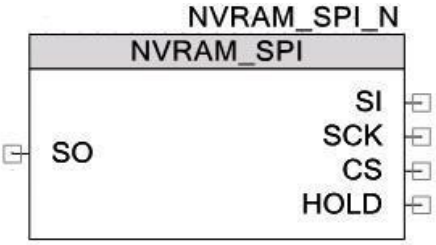
# NVRAM SPI PSoC 3/5 Creator Component User Guide

## Features




- Mode 0, Mode 3 SPI operating modes
- Supports up to 4 MHz of SPI operation[1]
- nvSRAM / F-RAM Read/Write
- nvSRAM RTC / F-RAM Processor Companion Read/Write[2]
- nvSRAM Commands (STORE, RECALL, ASENB, ASDISB)
- nvSRAM / F-RAM SLEEP Command[2]
- nvSRAM / F-RAM Product ID Read[2]
- nvSRAM / F-RAM Serial Number Read[2]
- nvSRAM / F-RAM Serial Number Write[2]

Notes:

1. This library component is tested up to 4 MHz of SPI Clock generation. This can go up the maximum frequency supported by PSoC 3/5 SPI component.
2. These features are not supported by all the parts. Please refer to the specific device datasheet for supportability.

## General Description

The NVRAM_SPI component provides a PSoC 3/5 SPI Master interface to SPI nvSRAM and F-RAM devices. This component can be directly used in user's PSoC 3/5 designs. High level API's are provided for different nvSRAM / F-RAM operations. User can directly use these API's as it reduces the time to integrate nvSRAM / F-RAM in the user application.

## Input/output Connections

This section describes the various input and output connections for the NVRAM (nvSRAM / F-RAM) component.

### SO – Input

The SO input carries the master input (PSoC 3/5) – slave output serial data from the nvSRAM / F-RAM device.

### SI – Output

The SI output carries the master output – slave (nvSRAM / F-RAM) input serial data from the master device (PSoC 3/5).

### SCK– Output

The SCK output carries the master (PSoC 3/5) synchronization clock output to the nvSRAM / F-RAM.

### CS – Output

The CS output carries the nvSRAM / F-RAM select signal to the nvSRAM device.

### HOLD – Output

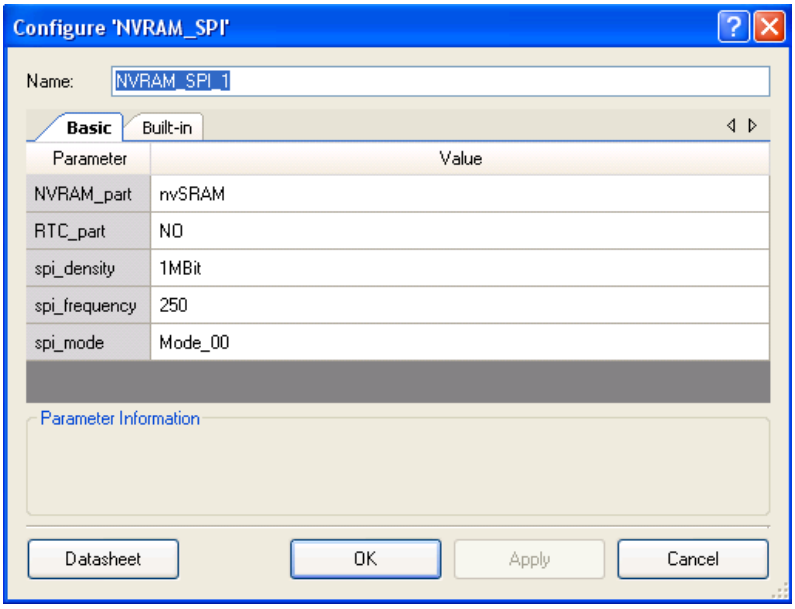
The HOLD output carries the SPI HOLD signal to the nvSRAM / F-RAM device.

## Parameters

Drag an NVRAM_SPI component onto your design and double-click on it to open the Configure dialog.

### Basic Tab

This tab is used to set the NVRAM part, RTC part, density, SPI frequency and mode of operation. Eg: nvSRAM part is selected by configuring NVRAM_part to nvSRAM, non-RTC part is selected by configuring RTC_part to “NO”, 1-MBit density, 250 for 250 KHz and Mode 0.



# Hardware Configuration

NVRAM_SPI component has following default hardware configuration.

Shift Direction: MSB first

Data Bits: 8

Data Lines: MOSI + MISO

Clock for SPI block: External

To change the above configuration user has to import the component instead of using ‘dependencies’.

# Application Programming Interface

Application Programming Interface (API) routines allow you to configure the component using software. The following table lists and describes the interface to each function. The subsequent sections cover each function in more detail.

