AN204454

How to Use S6J3200 Quad Flash, Traveo™ Family

Associated Part Family:	Series Name	Product Number
	S6J3200	S6J323B/3C/4B/4C/5B/5C/6B/6C S6J327B/7C28B/8C S6J32A9/2AA/2B9/2BA/2C9/2CA/2D9/2DA

This application note describes an example of system configuration and setting for using of DDR HSSPI (GDC side) in the S6J3200 series.

Contents

1 Introduction

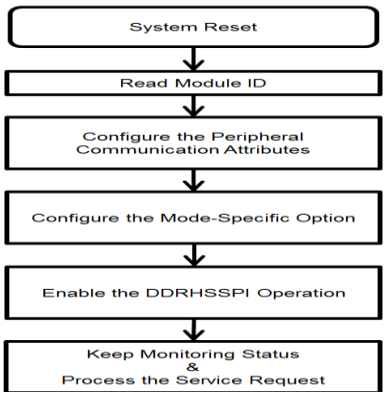
This application note describes an example of system configuration and setting for using of DDR HSSPI (GDC side) in the S6J3200 series.

2 Overview

2.1 General Steps

Figure 1 shows the general steps a programmer shall follow while using the DDRHSSPI.

Figure 1. Programmer's Flowchart: General Steps



1. After the System Reset, the software shall detect the Module ID number of DDRHSSPI, by reading the DDRHSSPIn_MID Register. This would help it in identifying the attributes and capabilities supported by the DDRHSSPI.
2. The next step is to configure the Attributes related to the Peripheral Communication with the Serial Flash Memory(Memories) connected with DDRHSSPI. DDRHSSPI can be interfaced with up to 4 Serial Flash Memories. Serial communication related attributes like Transfer Frequency (i.e. Clock Division Ratio bits), etc. shall be configured in the registers: DDRHSSPIn_PCC0-3. It is very important that these attributes shall be the same as the Serial Flash Memory, which is connected with DDRHSSPI. These configurations shall not be modified while the DDRHSSPI is active. In case the software has to re-program any of these values, the software shall first disable the DDRHSSPI and wait until the current serial transfer is finished.
3. DDRHSSPI can be configured either in Direct Mode or in Command Sequencer Mode, through the DDRHSSPIn_MCTRL.CSEN bit. Depending on which mode is to be used, the software shall configure the mode-specific registers. The registers specific to the Direct Mode are: (DDRHSSPIn_DMCFG, DDRHSSPIn_DMBCC, DDRHSSPIn_DMBCS, DDRHSSPIn_DMTRP, DDRHSSPIn_DMPSEL and DDRHSSPIn_DMFIFOCFG) and the registers specific to the Command Sequencer Mode are: (DDRHSSPIn_RDCSDC0-11, DDRHSSPIn_CSCFG, DDRHSSPIn_CSITIME, DDRHSSPIn_CSAEXT and DDRHSSPIn_CSPBUFFERCFG).
4. Only after all module-specific configurations are programmed, the DDRHSSPI shall be enabled (by setting the DDRHSSPIn_MCTRL.MEN to "1").
5. Once the DDRHSSPI is enabled, its normal working begins. The software shall keep monitoring the status of the DDRHSSPI using the various status bits. If the DDRHSSPI is configured for initiating the service requests, it would periodically trigger the service requests (i.e. Interrupts and/or DMA Service Requests). The software would service those requests, in order to ensure the normal working of DDRHSSPI.

More detail information about Direct Mode and Command Sequencer Mode, please see Chapter3 or Chapter4 on this document.

2.2 Evaluation Board Setting

Evaluation boards (S6T3J200261A216A2, S6T3J200261A208A2) mount two external Quad SPI(QSPI) Serial Flash ROMs(S25FL256S).

When you use external QSPI Serial Flash ROM on our evaluation board, please set the DIP switch as below.

- Bit1 of DIP-SW13 (BUF_1) is ON (QUAD_EN: ON Quad SPI Flash)

Then, when you use S6T3J200261A208A2 and external QSPI Serial Flash ROM, please change jumper pins as below.

Figure 2. Jumper Configuration for 208 Pin Evaluation Board

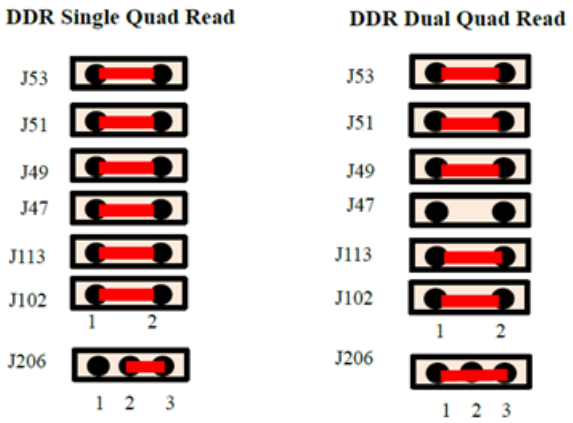
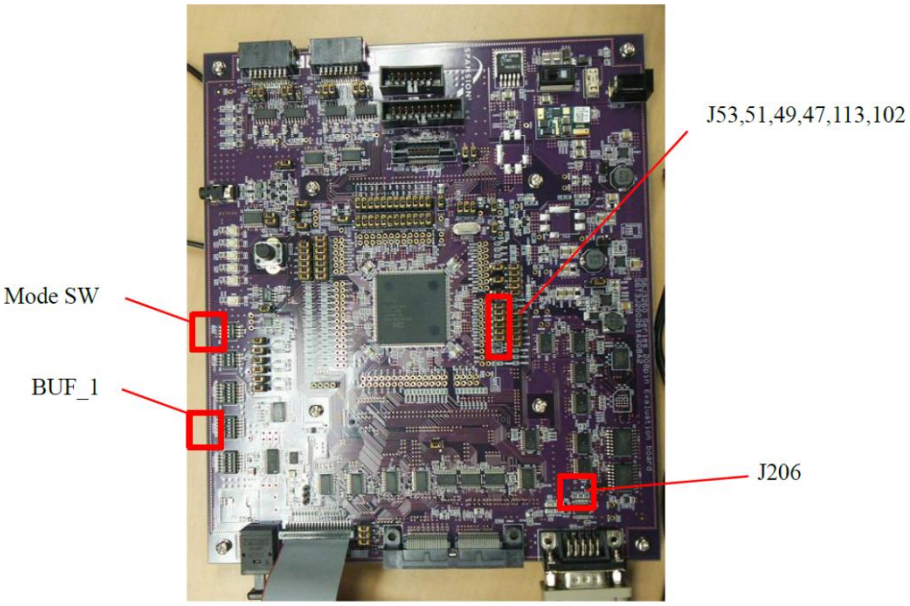
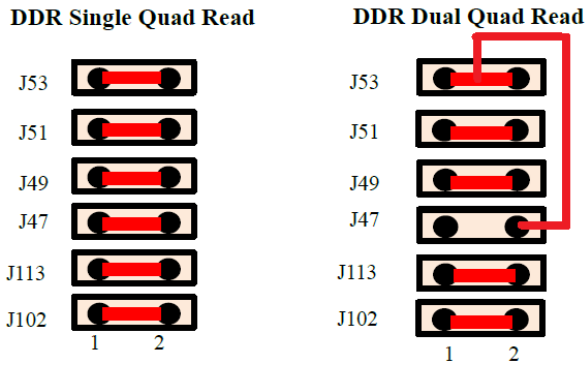


Figure 3. Evaluation Board



Then, when you use S6T3J200261A216A2 and external QSPI Serial Flash ROM, please change jumper pins as below.

