Renesas Synergy™ Platform

S3A7 MCU Safety Application Note

r11an0191eu0100
Rev.1.00
June 29, 2017

Introduction

IEC 61508 is an international standard governing a range of electrical, electromechanical, and electronic safety related systems. It defines the requirements needed to ensure that systems are designed, implemented, operated, and maintained at the required Safety Integrity Level (SIL). Four SIL levels have been defined to indicate the risks involved in any particular system, with SIL4 being the highest risk level.

At the heart of the majority of safety related systems nowadays is a sophisticated and often highly integrated Microcontroller (MCU). An integral part of meeting the requirements of IEC61508 is the ability to verify the correct operation of critical areas of the MCU.

The purpose of this document is to provide guidelines to use the Renesas S3A7 Series MCUs within a safety context.

Target Device

S3A7 MCUs

Contents

1. Common Terminology

This section defines some common terms and acronyms used throughout the rest of the document and provides references to other Renesas complementary documents.

1.1 Acronyms

Table 1.1 Terminology and acronyms

Acronym	Description
ADC	Analog to Digital Converter
BUSERR	Bus Error
CAN	Control Area Network
CRC	Cyclic Redundancy Check
CM	Compare Match
DAC	Digital to Analog Converter
DRW	Drawing Engine
E2E	End To End
ECC	Error Correction Code
EOC	End of Conversion
EOT	End of Transfer
GPT	General PWM Timer
HOCO	High speed On Chip Oscillator
IC	Input Capture
IrDA	Infrared Data Association
ISR	Interrupt Service Routine
IWDT	Independent Watchdog Timer
KINT	Key Interrupt Function
LOCO	Low-speed On-chip Oscillator
MMC	Multi Media Card
MOCO	Middle-speed On-chip Oscillator
MPU	Memory Protection Unit
NSR	Non Safety Relevant
OS	Output Short
PST	Process Safety Time
PTP	Precision Time Protocol
PWM	Pulse Width Modulation
Q&A	Question and Answer
QSCI	Quad Serial Communication Interface
RAM	Random Access Memory
SDHI	Secure Digital Host Interface (SDHI)
SCI	Serial Communication Interface
SIL	Safety Integrity Level
SM	Safety Mechanism
SPI	Serial Peripheral Interface
SR	Safety Relevant
SRAM	Static Random Access Memory
SRC	Sampling Rate Converter
SSI	Serial Sound Interface
USB	Universal Serial Bus
VM12	Voltage Monitoring 1/2
WDT	Watch Dog Timer
WUNI	Watch dog Underflow Non-maskable Interrupt

1.2 Document References

[1] User's Manual: Microcontrollers Renesas Synergy™ Family/S3 Series, Rev.1.2.

[2] ARM® Cortex®-M4 Devices Generic User Guide, revision 16 December 2010.

[3] S3 Series MCU Diagnostic Software User Guide, ID = r11an0188eu0100.

[4] Diagnostic Software Safety Manual, ID= r11um0066eu0100.

2. Considerations of External Protections

Table 2.1 describes recommendations related to elements external to the MCU.

Table 2.1 Considerations for external protection

ID	Description	Limitation/ Comment
Ext_1	It is recommended to use an external Watch Dog Timer (WDT). See section 2.1 for more information.	None
Ext_2	The user shall follow the Electrical Characteristics specified in section 51 of [1].	None
Ext_3	It is recommended to use external measures to protect the power supply to the MCU pins. See section 51 of [1] for a safe power supply range. These measures shall be used in conjunction with the internal LVD_Monitoring mechanism.	None
Ext_4	When using the ADC (ADC140 module), it is recommended to externally monitor the conversion reference voltage applied to pins VREFH0 and VREFL0.	None

2.1 External Watchdog

When designing the monitoring provided by an external Watch Dog, consider the following recommendations:

- Allow refresh only if all monitored tasks are executed, and in the proper order
- Allow refresh only if the expected time window for each task is followed
- Implement a Q&A-based exchange of messages before the refresh can take place (if supported by the external component).

Note: Two internal Watch Dog Timers (WDTs) – IWDT and WDT, are provided as MCU internal modules. If the internal WDTs are in use, the design of the external Watch Dog Timer can be simplified because the software (SW) control flow monitoring is also performed by the internal module(s).

Note: When complementing the external WDT with the internal one, consider that both IWDT and WDT are not able to send an external error notification when the MCU status cannot be recovered through a SW routine (IRQ) or a reset procedure. This has to be considered during system design. The external WDT can effectively compensate for this issue.

3. Hardware (HW) Module Recommended Settings/ Usage

3.1 Processing Module

3.1.1 CPU

This module is the core used to run the safety application.

(1) **Recommended settings**

Table 3.1 describes the recommended settings for the CPU module.

Table 3.1 Recommended settings/usage for the CPU module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.2 describes the safety mechanism to protect the safety application from permanent failure of the CPU.

Table 3.2 Recommended protection from permanent failure of CPU

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/ Comment
N/A	Medium	CPU_SW_Test	Run-time	To be used in combination with External WDT, see section 2.1 for more details.

(3) **Consideration for transient faults**

It is recommended to protect the CPU from transient failures. In order to do this, consider the following information.

When protection CPU_prot_1 is used:

a. Coverage is decreased to a low level since the effectiveness of CPU_SW_Test is affected by the test frequency.
b. To improve the coverage level, adoption of application level mitigations is suggested for multiple operation processing, where elaboration of tasks is duplicated and the results of the redundant elaborations are compared to look for mismatches that indicate the presence of faults.

(4) **Support of system level functional redundancy**

None.

3.2 Data I/O Modules

3.2.1 I/O Ports

(1) **Recommended settings**

Table 3.3 to Table 3.5 describe the recommended settings for the I/O Ports module.

Table 3.3 Recommended settings/usage for the I/O Ports module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.4 Recommended settings/usage for interference minimization of the I/O Ports module

ID	Description	Limitation/ Comment
IOPorts_sett_I1	Follow recommendation for unused pins given in section 20.4 of [1]	N/A

Table 3.5 Recommended settings/usage for protection against systematic System/SW faults

ID	Description	Limitation/ Comment
IOPorts_recc_S1	If pins belonging to the same General Purpose IO group are used for both SR and NSR functions, then a suitable SW handling is required to avoid compromising the safety functions	N/A

(2) **Recommended protection against permanent faults**

Table 3.6 describes the safety mechanisms to protect the safety application from permanent failures of the I/O Ports.

Note: The coverage information reported is only valid for General Purpose I/O. For other usage purposes, refer to the module specific sections.

Table 3.6 Recommended protection from permanent failure of I/O Ports

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
IOPorts_prot_1	Medium	Config_reg_read_back, GPIO_out_redundancy, GPIO_in_redundancy	Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the I/O Ports from transient failures. In order to do this, see the following information.

When protection IOPorts_prot_1 is used:

a. Overall coverage is still kept High because the GPIO_out_redundancy and GPIO_in_redundancy mechanisms are always active and also effective for transient faults.

(4) **Support for system level functional redundancy**

Considerations are already provided in section 3.2.1.2. In particular, see GPIO_in_redundancy.

Note: When using GPIO to achieve redundancy, special considerations need to be taken to mitigate possible dependencies. In particular, the following measures are recommended:

- Use pins belonging to different port groups
- Use pins that are not adjacent on the package
- Use pins whose data is located in different bits of the data bus.

