NXP Semiconductors
Application Note

Document Number: AN5288
Rev. 0, 09/2016

MPC5777C STCU Quick Start Guide

by: Steven Becerra and Jamaal Fraser

1 Introduction

This Application Note gives the user a simple overview on how to program the Self-Test Control Unit (STCU) to perform a full off-line and on-line Built-In Self-Test (BIST) on all memory and logic partitions. The term Built-In Self-Test is used to describe the set of on-chip hardware mechanisms that can be used to detect latent faults within the MCU. As the name suggests, the BIST allows the MCU to self-test and identify faults with the aim of taking appropriate action. On the MPC5777C, the BIST provides the ability to meet the latent fault detection requirements defined by the ISO 26262 functional safety standard. There are many similarities between the on-line and off-line BIST, and any differences will be noted in this document. If the user programs the registers as described in the Clock configuration and STCU Registers sections in the order they are presented, the STCU will be configured to run a BIST on all memory and logic partitions.

NOTE

STCU BIST configurations that differ from the configurations described in this document must be evaluated by the user, in the user’s application, to ensure proper functionality.

Contents

1.1 Acronyms and definitions

The following table contains acronyms and abbreviations used in this application note.

Table 1. Acronyms and definitions

Term	Meaning
BIST	Built-In Self-Test
LBIST	Logic Built-In Self-Test
MBIST	Memory Built-In Self-Test
WDG	Watchdog
MISR	Multiple Input Signature Register
POR	Power On Reset
SSCM	System Status and Configuration Module
FCCU	Fault Collection and Control Unit
FOSU	FCCU Output Supervision Unit
DCF	Device Configuration Format
PLL	Phase-Locked Loop
Partition	Segment of memory or logic

1.2 Self-Test Control Unit (STCU)

The STCU controls the execution of BISTs.

The STCU manages two primary types of BISTs:

- MBIST: Memory BIST (SRAM/ROM)
- LBIST: Logic BIST (digital logic)

The STCU has two sets of conditions under which it applies a self-test sequence:

- Off-line: After the user stores self-test parameters as DCF records in UTEST flash and a reset cycle is initiated by a power-up, RESET pin assertion, or FOSU reset, the STCU loads the stored parameters, executes a self-test, and then resets the chip.
- On-line: After the user initiates one or more self-tests during normal chip operation, the STCU executes the self-tests and then resets the chip.

The STCU controls the PLL configuration for the off-line BIST and can monitor the PLL lock signal to check if the PLL is working properly during the on-line BIST. The STCU controls the reset of the chip during both off-line and on-line self-tests. If both MBISTs and LBISTs are scheduled to be executed, the MBIST should be executed first, and the LBIST should be executed last. The order of BIST execution is critical, because an STCU reset will occur after the last scheduled LBIST.

2 Built-In Self-Test (BIST)

The term Built-In Self-Test (BIST) is used to describe the on-chip hardware mechanisms that can be used to detect latent faults within the MCU. The BIST allows the MCU to conduct periodic self-tests to identify faults. The results of these self-tests can then be used by the MCU to handle the faults and ensure that the device remains in a safe state. On MPC5777C, BIST provides the ability to meet the latent fault detection requirements defined by the ISO 26262 functional safety standard.

2.1 Memory BIST

MBIST is implemented for each of the memories on the MCU listed in Table 2, including memory contained in the peripheral modules. For MBIST testing purposes, each of the memories is segmented into individual MBIST partitions. MPC5777C implements 65 memory partitions, and each partition has an individual MBIST controller. The STCU controls the MBIST execution of all partitions. Each partition is a member of a partition group, so for example, MBIST0 through MBIST3 are a part of partition group 1 as shown in Table 2.

The MBIST partitions and their partition groups are shown in Table 2. The labels of partition groups are described in MBIST scheduling.

Table 2. MBIST partitions

Partition Group	MBIST Partition	Memory
1 (sequential)	MBIST0	SYSRAM 1_4
	MBIST1	SYSRAM 1_3
	MBIST2	SYSRAM 1_2
	MBIST3	SYSRAM 1_1
2 (sequential)	MBIST4	SYSRAM 2_4
	MBIST5	SYSRAM 2_3
	MBIST6	SYSRAM 2_2
	MBIST7	SYSRAM 2_1
3 (dedicated)	MBIST8	FEC FIFO RAM
4 (dedicated)	MBIST9	FEC MIB RAM
5 (sequential)	MBIST10	DMA RAM 2
	MBIST11	DMA RAM 1
6 (sequential)	MBIST12	ETPU N3 SCM RAM 2
	MBIST13	ETPU N3 SCM RAM 1
7 (dedicated)	MBIST14	ETPU S32D SCM RAM
8 (sequential)	MBIST15	ETPU N3 SPM RAM 3
	MBIST16	ETPU N3 SPM RAM 2
	MBIST17	ETPU N3 SPM RAM 1
9 (sequential)	MBIST18	ETPU S32D SPM RAM 2
	MBIST19	ETPU S32D SPM RAM 1
10 (dedicated)	MBIST20	CSE RAM
11 (parallel)	MBIST21	CORE1 CACHE DATA RAM 1_4
	MBIST22	CORE1 CACHE DATA RAM 1_3
	MBIST23	CORE1 CACHE DATA RAM 1_2
	MBIST24	CORE1 CACHE DATA RAM 1_1
12 (parallel)	MBIST25	CORE1 CACHE DATA RAM 2_4
	MBIST26	CORE1 CACHE DATA RAM 2_3
	MBIST27	CORE1 CACHE DATA RAM 2_2
	MBIST28	CORE1 CACHE DATA RAM 2_1
13 (parallel)	MBIST29	CORE0 CACHE DATA RAM 1_4

Table continues on the next page...

Table 2. MBIST partitions (continued)

