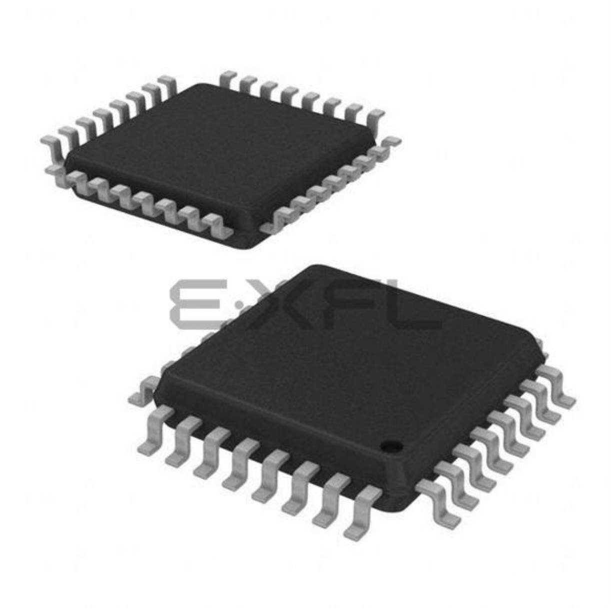
NXP USA Inc. - S9S12VR16F0VLC Datasheet



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	12V1
Core Size	16-Bit
Speed	25MHz
Connectivity	IrDA, LINbus, SCI, SPI
Peripherals	LVD, POR, PWM, WDT
Number of I/O	16
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3.13V ~ 5.5V
Data Converters	A/D 2x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LQFP
Supplier Device Package	32-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/s9s12vr16f0vlc

Chapter 5
Background Debug Module (S12SBDMV1)

Chapter 6
S12S Debug Module (S12DBGV2)

2.3.34 Port L Input Register (PTIL)

Address 0x0269 Access: User read only[1]

	7	6	5	4	3	2	1	0
R	0	0	0	0	PTIL3	PTIL2	PTIL1	PTIL0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-49. Port L Input Register (PTIL)

[1] Read: Anytime
Write: No Write

Table 2-33. PTIL - Register Field Descriptions

Field	Description
3-0 PTIL	**PorT Input data register port L** — A read returns the synchronized input state if the associated pin is used in digital mode, that is the related DIENL bit is set to 1 and the pin is not used in analog mode (PTAL[PTAENL]=0). See Section 2.3.36, "Port L Analog Access Register (PTAL)". A one is read in any other case[1].

[1] Refer to PTTEL bit description in Section 2.3.36, "Port L Analog Access Register (PTAL) for an override condition.

2.3.35 Port L Digital Input Enable Register (DIENL)

Address 0x26A Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	0	0	0	0	DIENL3	DIENL2	DIENL1	DIENL0
W								
Reset	0	0	0	0	0	0	0	0

Figure 2-50. Port L Digital Input Enable Register (DIENL)

[1] Read: Anytime
Write: Anytime

Table 2-34. DIENL Register Field Descriptions

Field	Description
3-0 DIENL	**Digital Input ENable port L** — Input buffer control This bit controls the HVI digital input function. If set to 1 the input buffers are enabled and the pin can be used with the digital function. If the analog input function is enabled (PTAL[PTAENL]=1) the input buffer of the selected HVI pin is forced off[1] in run mode and is released to be active in stop mode only if DIENL=1. 1 Associated pin digital input is enabled if not used as analog input in run mode[1] 0 Associated pin digital input is disabled[1]

[1] Refer to PTTEL bit description in Section 2.3.36, "Port L Analog Access Register (PTAL) for an override condition.

The page value for the program page window is stored in the PPAGE register. The value of the PPAGE register can be read or written by normal memory accesses as well as by the CALL and RTC instructions.

Control registers, vector space and parts of the on-chip memories are located in unpaged portions of the 64KB local CPU address space.

The starting address of an interrupt service routine must be located in unpaged memory unless the user is certain that the PPAGE register will be set to the appropriate value when the service routine is called. However an interrupt service routine can call other routines that are in paged memory. The upper 16KB block of the local CPU memory space (0xC000–0xFFFF) is unpaged. It is recommended that all reset and interrupt vectors point to locations in this area or to the other unmapped pages sections of the local CPU memory map.

Expansion of the BDM Local Address Map

PPAGE and BDMPPR register is also used for the expansion of the BDM local address to the global address. These registers can be read and written by the BDM.