3.2.2 14-Bit A/D Converter (ADC14)

This module can be used by the safety application when acquisition of analogue signals is necessary.

(1) **Recommended settings**

Table 3.7 to Table 3.9 describe the recommended settings for the 14-Bit ADC.

Table 3.7 Recommended settings/usage for the ADC14 module

ID	Description	Limitation/ Comment
ADC14_sett_U1	Enable the automatic clearing of A/D registers (ADDRy, ADRD, ADDBLDR, ADDBLDRA, ADDBLDRB, ADTSDR, or ADOCDR) after any of these registers is read by the CPU, DTC, or DMAC. ADCER.ACE = 1b.	N/A

Table 3.8 Recommended settings/usage for interference minimization for the ADC14 module

ID	Description	Limitation/ Comment
ADC14_sett_I1	Enable Module-Stop state, when the peripheral is not used by setting the following: MSTPCRD.MSTPD16 = 1b (ADC140).	N/A

Table 3.9 Recommended settings/usage for protection against systematic System/SW faults for the ADC14 module

ID	Description	Limitation/ Comment
ADC14_recc_S1	If at least one ADC channel is used for NSR functions while the other channels are used for SR functions, then a suitable SW handling is required to avoid compromising the safety functions	N/A
ADC14_recc_S2	The A/D conversion cannot be selected as ACMPHS input for targets listed in section 39.8.13 of [1] during A/D conversion because these pins are multiplexed with the A/D converter and ACMPHS	N/A

(2) **Recommended protection against permanent faults**

Table 3.10 describes the safety mechanisms to protect the safety application from permanent failures of the 14-Bit A/D Converter module.

Table 3.10 Recommended protection from permanent failure of ADC14

ID	Estimated Coverage Length	Suggested Safety Mechanism	Suggested Protection Type	Limitation/ Comment
ADC14_prot_1a	Medium	**Error! Reference source not found.**ADC14_Comparator_SW, Config_reg_read_back	Run-time	If DMA is used in connection with the ADC, then the self-test provided by ADC14_TriggerDMA shall be adopted, as described in ADC14_prot_1b. When using**Error! Reference source not found.**, take care to mitigate dependency issues between the sources of information.
ADC14_prot_1b	Medium	**Error! Reference source not found.**ADC14_Comparator_SW, ADC14_TriggerDMA, Config_reg_read_back	Self-test	When using **Error! Reference source not found.**, take care to mitigate dependency issues between the sources of information

(3) **Consideration for transient faults**

It is recommended to protect the ADC14 from transient failures. In order to do this, see the following information.

When protection ADC14_prot_1a is used:

a. Overall coverage can be kept at a medium level despite the negative impact of the test frequency on ADC14_Comparator_SW and Config_reg_read_back.

(4) **Support of system level functional redundancy**

The MCU allows the possibility of using multiple channels belonging to the same unit to support system level redundancy. Usage of ADC14_Comparator_SW is highly recommended in the latter case to mitigate dependency issues related to the common analogue comparator.

3.2.3 12-Bit D/A Converter (DAC12)

This module can be used by the safety application when Digital to Analogue conversion is necessary.

(1) **Recommended settings**

Table 3.11 to Table 3.13 describe the recommended settings for the D/A Converter module.

Table 3.11 Recommended settings/usage for the DAC12 module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.12 Recommended settings/usage for interference minimization for the DAC12 module

ID	Description	Limitation/ Comment
DAC12_sett_I1	As a measure against interference between D/A and A/D conversion, write DAADSCR.DAADST = 1b while ADCSR.ADST = 0b. See section 40.3.1 of [1] for more details.	N/A
DAC12_sett_I2	Enable Module-Stop State when the peripheral is not used, by setting MSTPCRD.MSTPD20 =1b	N/A

Table 3.13 Recommended settings/usage for protection against systematic System/SW faults for the DAC12 module

ID	Description	Limitation/ Comment
DAC12_recc_S1	If one channel (for example, DA0) is used for SR functions while the other (for example, DA1) is used for NSR functions, then, suitable SW handling is required to avoid compromising the safety functions	N/A

(2) **Recommended protection against permanent faults**

Table 3.14 describes the safety mechanisms to protect the safety application from permanent failures of the D/A Converter module.

Table 3.14 Recommended protection from permanent failure of DAC12

ID	Estimated Coverage Length	Suggested Safety Mechanism	Suggested Protection Type	Limitation/ Comment
DAC12 _prot_1	High	DAC12_Loop_ADC14, Config_reg_read_back	Run-time	Detection of an analog signal not updated in sync with a conversion start is partially covered because of the limitation in performing high frequency monitoring of the analog input (for example, due to conversion latency, SW elaboration delays)
DAC12_prot_2	High	DAC12_system_redundancy	Run-time	Analog signal comparison and feedback to MCU as part of fault detection can be difficult to implement and is costly in terms of system design. Fault detection shall be implemented external to the MCU.

(3) **Consideration for transient faults**

It is recommended to protect the DAC12 from transient failures. In order to do this, see the following information.

When protection DAC12_prot_1 is used:

a. Overall coverage is decreased to medium because DAC12_Loop_ADC14 has a limitation in performing high frequency monitoring of the analog input (for example, due to conversion latency, SW elaboration delays). This can be mitigated by using multiple samples.

When protection DAC_prot_2 is used:

a. Overall coverage can be kept high assuming high frequency comparison of the outputs of the DA channels.

(4) **Support of system level functional redundancy**

The possibility of using two output channels DA0 and DA1 to support system level redundancy (DACR.DAE = 1b, DACR.DAOE0 = 1b, DACR.DAOE1 = 1b) has already been proposed in section 3.2.3.2.

In relation to dependencies, consider that the two channels share the same converter.

As a mitigation, consider using DAC12_Loop_ADC14 to monitor the correct behavior of the converter.

3.2.4 Temperature Sensor (TSN)

This module can be used by the safety application to get information about the MCU temperature.

(1) **Recommended settings**

Table 3.15 to Table 3.17 describe the recommended settings for the Temperature Sensor module.

Table 3.15 Recommended settings/usage for the TSN module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.16 Recommended settings/usage for interference minimization for the TSN module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.17 Recommended settings/usage for protection against systematic System/SW faults for the TSN module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.18 describes the safety mechanisms to protect the safety application from permanent failures of the Temperature Sensor module.

Table 3.18 Recommended protection from permanent failure of TSN

ID	Estimated coverage level	Suggested safety mechanism	Suggested Protection Type	Limitation/comment
TSN_prot_1	Low	ADC14_TempSens	Self-test or Run-time	The mechanism can only check if the temperature reading is within a plausible range defined by the application in use. Note: If the MCU is operated after being exposed to temperature outside of the allowed range, this safety mechanism may provide insufficient protection.
TSN_prot_2	Medium	Ext_TempSens	Self-test or Run-time	The use of an external sensor is required. For this solution, system level impact has to be judged (especially if used as run-time test).

(3) **Consideration for transient faults**

It is recommended to protect the Temperature Sensor module from transient failures. In order to do this, see the following information.

When protection TSN_prot_1 is used:

a. Overall coverage is kept at a low level.
b. The test frequency negatively impacts the effectiveness of ADC14_TempSens but a low level of coverage can still be claimed. Usage of multiple samples can improve the coverage back to medium.