Partition Group	MBIST Partition	Memory
	MBIST30	CORE0 CACHE DATA RAM 1_3
	MBIST31	CORE0 CACHE DATA RAM 1_2
	MBIST32	CORE0 CACHE DATA RAM 1_1
14 (parallel)	MBIST33	CORE0 CACHE DATA RAM 2_4
	MBIST34	CORE0 CACHE DATA RAM 2_3
	MBIST35	CORE0 CACHE DATA RAM 2_2
	MBIST36	CORE0 CACHE DATA RAM 2_1
15 (parallel)	MBIST37	CORE1 CACHE TAG RAM 1_4
	MBIST38	CORE1 CACHE TAG RAM 1_3
	MBIST39	CORE1 CACHE TAG RAM 1_2
	MBIST40	CORE1 CACHE TAG RAM 1_1
16 (parallel)	MBIST41	CORE1 CACHE TAG RAM 2_4
	MBIST42	CORE1 CACHE TAG RAM 2_3
	MBIST43	CORE1 CACHE TAG RAM 2_2
	MBIST44	CORE1 CACHE TAG RAM 2_1
17 (parallel)	MBIST45	CORE0 CACHE TAG RAM 1_4
	MBIST46	CORE0 CACHE TAG RAM 1_3
	MBIST47	CORE0 CACHE TAG RAM 1_2
	MBIST48	CORE0 CACHE TAG RAM 1_1
18 (parallel)	MBIST49	CORE0 CACHE TAG RAM 2_4
	MBIST50	CORE0 CACHE TAG RAM 2_3
	MBIST51	CORE0 CACHE TAG RAM 2_2
	MBIST52	CORE0 CACHE TAG RAM 2_1
19 (dedicated)	MBIST53	CORE1 CACHE DIRTY RAM
20 (dedicated)	MBIST54	CORE0 CACHE DIRTY RAM
21 (sequential)	MBIST55	FLEXCAN RAM 1_2
	MBIST56	FLEXCAN RAM 1_1
22 (sequential)	MBIST57	FLEXCAN RAM 2_2
	MBIST58	FLEXCAN RAM 2_1
23 (dedicated)	MBIST59	MCAN RAM
24 (sequential)	MBIST60	ETPU NEXUS RAM 1_2
	MBIST61	ETPU NEXUS RAM 1_1
25 (sequential)	MBIST62	ETPU NEXUS RAM 2_2
	MBIST63	ETPU NEXUS RAM 2_1
26 (dedicated)	MBIST64	BAM ROM

2.1.1 MBIST coverage

The STCU can be configured to run one of three Test Modes; Full, Reduced, and Auto. The different tests allow for different coverage levels. The test that allows for the most coverage is the Full Test Mode followed by the Reduced Test Mode. The test with the least coverage but fastest execution time is the Auto Test Mode.

- Full Test Mode tests memory using all algorithms including the open PMOS algorithm.
- The Reduced Test Mode tests memory using all algorithms except the open PMOS algorithm and is not recommended unless the Full Test does not meet time contraints.
- The Auto Test Mode uses a smaller set of algorithms which target latent defect mechanisms.

The Full Test Mode is what is used in the NXP factory to test the RAM. Since NXP has already tested the parts prior to delivery, the primary concern for the user should be to detect latent defects. The Full Test is recommended for the on-line BIST since time contraints are typically not as critical in the user's on-line use case.

The Auto Test mode is designed to quickly test the RAM with good fault coverage and is recommended for the off-line BIST since this mode has an optimum balance of test execution time versus fault coverage.

The recommended off-line and on-line configurations are defined in STCU2_CFG . See BIST execution time for details on overall execution time. An interactive spreadsheet detailing BIST execution time per partition is attached to this document, BIST Execution Time Spreadsheet.

Table 3 shows the bit-field values necessary to program each Test Mode.

Table 3. Test mode configuration

Bit-field	Full Test	Reduced Test	Auto Test
STCU_CFG[MBU]	0	0	1
STCU_CFG[PMOSEN]	1	0	X (Don't care)

For example, to configure MBIST with the Full Test, STCU_CFG[MBU] should equal 0 and STCU_CFG[PMOSEN] should equal 1.

2.1.2 MBIST scheduling

To minimize overall MBIST execution time all MBIST partitions should be scheduled to execute concurrently except for the last MBIST. If the next BIST is a LBIST the last MBIST should be scheduled to execute sequentially with the first LBIST by writing 0 to STCU2_MB_CTRLn[CSM] as described in Table 12.

All MBIST partitions should be programmed to execute concurrently, however, not all MBIST partitions are capable of executing concurrently. The capability of an MBIST partion to run concurrently is determined by the partition group's label as shown in Table 2.

- Partitions in a partition group labeled "sequential" can only run sequentially within their partition group even if programmed to run concurrently.
- Partitions in a partition group labeled "parallel" can run concurrently within their partion group.
- Partitions from different partion groups can run concurrently no matter what their partion group label is.

Table 4 shows three examples of sequential and concurrent MBIST scheduling and how the partition group label affects overall execution time. Let the execution time of each MBIST be equal to *T* for this example.

- Example 1: MBIST0 and MBIST1 are programmed to run sequentially, and the total execution time is 2*T*.
- Example 2: MBIST0 and MBIST1 are programmed to run concurrently, but the overall execution time is still 2*T* since MBIST0 and MBIST1 cannot be run concurrently because both MBIST partitions are from the same partition group thatis labeled "sequential". Both MBISTs should still be configured to run concurrently because the programmed mode has an affect on subsequently programmed MBISTs.
- Example 3: MBIST0 and MBIST4 are programmed to run concurrently, and the overall execution time is *T* since the partitions are from separate partition groups.

Table 4. MBIST scheduling example

Example Number	MBIST scheduled to run	Partition Group	Programmed Mode (STCU2_MB_CTRLn[CSM])	Overall execution time (*T* = individual execution time of MBIST0, MBIST1, MBIST4)
1	MBIST0, MBIST1 (same partition group)	1 (sequential)	Sequential	2*T*
2	MBIST0, MBIST1 (same partition group)	1 (sequential)	Concurrent	2*T*
3	MBIST0, MBIST4 (different partition groups)	1 and 2	Concurrent	*T*

2.2 Logic BIST

The LBIST tests operate on the digital logic of the device and use scan test techniques to provide high coverage defect detection. The logic is divided up into multiple partitions, with each partition containing multiple user recognizable logic modules (CPU, XBAR, FlexCAN, etc). MPC5777C implements six logic partitions, and each partition has an individual LBIST controller. The STCU controls the LBIST execution of all partitions. All LBISTs must run sequentially.

The LBIST partitions and their contents are shown in Table 5.

Table 5. LBIST partitions

LBIST Partition	Module
LBIST0	Core1
LBIST1	Core0
LBIST2	Checker core, RCCU
LBIST3	Peripheral Bridge B (AIPS_0) and associated peripherals: • SDADC_1 and SDADC_3 • M_CAN Subsystem • CRC • DECFILTER_A to DECFILTER_L • LFAST • DSPI_A, DSPI_B, and DSPI_C • eMIOS_0 • EQADC_A • eSCI_A and eSCI_B • eTPU_C • FCCU including FOSU • FlexCAN_0 and FlexCAN_1 • IGF_0 • PSI5_0 • REACM2 • SRX_0 • SIPI
LBIST4	Peripheral Bridge A (AIPS_1) and associated peripherals: • SDADC_2 and SDADC_4 • CMU_0 to CMU_8 • DSPI_D and DSPI_E • EBI

Table continues on the next page...

