglected with respect to external resistances.

Figure 10. Input Equivalent Circuit



Legend:

R_S Source Impedance

R_F Filter Resistance

C_F Filter Capacitance

R_L Current Limiter Resistance

R_{SW1} Channel Selection Switch Impedance

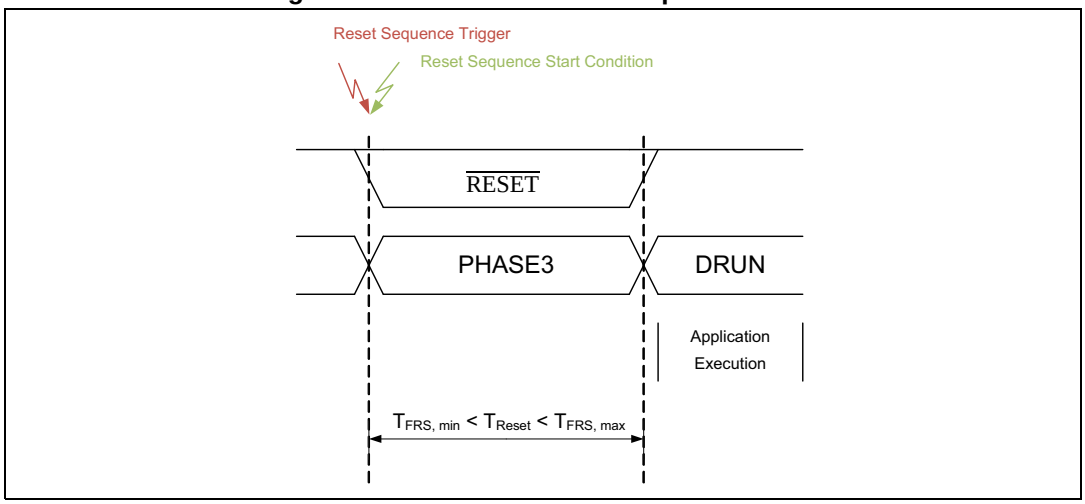
R_{AD} Sampling Switch Impedance

C_P Pin Capacitance (two contributions, C_{P1} and C_{P2})

C_S Sampling Capacitance

A second aspect involving the capacitance network shall be considered. Assuming the three capacitances C_F, C_{P1} and C_{P2} are initially charged at the source voltage V_A (refer to the equivalent circuit reported in *Figure 10*): A charge sharing phenomenon is installed when the sampling phase is started (A/D switch close).

Figure 18. Functional Reset Sequence Short



The reset sequences shown in *Figure 17* and *Figure 18* are triggered by functional reset events. RESET is driven low during these two reset sequences ***only if*** the corresponding functional reset source (which triggered the reset sequence) was enabled to drive RESET low for the duration of the internal reset sequence(c).

3.20.3 Reset sequence trigger mapping

The following table shows the possible trigger events for the different reset sequences. It specifies the reset sequence start conditions as well as the reset sequence end indications that are the basis for the timing data provided in *Table 33*.

Table 34. Reset sequence trigger — reset sequence

Reset Sequence Trigger	Reset Sequence Start Condition	Reset Sequence End Indication	Reset Sequence				
			Destructive Reset Sequence, BIST enabled(1)	***Destructive Reset Sequence, BIST disabled***(1)	***External Reset Sequence Long, BIST enabled***	***Functional Reset Sequence Long***	***Functional Reset Sequence Short***
All internal destructive reset sources (LVDs or internal HVD during power-up and during operation)	*Destructive reset*	Release of RESET(2)	triggers		cannot trigger	cannot trigger	cannot trigger
Assertion of RESET(3)	*External reset via RESET*		cannot trigger		triggers(4)	triggers(5)	triggers(6)

c. See RGM_FBRE register for more details.