The BDM expansion scheme is the same as the CPU expansion scheme.

The four BDMPPR Program Page index bits allow access to the full 256KB address map that can be accessed with 18 address bits.

The BDM program page index register (BDMPPR) is used only when the feature is enabled in BDM and, in the case the CPU is executing a firmware command which uses CPU instructions, or by a BDM hardware commands. See the BDM Block Guide for further details. (see Figure 3-10).

4.3.2.15 Autonomous Periodical Interrupt Control Register (CPMUAPICTL)

The CPMUAPICTL register allows the configuration of the autonomous periodical interrupt features.

0x02F2

	7	6	5	4	3	2	1	0
R	APICLK	0	0	APIES	APIEA	APIFE	APIE	APIF
W		▨	▨					
Reset	0	0	0	0	0	0	0	0

▨ = Unimplemented or Reserved

Figure 4-19. Autonomous Periodical Interrupt Control Register (CPMUAPICTL)

Read: Anytime

Write: Anytime

Table 4-18. CPMUAPICTL Field Descriptions

Field	Description
7 APICLK	**Autonomous Periodical Interrupt Clock Select Bit** — Selects the clock source for the API. Writable only if APIFE = 0. APICLK cannot be changed if APIFE is set by the same write operation. 0 Autonomous Clock (ACLK) used as source. 1 Bus Clock used as source.
4 APIES	**Autonomous Periodical Interrupt External Select Bit** — Selects the waveform at the external pin API_EXTCLK as shown in Figure 4-20. See device level specification for connectivity of API_EXTCLK pin. 0 If APIEA and APIFE are set, at the external pin API_EXTCLK periodic high pulses are visible at the end of every selected period with the size of half of the minimum period (APIR=0x0000 in Table 4-22). 1 If APIEA and APIFE are set, at the external pin API_EXTCLK a clock is visible with 2 times the selected API Period.
3 APIEA	**Autonomous Periodical Interrupt External Access Enable Bit** — If set, the waveform selected by bit APIES can be accessed externally. See device level specification for connectivity. 0 Waveform selected by APIES can not be accessed externally. 1 Waveform selected by APIES can be accessed externally, if APIFE is set.
2 APIFE	**Autonomous Periodical Interrupt Feature Enable Bit** — Enables the API feature and starts the API timer when set. 0 Autonomous periodical interrupt is disabled. 1 Autonomous periodical interrupt is enabled and timer starts running.
1 APIE	**Autonomous Periodical Interrupt Enable Bit** 0 API interrupt request is disabled. 1 API interrupt will be requested whenever APIF is set.
0 APIF	**Autonomous Periodical Interrupt Flag** — APIF is set to 1 when the in the API configured time has elapsed. This flag can only be cleared by writing a 1.Writing a 0 has no effect. If enabled (APIE = 1), APIF causes an interrupt request. 0 API time-out has not yet occurred. 1 API time-out has occurred.

Table 6-30. DBGADHM Field Descriptions

Field	Description
7–0 Bits[15:8]	**Comparator Data High Mask Bits** — The Comparator data high mask bits control whether the selected comparator compares the data bus bits [15:8] to the corresponding comparator data compare bits. Data bus comparisons are only performed if the TAG bit in DBGACTL is clear 0 Do not compare corresponding data bit Any value of corresponding data bit allows match. 1 Compare corresponding data bit

6.3.2.8.8 Debug Comparator Data Low Mask Register (DBGADLM)

Address: 0x002F

	7	6	5	4	3	2	1	0
R W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset	0	0	0	0	0	0	0	0

Figure 6-22. Debug Comparator Data Low Mask Register (DBGADLM)

Read: If COMRV[1:0] = 00

Write: If COMRV[1:0] = 00 and DBG not armed.

Table 6-31. DBGADLM Field Descriptions

Field	Description
7–0 Bits[7:0]	**Comparator Data Low Mask Bits** — The Comparator data low mask bits control whether the selected comparator compares the data bus bits [7:0] to the corresponding comparator data compare bits. Data bus comparisons are only performed if the TAG bit in DBGACTL is clear 0 Do not compare corresponding data bit. Any value of corresponding data bit allows match 1 Compare corresponding data bit

6.4 Functional Description

This section provides a complete functional description of the DBG module. If the part is in secure mode, the DBG module can generate breakpoints but tracing is not possible.