Table 5. LBIST partitions (continued)

LBIST Partition	Module
	• eMIOS_1 • EQADC_B • eSCI_D, eSCI_E, and eSCI_F • eTPU_A/B • FCCU including FOSU • FlexCAN_2 and FlexCAN_3 • IGF_1 • PIT_RTI • PSI5_1 • SRX_1
LBIST5	Core platform modules: • XBAR • EIM • ERM • PCU • MPU_0 • MPU_1 • CSE • FEC • PFLASH • INTC • PRAMC_0 • PRAMC_1 • eDMA_A • eDMA_B • STM • SWT_A • SWT_B • C55FMC flash memory array

2.2.1 LBIST coverage

The percentage of the logic gates that will be tested during the BIST, known as the test coverage level, is determined by the number of patterns that are run on each partition. The number of patterns that are run is controlled by configuring the PCS bit-field in the STCU2_PCSn register. As the number of gates varies in each partition, the number of patterns required to achieve a determined level of coverage will also vary. The test level coverage achieved for the examples in the applications note meet at least 90% coverage of all logic in the logic partitions. Reducing the number of patterns run on a partition reduces the time taken to complete the BIST for that partition, but also reduces the coverage level. The pattern count for each partition is configured within the STCU.

2.3 BIST execution time

To calculate the execution time of an individual LBIST partition, the following formula should be used

$$\mathrm{LBIST\_RUNTIME(ms)} = \frac{\mathrm{STCU2\_LB\_PCSn[PCS]} \times (\mathrm{Number\ of\ Shifts} + \mathrm{Number\ of\ Captures}) \times (\mathrm{STCU2\_LB\_CTRLn[SHS]} + 1)}{f_{\mathrm{PLL0\_PHI}}}$$

Equation 1

Where

- STCU2_LB_PCSn[PCS] : Pattern counter stop (value entered in PCS bit-field)
- Number of Shifts = 50 (fixed by hardware)
- Number of Captures = 25 (fixed by hardware)
- STCU2_LB_CTRLn[SHS] : Shift Speed (value entered in SHS bit-field)

To calculate the execution time of an individual MBIST partition, the following formula should be used

$$\text{MBIST\_RUNTIME}(\text{ms}) = \frac{\text{Memory\_Clocks}}{\text{Memory\_Clock\_Frequency}}$$

Equation 2

Where

- Memory_Clocks: Value dependent on algorithm used and partition size. See attached spreadsheet for values, BIST Execution Time Spreadsheet.
- Memory_Clock_Frequency: A factor of the BIST Frequency. See attached spreadsheet for values, BIST Execution Time Spreadsheet.

The overall execution time and current (mA) profile of off-line BIST is shown in Figure 1. The figure's current profile is only an estimate but is in the range of 100s of milliamps. NXP does not provide a detailed current characterization of the BIST. The off-line BIST in this example uses a 2N45H part and is running MBIST Auto Test mode with PLL0_PHI at 50 MHz for total MBIST execution time of 25.2 milliseconds. For the off-line BIST, the 16 MHz Internal RC oscillator (IRC) is used as the clock source when an LBIST or MBIST is not running. For example, in the time between the last MBIST execution and the first LBIST execution, the STCU clock is sourced by the IRC since a BIST on a partition is not being executed. All number values in the figure show time duration of the segments and are in milliseconds. See MPC5777C_off-line-40MHzXOSC_PLL50MHz_2N45H.cmm script attached to this document for full configuration.

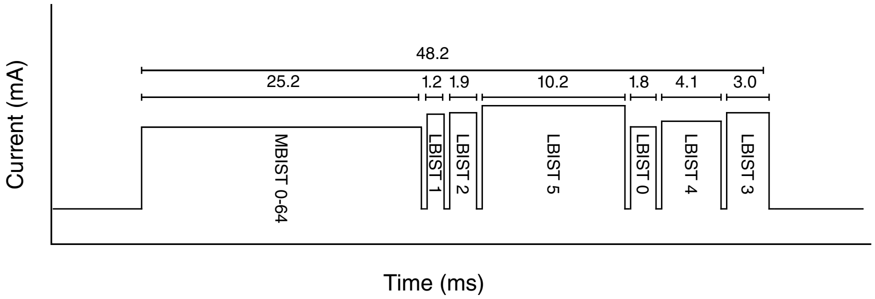


Figure 1. Off-line BIST Current Profile and Timing

The overall execution time and current profile of on-line BIST is shown in Figure 2. The figure's current profile is only an estimate but is in the range of 100s of milliamps. NXP does not provide a detailed current characterization of the BIST. The on-line BIST in this example uses a 2N45H part and is running MBIST Full Test mode with PLL0_PHI at 50 MHz for a total MBIST execution time of 168.3 milliseconds. For the on-line BIST the PLL is used as the clock source when an LBIST or MBIST is not running. For example, in the time between the last MBIST execution and the first LBIST execution, the STCU clock is sourced by the PLL since a BIST on a partition is not being executed. All number values in the figure show time duration of the segments and are in milliseconds. See MPC5777C_on-line-bist_2N45H.c file attached to this document for full configuration.

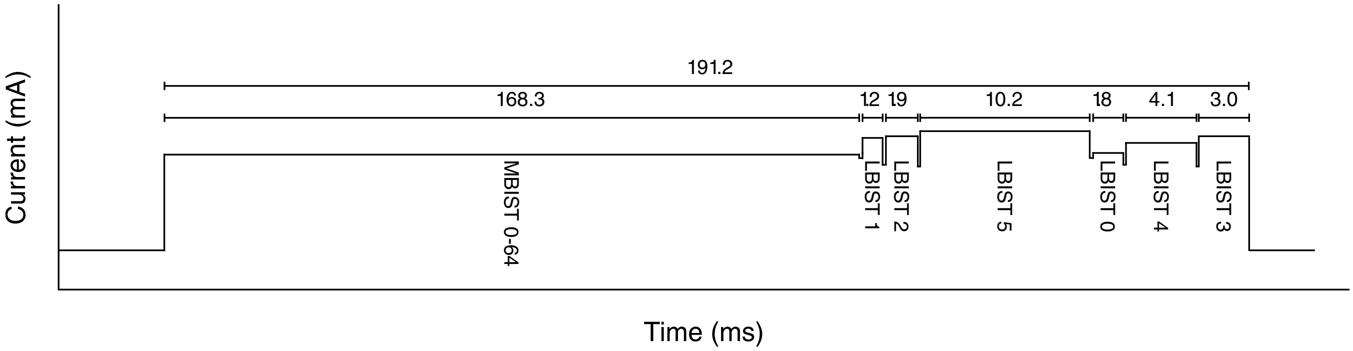


Figure 2. On-line BIST Current Profile and Timing

3 Initializing the STCU

The method to program the STCU for an off-line BIST is different from programming the STCU for an on-line BIST. When programming the STCU for the off-line BIST, DCF records are used to specify off-line self-test parameters in flash. The SSCM processes these records during the system reset sequence before the CPU leaves reset.

3.1 Off-line BIST configuration

A BIST can be configured to execute after a power-up event, reset pin assertion, or an FOSU reset. The BIST is performed before the application code runs while the MCU is powered and held in reset. To configure the off-line BIST, user configurable Device Configuration Format (DCF) records are loaded at start-up by the SSCM module into the STCU to configure the self-test procedure. See DCF Addresses and LTB Commands Spreadsheet for off-line example code and description.