By default, PSoC Creator assigns the instance name "NVRAM_SPI_1" to the first instance of a component in a given design. You can rename the instance to any unique value that follows the syntactic rules for identifiers. The instance name becomes the prefix of every global function name, variable, and constant symbol. For readability, the instance name used in the following table is "NVRAM_SPI".

| Function | Description | Applicability (nvSRAM / F-RAM) |
|---|---|---|
| NVRAM_SPI_Init | Initialization routine for NVRAM_SPI component | nvSRAM & F-RAM |
| NVRAM_SPI_Write | SPI NVRAM SRAM Write | nvSRAM & F-RAM |
| NVRAM_SPI_Read | SPI NVRAM SRAM Read | nvSRAM & F-RAM |
| NVRAM_SPI_RTC_Write | SPI NVRAM RTC Write | nvSRAM and F-RAM Processor companion |
| NVRAM_SPI_RTC_Read | SPI NVRAM RTC Read | nvSRAM and F-RAM Processor companion |
| NVRAM_SPI_nvCommand | Send NVRAM Command | nvSRAM only |
| NVRAM_SPI_Sleep | NVRAM Sleep mode entry | nvSRAM & F-RAM |
| NVRAM_SPI_Status_Reg_Read | NVRAM Status Register Read | nvSRAM & F-RAM |
| NVRAM_SPI_Status_Reg_Write | NVRAM Status Register Write | nvSRAM & F-RAM |
| NVRAM_SPI_Serial_No_Write | NVRAM Serial Number Write | All nvSRAM & F-RAM processor companions |
| NVRAM_SPI_Serial_No_Read | NVRAM Serial Number Read | All nvSRAM & F-RAM 1-MBit & F-RAM Processor companions |
| NVRAM_SPI_Device_ID_Read | NVRAM Device ID Read | All nvSRAM & F-RAMs above 128-Kbit |

## void NVRAM_SPI_Init(void)

**Description:** Initialization routine for NVRAM_SPI component. Initializes the SPI block, CS and Hold pin.

**Parameters:** None

**Return Value:** None

**Side Effects:** None

## uint8 NVRAM_SPI_Write(uint32 addr, uint8 *data_write_ptr, uint32 total_data_count )

**Description:** Write total_data_count number of data into NVRAM.

**Parameters:** uint32 addr: 32 bit NVRAM address for write.
uint8 *data_write_ptr: Pointer to an array of data bytes to be written.
uint32 total_data_count: Number of data bytes to be written.

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_Read(uint32 addr, uint8 *data_read_ptr, uint32 total_data_count )

**Description:** Read total_data_count number of data from NVRAM.

**Parameters:** uint32 addr : 32 bit NVRAM address for read.
uint8 *data_read_ptr: Pointer to an array for storing data bytes.
uint32 total_data_count: Number of data bytes to be read.

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_RTC_Write(uint8 addr, uint8 *data_write_ptr, uint8 total_data_count )

**Description:** Write total_data_count number of data into NVRAM RTC / F-RAM Processor Companion registers.

**Parameters:** uint8 addr : 8 bit NVRAM RTC / F-RAM Proc companion register address for write.
uint8 *data_write_ptr: Pointer to an array of RTC data bytes to be written.
uint8 total_data_count: Number of RTC data bytes to be written.

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_RTC_Read(uint8 addr, uint8 *data_read_ptr, uint8 total_data_count )

**Description:** Read total_data_count number of data from nvSRAM RTC / F-RAM Processor Companion registers.

**Parameters:** uint8 addr : 8 bit nvSRAM RTC / F-RAM Proc companion register address for read.
uint8 *data_read_ptr: Pointer to an array for storing RTC data bytes.
uint8 total_data_count: Number of RTC data bytes to be read.

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_ nvCommand (uint8 nvcmd)

**Description:** Send NVRAM Command. This is supported by nvSRAM only.

**Parameters:** uint8 nvcmd: 8 bit NVRAM Command

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_ Sleep (void)

**Description:** Send NVRAM Command for Sleep.

**Parameters:** None

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_ Status_Reg_Read (uint8 *data_byte)

**Description:** NVRAM Status Register Read

**Parameters:** uint8 *data_byte : Pointer to status register data to be written

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_ Status_Reg_Write (uint8 data_byte)

**Description:** NVRAM Status Register Write

**Parameters:** uint8 data_byte: 1 byte Status register data to be written

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_ Serial_No_Write (uint8 *data_ptr)

**Description:** NVRAM Serial Number Write. This is supported by nvSRAM and F-RAM Processor companions only.

**Parameters:** uint8 *data_ptr: pointer to an array of serial number data to be written.

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_ Serial_No_Read (uint8 *data_ptr)

**Description:** NVRAM Serial Number Read. This is supported by nvSRAM, 1-MBit F-RAM and F-RAM processor companions only.

**Parameters:** uint8 *data_ptr: pointer to an array for storing serial number data.

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

## uint8 NVRAM_SPI_ Device_ID_Read (uint8 *data_ptr)

**Description:** NVRAM Device ID Read. This is supported by nvSRAM and F-RAM above 128-Kbit.

**Parameters:** uint8 *data_ptr: pointer to an array for storing device id.

**Return Value:** uint8: returns the SPI Communication Status
0 : SPI Communication Error
1 : SPI Communication Success

**Side Effects:** None

# Component Changes

This section lists the major changes in the component from the previous version.

| Version | Description of Changes | Reason for Changes / Impact |
|---|---|---|
| 1.0 | Created APIs for interfacing nvSRAM with PSoC 3 | Initial Version |
| 2.0 | Added F-RAM support<br>Renamed component name from nvSRAM to nvRAM.<br>Added configuration options to select nvSRAM / F-RAM, Density and RTC/non-RTC | Changes made for F-RAM support.<br>No user impact for the existing nvSRAM APIs. For F-RAM, existing APIs are updated and new APIs are added. |
| 3.0 | PSoC 3/5 Support | Updated for PSoC 3/5 support. Updated APIs to return communication errors. |

© Cypress Semiconductor Corporation, 2015. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

This Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.