Figure 4. Jumper Configuration for 216 Pin Evaluation Board



3 DDRHSSPI Operation in Direct Mode

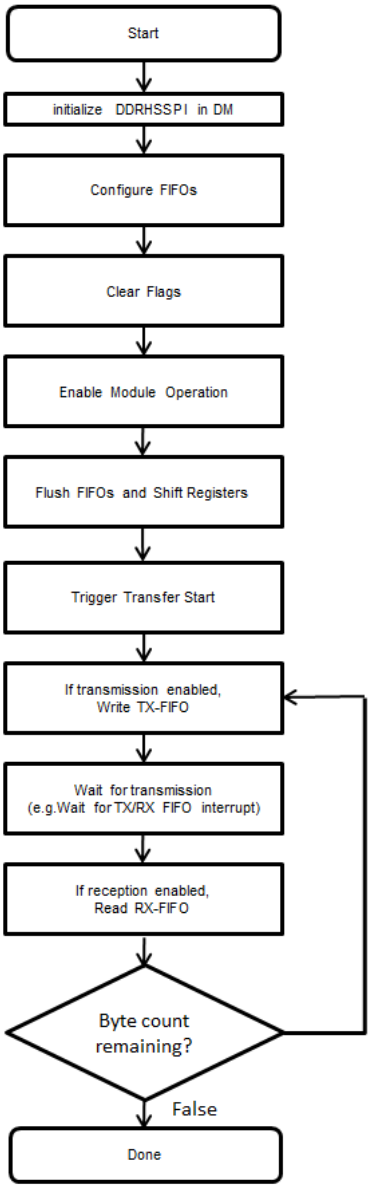
In general, you want to erase, write and configure to external QSPI Serial Flash ROM in Direct Mode.

More detail, please see section 4.1 of chapter 50 in S6J3200 Series Platform Hardware Manual.

3.1 Programmer's Flowchart

Figure 5 shows gives the general steps which the SW shall follow while using the DDRHSSPI in Direct Mode.

Figure 5. Programmer's Flowchart: DDRHSSPI in Direct Mode of Operation



1. After the System Reset, the software shall initialize the DDRHSSPI by reading the DDRHSSPIn_MID Register and setting the Peripheral Communication related attributes in the DDRHSSPIn_PCC0, DDRHSSPIn_PCC1, DDRHSSPIn_PCC2 and DDRHSSPIn_PCC3 Registers. Please make sure that the DDRHSSPIn_MCTRL.CSEN bit is cleared to "0".
2. The next step is to configure the transfer protocol (i.e. whether the DDRHSSPI serial transfers use the Legacy or the Quad Protocol and whether the DDRHSSPI would be used only for transmission, or both for transmission and reception) in the DDRHSSPIn_DMTRP.TRP. DDRHSSPI loads the DDRHSSPIn_DMBCC.BCC with the number of bytes to be serially transferred.
3. Configure the DDRHSSPIn_DMFIFOCFG Register, to set the FIFO threshold levels. By programming these levels, the assertion of the service requests can be controlled. Also configure the DDRHSSPIn_DMFIFOCFG.FWIDTH, to select the width of the FIFOs. Configure the service requests: DDRHSSPI supports both Interrupt Service Request and DMA Service Request, for the normal data read operations from RX-FIFO or write operations to TX-FIFO. For normal operation, either the Interrupt Service Requests or the DMA Service Requests shall be enabled by the software. To enable the Interrupt Service Requests for writing TX-FIFO, please program the bits in the DDRHSSPIn_TXE Register. To enable the Interrupt Service Requests for reading RX-FIFO, please program the bits in the DDRHSSPIn_RXE Register. To enable the DMA Service Request (for writing and/or reading), please program either/both of the DDRHSSPIn_DMAEN.TXDMAEN and the DDRHSSPIn_DMAEN.RXDMAEN bits. The DMA Read Channel must be setup to perform a block transfer of "DDRHSSPIn_DMFIFOCFG.RXFTH + 1" transfers. The DMA Write Channel must be setup to perform a block transfer of "24-DDRHSSPIn_DMFIFOCFG.TXFTH" transfers. Select the peripheral (in DDRHSSPIn_DMPSEL.PSEL) on which DDRHSSPI shall initiate the transfer.
4. Clear all relevant flags. This finishes the steps in initialization of DDRHSSPI for Direct Mode.
5. Set the DDRHSSPIn_MCTRL.MEN bit, to enable the module.
6. Flush FIFOs to ensure data consistency and avoid any data corruption from previous transfers.
7. When DDRHSSPI is configured, setting the DDRHSSPIn_DMSTART.START bit triggers the start of the serial transaction. Once the serial transaction starts, if transmission is enabled in the
8. DDRHSSPIn_DMTRP.TRP, the DDRHSSPI reads data from TX-FIFO and loads them to the Shift Register. The Shift Register is shifted left and the transmit data is shifted-out onto the Serial Interface. If DDRHSSPI is enabled for Receive operation (in DDRHSSPIn_DMTRP.TRP), the DDRHSSPI receives the serial data with shifting the Shift Register. The received data assembled in the Shift Register is pushed into the RX-FIFO.
9. Write the data to be transmitted into the TX-FIFO via DDRHSSPIn_TXFIFO0-23 Register address. Before writing to the DDRHSSPIn_TXFIFO0-23 Register, modify the value of the DDRHSSPIn_DMFIFOCFG.TXCTRL bit appropriately. Generally (i.e. when the data being written to the TX-FIFO is to be transmitted as it is), the DDRHSSPIn_DMFIFOCFG.TXCTRL bit shall be "0". Only when in adding some kind of controls such as dummy cycles, the DDRHSSPIn_DMFIFOCFG.TXCTRL bit shall be set to "1".The write access toDDRHSSPIn_TXFIFO0-23 shall be performed after the control of the DDRHSSPIn_DMFIFOCFG.TXCTRL bit.
10. Service Requests are asserted by DDRHSSPI whenever the TX-FIFO level is below the threshold or whenever the DDRHSSPI RX-FIFO level is above the threshold. The software shall write TX-FIFO or read RX-FIFO, to ensure the serial data transfer of DDRHSSPI. After writing or reading the relevant FIFO, the software shall clear the Interrupt Service Requests by writing the DDRHSSPIn_TXC or the
11. DDRHSSPIn_RXC Register. DMA Service Requests are cleared by the DMA Controller.
12. If reception is enabled in DDRHSSPIn_DMTRP Register, then the software fetches the received data from the RX-FIFO.
13. Software judges if current serial transfer has finished, by checking (1)the DDRHSSPIn_TXF.TSSRS bit to be "1" or(2)the DDRHSSPIn_DMBCS Register value to be0x0000.In the normal course of operation, the software usually keeps repeating steps from 8 to 11 until the end of serial transfer.

When the software initiates a new serial transfer again, it starts this flow from step 2.

To switch between the Direct Mode and Command Sequencer Mode, or to re-program any of the parameter that directly affect the serial transfer, the software shall first stop the current transfer and disable the DDRHSSPI (by resetting DDRHSSPIn_MCTRL.MEN bit to "0"). The software can check the status bit

DDRHSSPIn_TXF.TSSRS to see if the current transfer has finished.

3.2 Common Use Case in Direct Mode

Generally, the Direct Mode is mainly used to erase (Bulk Erase) and write (Quad Page Program, QPP) the data to external QSPI Serial Flash ROM.
Also, the Direct Mode can be used to write/read some register for QSPI.

In this chapter, this document described some commonly-used commands.
More detail commands, please see the S25FL128S and S25FL256S Data Sheet.
Regarding the use case, please see the chapter 5 in this document.

3.2.1 Commonly-Used Commands

This section shows other commonly-used commands.

Write Enable (WREN 06h)

The Write Enable (WREN) command sets the Write Enable Latch (WEL) bit of the Status Register 1 (SR1[1]) to a 1. The Write Enable Latch (WEL) bit must be set to a 1 by issuing the Write Enable (WREN) command to enable write, program and erase commands.