When protection TSN_prot_2 is used:

a. Overall coverage is kept at a low level.
b. The test frequency negatively impacts the effectiveness of Ext_TempSens but a low level of coverage can still be claimed. Usage of multiple samples can improve the coverage back to medium.

(4) **Support of system level functional redundancy**

None

3.2.5 High-Speed Analog Comparator (ACMPHS)

This module can be used by the safety application to compare a test voltage with a reference voltage, and to provide a digital output based on the result of conversion.

(1) **Recommended settings**

Table 3.19 to Table 3.21 describe the recommended settings for the High-Speed Analog Comparator module.

Table 3.19 Recommended settings/usage for the ACMPHS module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.20 Recommended settings/usage for interference minimization for the ACMPHS module

ID	Description	Limitation/ Comment
ACMPHS_sett_I1	Enable Module Stop State when the peripheral is not used by setting MSTPCRD.MSTPD30 = 1b (ACMPHS1), MSTPCRD.MSTPD28 = 1b (ACMPHS0).	N/A

Table 3.21 Recommended settings/usage for protection against systematic System/SW faults for the ACMPHS module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.22 describes the safety mechanisms to protect the safety application from permanent failures of the High-Speed Analog Comparator module.

Table 3.22 Recommended protection from permanent failure of ACMPHS

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
ACMPHS_prot_1	Medium	ACMPHS_triggerISR, Config_reg_read_back	Self-test or Run-time	This test requires the utilization of both DAC channels

(3) **Consideration for transient faults**

It is recommended to protect the ACMPHS module from transient failures. In order to do this, see the following information.

When protection ACMPHS_prot_1 is used:

a. Overall coverage is kept a medium level:
   Transient faults of the analog part of the peripheral are not expected to impact the main function. However, faults of the digital processing functionality could have an impact, but in a delimited time window (for example, one clock cycle). As for configuration, the Config_reg_read_back is effective. A medium level of coverage can still be claimed.

(4) **Support of system level functional redundancy**

ACMPHS provides two comparators that can be used redundantly, but only one output pin VCOUT is provided.

3.2.6 Operational Amplifier (OPAMP)

This module can be used to amplify small analog input voltages and output the amplified voltages.

This module is classified as non-safety relevant and only the information related to interference minimization is recommended and shown in Table 3.23.

Table 3.23 Recommended settings/usage for interference minimization for the ACMPHS module

ID	Description	Limitation/ Comment
OPAMP_sett_I1	Enable Module Stop State when the peripheral is not used by setting MSTPCRD.MSTPD31 = 1b.	N/A

3.2.7 Low-Power Analog Comparator (ACMPLP)

This module can be used by the safety application to compare a reference input voltage and an analog input voltage, and to provide a digital output based on the result of conversion.

(1) **Recommended settings**

Table 3.19 to Table 3.21 describe the recommended settings for the Low Power Analog Comparator module.

Table 3.24 Recommended settings/usage for the ACMPLP module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.25 Recommended settings/usage for interference minimization for the ACMPLP module

ID	Description	Limitation/ Comment
ACMPLP_sett_I1	Enable Module Stop State when the peripheral is not used by setting MSTPCRD.MSTPD29 = 1b	N/A

Table 3.26 Recommended settings/usage for protection against systematic System/SW faults for the ACMPLP module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.22 describes the safety mechanisms to protect the safety application from permanent failures of the Low Power Analog Comparator module.

Table 3.27 Recommended protection from permanent failure of ACMPLP

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
ACMPLP_prot_1	Medium	ACMPLP_triggerISR ACMPLP_win_triggerISR, Config_reg_read_back	Self-test or Run-time	This tests require the utilization of external hardware

(3) **Consideration for transient faults**

It is recommended to protect the ACMPLP module from transient failures. In order to do this, see the following information.

When protection ACMPLP_prot_1 is used:

b. Overall coverage is kept a medium level:
Transient faults of the analog part of the peripheral are not expected to impact the main function. However, faults of the digital processing functionality could have an impact, but in a delimited time window (for

example, one clock cycle). As for configuration, the Config_reg_read_back is effective. A medium level of coverage can still be claimed.

(4) **Support of system level functional redundancy**

ACMPLP provides two comparators that can be used redundantly, but only one output pin VCOUT is provided.

3.3 Protection against HW faults

These modules are primarily intended for MCU protection against HW faults.

3.3.1 Low Voltage Detection Circuit (LVD)

This module can be used by the safety application to detect under voltage conditions in the power supply.

(1) **Recommended settings**

Table 3.28 to Table 3.30 describe the recommended settings for the Low Voltage Detection Circuit module.

Table 3.28 Recommended settings/usage for the LVD module

ID	Description	Limitation/ Comment
	N/A[1]	

Table 3.29 Recommended settings/usage for interference minimization for the LVD module

ID	Description	Limitation/ Comment
	N/A	

Table 3.30 Recommended settings/usage for protection against systematic System/SW faults for the LVD module

ID	Description	Limitation/ Comment
	N/A	

(2) **Recommended protection against permanent faults**

Table 3.31 describes the safety mechanisms to protect the safety application from permanent failures of the Low Voltage Detection Circuit module.

Table 3.31 Recommended protection from permanent failure of LVD

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
LVD_prot_1	Medium	LVD_check	Self-test	This mechanism requires external HW acting on the power supply. Impact of run-time failures of such HW has to be considered during system level analysis.
LVD_prot_2	Medium	Config_reg_read_back	Run-time	Config_reg_read_back offers a medium level of coverage on configuration registers at run-time. Note: LVD_Monitoring[2] itself provides additional run-time protection.

(3) **Support of system level functional redundancy**

None.

---

[1] A description of the LVD module as a safety mechanism is given in section 4 for LVD_Monitoring mechanisms.

[2] The **Error! Reference source not found.** mechanism can provide auto-diagnosis on the LVD module itself.

3.3.2 Clock Frequency Accuracy Measurement Circuit (CAC)

This module can be used by the safety application to detect failures in clock frequency generation and then to protect the MCU.

(1) **Recommended settings**

Table 3.32 to Table 3.34 describe the recommended settings for the Clock Frequency Accuracy Measurement Circuit module.

Table 3.32 Recommended settings/usage for the CAC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.33 Recommended settings/usage for interference minimization for the CAC module

ID	Description	Limitation/ Comment
CAC_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRC.MSTPC0 =1b	N/A

Table 3.34 Recommended settings/usage for protection against systematic System/SW faults for the CAC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.35 describes the safety mechanisms to protect the safety application from permanent failures of the Clock Frequency Accuracy Measurement Circuit.

Table 3.35 Recommended protection from permanent failure of CAC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
CAC_prot_1	Medium	CAC_self_test, Config_reg_read_back	Self-test	N/A
CAC_prot_2	Medium	Config_reg_read_back	Run-time	Config_reg_read_back offers a medium level of coverage on configuration registers at run-time

(3) **Support of system level functional redundancy**

None.

3.3.3 Port Output Enable Module for GPT (POEG)

This module can be used by the safety application to detect failures in the delivery of PWM overlapping phases.

(1) **Recommended settings**

Table 3.36 to Table 3.38 describe the recommended settings for the Port Output Enable for the GPT module.