Below are items to consider when programming the STCU for off-line BIST.

- Security key code Key1 and Key2 need to be written to the STCU2_SCK register to unlock STCU2 access. Once unlocked, the hard-coded WDG time-out starts decrementing. The hard-coded WDG timer is a security feature that limits the time the STCU can be accessed. To prevent the WDG time-out, security key code Key2 must be written to the STCU2_SCK register once every 30 instructions.
- In the event that a fault is found by the BIST during an off-line BIST, the MCU can be configured to remain in reset and communicate the unrecoverable fault externally. If the fault is regarded as recoverable, the MCU can exit reset and inform the system of the failure. All faults are mapped as recoverable in this application note, so for further information on fault handling, please see the MPC5777C reference manual.
- When the off-line BIST executes successfully, the device performs an STCU reset, and the application software is executed.

3.2 On-line BIST configuration

The method to program the STCU for an on-line BIST is different from programming the STCU for an off-line BIST. When programming the STCU for the on-line BIST, the MCU allows software to write to the STCU during run time to configure and trigger the execution of MBIST and LBIST. Since MBIST and LBIST are destructive, a reset of the MCU is performed at the end of the test sequence before going into operation again.

Below are items to consider when programming the STCU for on-line BIST.

- Security key code Key1 and Key2 need to be written to the STCU2_SCK register to unlock STCU2 access. Once unlocked, the hard-coded WDG time-out starts decrementing. The hard-coded WDG timer is a security feature that limits the time the STCU can be accessed. To prevent the WDG time-out, security key code Key2 must be written to the STCU2_SCK register once every 30 instructions.
- It is recommended to never run MBIST only since MBIST doesn't issue a reset when it finishes. User should at least program one LBIST partion to follow MBIST execution.
- When the STCU is programmed to run both MBISTs and LBISTs and an error occurs during the MBIST, a reset will not occur. An STCU reset will only occur in the LBIST phase, but the STCU can be configured to handle faults as unrecoverable. The FCCU can reset the device once an MBIST abort has been detected. Details on fault collection and handling is not addressed in this application note. See the Fault Collection Control Unit (FCCU) chapter in the MPC5777C reference manual for details on fault handling.
- Faults can be mapped to unrecoverable which will cause a reset in the event of a fault by setting the desired bit/s in the STCU2_ERR_FM register. All faults are set as recoverable by default.
- If a reset occurs during the LBIST, the LBIST will be aborted, and the results of the LBIST will be loaded into the STCU registers before an STCU reset occurs.
- It is recommended that the DMA, cache, and ETPU are disabled and to hold user code in a while loop for the expected duration of the BIST. The expected duration of the BIST can be calculated by using the attached spreadsheet, see BIST Execution Time Spreadsheet.

3.2.1 Clock configuration

On-line clock configuration should occur before writing to the STCU registers in on-line BIST. The example values that follow program PLL0_PHI to 50 MHz which is the only validated frequency when running MBIST and LBIST together on-line. We have also validated PLL0_PHI at 200 MHz if only running LBIST on-line. Register values are based on using a 40 MHz XOSC as the PLL0 source. Off-line clock configuration is handled in STCU2_PLL_CFG *(Off-line only)* and STCU2_RUN*(Off-line only)*.

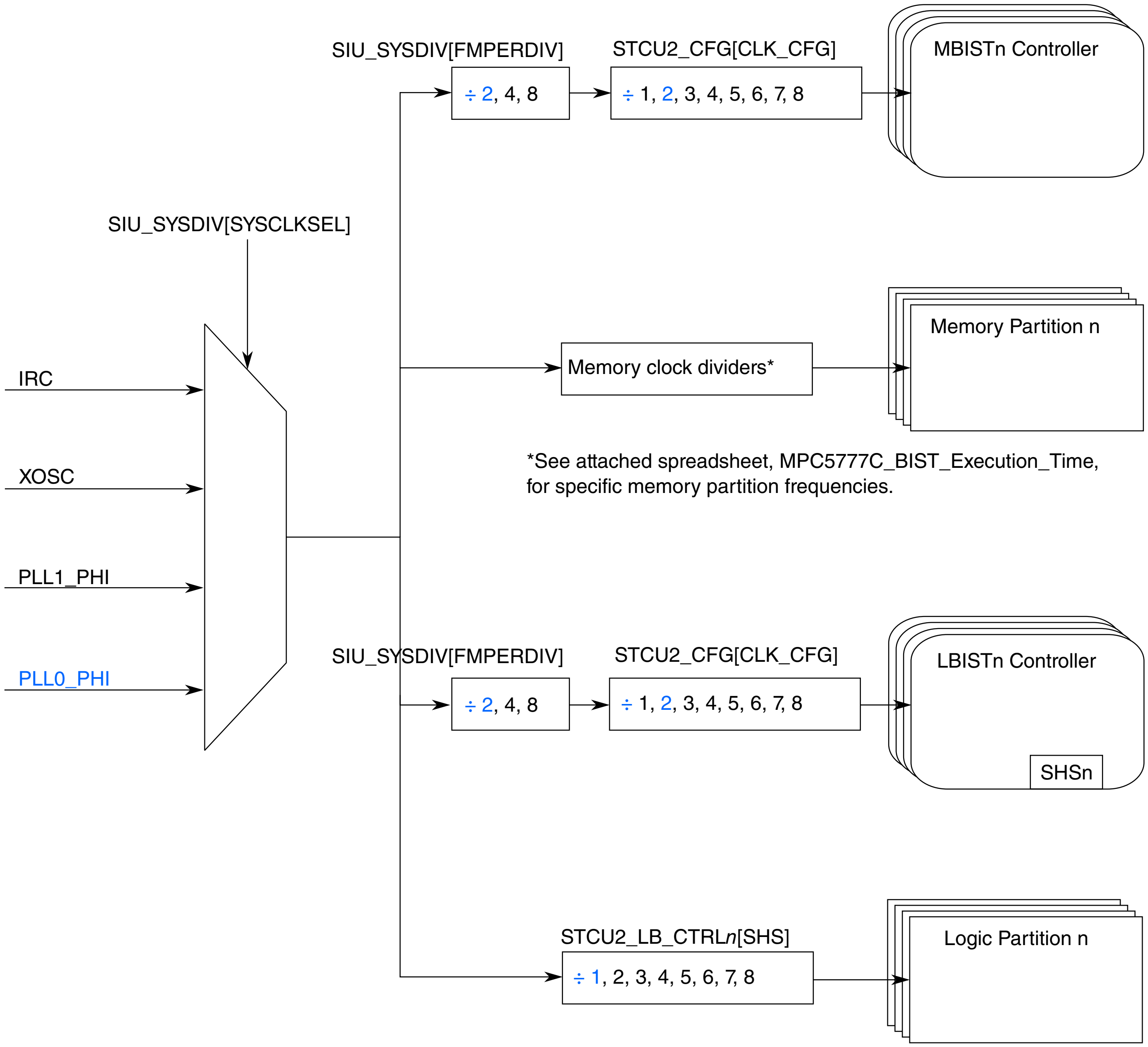


Figure 3. On-line Clock Tree

3.2.1.1 SIU_SYSDIV

Select clock dividers and sources. If the bit-field values used in this register are different than those described below, incorrect MISR values can occur resulting in failed LBISTs.