Write Register (WRR 01h)

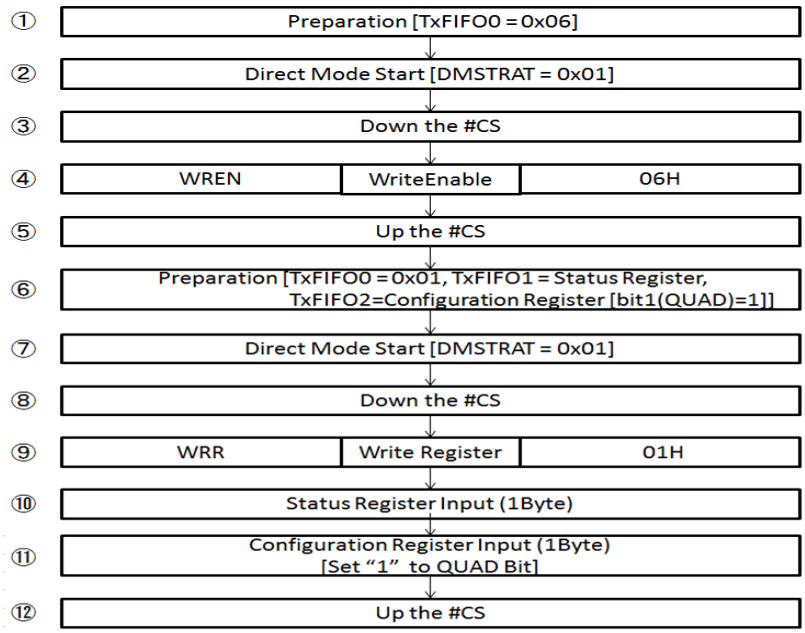
In order to use Quad Page Program the Quad Enable Bit in the Configuration Register must be set bit1 which is Quad Enable bit.
If this bit doesn't be set, you cannot use Quad I/O.

Table 1. Initialization in Direct Mode for BulkErase

	Bit	Name	Bit Function	Description	
				Bit=1	Bit=0
FL-S FL-P	1	QUAD	Puts the device into Quad I/O mode (Non volatile)	Enable to use Quad I/O. Also enable to use Single and Dual I/O	Enable to use Single and Dual I/O

Figure 6 shows gives the Write Register follow.

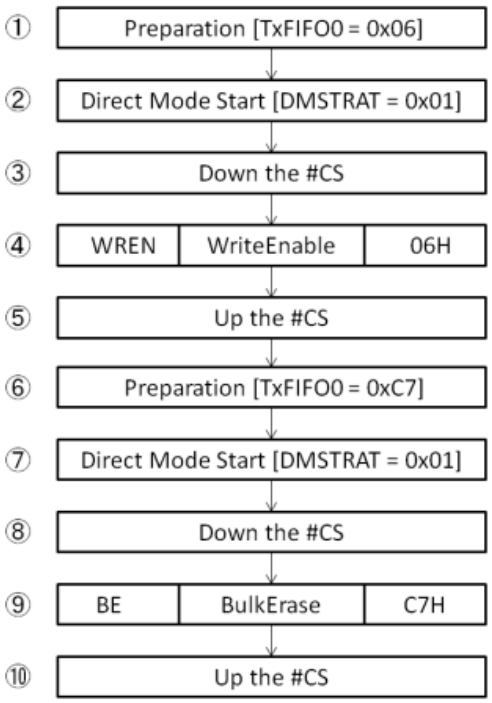
Figure 6. Programmer's Flowchart: Write Register



Bulk Erase (BE C7h)

The Bulk Erase (BE) command sets all the bits to 1 (all bytes are FFh) inside the entire flash memory array. Before the BE command can be accepted by the device, Write Enable (WREN) command must be issued and decoded by the device.
Figure 7 shows gives the Bulk Erase follow.

Figure 7. Programmer's Flowchart: Bulk Erase

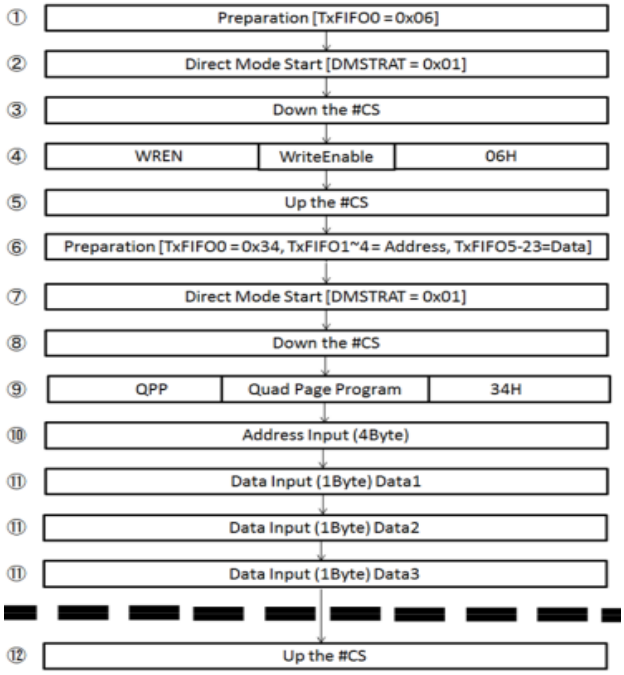


Quad Page Program (QPP 34h)

The Quad-input Page Program (QPP) command allows bytes to be programmed in the memory. To use Quad Page Program the Quad Enable Bit in the Configuration Register must be set (QUAD=1). A Write Enable command must be executed before the device will accept the QPP command (Status Register 1, WEL=1).

Figure 8 shows gives the Quad Page Program follow.

Figure 8. Programmer's Flowchart: Quad Page Program



Read Configuration Register (RDCR 35h)

The Read Configuration Register (RDCR) command allows the Configuration Register contents to be read. Usually, it is used to read and check the QUAD bit.
To receive the data, Transfer protocol in DDRHSSPIn_DMTRP register is changed TX and RX mode.

Figure 9 shows gives the Write Register follow.

Figure 9. Programmer's Flowchart: Write Register

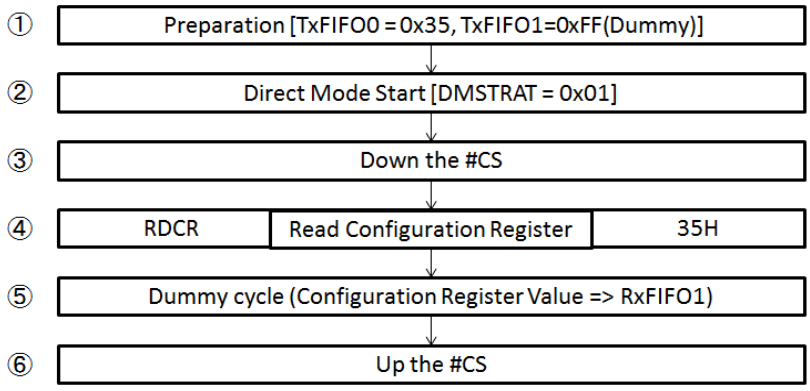


Table 2 is shown Configuration Register 1 (CR1) information.

The newest and detail information, please see the S25FL128S and S25FL256S Data Sheet.

Table 2. Configuration Register 1(CR1)

Bits	Field Name	Function	Type	Default State	Description
7	LC1	Latency Code	Non-Volatile	0	Selects number of initial read latency cycles See Latency Code Tables
6	LC0			0	
5	TBPROT	Configures Start of Block Protection	OTP	0	1 = BP starts at bottom (Low address) 0 = BP starts at top (High address)
4	RFU	RFU	OTP	0	Reserved for Future Use
3	BPNV	ConfiguresBP2-0 in Status Register	OTP	0	1 = Volatile 0 = Non-Volatile
2	TBPARM	Configures Parameter Sectors location	OTP	0	1 = 4-kB physical sectors at top. (high address) 0 = 4-kB physical sectors at bottom (Low address) RFU in unform sector devices
1	QUAD	Puts the device into Quad I/O operation	Non-Volatile	0	1 = Quad 0 = Dual or Serial
0	FREEZE	Lock current state of BP2-0 bits in Status Register, TBPROT and TBPARM in Configuration Register and OTP regions	Volatile	0	1 = Block Protecton and OTP locked 0 = Block Protecton and OTP un-locked

Read Status Register (RDSR 05h)

The Read Status Register-1 (RDSR1) command allows the Status Register-1 contents to be read. The Status Register-1 contents may be read at any time, even while program, erase, or write operation is in progress. It is possible to read Status Register-1 continuously by providing multiples of eight clock cycles.
Usually, it is used to read and check the WIP bit to confirm the status of the device.