Table 45. Document revision history

Date	Revision	Changes
01-Aug-2012	8	Editorial changes. In the “Block diagram” section, removed one PMU from the figure. In the 257-pin pinout figure, changed cut2 to cut2/3 in Notes. In the pin function summary table, changed cut2 to cut2/3. In the “System pins” table: – Added Note regarding Open Drain Enable. – Added description to RESET pin. In the pin-muxing table: – Added Note about Open Drain. – Changed cut2 to cut2/3. – Changed all entries of column 'Weak pull config during reset’ to ' - ' , except for PCR[2], PCR[3], PCR[4] and PCR[21]. In the “Absolute maximum ratings” table: – Removed the “$V_{SS\_HV\_REG}$” row. – Added the footnote “Internal structures hold the input voltage...” to the V_{IN} maximum specifications. In the “Recommended operating conditions” table, removed the “$V_{SS\_HV\_REG}$” row. In the “Thermal characteristics” section: – Added the “Thermal characteristics for LQFP100 package“ table. – Updated values and footnote 1 in the 144 package table. – Updated footnote 1 in the 257 package table. In the “Supply current characteristics“ table: – Added footnote 1 to parameter “$I_{DD\_LV\_TYP}$ + $I_{DD\_LV\_PLL}$“ (symbol “T”). – Changed “$I_{DD\_LV\_STOP}$” at 150C from 80mA to 72mA. – Changed “$I_{DD\_LV\_HALT}$” at 150C from 72mA to 80mA. In the “FMPLL electrical characteristics” table: – Deleted the footnote “This value is true when operating at frequencies above 60 MHz...” from the specification for f_{CS} and f_{DS}. – Changed “f_{SYS}” to “$f_{FMPLLOUT}$” in the entries for the C_{JITTER}, f_{LCK}, f_{UL}, f_{CS}, and f_{DS} specifications. In the “ADC conversion characteristics” table: – Revised the entry for $TUE_{IS1WINJ}$ (was P/T and “Total unadjusted error for IS1WINJ”, is T and “Total unadjusted error for IS1WINJ (single ADC channels)”). – Revised the entry for $TUE_{IS1WWINJ}$ (was “Total unadjusted error for IS1WWINJ”, is “Total unadjusted error for IS1WWINJ (double ADC channels)”). In the “Temperature sensor electrical characteristics“ table, for T_J = T_A to 125 °C, changed Min/Max from values -7/+7 to -10/+10. In the “Input Impedance and ADC Accuracy“ section: – Changed C_S in the text from 3 pF to 7.5 pF. – Changed R_{eq} in the text from 330 kΩ to 133 kΩ. – Removed R_L, R_{SW}, and R_{AD} from the external network design constraint equation and the sentence immediately preceding it. – Changed the C_F constraint value equation constant from 2048 to 8192. In the “ADC conversion characteristics“ table, changed INL Min/Max values from -2/+2 to -3/+3.

Table 45. Document revision history

Date	Revision	Changes
01-Aug-2012	8 (cont.)	Changed min value to '-72' for symbol 'THD'. – In *Table 27: ADC conversion characteristics*, changed ADC specification parameter 'THD' minimum limit from -72 to -65dB. – In *Table 28: Flash memory program and erase electrical specifications*, changes done are as follows: $T_{DWPROGRAM}$, changed typical value from '39' to '38'. $T_{PPROGRAM}$, changed typical value from '48' to '45' and initial max value from '100' to '160'. $T_{16KPPERASE}$, inserted typical value '270' and factory avg '1000'. $T_{48KPPERASE}$, inserted typical value '625' and factory avg '1500'. $T_{64KPPERASE}$, inserted typical value '800' and factory avg '1800'. $T_{128KPPERASE}$, inserted typical value '1500' and factory avg '2600'. $T_{256KPPERASE}$, inserted typical value '3000' and factory avg '5200'. Updated table footnote and removed min column in *Table 28: Flash memory program and erase electrical specifications* – In *Table 29: Flash memory timing*, added symbol T_{PSRT} ,T_{ESRT} and added table footnote for T_{PSRT} ,T_{ESRT} . – Added *Table 31: SPC56XL60/54 SWG Specifications* – In *Table 31: SPC56XL60/54 SWG Specifications* Added table footnote for Common Mode. Changed text from "internal device pad resistance" to "internal device routing resistance". – Added *Figure 28: Nexus EVTI Input Pulse Width* in *Section 3.21.4: Nexus timing*". – In *Table 32: Pad AC specifications (3.3 V, IPP_HVE = 0)*, removed the row of pad "Pull Up/Downc(3.6 V max)". – In *Figure 44*, updated part numbers (changed 'PPC' to 'SPC' and 'F0' to 'F2'). – Replaced *Figure 42*, *Figure 43* with the new versions. – In*Table 20*,changed the symbol of spec external decoupling capacitor from SR to C_{ext}. In *Table 6*, changed the ESR range in note text to 1 mΩ to 100 mΩ from 30 mΩ to 150 mΩ. – In *Section 1.5.32: Sine Wave Generator (SWG)*" removed the following text: Frequency range from 1kHz to 50kHz. Sine wave amplitude from 0.47 V to 2.26 V. – In *Table 22*,changed symbol from 'C' to 'T' , added "operating current" to the parameter and updated the maximum value for five additional RunIDD parameters. – In *Table 22*, changed "Conditions" from '1.2 V supplies' to '1.2 V supplies during LBIST (full LBIST configuration)' for all the parameters. Removed Table "SWG electrical characteristics".