Recommended Register Value

- **SIU_SYSDIV: 0x0000B010**

Table 6 describes the SIU_SYSDIV bit-fields and their recommended value.

Table 6. SIU_SYSDIV Bit-field Description

Bit-field	Value	Description
PLL0SEL	0x00	The crystal oscillator (XOSC) is the clock source for PLL0.
PLL1SEL	0x00	The PH1 output of PLL0 is the clock source for PLL1.
PERCLKSEL	0x00	Peripheral Clock is connected to the output of the SYSCLKDIV divider
FMPERDIV	0x00	Sets the FM Clock Peripheral Divider to Divide by 2.
PERDIV	0x00	Sets the Non-FM Clock Peripheral Divider to Divide by 2.
MCANSEL	0x01	The non-FM peripheral clock is the clock source for the MCAN modules.
SYSCLKSEL	0x03	The system clock (SYS_CLK) is driven by the PLL0_PHI.
ETPUDIV	0x00	Sets the eTPU Clock Divider to Divide by 2
SYSCLKDIV	0x04	Sets the System Clock Divider to Divide by 1.
PCS	0x00	Progressive Clock Switch is disabled.

3.2.1.2 PLLDIG_PLL0DV

Set PLL0 dividers to output 50 MHz at PLL0_PHI

Recommended Register Value

- **PLLDIG_PLL0DV: 0x2006200F**

Table 7 describes the PLLDIG_PLL0DV bit-fields and their recommended value.

Table 7. PLLDIG_PLL0DV Bit-field Description

Bit-field	Value	Description
RFDPHI	0x06	Set VCO clock post-divider to Divide by 6
PREDIV	0x02	Set the input clock divider to Divide by 2
MFD	0x0F	Set multiplication factor applied to reference frequency

The relationship between input and output frequency of the PLL is determined by programming the PLL0DV register and is calculated according to the following formula.

$$f_{\mathrm{PLL0\_PHI}} = f_{\mathrm{PLL0\_REF}} \times \frac{\mathrm{PLL0DV[MFD]}}{\mathrm{PLL0DV[PREDIV]} \times \mathrm{PLL0DV[RFDPHI]}}$$

Equation 3

Where

- $f_{\mathrm{PLL0\_PHI}}$ = output frequency of PLL0_PHI
- $f_{\mathrm{PLL0\_REF}}$ = frequency of PLL0 reference clock
- PLL0DV[MFD] = Loop multiplication factor divider
- PLL0DV[PREDIV] = Input clock predivider
- PLL0DV[RFDPHI] = PHI reduced frequency divider

4 STCU Registers

All registers should be written in the order listed below. The values used reflect the attached example configurations of running both MBSIT and LBIST off-line (MPC5777C_off-line_40MHzXOSC_PLL50MHz_*n*N45H.cmm) and on-line (MPC5777C_on-line_bist_*n*N45H.c) with PLL0_PHI set to 50 MHz which is the only validated MBIST plus LBIST configuration. In some cases, off-line and on-line BISTs require different values, and any differences will be indicated in the register descriptions.

4.1 STCU2_SKC

Unlock write access to the STCU2. Key1 and Key2 should be written before attempting to write to any other STCU register. Writing Key1 and Key2 to the STCU2_SKC register intializes the WDG time-out counter. After initializing the WDG time-out counter, Key2 should be written once every 30 instructions to re-initialize the counter. If the WDG time-out counter expires before either STCU2_RUN[RUN] or STCU2_RUN[BYP] are set, the STCU2_ERR_STAT[WDTO] flag will assert in the off-line case and STCU2_ERR_STAT[WDTOSW] flag will assert in the on-line case.

Table 8 describes the STCU2_SKC bit-fields for off-line BIST and their recommended value.

Table 8. STCU2_SKC Bit-field Description *(Off-line only)*

Bit-field	Value	Description
SKC	0xD3FEA98B	Key1 to unlock write access to the STCU
SKC	0x2C015674	Key2 to unlock write access to the STCU and re-initialize the WDG time-out counter

Table 9 describes the STCU2_SKC bit-fields for on-line BIST and their recommended value.

Table 9. STCU2_SKC Bit-field Description *(On-line only)*

Bit-field	Value	Description
SKC	0x753F924E	Key1 to unlock write access to the STCU
SKC	0x8AC06DB1	Key2 to unlock write access to the STCU and re-initialize the WDG time-out counter

4.2 STCU2_CFG

Start at MBIST 0, run Auto Test for off-line BIST and Full Test for on-line BIST. The Auto Test is designed to quickly test the RAM with good fault coverage. The Full Test is a more comprehensive test, but is recommended for on-line BIST as the time requirement is greater than the Auto Test.

Recommended Register Value

- **STCU2_CFG: 0x10000009 *(off-line)***
- **STCU2_CFG: 0x10000011 *(on-line)***

Table 10 describes the STCU2_CFG bit-fields and their recommended value.

Table 10. STCU2_CFG Bit-field Description

Bit-field	Value	Decription
PTR	0x10	Set pointer to run MBIST0 first.
MBU	0x1 (off-line) 0x0 (on-line)	Run the Auto Test (off-line only). During on-line BIST, this bit should be set to 0 to run the Full Test. See Table 3 for more details.
PMOSEN	0x0 (off-line) 0x1 (on-line)	Run the Auto Test (off-line only). During on-line BIST, this bit should be set to 1 to run the Full Test. See Table 3 for more details.

4.3 STCU2_PLL_CFG *(Off-line only)*

Set PLL0 dividers to output 50 MHz at PLL0_PHI. When using the PLL for the Off-Line BIST, the XOSC is the PLL0 reference clock by default and cannot be changed. Register values are based on a 40 MHz XOSC. For details on how to configure the PLL to different frequencies and using an oscillator with a value other than 40 MHz, see the Functional description section of the Dual PLL Digital Interface chapter of the MPC5777C Reference Manual.