Table 3 is shown Status Register 1 (SR1) information.
The newest and detail information, please see the S25FL128S and S25FL256S Data Sheet.

Table 3. Status Register 1(SR1)

Bits	Field Name	Function	Type	Default State	Description
7	SRWU	Status Register Write Disable	Non-Volatile	0	1 = Locks state of SRWD, BP, and configuration register bits when WP# is low by ignoring WRR command 0 = No protection, even when WP# is low
6	P_ERR	Programming Error Occurred	Volatile, Read only	0	1 = Error occurred 0 = No Error
5	E_ER	Erase Error Occurred	Volatile, Read only	0	1 = Error occurred 0 = No Error
4	BP2	Block Protection	Volatile if CR1[3]=1, Non-Volatile if CR1[3]=0	1 if CR1[3]=1, 0 shipped from Cypress	Protects selected range of sectors (Block) from Program or Erase
3	BP1				
2	BP0				
1	WEL	Write Enable Latch	Volatile	0	1 = Device accepts Write Registers (WRR), program or erase commands 0 = Device ignores Write Registers (WRR), program or erase commands This bit is not affected by WRR, only WREN and WRDI commands affect this bit
0	WIP	Write in Progress	Volatile, Read only	0	1 = Device Busy, a Write Register (WRR), program, erase or other operation is in progress 0 = Ready Device is in standby mode and can accept commands

4 DDRHSSPI Operation in Command Sequencer Mode

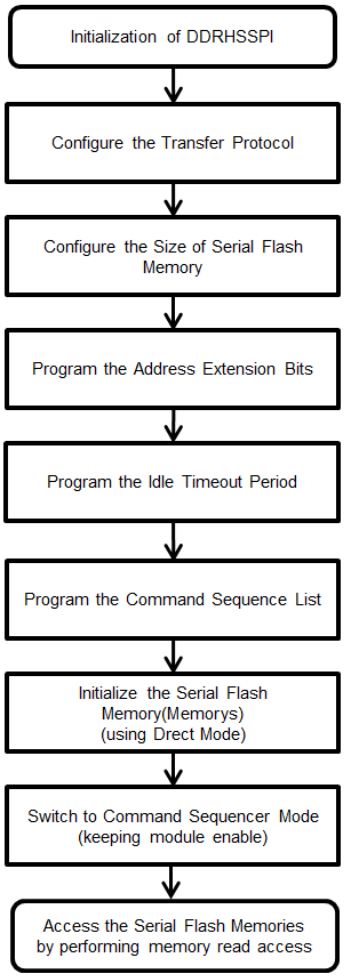
In general, you can use Command Sequencer Mode when you want to read external QSPI Serial Flash ROM data which is mapping on 0x4000000 after some configurations using Direct Mode.

More detail information about Command Sequencer Mode, please see section 4.2 of chapter 50 in S6J3200 Series Platform Hardware Manual.

4.1 Flowchart

Figure 10 shows the general steps which the SW shall follow while using the DDRHSSPI in Direct Mode.

Figure 10. Programmer's Flowchart: Memory Mapping of Serial Flash Memories



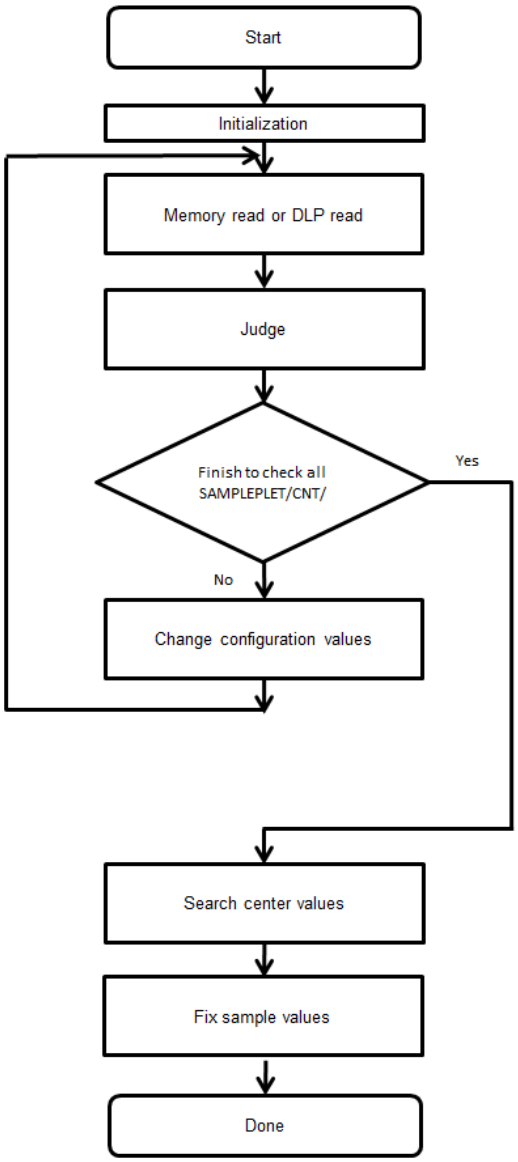
1. After the System Reset, the software shall initialize the DDRHSSPI by setting the Peripheral Communication related attributes in the DDRHSSPIn_PCC0-3Registers. It is very important that these attributes shall be the same as being used by the Serial Flash Memory with which DDRHSSPI is interfaced. When Serial Flash Memories are to be memory-mapped using Command Sequencer Mode, all Serial Flash Memories shall be of same family. Therefore, all ofDDRHSSPIn_PCC0-3 Registers shall have same configuration values.
2. The next step is to configure the transfer protocol (i.e. whether the DDRHSSPI serial transfers use the Quad or Octal Protocol in the DDRHSSPIn_CSCFG.MBM). The DDRHSSPIn_CSCFG.DDRMODE bits hall be set same as DDRHSSPIn_DMFIFOCFG.DDRM bit.

3. Program the DDRHSSPIn_CSCFG.MSEL, with the size of the System Bus address space which must be used in selection of the Serial Flash Memory on which the serial transfer must be initiated. Please refer to Section 4.2 chapter 50 of PF HWM for details of the Slave Select.
4. If the addresses generated for the memory-mapped accesses are to be virtually extended to cover a memory range of virtually 16GB, the DDRHSSPIn_CSAEXT Register value gives the upper bits of the address. Please refer to Section 4.2 in chapter 50 of PF HWM for details of address generation.
5. The DDRHSSPIn_CSITIME.ITIME helps DDRHSSPI enhance the performance of memory accesses, by continuing previous serial transfer. If DDRHSSPI detects a consecutive memory access during ITIMER period (Slave Select is kept asserted and SCLK is halted), it extends the data transfer without de-asserting current Slave Select. This feature reduces the access time by omitting a new Command Sequence. Program the DDRHSSPIn_CSITIME.ITIME with appropriate idle time-out value.
6. Program the list of Read Command Sequence Registers (i.e. DDRHSSPIn_RDCSDC0-11) with the sequence of the memory read command for the Serial Flash Memory which is interfaced. Please refer to the datasheet of the Serial Flash Memory for details of the Read Command Sequence.
7. The next step is to initialize the Serial Flash Memory that is to be memory mapped. The initialization is specific for the Serial Flash Memory, including the setting of some control or status bits in its register set. e.g. To use a Serial Flash Memory in a high-performance Quad Mode. Please refer to the datasheet of the Serial Flash Memory to be interfaced. This initialization of the Serial Flash Memory shall be performed using Direct Mode of DDRHSSPI.
8. With this, DDRHSSPI has been configured for accessing the memory-mapped devices. Switch the DDRHSSPI to Command Sequencer Mode, so that it starts generating the Read Command Sequences on the Serial Interface, by mapping the System Bus accesses to the memory-mapped locations.