Table 3.36 Recommended settings/usage for the POEG module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.37 Recommended settings/usage for interference minimization for the POEG module

ID	Description	Limitation/ Comment
POEG_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRD.MSTPD14 = 1b	N/A

Table 3.38 Recommended settings/usage for protection against systematic System/SW faults for the POEG module

ID	Description	Limitation/ Comment
POEG_sett_S1	If compatible with application requirements, the noise cancellation filter (POEGGn.NFEN = 1 (n = A to D)) shall be enabled	N/A

(2) **Recommended protection against permanent faults**

Table 3.39 describes the safety mechanisms to protect the safety application from permanent failures of the Port Output Enable for the GPT module.

Table 3.39 Recommended protection from permanent failure of POEG

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
POEG_prot_1	Medium	POEG_ExtLoop, Config_reg_read_back	Self-test or Run-time	It is not possible to force (and then test) the high impedance control in the case of a phase-overlap of the complementary PWM outputs or oscillation stop condition. Run-time failures of this module will affect the PWM outputs of the GPT module and they will be covered by the loop-back mechanisms recommended to protect the timer. Run-time test is possible if compatible with application requirements.
POEG_prot_2	Medium	Config_reg_read_back	Run-time	Config_reg_read_back offers a medium level of coverage on configuration registers at run-time
POEG_prot_3	Medium	POEG_ExtLoop	Self-test	This mechanism implies the utilization of external HW

(3) **Support of system level functional redundancy**

None.

3.3.4 Independent Watchdog Timer (IWDT)

This module can be used by the safety application to detect the SW program running out of control.

(1) **Recommended settings**

Table 3.40 to Table 3.42 describe the recommended settings for the Independent Watchdog Timer module.

Table 3.40 Recommended settings/usage for the IWDT module

ID	Description	Limitation/ Comment
IWDT_sett_U1	The IWDT shall be used in auto-start mode (OFS0.IWDTSTRT= 0) to avoid the possibility that WDT protection (from both HW and SW faults) is not present after a first reset is generated	N/A
IWDT_sett_U2	If compatible with application requirements, "reset signal generation" shall be selected as an action for counter underflow (OFS0.IWDTRSTIRQS=1). Alternatively, interrupt request can be activated in order to perform specific recovery action in the ISR.	In case of the interrupt routine, when RAM and stack memory status are not guaranteed, limit the operations which rely on variable values
IWDT_sett_U3	If compatible with application requirements, set the refresh-permitted period to 25% (see OFS0 register description) of the count period in order to improve detection capabilities	This configuration limits the design of SW architecture and SW itself, in order to meet the refresh period
IWDT_sett_U4	A monitoring strategy shall be implemented to ensure that all tasks are executed in a predefined order (for example, through a shared monitoring variable). The WDT refresh strategy should allow for refresh only if the task monitoring shows that all the tasks have been correctly executed.	N/A
IWDT_sett_U5	In order to avoid that a fault on the debug logic can stop the IWDT, IWDTDBGSTOPCR.DBGSTOP_IWDT = 0b	N/A

Table 3.41 Recommended settings/usage for interference minimization for the IWDT module

ID	Description	Limitation/ Comment
IWDT_sett_I1	When using the IWDT peripheral, use the Renesas IWDT Management SW [3] to enable IWDT NMI (through the IWDT_Init function), refresh IWDT counter (that is, through the IWDT_Kick function), and check whether the IWDT timed out (through the IWDT_DidReset function)	N/A

Table 3.42 Recommended settings/usage for protection against systematic System/SW faults for the IWDT module

ID	Description	Limitation/ Comment
N/A	N/A[3]	N/A

[3] The usage of the IWDT is already a measure to protect against SW faults.

(2) **Recommended protection against permanent faults**

Table 3.43 describes the safety mechanisms to protect the safety application from permanent failures of the Independent Watchdog Timer (IWDT) module.

Table 3.43 Recommended protection from permanent failure of IWDT

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
IWDT_prot_1	Medium	IWDT_Expire, Config_reg_read_back	Self-test	N/A
IWDT_prot_2	Medium	Config_reg_read_back	Run-time	Config_reg_read_back offers a medium level of coverage on configuration registers at run-time.

(3) **Support of system level functional redundancy**

None.

3.3.5 Battery Backup Function

This module is used by the safety application to partially support powering the MCU when the main power supply fails.

(1) **Recommended settings**

Table 3.44 to Table 3.46 describe the recommended settings for the Battery Backup Function module.

Table 3.44 Recommended settings/usage for the Battery Backup module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.45 Recommended settings/usage for interference minimization for the Battery Backup module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.46 Recommended settings/usage for protection against systematic System/SW faults for the Battery Backup module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.47 describes the safety mechanisms to protect the safety application from permanent failure of the Battery Backup Function module.

Table 3.47 Recommended protection from permanent failure of Battery Backup module

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
BattB_prot_1	Medium	Write_verify	Run-time	N/A
BattB_prot_2	Medium	BatteryBckp_check	Self-test	N/A

(3) **Support of system level functional redundancy**

None.

3.3.6 Parity check

Parity check is performed to detect potential issues in the parity logic, protecting SRAM0 (addresses from 20004000h to 2001FFFFh) and SRAM1 (see section 47.1 of [1]).

(1) **Recommended settings**

Table 3.48 to Table 3.50 describe the recommended settings for the Parity Check.

Table 3.48 Recommended settings/usage for the Parity Check

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.49 Recommended settings/usage for interference minimization for the Parity Check

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.50 Recommended settings/usage for protection against systematic System/SW faults for the Parity Check

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

The Parity Check function cannot be tested. So, any fault affecting this function cannot be detected. The user must take care about this constraint.

(3) **Support of system level functional redundancy**

None.

3.3.7 ECC

ECC (Error Correction Code) is performed to detect both single bit and double bit error in SRAM0 (ECC area) (addresses from 2000 0000h to 2000 3FFFh) (see [1]).

(1) **Recommended settings**

Table 3.51 to Table 3.53 describe the recommended settings for the ECC.

Table 3.51 Recommended settings/usage for the ECC

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.52 Recommended settings/usage for interference minimization for the ECC

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.53 Recommended settings/usage for protection against systematic System/SW faults for the ECC

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

The ECC function can be tested following the strategy described in Section 47.3.4 of [1].

Table 3.54 Recommended protection from permanent failure of ECC module

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
ECC_prot_1	High	ECC Decoder Testing (see Section 47.3.4 of [1])	Self-test or Run-time	N/A

(3) **Support of system level functional redundancy**

None.

3.4 Protections against SW faults

These modules are primarily intended for MCU protection against SW faults.

3.4.1 Memory Protection Unit (MPU)

This module can be used by the safety application to detect memory access violations.

(1) **Recommended settings**

Table 3.55 to Table 3.57 describe the recommended settings for the Memory Protection Unit module.

This MCU incorporates a CPU stack pointer monitor that detects underflows and overflows of the two stack pointers, Main Stack Pointer (MSP) and Process Stack Pointer (PSP). The CPU stack pointer monitor is enabled by setting the Main Stack Pointer monitor enable bit or the Process Stack Pointer monitor enable bit in the Access Control Register (MSPMPUCTL, PSPMPUCTL), to 1.

Moreover, the MCU incorporates a bus master MPU that monitors the addresses of the bus master access to the overall address space (0000 0000h to FFFF FFFFh). If access to the protected region is detected, the bus master MPU generates a reset or a non-maskable interrupt. The supported access control information for the individual regions consists of permissions to read and write.