Recommended Register Value

- **STCU_PLL_CFG: 0x0602000F**

Table 11. STCU2_PLL_CFG Bit-field Description

Bit-field	Value	Description
PLLODF	0x06	PLLODF is equivalent to RFDPHI in the PLLDIG PLL0 Divider Register (PLLDIG_PLL0DV)
PLLIDF	0x02	PLLIDF is equivalent to PREDIV in the PLLDIG PLL0 Divider Register (PLLDIG_PLL0DV)
PLLLDF	0x0F	PLLLDF is equivalent to MFD in the PLLDIG PLL0 Divider Register (PLLDIG_PLL0DV)

The relationship between input and output frequency of the PLL is determined by programming the STCU2_PLL_CFG register and is calculated according to the following formula.

$$f_{\mathrm{PLL0\_PHI}} = f_{\mathrm{PLL0\_REF}} \times \frac{\mathrm{STCU2\_PLL\_CFG[PLLLDF]}}{\mathrm{STCU2\_PLL\_CFG[PLLIDF]} \times \mathrm{STCU2\_PLL\_CFG[PLLODF]}}$$

Equation 4

Where

- $f_{\mathrm{PLL0\_PHI}}$ = output frequency of PLL0_PHI
- $f_{\mathrm{PLL0\_REF}}$ = frequency of PLL0 reference clock
- STCU2_PLL_CFG[PLLLDF] = Loop multiplication factor divider
- STCU2_PLL_CFG[PLLIDF] = Input clock predivider
- STCU2_PLL_CFG[PLLODF] = PHI reduced frequency divider

4.4 STCU2_WDG

Below are the recommended watchdog timer settings for BIST executionn in off-line (61.44 ms) and on-line (300 ms) modes as described in this application note. The watchdog runs off the BIST controller clocks. In off-line mode the STCU watchdog timer starts counting as soon as the STCU2_RUN[RUN] bit is set. However the BIST execution won't start until the RESET pin is released and the PLL is locked. If the user's application holds the RESET pin low for any reason then that amount of time needs to be added into the off-line watchdog value. The oscillator is powered by VDDEH6 so if the user's application delays or brings up VDDEH6 last then the oscillator might not be started and as a result the PLL not locked before the device has started the BIST test execution. In this case the user may need to add extra time ot the off-line watchdog value as well.

Recommended Register Value

- **STCU2_WDG: 0x0000BB80** ***(off-line)***
- **STCU2_WDG: 0x000395F8** ***(on-line)***

4.5 STCU2_ERR_FM

Map all STCU2 faults as recoverable.

Recommended Register Value

- **STCU2_ERR_FM: 0x00000000**

4.6 STCU2_LBRMSW *(On-line only)*

Set the global functional reset to be executed at the end of last scheduled LBIST. Writing a 1 to any of the STCU2_LBRMSW[LBRMSWn] bit fields will cause a global reset after the last scheduled LBIST since we are executing all of the LBIST partions.

Recommended Register Value

- **STCU2_LBRMSW: 0x0000003F**

4.7 STCU2_LBUFM

Map all LBIST faults as recoverable.

Recommended Register Value

- **STCU2_LBUFM: 0x00000000**

4.8 STCU2_MBUFMn

Map all MBIST faults as recoverable.

Recommended Register Value

- **STCU2_MBUFML: 0x00000000**
- **STCU2_MBUFMM: 0x00000000**
- **STCU2_MBUFMH: 0x00000000**

4.9 STCU2_MB_CTRLn

Set MBISTs to run concurrently and set the pointer for the next MBIST. MBIST can run sequentially, but running sequentially would greatly increase the overall execution time without any benefit. MBIST 64 pointer should point to LBIST 1. Let n equal 0 to 64.

Recommended Register Values

- **STCU2_MB_CTRL0: 0x91000000**
- **STCU2_MB_CTRL1: 0x92000000**
- **...**
- **STCU2_MB_CTRL63: 0xD0000000**
- **STCU2_MB_CTRL64: 0x01000000**

Table 12. STCU2_MB_CTRLn Bit-field Description

Bit-field	Value	Description
CSM	0x1	Sets MBISTn to run concurrently with next scheduled BIST. **NOTE:** Value should be set to 0 on the last scheduled MBIST, MBIST64, to run sequentially with the first LBIST .
PTR	0x10 + (n + 1)	Sets pointer for next BIST. When n equals 64, PTR value should be set to 0x01 to point to the first LBIST to be excecuted, LBIST1.

4.10 STCU2_LB_CTRLn

Set LBISTs to run sequentially, pointer to next LBIST, and shift speed. LBIST should not be run concurrently. LBISTs should be run in the order: LBIST1, LBIST2, LBIST5, LBIST0, LBIST4, and then LBIST3 to minimize current spikes. Because LBIST 3 is the last to run, its pointer should have the value 0x7F to indicate the end of the LBISTs.

Recommended Register Value

- **STCU2_LB_CTRL0: 0x04004504**

Table 13 describes the STCU2_LB_CTRL0 bit-fields and their recommended value.

Table 13. STCU2_LB_CTRL0 Bit-field Description

Bit-field	Value	Description
PTR	0x04	Sets pointer for next BIST to LBIST4.
SHS	0x00	Sets shift rate in relation to SYS_CLK. See Equation 5 on page 17 for more information.

Recommended Register Value

- **STCU2_LB_CTRL1: 0x02004504**

Table 14 describes the STCU2_LB_CTRL1 bit-fields and their recommended value.

Table 14. STCU2_LB_CTRL1 Bit-field Description

Bit-field	Value	Description
PTR	0x02	Sets pointer for next BIST to LBIST2.
SHS	0x00	Sets shift rate in relation to SYS_CLK. See Equation 5 on page 17 for more information.

Recommended Register Value

- **STCU2_LB_CTRL2: 0x05004504**

Table 15 describes the STCU2_LB_CTRL2 bit-fields and their recommended value.

Table 15. STCU2_LB_CTRL2 Bit-field Description

Bit-field	Value	Description
PTR	0x05	Sets pointer for next BIST to LBIST5.
SHS	0x00	Sets shift rate in relation to SYS_CLK. See Equation 5 on page 17 for more information.

Recommended Register Value

- **STCU2_LB_CTRL3: 0x7F004504**

Table 16 describes the STCU2_LB_CTRL3 bit-fields and their recommended value.

Table 16. STCU2_LB_CTRL3 Bit-field Description

Bit-field	Value	Description
PTR	0x7F	Sets pointer for next BIST to null. BIST execution will end after LBIST3 completes.
SHS	0x00	Sets shift rate in relation to SYS_CLK. See Equation 5 on page 17 for more information.

Recommended Register Value

- **STCU2_LB_CTRL4: 0x03004504**

Table 17 describes the STCU2_LB_CTRL4 bit-fields and their recommended value.

Table 17. STCU2_LB_CTRL4 Bit-field Description

Bit-field	Value	Description
PTR	0x03	Sets pointer for next BIST to LBIST3.
SHS	0x00	Sets shift rate in relation to SYS_CLK. See Equation 5 on page 17 for more information.

Recommended Register Value

- **STCU2_LB_CTRL5: 0x00004504**

Table 18 describes the STCU2_LB_CTRL5 bit-fields and their recommended value.