4.2 Example Configuration of Sampling Point

Figure 4 2 shows the example configuration of sampling point as below.

Figure 11. Flowchart about Example Configuration of Sampling Point



1. Before the adjustment of sampling point, please finish the Direct Mode configuration.
2. Then, write “0” to DDRHSSPIn_SDATA SAMPLEPTLFT/CNT/RGH, at first.
3. There are two comparison methods.

[Non-DLP function]

- You make a comparison value and expected value. (ex: 0x34AD56CD)
  Specified address such as 0x40000000 is already written comparison value in Direct Mode.
  If compare match about both values, you hold 0 in array variable (e.g. ddr_smpl_mapx[i]) which you make.
  If compare miss about both values, you hold 1 in array variable (e.g. ddr_smpl_mapx[i]) which you make.

Then, increase the DDRHSSPIn_SDATA SAMPLEPTLFT/CNT/RGH value and compare again.

Finally, we get the data such as following table.Then, you read value in specified address and then compare to expected value in RAM.

Table 4. Comparison Result by Using Non-DLP Function

	i																																
	0	1	2	…	25	26	27	28	29	30	31	32	…	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
ddr_smpl_map0[i]	0	0	0	…	0	0	0	0	1	1	1	1	…	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map1[i]	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map2[i]	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map3[i]	0	0	0	…	0	0	0	0	0	0	1	1	…	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map4[i]	0	0	0	…	0	0	0	0	0	0	1	1	…	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map5[i]	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map6[i]	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map7[i]	0	0	0	…	0	0	0	1	1	1	1	1	…	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[DLP function (in DDR mode)]

- You check DDRHSSPIn_DLP SAMPLESTATUS register in DDR mode.

  You hold the result in array variable (e.g. ddr_smpl_mapx[i]) which you make.

  Then, increase the DDRHSSPIn_SDATA SAMPLEPTLFT/CNT/RGH value and compare again.

  Finally, we get the data such as following table.

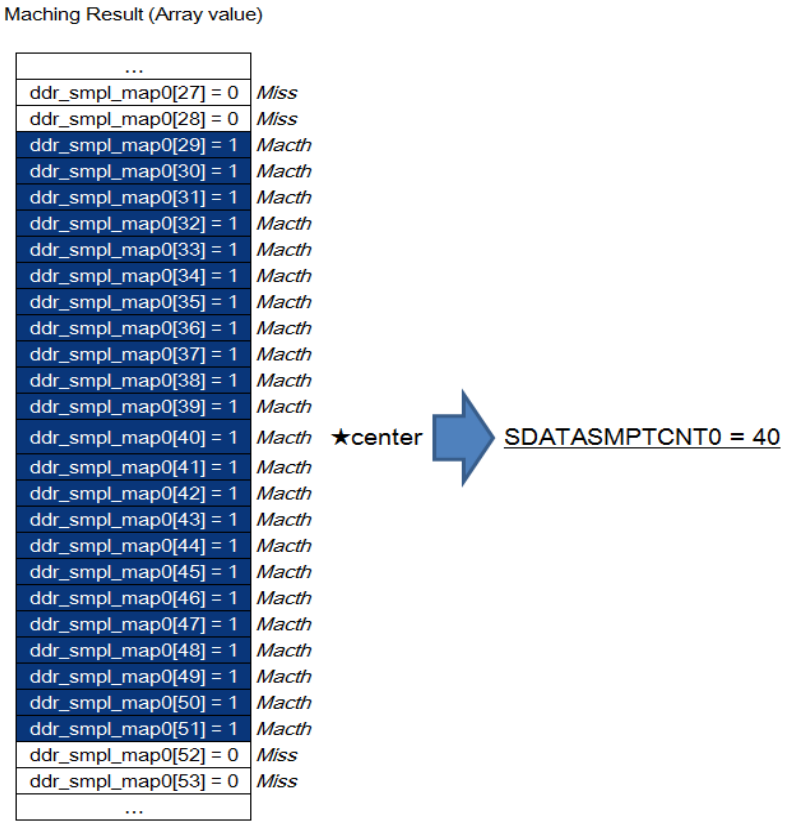
Table 5. Comparison Result by Using DLP Function

	i																																							
	0	1	2	…	25	26	27	28	29	30	31	32	…	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
ddr_smpl_map0[i] (DLPSMPLST0C)	0	0	0	…	0	0	0	0	1	1	1	1	…	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map1[i] (DLPSMPLST1C)	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map2[i] (DLPSMPLST2C)	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map3[i] (DLPSMPLST3C)	0	0	0	…	0	0	0	0	0	0	1	1	…	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map4[i] (DLPSMPLST4C)	0	0	0	…	0	0	0	0	0	0	1	1	…	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map5[i] (DLPSMPLST5C)	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map6[i] (DLPSMPLST6C)	0	0	0	…	0	0	0	0	0	1	1	1	…	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ddr_smpl_map7[i] (DLPSMPLST7C)	0	0	0	…	0	0	0	1	1	1	1	1	…	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4. If you finished checking all DDRHSSPIn_SDATA SAMPLEPTLFT/CNT/RGH, please go to 5.
   If not, please go to 3.
5. You configure each center value to DDRHSSPIn_SDATASAMPLEPTLFT/CNT/RGH.

Figure 14 is described how to fix center value of sampling point.

Figure 12. How to Fix Center Value of Sampling Point



4.3 Calibration

When condition will be changed such as temperature, voltage etc., SAMPLEPTLFT/CNT/RGH values need to be modified (Calibration).

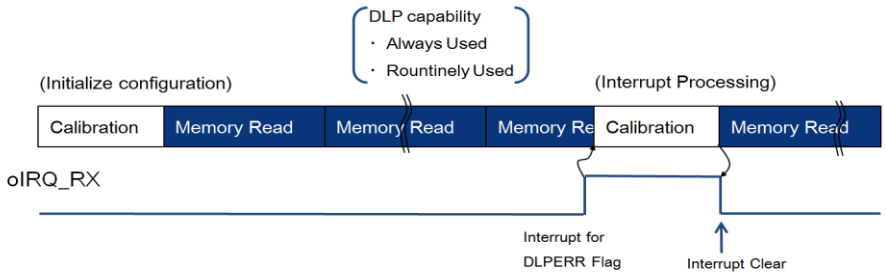
Calibrations of using DLP function in DDR or non-DLP function are deference how to modify these values.

4.3.1 Using DLP Capability in DDR (Dual Data Rate)

When you use external QSPI Serial Flash ROM, you can use DLP capability in DDR.

Please calibrates when DLPERR interrupt occurs for DDR (Dual Data Rate) with DLP capability.

Figure 13. Calibration and Memory Access for DDR with DLP Capability



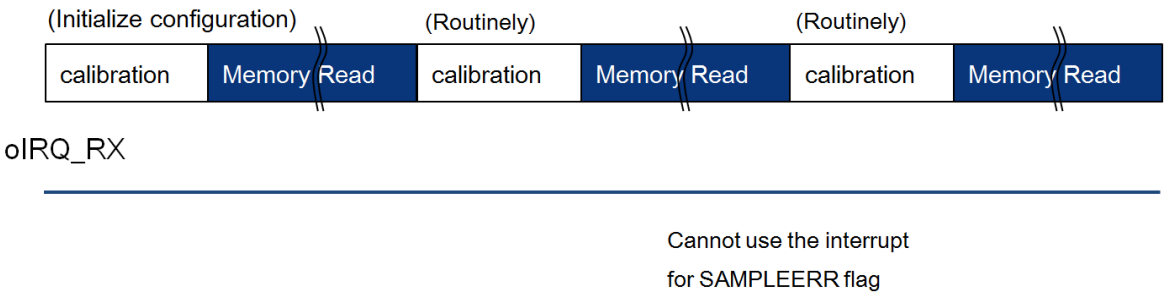
Note:

- Please attention as below in DLP capability.
  - DDRHSSPIn_MCTRL.DLPEN bit is set to 1.
  - You need to set DLP value in Serial Flash Memory.(Recommended value = 0x34)
    This value can be set in Direct Mode.
- DDRHSSPIn_DLP.DLP is also set to same DLP value.
- DDRHSSPIn_RXC.DLPERRC is set to 1.
- When DLPERR interrupt occurs, the data abort is always happened.
  In this time, please write DDRHSSPIn_FAULTC_DLPFC=1 in Data abort exception.
  If NMI is happened instead of Data abort for GFX side, please disable BUS Monitor Interrupt (BUSM_MonitorInterruptEnable=0).