6.4.1 S12DBGV2 Operation

Arming the DBG module by setting ARM in DBGC1 allows triggering the state sequencer, storing of data in the trace buffer and generation of breakpoints to the CPU. The DBG module is made up of four main blocks, the comparators, control logic, the state sequencer, and the trace buffer.

The comparators monitor the bus activity of the CPU. All comparators can be configured to monitor address bus activity. Comparator A can also be configured to monitor databus activity and mask out individual data bus bits during a compare. Comparators can be configured to use R/W and word/byte access qualification in the comparison. A match with a comparator register value can initiate a state sequencer transition to another state (see Figure 6-24). Either forced or tagged matches are possible. Using

8.3.2.12.2 Right Justified Result Data (DJM=1)

Module Base +
0x0010 = ATDDR0, 0x0012 = ATDDR1, 0x0014 = ATDDR2, 0x0016 = ATDDR3
0x0018 = ATDDR4, 0x001A = ATDDR5

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	0	0	0	0	Result-Bit[11:0]											
W	▨	▨	▨	▨												
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

▨ = Unimplemented or Reserved

Figure 8-15. Right justified ATD conversion result register (ATDDR*n*)

Table 8-22 shows how depending on the A/D resolution the conversion result is transferred to the ATD result registers for right justified data. Compare is always done using all 12 bits of both the conversion result and the compare value in ATDDRn.

Table 8-22. Conversion result mapping to ATDDRn

A/D resolution	DJM	conversion result mapping to ATDDR*n*
8-bit data	1	Result-Bit[7:0] = result, Result-Bit[11:8]=0000
10-bit data	1	Result-Bit[9:0] = result, Result-Bit[11:10]=00

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
PWMPER1[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMPER2[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMPER3[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMPER4[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMPER5[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMPER6[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMPER7[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY0[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY1[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY2[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY3[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY4[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY5[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY6[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								
PWMDTY7[2]	R	Bit 7	6	5	4	3	2	1	Bit 0
	W								

▨ = Unimplemented or Reserved

Figure 9-2. The scalable PWM Register Summary (Sheet 3 of 4)

Clock Source = bus clock, where bus clock= 10 MHz (100 ns period)

PPOLx = 0

PWMPERx = 4

PWMDTYx = 1

PWMx Frequency = 10 MHz/8 = 1.25 MHz

PWMx Period = 800 ns

PWMx Duty Cycle = 3/4 *100% = 75%

Shown in Figure 9-20 is the output waveform generated.

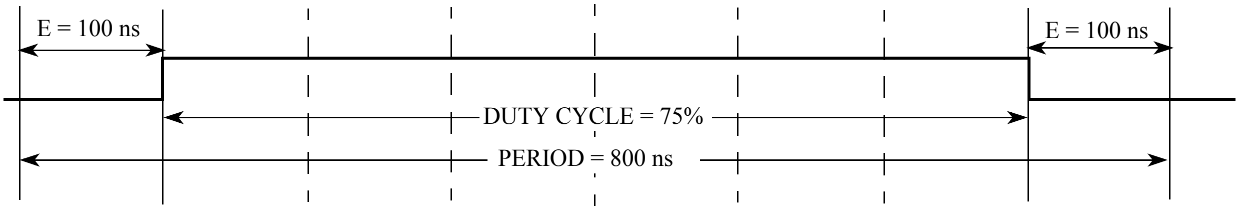


Figure 9-20. PWM Center Aligned Output Example Waveform

9.4.2.7 PWM 16-Bit Functions

The scalable PWM timer also has the option of generating up to 8-channels of 8-bits or 4-channels of 16-bits for greater PWM resolution. This 16-bit channel option is achieved through the concatenation of two 8-bit channels.

The PWMCTL register contains four control bits, each of which is used to concatenate a pair of PWM channels into one 16-bit channel. Channels 6 and 7 are concatenated with the CON67 bit, channels 4 and 5 are concatenated with the CON45 bit, channels 2 and 3 are concatenated with the CON23 bit, and channels 0 and 1 are concatenated with the CON01 bit.

NOTE

Change these bits only when both corresponding channels are disabled.