This MCU incorporates a bus slave MPU that checks access to the bus slave function such as flash or SRAM. The bus slave function can be accessed from four bus masters (CPU, bus master MPU group A, bus master MPU group B, and bus master MPU group C). The bus slave MPU has a separate protection register for each of the four bus masters and can protect access individually.

The supported access control information for the individual regions consists of permissions to read and write.

Enable the protection (reading and writing) of the Access Control Register for the memory bus by properly setting the registers as detailed in [1], section 16.5.

Table 3.55 Recommended settings/usage for the MPU module

ID	Description	Limitation/ Comment
MPU_recc_U1	Enable the Main Stack Pointer monitor by setting MSPMPUCTL.ENABLE = 1	When the MSPMPUCTL.ENABLE bit is set to 1, the available registers are MSPMPUSA, MSPMPUEA, and MSPMPUOAD
MPU_recc_U2	Enable the Process Stack Pointer monitor by setting PSPMPUCTL.ENABLE = 1	When the PSPMPUCTL.ENABLE bit is set to 1, the available registers are PSPMPUSA, PSPMPUEA, and PSPMPUOAD
MPU_recc_U3	Protect Stack Pointer Monitor registers (MSPMPUCTL, MSPMPUSA, and MSPMPUEA) from writing, while allowing reading, by setting MSPMPUPT.PROTECT = 1	
MPU_recc_U4	Protect Stack Pointer Monitor registers (PSPMPUCTL, PSPMPUSA, and PSPMPUEA) from writing, while allowing reading, by setting PSPMPUT.PROTECT = 1	
MPU_recc_U5	Enable the control of access permissions or protection by setting MMPUACAn.ENABLE = 1 (*n* = 0 to 15), where *n* is the Access Control Register	To have MMPUACAn available, the bus master MPU control register shall be enabled by setting MMPUCTLA.ENABLE = 1 (more register details are in [1], section 16.4.1)
MPU_recc_U6	Enabling the Read and/or Write Protection of group *A* region *n* unit, by setting MMPUACAn.RP = 1 MMPUACAn.WP = 1	This is possible only when MMPUACAn.ENABLE = 1 (more registers details are in [1], section 16.4.1)

Table 3.56 Recommended settings/usage for interference minimization for the MPU module

ID	Description	Limitation/ Comment

MPU_recc_I1	If functions of different SIL levels are present and/or NSR functions are present, then the highest SIL level shall be assigned for the MPU configuration/usage	N/A

Table 3.57 Recommended settings/usage for protection against systematic System/SW faults for the MPU module

ID	Description	Limitation/ Comment
MPU_sett_S1	The MPU can be used in a safety application as a means to partially support independence between two SW implemented functions or two groups of functions. This can be achieved by running one function (one group of functions) in unprivileged mode and the other function (the other group of functions) in privileged mode. The following restrictions can then be adopted: Limit the access to data reserved for tasks with privileged mode privileges (permission to read). Do not overwrite data reserved for tasks with privileged mode privileges (permission to write). Do not execute functions (for example, libraries) reserved for tasks with privileged mode privileges (permission to execute).	The following limitations apply: Function(s) with unprivileged mode privileges will inherit the permissions granted for such mode (for example, it is not possible to change the processor Interrupt Priority Level setting). The behavior is asymmetric, that is, the protection works in one direction only to limit the interference between the tasks with unprivileged mode privileges and those with privileged mode privileges. On the other hand, this is useful when different integrity levels are allocated to different functions with the higher integrity level functions executed in privileged mode. Note that the MPU cannot protect from inadvertent accesses by bus masters which are not the CPU (for example, DMA accesses).

(2) **Recommended protection against permanent faults**

Table 3.58 describes the safety mechanisms to protect the safety application from permanent failures of the Memory Protection Unit module.

Table 3.58 Recommended protection from permanent failure of MPU

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
MPU_prot_1	Medium	MPU_region_check, MSPMPU_monitor, PSPMPU_monitor	Self-test	MPU_region_check: A faulty MPU can block valid accesses during run-time, preventing the normal SW execution. In such cases, the external WDT is effective and provides additional protection that has been considered in the coverage estimation. MSPMPU_monitor and PSPMPU_monitor: A faulty stack pointer monitor can prevent the normal SW execution. In such cases, the external WDT is effective and provides additional protection that has been considered in the coverage estimation.
MPU_prot_2	Medium	Config_reg_read_back	Run-time	Config_reg_read_back offers a medium level of coverage on configuration registers at run-time

(3) **Support of system level functional redundancy**

None.

3.4.2 Watchdog Timer (WDT)

This module can be used by the safety application to detect the SW program running out of control.

(1) **Recommended settings**

Table 3.59 to Table 3.61 describe the recommended settings for the Watchdog Timer module.

Table 3.59 Recommended settings/usage for the WDT module

ID	Description	Limitation/ Comment
WDT_sett_U1	The WDT shall be used in auto-start mode (OFS0.WDTSRT=0) to avoid the possibility that the protection of the WDT (due to both HW and SW faults) is not present after a terminal reset	N/A
WDT_sett_U2	If compatible with application requirements, it is recommended to select reset signal generation as an action for counter underflow (OFS0.WDTRSTIRQS=1). Alternatively, interrupt request can be activated in order to perform specific recovery action within the ISR.	In the case of an interrupt routine where RAM and stack memory operation are not guaranteed, limit the operations which rely on variables
WDT_sett_U3	If compatible with application requirements, set the refresh-permitted period to 25% (see OFS0 and WDTCR registers) of the count period in order to improve detection capabilities	This configuration limits the design of SW architecture and SW itself in order to meet the refresh period
WDT_sett_U4	The WDT is stopped when a low power consumption mode is entered. If entering low power is an application requirement, use the IWDT module instead.	N/A
WDT_sett_U5	A monitoring strategy shall be implemented to see if all tasks are executed in the predefined order (for example, through a shared monitoring variable). The WDT refresh strategy should enable the refresh only if the task monitoring shows that all the tasks have been correctly executed.	N/A
WDT_sett_U5	In order to avoid a fault on the debug logic from stopping the WDT, DBGSTOPCR.DBGSTOP_WDT = 0b	N/A

Table 3.60 Recommended settings/usage for interference minimization for the WDT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.61 Recommended settings/usage for protection against systematic System/SW faults for the WDT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A[4]

[4] Please note that the usage of the WDTA is already a measure to protect against SW faults.

(2) **Recommended protection against permanent faults**

Table 3.62 describes the safety mechanisms to protect the safety application from permanent failures of the Watchdog Timer module.

Table 3.62 Recommended protection from permanent failure of WDT

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
WDT_prot_1	Medium	WDT_Counter_Monitoring, WDT_Expire, Config_reg_read_back	Self-test	N/A
WDT_prot_2	Medium	Config_reg_read_back	Run-time	Config_reg_read_back offers a medium level of coverage on configuration registers at run-time

(3) **Support of system level functional redundancy**

None.

3.5 Communication Module(s)

3.5.1 Buses

This module is used by the safety application to support communication between internal modules.

(1) **Recommended settings**

Table 3.63 to Table 3.65 describe the recommended settings for the Buses module.

Table 3.63 Recommended settings/usage for the BSC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.64 Recommended settings/usage for interference minimization for the BSC module

ID	Description	Limitation/ Comment
BSC_settl_1	If the MMF is disabled (see section 3.9.3) and the CPU accesses memory addresses from 0200 0000h to 027F FFFFh, the bus module can raise an access error	N/A

Table 3.65 Recommended settings/usage for protection against systematic System/SW faults for the BSC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.66 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Buses.