4.3.2 Using Non-DLP Capability

Please calibrates routinely for using Non-DLP capability.

Figure 14. Calibration and Memory Access Using Non-DLP Capability



5 Sample Program

This sample program which is included in the Sample SW project (SampleSW_S6J3200_20150522.zip) can be erase, write 1 word, and read for one external Quad Flash.

In this document, explanation excluding DDRHSSPI isn't included such as port/clock configuration, interrupt etc.

This sample program doesn't be used interrupt.

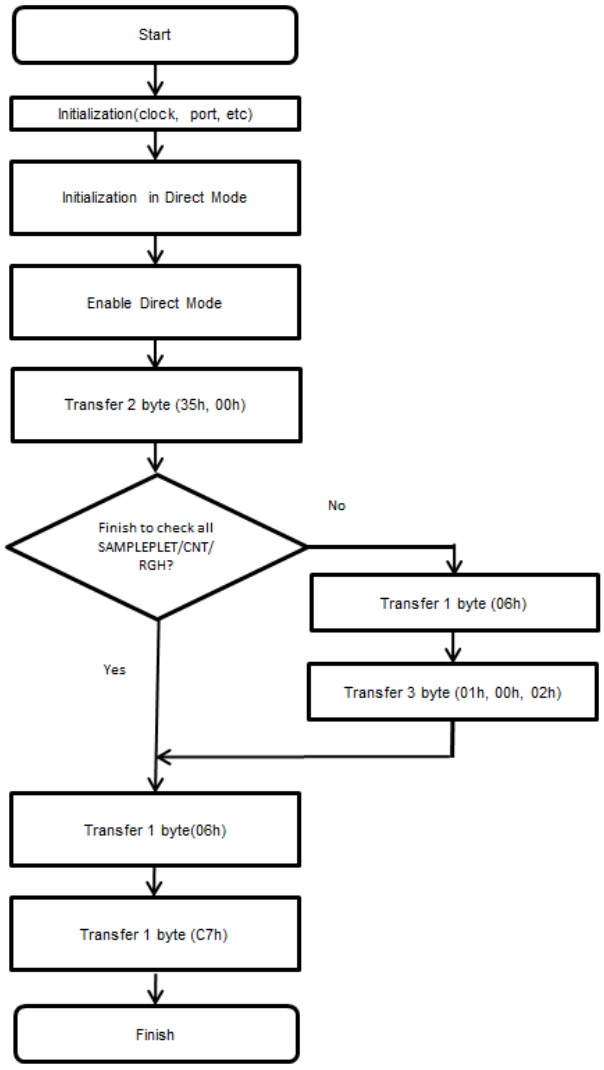
5.1 Erase

This program can be erasing about external Flash by SDR Legacy.

5.1.1 Flowchart

Figure 15 shows sample program flowchart below.

Figure 15. Flowchart for Sample Program for Erase



5.1.2 Initialization in Direct Mode

Sample program is configured about Direct Mode as below.

Table 6. Initialization in Direct Mode for BulkErase

	Register	Detail
1	MCTRL	CSEN=0b, MEN=0b
2	DMTRP	DDRM=0b, TRP=0000b
3	PCC0	SSELDEASRT=11111b, CDRS=1111b, SS2CD =01b
4	DMCFG	SSDC=1b
5	DMPSEL	PSEL=00b
6	DMFIFOCFG	TXCTRL=1b, FWIDTH=00b
7	MCTRL	CSEN=0b, MEN=1b

5.1.3 Transfer to External Flash in Direct Mode

At first, DMBCC is set to number of FIFO in just one transmitting.

The content of transmitting is as below,

1. Transfer 2bytes (35h, 00h)

   At first, transfer command 35h(Configuration Register) and then transfer 00h(Dummy) to get Quad bit in external Flash.

   If Quad bit in external Flash is set, go to No.4.

   If Quad bit in external Flash isn't set, go to No.2.
2. Transfer 1byte (06h)

   Transfer command 06h(WriteEnable) for No.3.
3. Transfer 3bytes (01h,00h,02h)

   Transfer command 01h (Write Register), then transfer 00h(Status Register-1) and 02h(Configuration Register) to set Quad bit in external Flash.
4. Transfer 1bytes (06h)

   Transfer command 06h(WriteEnable) for No.5.
5. Transfer 1bytes (C7h)

   Transfer command C7h(BulkErase).

Note: Please confirm bit0:WIP in Status Register about end of writing.

6 Reference Documents

Other manuals that relate this function are shown below. Refer to the relevant manual as needed.

The content of these manuals is subject to change without notice. For assistance, please contact your sales representative.

Traveo Manuals

- S6J3200 Series 32-BIT MICROCONTROLLER Spansion Traveo Family HARDWARE MANUAL
  (Hereafter referred to as the "S6J3200 Series Hardware Manual.")
- Spansion Traveo Family 32-BIT MICROCONTROLLER Platform Part HARDWARE MANUAL
  (Hereafter referred to as the "S6J3200 Series Platform Hardware Manual.")
- S6J3200 Series 32-BIT MICROCONTROLLER Spansion Traveo Family DATA SHEET
  (Hereafter referred to as the "S6J3200 Series Data Sheet.")

Flash Memory Manuals

- S25FL128S and S25FL256S MirrorBit® Flash Non-Volatile Memory DATA SHEET
  (Hereafter referred to as the "S25FL128S and S25FL256S Data Sheet.")

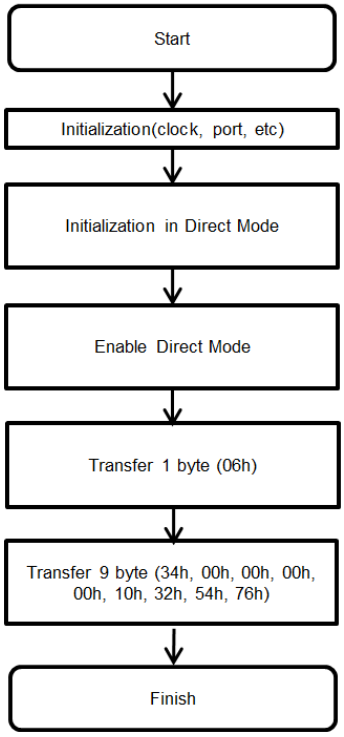
6.1 Write

This program can be written 1 word data (0x76543210) to external Flash by SDR Quad.

6.1.1 Flowchart

Figure 16 shows sample program flowchart below.

Figure 16. Flowchart for Sample Program for Write



6.1.2 Initialization in Direct Mode

Sample program is configured about Direct Mode as below

Table 7. Initialization in Direct Mode for SDR Quad Write

	Register	Detail
1	MCTRL	CSEN=0b, MEN=0b
2	DMTRP	DDRM=0b, TRP=1010b
3	PCC0	SSELDEASRT=11111b, CDRS=1111b, SS2CD =01b
4	DMCFG	SSDC=1b
5	DMPSEL	PSEL=00b
6	DMFIFOCFG	TXCTRL=1b, FWIDTH=00b
7	MCTRL	CSEN=0b, MEN=1b

6.1.3 Transfer to External Flash in Direct Mode

At first, DMBCC is set to number of FIFO in just one transmitting.