Table 18. STCU2_LB_CTRL5 Bit-field Description

Bit-field	Value	Description
PTR	0x00	Sets pointer for next BIST to LBIST0.
SHS	0x00	Sets shift rate in relation to SYS_CLK. See Equation 5 on page 17 for more information.

SHS value is based on a system clock at 50 MHz. If the system clock is different than 50 MHz, then SHS should be adjusted to validate the following formula

$$\frac{\mathrm{SYSCLK}}{\mathrm{STCU2\_LB\_CTRLn[SHS]}+1} \leq 50\ \mathrm{MHz}$$

Equation 5

Where

- SYSCLK = core_clk = BIST clock = PLL0 (in this configuration)
- STCU2_LB_CTRLn[SHS] = Shift speed (value programmed in SHS bit-field)

4.11 STCU2_PCSn

Set the Pattern Counter Stop value. If PCS values provided below are programmed into the STCU2 LBIST PC Stop Register (STCU2_LB_PCS*n*), 90 percent of logic will be covered. Different PCS values will output different Multiple Input Signature Register (MISR) values, which will affect LBIST passing criteria.

Recommended Register Value

	2N45H	3N45H
STCU2_PCS0	0x000004C0	0x000004C0
STCU2_PCS1	0x00000340	0x00000340
STCU2_PCS2	0x00000500	0x00000500
STCU2_PCS3	0x00000780	0x00000A00
STCU2_PCS4	0x00000A40	0x00000A40
STCU2_PCS5	0x00001776	0x00001740

4.12 STCU2_LB_MISRELn, STCU2_LB_MISREHn *(Off-line only)*

Set the expected MISR values for off-line BISTs. After completing an LBIST, the MISR Expected values are compared with the MISR Read values to determine if the LBIST has passed. If expected and read values are equivalent, the LBIST has passed, otherwise, the LBIST has failed.

Recommended Register Values

	2N45H	3N45H
STCU2_LB_MISREL0	0x638CABD3	0x638CABD3
STCU2_LB_MISREH0	0xF6FEF134	0xF6FEF134
STCU2_LB_MISREL1	0x29CAE1DA	0x29CAE1DA
STCU2_LB_MISREH1	0xF547396A	0xF547396A
STCU2_LB_MISREL2	0x1AC8F17C	0x1AC8F17C
STCU2_LB_MISREH2	0x84F64979	0x84F64979
STCU2_LB_MISREL3	0x2A96C340	0xC59BF565
STCU2_LB_MISREH3	0x84756583	0x555F4ACF
STCU2_LB_MISREL4	0x15C9A73F	0x15C9A73F
STCU2_LB_MISREH4	0xBD55BE6A	0xBD55BE6A
STCU2_LB_MISREL5	0x9B9F264C	0x6A3C6D9C
STCU2_LB_MISREH5	0xE85AA33F	0x3AB9877C

4.13 STCU2_LB_MISRELSWn, STCU2_LB_MISREHSWn *(On-line only)*

Set the expected MISR values for on-line BISTs. After completing an LBIST, the MISR Expected values are compared with the MISR Read values to determine if the LBIST has passed. If expected and read values are equivalent, the LBIST has passed, otherwise, the LBIST has failed.

Recommended Register Values

	2N45H	3N45H
STCU2_LB_MISRELSW0	0x638CABD3	0x638CABD3
STCU2_LB_MISREHSW0	0xF6FEF134	0xF6FEF134
STCU2_LB_MISRELSW1	0x29CAE1DA	0x29CAE1DA
STCU2_LB_MISREHSW1	0xF547396A	0xF547396A
STCU2_LB_MISRELSW2	0x1AC8F17C	0x1AC8F17C
STCU2_LB_MISREHSW2	0x84F64979	0x84F64979
STCU2_LB_MISRELSW3	0x2A96C340	0xC59BF565
STCU2_LB_MISREHSW3	0x84756583	0x555F4ACF
STCU2_LB_MISRELSW4	0x15C9A73F	0x15C9A73F
STCU2_LB_MISREHSW4	0xBD55BE6A	0xBD55BE6A
STCU2_LB_MISRELSW5	0x9B9F264C	0x6A3C6D9C
STCU2_LB_MISREHSW5	0xE85AA33F	0x3AB9877C