Table 3.66 Recommended protection from permanent failure of BSC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
BSC_prot_1	Medium	ExtWDT (see section 2.1), BSC_Monitoring, Config_reg_read_back	Run-time	Application level measures reported in other sections of the document, as the E2E mechanisms, also contribute to the protection of this module. These are considered in the estimation of the coverage. It is not possible to perform self-testing on the BSC_Monitoring mechanism. Its usage is only intended as a complementary mitigation measure.

(3) **Consideration for transient faults**

It is recommended to protect the BSC from transient failures. In order to do this, see the following information.

When protection BSC_prot_1 is used:

a. Overall coverage can be kept at medium level despite the decreased effectiveness of Config_reg_read_back due to the frequency limitation of the read back operation.
b. Other mechanisms remain effective, as the external WDT for which the following considerations hold:

— Instruction(s) corrupted by a transient fault may result in the program flow running out of control and this will be detected by the WDT
— Instruction(s) missed due to a transient fault may result in the program flow running out of control and this will be detected by the WDT
— The case of the operand bus not being accessible due to a transient fault may result in the program flow running out of control and this will be detected by the WDT
— Data corruption due to a transient fault may result in the program flow running out of control and this will be detected by the WDT
— The case of not accessing the peripherals due to a transient fault may result in WDT detection.

(4) **Support of system level functional redundancy**

None.

3.5.2 USB 2.0 Full-Speed Module (USBFS)

This module can be used to support communication over the USB.

This module is classified as non-safety relevant and only the information related to interference minimization is recommended and shown in Table 3.67.

Table 3.67 Recommended settings/usage for interference minimization for the USBFS module

ID	Description	Limitation/ Comment
USBFS_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRB.MSTPB11=1b	N/A

Table 3.68 Recommended settings/usage for protection against systematic System/SW faults for the USBFS module

ID	Description	Limitation/ Comment
USBFS_recc_S1	If the USB channel is used to transmit data for both NSR and SR functions, then a suitable SW handling is required to avoid compromising the safety functions	N/A

3.5.3 Serial Communication Interface (SCI)

This module can be used by the safety application to support serial communications over the SCI bus.

(1) **Recommended settings**

Table 3.69 to Table 3.71 describe the recommended settings for the Serial Communication Interface module.

Table 3.69 Recommended settings/usage for the SCI module

ID	Description	Limitation/ Comment
SCI_sett_U1	If compatible with application requirements, parity bit generation function and parity bit check function (SMR.PE = 1) shall be enabled	N/A
SCI_sett_U2	If compatible with application requirements, Block Transfer Mode (SMR.BLK = 1, if SCMR.SMIF = 1) shall be disabled in order not to ignore parity errors	N/A

Table 3.70 Recommended settings/usage for interference minimization for the SCI module

ID	Description	Limitation/ Comment
SCI_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRB.MSTPB22=1b (SCI9), MSTPCRB.MSTPB27=1b (SCI4), MSTPCRB.MSTPB28=1b (SCI3), MSTPCRB.MSTPB29=1b (SCI2), MSTPCRC.MSTPC30=1b (SCI1), MSTPCRC.MSTPC31=1b (SCI0)	N/A

Table 3.71 Recommended settings/usage for protection against systematic System/SW faults for the SCI module

ID	Description	Limitation/ Comment
SCI_sett_S1	If compatible with application requirements, the noise cancellation filter (SEMR.NFEN = 1) shall be enabled	N/A
SCI_recc_S1	If the serial channel is used to transmit data for both NSR and SR functions, then a suitable SW handling is required to avoid compromising the safety functions	N/A

(2) **Recommended protection against permanent faults**

Table 3.72 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Serial Communication Interface module.

Table 3.72 Recommended protection from permanent failure of SCI

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
SCI _prot_1	High	EndToEnd, Config_reg_read_back	Self-test or Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the SCI from transient failures. In order to do this, see the following information.

When protection SCI_prot_1 is used:

a. Overall coverage can still be considered high even if the effectiveness of Config_reg_read_back is decreased because a low frequency of the read back operation is not expected to have negative impact on application performance. This is because the EndToEnd protection can also cope with some faults in the module configuration, and because the negative impact on the aggregated coverage is negligible.

(4) **Support of system level functional redundancy**

Different units can be used to support system level redundancy.

3.5.4 IrDA Interface

This module can be used to support data communications on the Infrared Data Association (IrDA).

This module is classified as non-safety relevant and only the information related to interference minimization is recommended and shown in Table 3.73.

Table 3.73 Recommended settings/usage for interference minimization for the IrDA module

ID	Description	Limitation/ Comment
IrDA_sett_I1	Enable Module-Stop State when the peripheral is not used by setting MSTPCRB.MSTPB5=1b	N/A

3.5.5 I2C Bus Interface (IIC)

This module can be used by the safety application to support communication over the I2C bus.

(1) **Recommended settings**

Table 3.74 to Table 3.76 describe the recommended settings for the I2C Bus Interface module.

Table 3.74 Recommended settings/usage for the IIC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.75 Recommended settings/usage for interference minimization for the IIC module

ID	Description	Limitation/ Comment
IIC_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRB.MSTPB7 = 1b (IIC2), MSTPCRB.MSTPB8 = 1b (IIC1), and MSTPCRC.MSTPC9 = 1b (IIC0)	N/A

Table 3.76 Recommended settings/usage for protection against systematic System/SW faults for the IIC module

ID	Description	Limitation/ Comment
IIC_sett_S1	The Digital Noise Filter shall be enabled (ICFER.NFE = 1b, use ICMR3.NF[1:0] bits to set up the filter)	Check the ratio between the frequency of the internal operating clock (PCLKB) and the transfer rate (see section 31.6 of [1])
IIC_sett_S2	If compliant with application requirements, in order to improve robustness of the communication, Timeout Detection Time Selection shall be set to Short mode (ICMR2.TMOS=1b)	N/A
IIC_sett_S3	The timeout function shall be enabled (ICFER.TMOE=1) to detect long-interval stop of the SCL (clock signal)	N/A
IIC_sett_S4	The arbitration-lost detection function shall be enabled (ICFER.MALE=1b) in order to detect master arbitration-lost	N/A
IIC_sett_S5	The NACK Transmission Arbitration-Lost Detection shall be enabled (ICFER.NALE=1b) in order to improve error detection functionalities	N/A
IIC_sett_S6	The Slave Arbitration-Lost Detection shall be enabled (ICFER.SALE=1b) in order to improve error detection functionalities	N/A
IIC_sett_S7	The NACK Reception Transfer Suspension shall be enabled (ICFER.NACKE=1b) in order to suspend data transmission/reception after NACKF flag is set to 1	N/A
IIC _recc_S1	If the I2C channel is used to transmit data for both NSR and SR functions, then a suitable SW handling is required to avoid compromising the safety functions	N/A

(2) **Recommended protection against permanent faults**

Table 3.77 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the I^2C Bus Interface module.

Table 3.77 Recommended protection from permanent failure of IIC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
IIC_prot_1	High	EndToEnd, Config_reg_read_back	Self-test or Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the IIC from transient failures. In order to do this, see the following information.

When protection IIC_prot_1 is used:

a. Overall coverage can still be considered high even if the effectiveness of Config_reg_read_back is decreased because a low frequency of the read back operation is not expected to negatively impact the application performance. This is because the EndToEnd protection can also cope with some faults in the module configuration, and because the negative impact on the aggregated coverage is negligible.