The content of transmitting is as below,

1. Transfer 1byte (06h)
   Transfer command 06h (WriteEnable) for No.2.
2. Transfer 9byte (34h, 00h, 00h, 00h, 00h, 10h, 32h, 54h, 76h)
   TxFIFO0 =0x00001032 [Command 34h (Command: Quad Page Program)]
   TxFIFO1 =0x00001000 [Address 00h (0x00XXXXXX)]
   TxFIFO2 =0x00001000 [Address 00h (0xXX00XXXX)]
   TxFIFO3 =0x00001000 [Address 00h (0xXXXX00XX)]
   TxFIFO4 =0x00001000 [Address 00h (0xXXXXXX00)]
   TxFIFO5 =0x00001210 [Data 10h by SDR Quad]
   TxFIFO6 =0x00001232 [Data 32h by SDR Quad]
   TxFIFO7 =0x00001254 [Data 54h by SDR Quad]
   TxFIFO8 =0x00001276 [Data 76h by SDR Quad]

Note: Please confirm bit0:WIP in Status Register about end of writing.

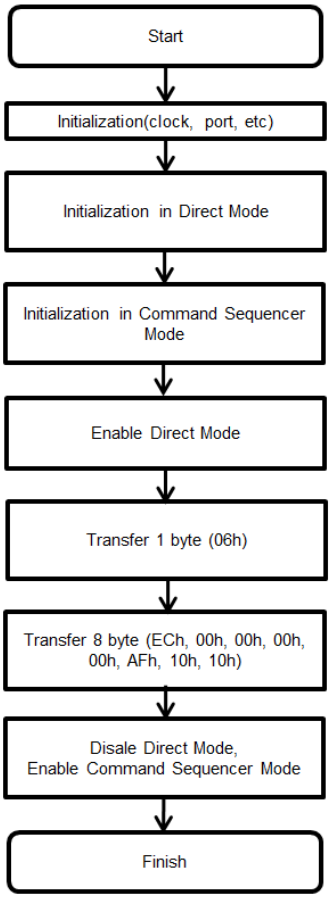
6.2 Read

This program can be read in Command Sequencer Mode to external Flash by SDR Quad.

6.2.1 Flowchart

Figure 17 shows sample program flowchart below.

Figure 17. Flowchart for Sample Program for SDR Quad Read



6.2.2 Initialization in Direct Mode

Sample program is configured about Direct Mode as below.

Table 8. Initialization in Direct Mode for SDR Quad Read

	Register	Detail
1	MCTRL	CSEN=0b, MEN=0b
2	DMTRP	DDRM=0b, TRP=1010b
3	PCC0	SSELDEASRT=11111b, CDRS=1111b, SS2CD =01b
4	DMCFG	SSDC=1b
5	DMPSEL	PSEL=00b
6	DMFIFOCFG	TXCTRL=1b, FWIDTH=00b
7	MCTRL	CSEN=0b, MEN=1b

6.2.3 Initialization in Command Sequencer Mode

Sample program is configured about Command Sequencer Mode as below.

Table 9. Initialization in Command Sequencer Mode for SDR Quad Read

	Register	Detail
1	CSCFG	NOCTAL=0b, MSEL=1100b, SSEL=4'b1111, DDRM=0, MBM=2'b10
2	RDCSDC0	DEC=0x03h
3	RDCSDC1	DEC=0x02h
4	RDCSDC2	DEC=0x01h
5	RDCSDC3	DEC=0x00h
6	RDCSDC4	DEC=0xA5h
7	RDCSDC5	DEC=0x1Ch(4Tri)
8	RDCSDC6	DEC=0x07h(EOF)

6.2.4 Transfer to External Flash in Direct Mode

At first, DMBCC is set to number of FIFO in just one transmitting.

The content of transmitting is as below,

1. Transfer 8byte (ECh,00h,00h,00h,00h,AFh,10h,10h)
   TxFIFO0 =0x000010EC [Command ECh(Command: Quad I/O High Performance Read)]
   TxFIFO1 =0x00001200 [Address=00h (0x00XXXXXX) by SDR Quad]
   TxFIFO2 =0x00001200 [Address=00h (0xXX00XXXX) by SDR Quad]
   TxFIFO3 =0x00001200 [Address=00h (0xXXXX00XX) by SDR Quad]
   TxFIFO4 =0x00001200 [Address=00h (0xXXXXXX00) by SDR Quad]
   TxFIFO5 =0x000012AF [Mode bit =AFh]
   TxFIFO6 =0x00000010 [Dummy or sclk]
   TxFIFO7 =0x00000010 [Dummy or sclk]

Note: Please confirm bit0:WIP in Status Register about end of writing.

A Appendix

A.1 Sampling Coordination

SDATASMPTCNT(SDATA Sample Point Center Control, Delayed sample Clock) is configured the clock which is generated by Delay Buffer.

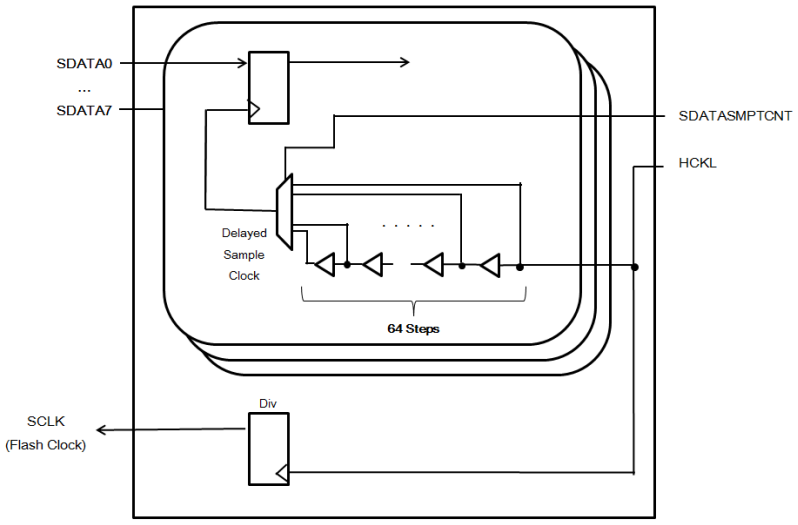
Table 10 is shown the delay time per 1step (1 SDATASMPTCNT).

Table 10. Operation Assurance Condition

	Min	Typ	Max	Unit
1 Step	0.2	0.39	0.5	ns

There is a variation in 0.3ns per 1step. When you set many steps, the variation is increased.

Figure 18. Circuit Construction in Delayed Sample Clock



A.2 Configuration of SDATASMPTCNT/LFT/RGH

Regarding configuration of SDATASMPTCNT/LFT/RGH, we recommend using following procedure.

1. Configuration of SDATASMPTCNT in Initialization

In initialization, you configure the SDATASMPTCNT.

In this time, you confirm Pass/Fail area in each step and each pin.

Finally, you set value of SDATASMPTCNT/LFT/RGH.

We recommend setting SDATASMPTLFT/RGH to ±2steps for SDATASMPTCNT in order to ensure Setup/Hold.

(ex: When you set SDATASMPTCNT = 30, SDATASMPTLFT = 28 and SDATASMPTRGH = 32.)

More information, please see 4.2 Example Configuration of Sampling Point.

The processing time is about 1.5ms by 80MHz in Flash Clock.

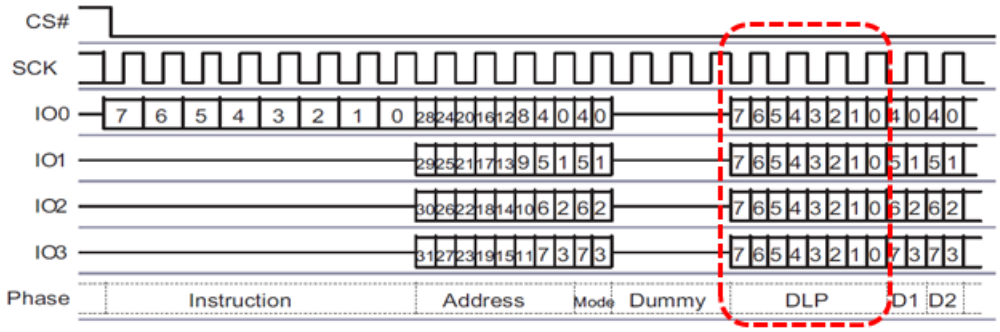
Also, we recommend configuring in CPU startup.