When channels 6 and 7 are concatenated, channel 6 registers become the high order bytes of the double byte channel, as shown in Figure 9-21. Similarly, when channels 4 and 5 are concatenated, channel 4 registers become the high order bytes of the double byte channel. When channels 2 and 3 are concatenated, channel 2 registers become the high order bytes of the double byte channel. When channels 0 and 1 are concatenated, channel 0 registers become the high order bytes of the double byte channel.

When using the 16-bit concatenated mode, the clock source is determined by the low order 8-bit channel clock select control bits. That is channel 7 when channels 6 and 7 are concatenated, channel 5 when channels 4 and 5 are concatenated, channel 3 when channels 2 and 3 are concatenated, and channel 1 when channels 0 and 1 are concatenated. The resulting PWM is output to the pins of the corresponding low order 8-bit channel as also shown in Figure 9-21. The polarity of the resulting PWM output is controlled by the PPOLx bit of the corresponding low order 8-bit channel as well.

10.1.2 Features

The SCI includes these distinctive features:

- Full-duplex or single-wire operation
- Standard mark/space non-return-to-zero (NRZ) format
- Selectable IrDA 1.4 return-to-zero-inverted (RZI) format with programmable pulse widths
- 16-bit baud rate selection
- Programmable 8-bit or 9-bit data format
- Separately enabled transmitter and receiver
- Programmable polarity for transmitter and receiver
- Programmable transmitter output parity
- Two receiver wakeup methods:
  - Idle line wakeup
  - Address mark wakeup
- Interrupt-driven operation with eight flags:
  - Transmitter empty
  - Transmission complete
  - Receiver full
  - Idle receiver input
  - Receiver overrun
  - Noise error
  - Framing error
  - Parity error
  - Receive wakeup on active edge
  - Transmit collision detect supporting LIN
  - Break Detect supporting LIN
- Receiver framing error detection
- Hardware parity checking
- 1/16 bit-time noise detection

10.1.3 Modes of Operation

The SCI functions the same in normal, special, and emulation modes. It has two low power modes, wait and stop modes.