(4) **Support of system level functional redundancy**

Different units (up to three channels) can be used to support system level redundancy.

3.5.6 Control Area Network Module (CAN)

This module can be used by the safety application to support communication over the CAN bus.

(1) **Recommended settings**

Table 3.78 to Table 3.80 describe the recommended settings for the CAN module.

Table 3.78 Recommended settings/usage for the CAN module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.79 Recommended settings/usage for interference minimization for the CAN module

ID	Description	Limitation/ Comment
CAN_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRB.MSTPB2=1b	N/A

Table 3.80 Recommended settings/usage for protection against systematic System/SW faults for the CAN module

ID	Description	Limitation/ Comment
CAN_recc_S1	If the CAN channel is used to transmit data for both NSR and SR functions, then a suitable SW handling is required to avoid compromising the safety functions	N/A

(2) **Recommended protection against permanent faults**

Table 3.81 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the CAN module.

Table 3.81 Recommended protection from permanent failure of CAN

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
CAN_prot_1	High	EndToEnd, Config_reg_read_back	Self-test or Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the communication over CAN from transient failures. In order to do this, see the following information.

When protection CAN_prot_1 is used:

a. The effectiveness of Config_reg_read_back is decreased because the test depends on the frequency of the read back operation as this is expected to be low so as to not negatively impact the application performance. The overall coverage can be still considered high because the EndToEnd protection can also cope with faults in the module configuration.

(4) **Support of system level functional redundancy**

None.

3.5.7 Serial Peripheral Interface (SPI)

This module can be used by the safety application to support communications over the SPI bus.

(1) **Recommended settings**

Table 3.82 to Table 3.84 describe the recommended settings for the Serial Peripheral Interface module.

Table 3.82 Recommended settings/usage for the SPI module

ID	Description	Limitation/ Comment
SPI_sett_U1	If compatible with application requirements, the setting for the detection of mode error (SPCR.MODFEN=1b) shall be enabled	N/A
SPI_sett_U2	Parity bit generation and parity bit check shall be enabled (SPCR2.SPPE=1b)	N/A

Table 3.83 Recommended settings/usage for interference minimization for the SPI module

ID	Description	Limitation/ Comment
SPI_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRB.MSTPB18=1b (SPI1) and MSTPCRB.MSTPB19=1b (SPI0)	When low power procedure is used, follow details provided in section 33.5.2 of [1]

Table 3.84 Recommended settings/usage for protection against systematic System/SW faults for the SPI module

ID	Description	Limitation/ Comment
SPI_recc_S1	If the SPI channel is used to transmit data for both NSR and SR functions, then a suitable SW handling is required to avoid compromising the safety functions	N/A

(2) **Recommended protection against permanent faults**

Table 3.85 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Serial Peripheral Interface module.

Table 3.85 Recommended protection from permanent failure of SPI

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
SPI_prot_1	High	EndToEnd, Config_reg_read_back	Self-test or Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the SPI from transient failures. In order to do this, see the following information.

When protection SPI_prot_1 is used:

a. Only the coverage of Config_reg_read_back is decreased to low because the effectiveness of the mechanism depends on the frequency of the read back operation as this is expected to be low so as to not negatively impact the application performance. The overall coverage can be still considered high because the EndToEnd protection can also cope with faults in the module configuration.

(4) **Support of system level functional redundancy**

Different SPI units (up to 2 are available) can be used to support system level redundancy.

3.5.8 Quad Serial Peripheral Interface (QSPI)

This module can be used by the safety application to support connection of memory modules that have an SPI-compatible interface.

(1) **Recommended settings**

Table 3.86 to Table 3.88 describe the recommended settings for the Serial Peripheral Interface module.

Table 3.86 Recommended settings/usage for the QSPI module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.87 Recommended settings/usage for interference minimization for the QSPI module

ID	Description	Limitation/ Comment
QSPI_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRB.MSTPB6=1b	N/A

Table 3.88 Recommended settings/usage for protection against systematic System/SW faults for the QSPI module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.89 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Quad Serial Peripheral Interface module.

Table 3.89 Recommended protection from permanent failure of QSPI

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
QSPI_prot_1	High	EndToEnd, Config_reg_read_back	Self-test or Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the QSPI from transient failures. In order to do this, see the following information.

When protection QSPI_prot_1 is used:

a. Only the coverage of Config_reg_read_back is decreased to low because the effectiveness of the mechanism depends on the frequency of the read back operation as this is expected to be low so as to not negatively impact the application performance. The overall coverage can be still considered high because the EndToEnd protection can also cope with faults in the module configuration.

(4) **Support of system level functional redundancy**

None.

3.5.9 Serial Sound Interface (SSI)

This module can be used by the safety application to support communication with digital audio devices. The assumption is that this peripheral is used as part of a protection mechanism to notify the user about the presence of error conditions.

(1) **Recommended settings**

Table 3.90 to Table 3.92 describe the recommended settings for the Serial Sound Interface module.

Table 3.90 Recommended settings/usage for the SSI module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.91 Recommended settings/usage for interference minimization for the SSI module

ID	Description	Limitation/ Comment
SSI_sett_I1	Enable Module-Stop State when the peripheral is not used by setting MSTPCRC.MSTPC7=1b (SSI1) and MSTPCRC.MSTPC8=1b (SSI0)	N/A

Table 3.92 Recommended settings/usage for protection against systematic System/SW faults for the SSI module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.93 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Serial Sound Interface module.

Table 3.93 Recommended protection from permanent failure of SSI

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
SSI_prot_1	High	SSI_check	Self-test	N/A
SSI_prot_2	Low	Config_reg_read_back	Run-time	A full run-time test is not possible

(3) **Consideration for transient faults**

Based on the assumption that this peripheral is used as part of a safety mechanism, protection against transient faults is not required.

(4) **Support of system level functional redundancy**

Different units (up to two channels) can be used to support system level redundancy.

3.5.10 Secure Digital Host Interface (SDHI)

This module can be used by the safety application to support connection with external memory cards.

(1) **Recommended settings**

Table 3.94 to Table 3.96 describe the recommended settings for the Secure Digital Host Interface module.

Table 3.94 Recommended settings/usage for the SDHI module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.95 Recommended settings/usage for interference minimization for the SDHI module

ID	Description	Limitation/ Comment
SDHI_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRC.MSTPC12=1b	N/A

Table 3.96 Recommended settings/usage for protection against systematic System/SW faults for the SDHI module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.97 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Secure Digital Host Interface module.

Table 3.97 Recommended protection from permanent failure of SDHI

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
SDHI_prot_1	Medium	SDHI_DMAC_triggerISR	Self-test or Run-time	Failures related to access error/command error detection interrupt request generation are not covered
	Medium	Config_reg_read_back	Self-test	N/A
SDHI_prot_3	Medium	SDHI_information_redundancy	Self-test or Run-time	N/A
SDHI_prot_4	Medium	Write_verify	Self-test or Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the SDHI from transient failures. In order to do this, see the following information.

When protections SDHI_prot_1/3/4 are used:

a. Overall coverage is kept medium.

(4) **Support of system level functional redundancy**

None.

3.5.11 Multi Media Card (MMC)

This module can be used by the safety application to support connection with external memory cards.