2. Error Check Using DLP(Data Learning Point) Capability

When DLP capability is ON, DLP value is output to dummy cycle in ROM reading.

(Please see Figure 19)

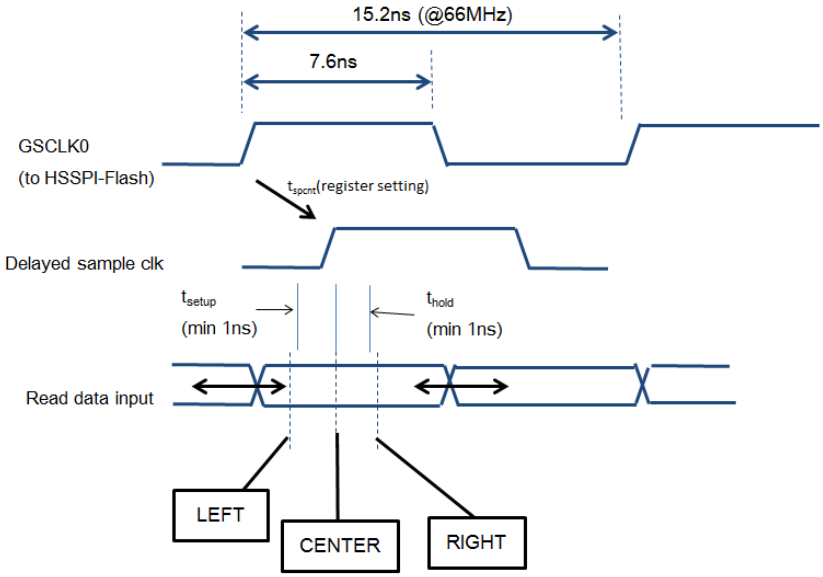
Figure 19. Example of Access Cycle of QSPI Flash



In this time, if you set SDATASMPTLFT and SDATASMPTRGH in before procedure (1), DLP reading value is checked error or not using DLPSAMPLESTATUS register.

(Please see Figure 20.)

Figure 20. Left/Center/Right in Sample Point



3. Change the SDATASMPTCNT/LFT/RGH Value

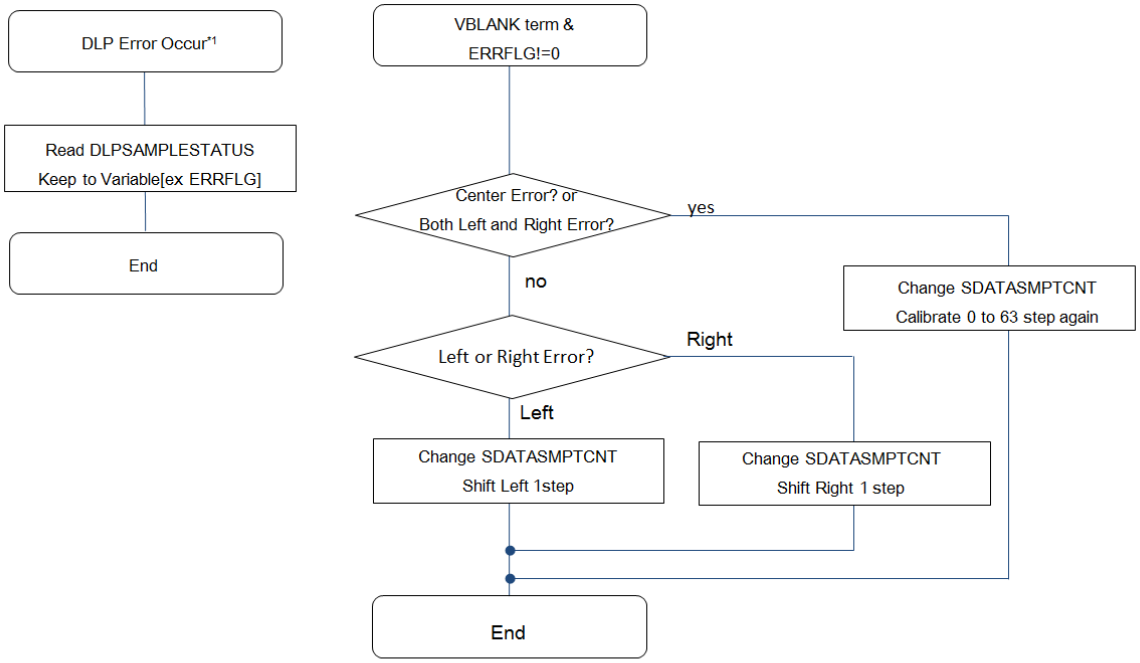
When the error is occurred in error check using DLP capability, you need to change the value of SDATASMPTCNT/LFT/RGH.

For example, DLPSMPLSTxL in DLPSAMPLESTATUS register is “1”, SDATASMPTCNTx/LFTx/RGHx are added “1”.

A.3 Flowchart in an Error Occurrence

Figure 21 is flowchart in the error occurrence.
In this figure, we assume that we want to change value of SDATASMPTCNT/LFT/RGH during VBLANK term.

Figure 21. Flowchart in an Error Occurrence

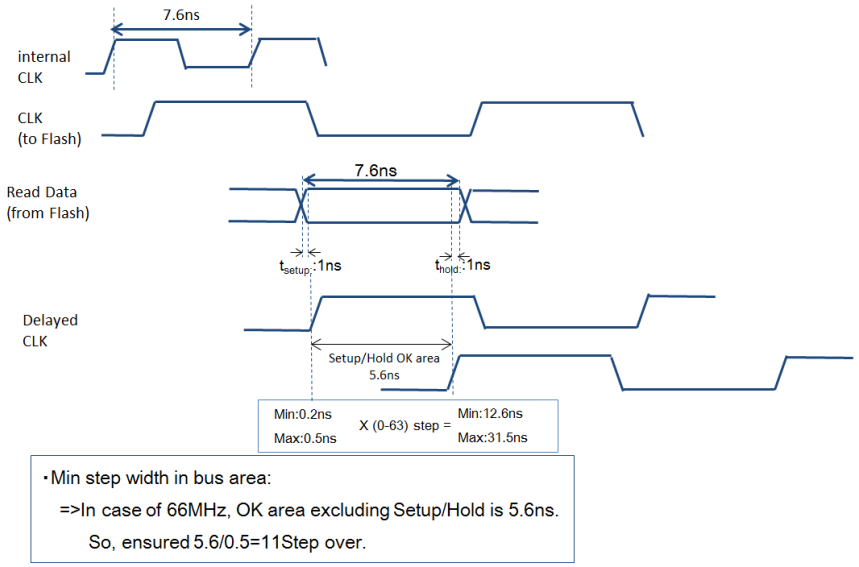


*1:Indicated [bit7] DLPERR : Data Learning Pattern Reception Error
in DDRHSSPI RX Interrupt Flag Register (DDRHSSPIn_RXF)

A.4 Example of Setup/Hold OK Area Calculation

Figure 22 is shown example of setup/Hold OK area calculation (HSSPI CLK = 66MHz).

Figure 22. Example of Setup/Hold OK Area Calculation



*Please consider for your examination about the step in your environment.

A.5 Abbreviations

This section explains abbreviations about S6J3200 Series.

Table 11. Abbreviations about S6J3200 Series

Abbreviations	Meaning
SDR	Single Data Rate
DDR	Dual Data Rate
DLP	Data Learning Pattern

7 Document History

Document Title: AN204454 - How to Use S6J3200 Quad Flash, Traveo™ Family

Document Number: 002-04454

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	-	KHAS	07/27/2015	Initial Release
*A	5041950	KHAS	12/08/2015	Migrated Spansion Application Note from AN708-00013-1v0-E to Cypress format
*B	5782386	AESATMP8	06/27/2017	Updated logo and Copyright.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

Cypress Developer Community

Forums | WICED IOT Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709