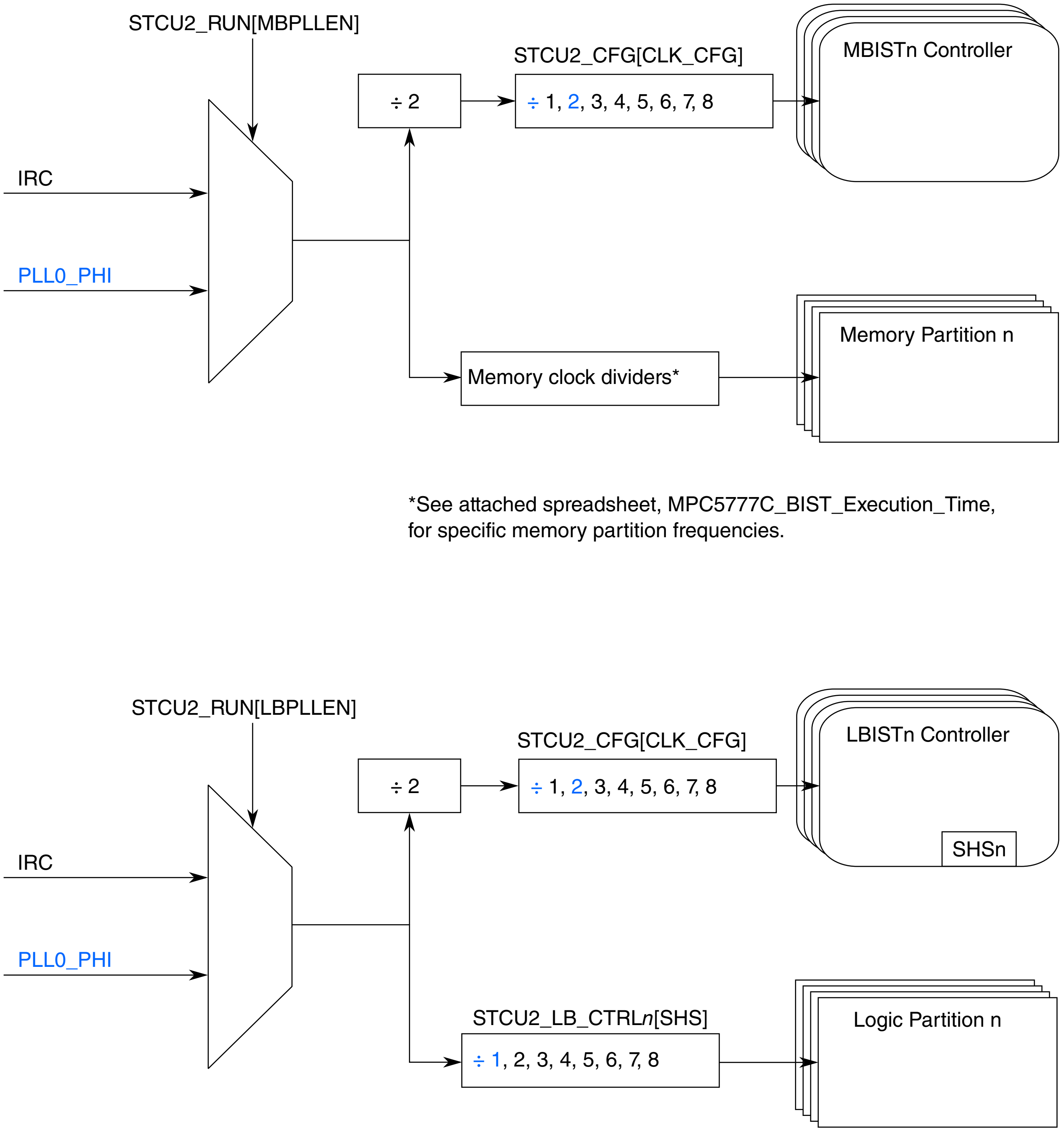
4.14 STCU2_RUN*(Off-line only)*

Set STCU to run with the on-chip PLL using the parameters defined in STCU2_PLL_CFG. For off-line BIST, the PLL is sourced only by XOSC and this source cannot be changed. Setting the BYP bit to enable Bypass mode will permanently bypass any BISTs. Attempting to overwrite the STCU2_RUN register by adding an additional DCF record will not overwrite the register. For additional details on DCF records, see the MPC5777C reference manual.

Recommended Register Values

- **STCU2_RUN: 0x00000301**

Figure 4 shows the clock sources that feed into the MBIST and LBIST controllers. Under the configuration presented in this document the 65 MBIST controllers and the 6 LBIST controllers are run at 12.5 MHz, one quarter of the SYS_CLK (50 MHz PLL0_PHI). The clock source of the MBIST and LBIST controllers is set to PLL0_PHI by setting MBPLLEN and LBPLLEN bits in STCU2_RUN to one, respectively.



*See attached spreadsheet, MPC5777C_BIST_Execution_Time, for specific memory partition frequencies.

Figure 4. STCU Off-line Clock Tree

4.15 STCU2_RUNSW *(On-line only)*

Set STCU to run with PLL monitoring enabled.

Recommended Register Values

- **STCU2_RUN: 0x00000301**

5 Processing BIST results

For the off-line BIST, the STCU configuration is completed by the SSCM, the RUN bit in the STCU2_RUN register is set, and, after a power on reset (POR), the BIST is run. After the BIST is complete, another reset will occur, and the device will go to user code.

For the on-line BIST, after the STCU is configured by user software and the RUN bit in the STCU2_RUNSW register is set, the BIST will run. After the BIST is complete, a functional reset will occur. The following section describes how to confirm a BIST has passed and how to handle any potential errors.

5.1 BIST status

To confirm the off-line and on-line BIST have passed successfully, check the following registers.

Table 19. Off-line BIST Status Registers

Status Register	Description
Off-line LBIST Status Register (STCU2_LBS)	LBS0 - LBS5 are asserted after a successful LBIST on their respective partitions
Off-line MBIST Status Low Register (STCU2_MBSL)	MBS0 - MBS31 are asserted after a successful MBIST on their respective partitions **NOTE:** MBS0 - MBS31 are meaningful when the related bit of the STCU2_MBEL registers reports the MBIST is finished.
Off-line MBIST Status Medium Register (STCU2_MBSM)	MBS32 - MBS63 are asserted after a successful MBIST on their respective partitions **NOTE:** MBS32 - MBS63 are meaningful when the related bit of the STCU2_MBEM registers reports the MBIST is finished.
Off-line MBIST Status High Register (STCU2_MBSH)	MBS64 is asserted after a successful MBIST on this partition **NOTE:** MBS64 is meaningful when the related bit of the STCU2_MBEH registers reports the MBIST is finished.

Table 20. On-line BIST Status Registers

Status Register	Description
On-line LBIST Status Register (STCU2_LBSSW)	LBSSW0 - LBSSW5 are asserted after a successful LBIST on their respective partitions
On-line MBIST Status Low Register (STCU2_MBSLSW)	MBSSW0 - MBSSW31 are asserted after a successful MBIST on their respective partitions
On-line MBIST Status Medium Register (STCU2_MBSMSW)	MBSSW32 - MBSSW63 are asserted after a successful MBIST on their respective partitions
On-line MBIST Status High Register (STCU2_MBSHSW)	MBSSW64 is asserted after a successful MBIST on this partition

If BIST is not successful, check the STCU2 Error Register (STCU2_ERR_STAT) to determine if an error occurred and details of the error.. If the STCU2_ERR_STAT register does not show any errors, it is possible that the BIST did not run, and the STCU configuration registers should be checked for accuracy.

6 Conclusion

After going through each section in this application note, the user should be able to configure both an off-line and on-line BIST and understand why the STCU is configured with the values presented. For STCU configurations other than those presented in this document, please see the MPC5777C reference manual for details on register bit-fields not explained in the STCU Registers chapter.

Appendix A BIST Execution Time Spreadsheet

Attached File: MPC5777C_BIST_Execution_Time_protected_v2.0.xlsx

Appendix B DCF Addresses and LTB Commands Spreadsheet

Attached File: MPC5777C_DCF_Addresses_and_LTB_commands_Example.xlsx

Appendix C Off-line Script Example

NOTE

For mask set 2N45H there exists errata *ERR009877*. Please review the errata as it may affect the example script below for 2N45H.

Attached File: MPC5777C_off-line_40MHzXOSC_PLL50MHz_2N45H.cmm

Attached File: MPC5777C_off-line_40MHzXOSC_PLL50MHz_3N45H.cmm

NOTE

The script below only runs MBSIT off-line at a PLL of 200 MHz and is provided for those who are willing to trade off speed vs. test coverage during the reset sequence. It will work for either 2N45H or 3N45H and is the only other NXP validated off-line BIST configuration. The user needs to evaluate their system design to insure the core regulator can handle the current load change during the start and end of MBIST.

Attached File: MPC5777C_off-line_40MHzXOSC_PLL200MHz_MBIST.cmm

Appendix D On-line Code Example

Attached File: MPC5777C_on-line_bist_2N45H.c

Attached File: MPC5777C_on-line_bist_3N45H.c

NOTE

The files below only run LBSIT on-line and are for using with a PLL of 200 MHz and is the only other NXP validated on-line BIST configuration. The user needs to evaluate their system design to insure the core regulator can handle the current change during the start and end of LBIST.

Attached File: MPC5777C_on-line_LBIST_2N45H.c

Attached File: MPC5777C_on-line_LBIST_3N45H.c

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support