- Run mode
- Wait mode
- Stop mode

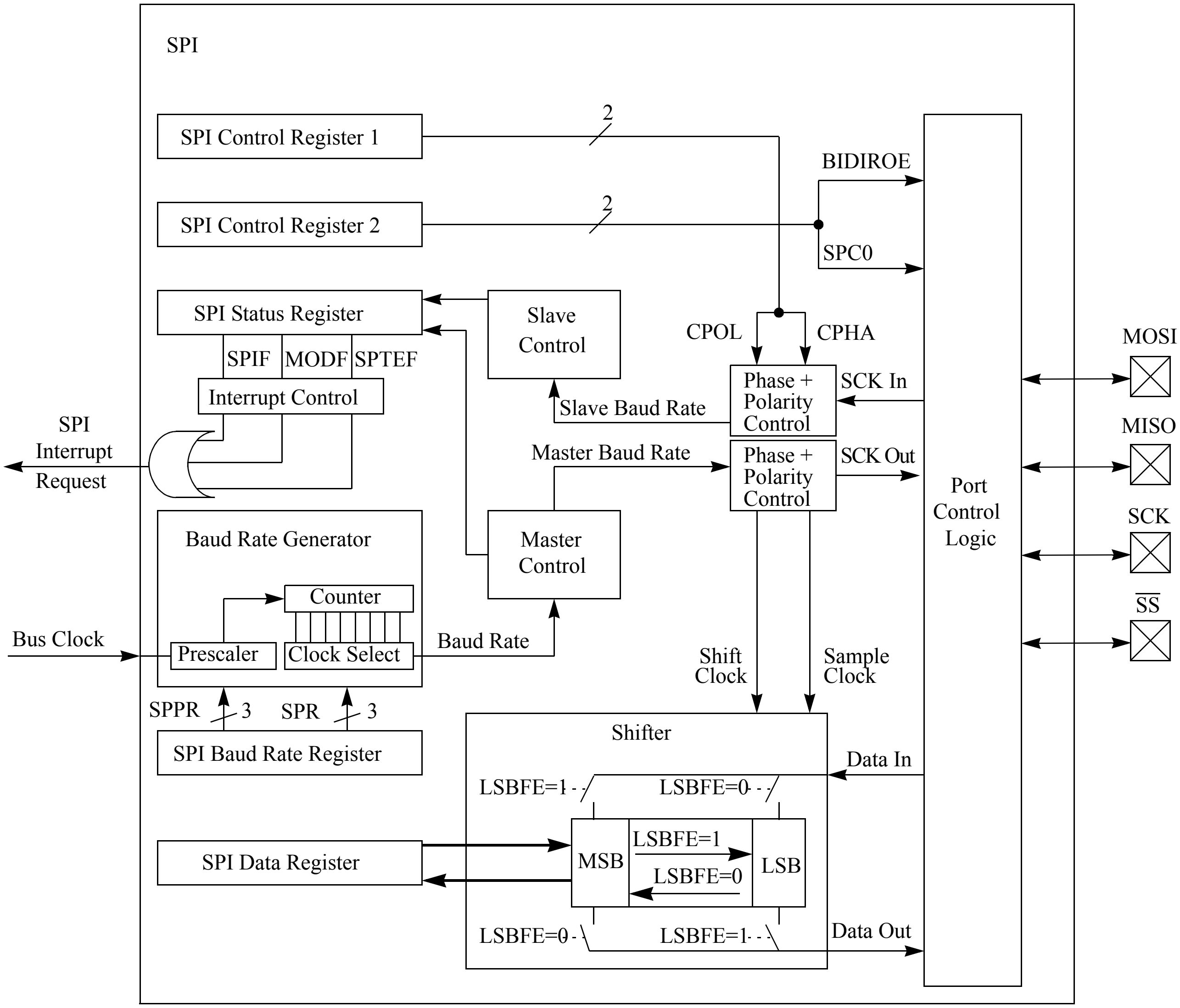


Figure 11-1. SPI Block Diagram

11.2 External Signal Description

This section lists the name and description of all ports including inputs and outputs that do, or may, connect off chip. The SPI module has a total of four external pins.

11.2.1 MOSI — Master Out/Slave In Pin

This pin is used to transmit data out of the SPI module when it is configured as a master and receive data when it is configured as slave.

11.2.2 MISO — Master In/Slave Out Pin

This pin is used to transmit data out of the SPI module when it is configured as a slave and receive data when it is configured as master.

Only bits related to implemented channels are valid.

Register Name		Bit 7	6	5	4	3	2	1	Bit 0
0x0000 TIOS	R W	RESERVED	RESERVED	RESERVED	RESERVED	IOS3	IOS2	IOS1	IOS0
0x0001 CFORC	R	0	0	0	0	0	0	0	0
	W	RESERVED	RESERVED	RESERVED	RESERVED	FOC3	FOC2	FOC1	FOC0
0x0004 TCNTH	R W	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8
0x0005 TCNTL	R W	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0
0x0006 TSCR1	R	TEN	TSWAI	TSFRZ	TFFCA	PRNT	0	0	0
	W								
0x0007 TTOV	R W	RESERVED	RESERVED	RESERVED	RESERVED	TOV3	TOV2	TOV1	TOV0
0x0008 TCTL1	R W	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
0x0009 TCTL2	R W	OM3	OL3	OM2	OL2	OM1	OL1	OM0	OL0
0x000A TCTL3	R W	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED	RESERVED
0x000B TCTL4	R W	EDG3B	EDG3A	EDG2B	EDG2A	EDG1B	EDG1A	EDG0B	EDG0A
0x000C TIE	R W	RESERVED	RESERVED	RESERVED	RESERVED	C3I	C2I	C1I	C0I
0x000D TSCR2	R	TOI	0	0	0	RESERVED	PR2	PR1	PR0
	W								
0x000E TFLG1	R W	RESERVED	RESERVED	RESERVED	RESERVED	C3F	C2F	C1F	C0F
0x000F TFLG2	R	TOF	0	0	0	0	0	0	0
	W								
0x0010–0x001F TCxH–TCxL[1]	R W	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
	R W	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0024–0x002B Reserved	R W								
0x002C OCPD	R W	RESERVED	RESERVED	RESERVED	RESERVED	OCPD3	OCPD2	OCPD1	OCPD0
0x002D Reserved	R								
0x002E PTPSR	R W	PTPS7	PTPS6	PTPS5	PTPS4	PTPS3	PTPS2	PTPS1	PTPS0
0x002F Reserved	R W								

Figure 12-3. TIM16B4CV3 Register Summary

13.3.2 Register Definition

13.3.3 Port HS Data Register (HSDR)

Module Base + 0x0000 Access: User read/write[1]

	7	6	5	4	3	2	1	0
R	0	0	0	0	0	0	HSDR1	HSDR0
W	▨	▨	▨	▨	▨	▨		
Altern. Read Function	—	—	—	—	—	—	OC[2]	OC[2]
	—	—	—	—	—	—	PWM[2]	PWM[2]
Reset	0	0	0	0	0	0	0	0

▨ = Unimplemented

Figure 13-2. Port HS Data Register (HSDR)

[1] Read: Anytime The data source (HSDRx or alternate function) depends on the HSE control bit settings.
Write: Anytime

[2] See PIM chapter for detailed routing description.

Table 13-4. PTHS Register Field Descriptions

Field	Description
1-0 HSDRx	**Port HS Data Bits**—Data registers or routed timer outputs or routed PWM outputs These register bits can be used to control the high-side drivers if selected as control source. See PIM section for routing details. If the associated HSEx bit is set to 0, a read returns the value of the Port HS Data Register (HSDRx). If the associated HSEx bit is set to 1, a read returns the value of the selected control source for the driver. When entering in STOP mode the Port HS Data Register (HSDRx) is cleared. 0 High-side driver is turned off 1 High-side driver is turned on **NOTE** After enabling the high-side driver with the HSEx bit in HSCR register, the user must wait a minimum settling time $t_{HS\_settling}$ before turning on the high-side driver.

13.4 Functional Description

13.4.1 General

The HSDRV module provides two high-side drivers able to drive LED or resistive loads. The driver can be controlled directly through register bits or alternatively by dedicated timer or PWM channels. See PIM chapter for routing details.

13.4.2 Over-Current Shutdown

Each high-side driver has an over-current shutdown feature with a current threshold of I_{OCTHSX}.

If an over-current is detected the related interrupt flag (HSOCIF1 or HSOCIF0) is set in the HSDRV Interrupt Flag Register (HSIF). As long as the over-current interrupt flag remains set, the related high-side driver is turned off to protect the circuit. Clearing the related over-current interrupt flag returns back the control to the selected source in the PIM module.

The over-current detection and driver shutdown can be masked for an initial T_{HSOCM} after switching the driver on. This can be achieved by writing the related HSOCME register bit in the HSCR register to 1. The HSOCME bits are only writable while the related driver is disabled (HSE=0).

13.4.3 Interrupts

This section describes the interrupt generated by HSDRV module. The interrupt is only available in MCU run mode. Entering and exiting MCU stop mode has no effect on the interrupt flags.

The HSDRV interrupt vector is named in Table 13-9. Vector addresses and interrupt priorities are defined at MCU level.

13.4.3.1 HSDRV Over Current Interrupt (HSOCI)

Table 13-9. HSDRV Interrupt Sources

Module Interrupt Source	Module Internal Interrupt Source	Local Enable
HSDRV Interrupt (HSI)	HSDRV Over-Current Interrupt (HSOCI)	HSOCIE = 1

If an over-current is detected the related interrupt flag HSOCIFx asserts. Depending on the setting of the HSDRV Error Interrupt Enable (HSOCIE) bit an interrupt is requested.

Chapter 17
Supply Voltage Sensor - (BATSV2)

Table 17-1. Revision History Table

Rev. No. (Item No.)	Data	Sections Affected	Substantial Change(s)
V01.00	15 Dec 2010	all	Initial Version
V02.00	16 Mar 2011	17.3.2.1 17.4.2.1	- added BVLS[1] to support four voltage level - moved BVHS to register bit 6

17.1 Introduction

The BATS module provides the functionality to measure the voltage of the battery supply pin VSENSE or of the chip supply pin VSUP.

17.1.1 Features

Either One of the voltage present on the VSENSE or VSUP pin can be routed via an internal divider to the internal Analog to Digital Converter. Independent of the routing to the Analog to Digital Converter, it is possible to route one of these voltages to a comparator to generate a low or a high voltage interrupt to alert the MCU.

17.1.2 Modes of Operation

The BATS module behaves as follows in the system power modes:

1. Run mode

   The activation of the VSENSE Level Sense Enable (BSESE=1) or ADC connection Enable (BSEAE=1) closes the path from the VSENSE pin through the resistor chain to ground and enables the associated features if selected.

   The activation of the VSUP Level Sense Enable (BSUSE=1) or ADC connection Enable (BSUAE=1) closes the path from VSUP pin through the resistor chain to ground and enables the associated features if selected.

   BSESE takes precedence over BSUSE. BSEAE takes precedence over BSUAE.

2. Stop mode

   During stop mode operation the path from the VSENSE pin through the resistor chain to ground is opened and the low voltage sense features are disabled.

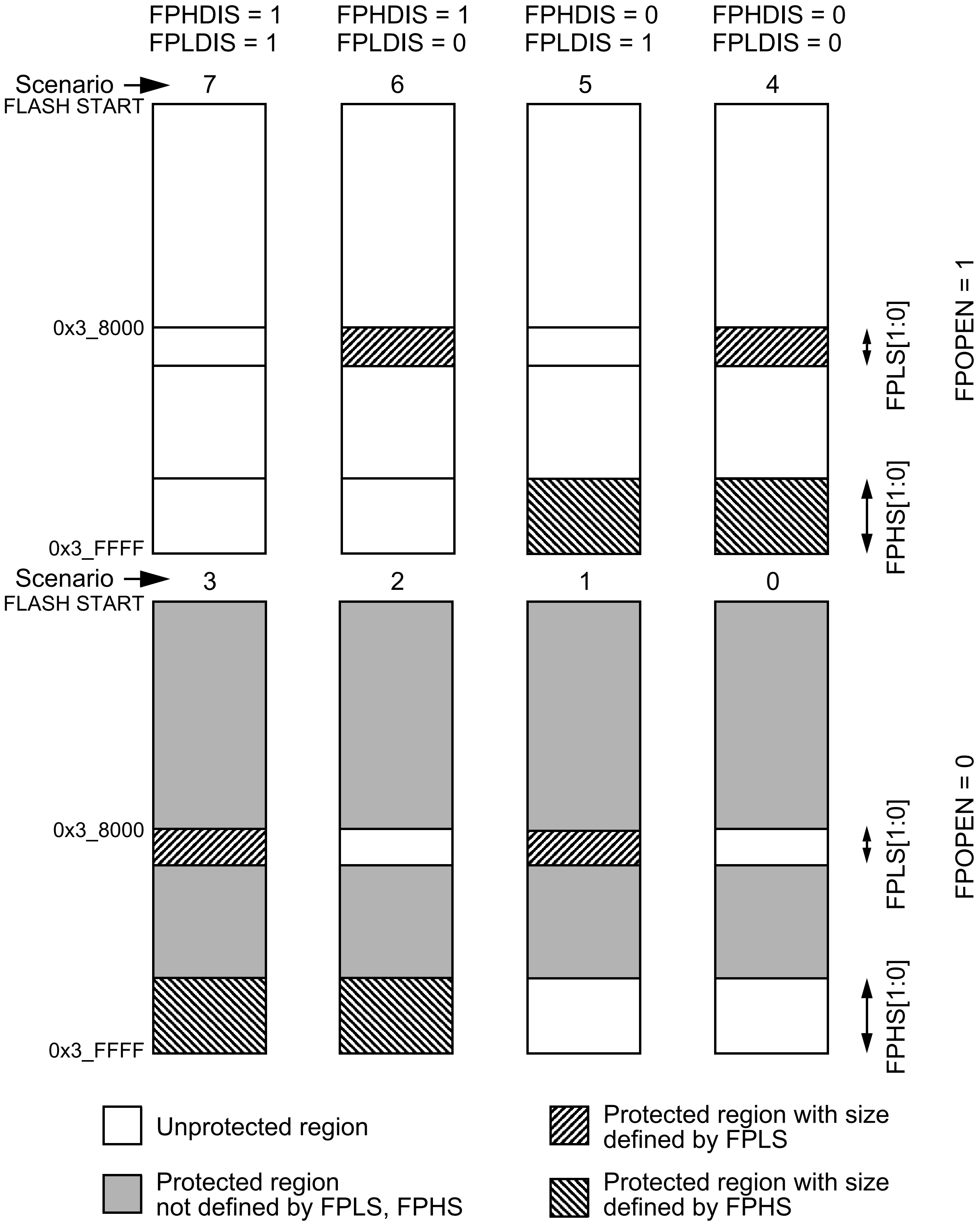


Figure 18-13. P-Flash Protection Scenarios

A.1.8 Supply Currents

This section describes the current consumption characteristics of the device as well as the conditions for the measurements.

A.1.8.1 Measurement Conditions

Current is measured on VSUP & VSUPHS pins. VDDX is connected to VDDA. It does not include the current to drive external loads. Unless otherwise noted the currents are measured in special single chip mode and the CPU code is executed from RAM. For Run and Wait current measurements PLL is on and the reference clock is the IRC1M trimmed to 1MHz. The bus frequency is 25MHz and the CPU frequency is 50MHz. Table A-9, Table A-10 and Table A-11 show the configuration of the CPMU module and the peripherals for Run, Wait and Stop current measurement.

Table A-9. CPMU Configuration for Pseudo Stop Current Measurement

CPMU REGISTER	Bit settings/Conditions

CPMUCLKS	PLLSEL=0, PSTP=1, CSAD=0
	PRE=PCE=RTIOSCSEL=COPOSCSEL=1

CPMUOSC	OSCE=1, External Square wave on EXTAL f_{EXTAL}=4MHz, V_{IH}= 1.8V, V_{IL}=0V
CPMURTI	RTDEC=0, RTR[6:4]=111, RTR[3:0]=1111;
CPMUCOP	WCOP=1, CR[2:0]=111

Table A-10. CPMU Configuration for Run/Wait and Full Stop Current Measurement

CPMU REGISTER	Bit settings/Conditions
CPMUSYNR	VCOFRQ[1:0]=01,SYNDIV[5:0] = 24
CPMUPOSTDIV	POSTDIV[4:0]=0
CPMUCLKS	PLLSEL=1
CPMUOSC	OSCE=0, Reference clock for PLL is f_{ref}=f_{irc1m} trimmed to 1MHz
API settings for STOP current measurement	
CPMUAPICTL	APIEA=0, APIFE=1, APIE=0
CPMUAPITR	trimmed to >=10Khz
CPMUAPIRH/RL	set to $FFFF

Table A-13. Stop Current Characteristics

Conditions are: V_{SUP}=V_{SUPHS}=12V API see CPMU Configuration for Pseudo Stop Current MeasurementTable A-9.						
Num	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
Stop Current all modules off						
1	$T_A = T_J$ = -40°C[1]	I_{SUPS}		29	60	µA
2	$T_A = T_J$ = 150°C[1]	I_{SUPS}		140	600	µA
3	$T_A = T_J$ = 25°C[1]	I_{SUPS}		33	65	µA
4	$T_A = T_J$ = 105°C[1]	I_{SUPS}		55	90	µA
Stop Current API enabled & LINPHY in standby (see 15.4.3.4 Standby Mode with wake-up feature)						
5	$T_A = T_J$ = 25°C[1]	I_{SUPS}		50	80	µA

[1] If MCU is in STOP long enough then $T_A = T_J$. Die self heating due to stop current can be ignored.

Table A-14. Pseudo Stop Current Characteristics

Conditions are: V_{SUP}=V_{SUPHS}=12V, API see CPMU Configuration for Pseudo Stop Current MeasurementTable A-9., COP & RTI enabled						
Num	**Rating**	**Symbol**	**Min**	**Typ**	**Max**	**Unit**
1	T_A= 25°C	I_{SUPPS}		358	480	µA

$$J(N) = \frac{j_1}{\sqrt{N(POSTDIV + 1)}}$$

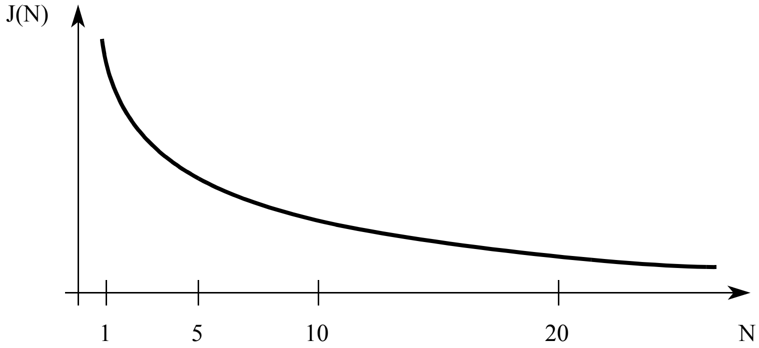


Figure E-2. Maximum Bus Clock Jitter Approximation (N=Number of Bus Cycles)

NOTE

On timers and serial modules a prescaler will eliminate the effect of the jitter to a large extent.