(1) **Recommended settings**

Table 3.98 to Table 3.100 describe the recommended settings for the Multi Media Card module.

Table 3.98 Recommended settings/usage for the MMC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.99 Recommended settings/usage for interference minimization for the MMC module

ID	Description	Limitation/ Comment
MMC_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRC.MSTPC12=1b	N/A

Table 3.100 Recommended settings/usage for protection against systematic System/SW faults for the MMC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.101 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Multi Media Card module.

Table 3.101 Recommended protection from permanent failure of MMC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
MMC_prot_1	Medium	SDHI_DMAC_triggerISR	Self-test or Run-time	Failures related to access error/command error detection interrupt request generation are not covered
MMC_prot_2	Medium	Config_reg_read_back	Self-test	To improve the coverage level, consider redundantly storing data/information
MMC_prot_3	Medium	SDHI_information_redundancy	Self-test or Run-time	N/A
MMC_prot_4	Medium	Write_verify	Self-test or Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the MMC from transient failures. In order to do this, see the following information.

When protections MMC_prot_1/3/4 are used:

a. Overall coverage is kept medium.

(4) **Support of system level functional redundancy**

None.

3.6 Internal interaction module(s)

3.6.1 Interrupt Controller (ICU)

This module can be used by the safety application to support interrupt based SW control flows.

(1) **Recommended settings**

Table 3.102 to Table 3.104 describe the recommended settings for the Interrupt Controller module.

Table 3.102 Recommended settings/usage for the ICU module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.103 Recommended settings/usage for interference minimization for the ICU module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.104 Recommended settings/usage for protection against systematic System/SW faults for the ICU module

ID	Description	Limitation/ Comment
ICU_sett_S1	To improve robustness of the system design, digital filters shall be used on IRQ pins (see IRQCRi register). Similar considerations are valid for the non-maskable interrupt pin (see the NMICR register).	Before adopting these settings, verify that the signal filtering effect and the delay introduced by the filtering itself are compatible with the application requirements
ICU_recc_S1	If interrupt requests are associated to both NSR and SR functions, then a suitable SW handling is required to avoid compromising the safety functions. Whenever compatible with real time constraints, this shall include a higher priority being associated to SR interrupt requests.	N/A

(2) **Recommended protection against permanent faults**

Error! Reference source not found. describes the safety mechanisms to protect the safety application from permanent failures of the ICU module.

Table 3.105 Recommended protection from permanent failure of ICU

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
ICU_prot_1	Medium	ICU_int_source_check, ICU_int_count, Config_reg_read_back	Run-time	When considering the fault handling strategy, note that registers such as the Non-Maskable Interrupt Enable Register (NMIER) cannot be re-written by SW. See [1] for more information. Application level measures reported in the other sections of the document such as the E2E mechanisms, also contribute to the protection of this module. Consider also that the following failure modes are left uncovered: • Interrupt priority order not respected • Clock restoration request not served when returning from low power modes.

Note: The medium level of coverage reported in **Error! Reference source not found.** also includes the contribution of the safety mechanisms such as LVD_Monitoring, DMAC_Crc, DTC_Crc, GPT_INT_generation, POEG_ExtLoop, AGT_INT_generation, AGT_TriggerDTC, and GPT_DMAC_connection.

This is because the ICU is also indirectly protected when testing interaction with other modules.

(3) **Consideration for transient faults**

It is recommended to protect the ICU from transient failures. In order to do this, see the following information.

When protection ICU_prot_1 is used:

a. Overall coverage can be kept at a medium level considering the positive impact of application level measures, as the E2E mechanisms, that are effective against transient failures.

(4) **Support of system level functional redundancy**

None.

3.6.2 Event Link Controller (ELC)

This module can be used to support direct interaction between peripheral modules.

This module is classified as non-safety relevant and only the information related to interference minimization is recommended and reported in Table 3.106**Error! Reference source not found.**.

Table 3.106 Recommended settings/usage for interference minimization for the ELC module

ID	Description	Limitation/ Comment
ELC_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRC.MSTPC14 = 1b	N/A

3.7 Sync module(s)

3.7.1 General PWM Timer (GPT)

This module can be used a general purpose timer.

(1) **Recommended settings**

Table 3.107 to Table 3.109 describe the recommended settings for the General PWM Timer module.

Table 3.107 Recommended settings/usage for the GPT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.108 Recommended settings/usage for interference minimization for the GPT module

ID	Description	Limitation/ Comment
GPT_settl_1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRD.MSTPD5 (GPT32x (x=0 to 3), and MSTPCRD.MSTPD6 (GPT32x (x=4 to 9))	N/A

Table 3.109 Recommended settings/usage for protection against systematic System/SW faults for the GPT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.110 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the General PWM Timer module.

Table 3.110 Recommended protection from permanent failure of GPT

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
GPT_prot_1a	Medium	GPT_INT_generation, GPT_DMAC_connection, GPT_DTC_connection, GPT_prot_1b	Self-test	To achieve the suggested level of coverage, both GPT_prot_1a and GPT_prot_1b protections have to be adopted. If compare match interrupts are not used, GPT_INT_generation can be omitted with no impact on the coverage. If the timer is not used in connection with the DMACA and/or DTC, GPT_DMAC_connection and/or GPT_DTC_connection can be omitted with no impact on the coverage.
GPT_prot_1b	Medium	GPT_redund_chan_input _capt, GPT_chan_loop_back, PWM_short_monitor, POEG_ExtLoop, Config_reg_read_back	Run-time	PWM_short_monitor checks only for the overlapping of the phases, while other inconsistencies are not detected The clock source is common for GPT_chan_loop_back and is not covered by the mechanism. In order to mitigate possible issues, the user should periodically monitor the timer counter value, checking that it is

				varying its value. If the input capture mode is not used, consider protection GPT_prot_2a/b as an alternative with similar coverage. If the timer is not used in connection with POEG, PWM_short_monitor can be omitted with no impact on the coverage. If the timer is not used to produce PWM signals, POEG_ExtLoop and GPT_chan_loop_back can be omitted with no impact on the coverage.
GPT_prot_2a	Medium	GPT_sync_chan, AGT_GPT_freq_check, GPT_INT_generation, GPT_DMAC_connection, GPT_DTC_connection, GPT_prot_2b	Self-test	To achieve the suggested level of coverage, both GPT_prot_2a and GPT_prot_2b protections have to be adopted. If compare match interrupts are not used, GPT_INT_generation can be omitted with no impact on the coverage. If the timer is not used in connection with the DMACA and/or DTC, GPT_DMAC_connection and/or GPT_DTC_connection can be omitted with no impact on the coverage.
MTU2a_prot_2b	Medium	GPT_chan_loop_back, PWM_short_monitor, POEG_ExtLoop, Config_reg_read_back	Run-time	The clock source is common for GPT_chan_loop_back and is not covered by the mechanism. In order to mitigate possible issues, the user should periodically monitor the timer counter value, checking that it is varying its value. If the timer is not used in connection with POEG, PWM_short_monitor can be omitted with no impact to the coverage. If the timer is not used to generate PWM signals, then POEG_ExtLoop and GPT_chan_loop_back can be omitted with no impact on the coverage.

(3) **Consideration for transient faults**

With the exception of the impact on configuration registers, no further mitigation is considered to be required for transient failures because their effects on the timer operations are confined in one period and impact on output signals (for example, PWM outputs) at system level is expected to be negligible.

As for the other modules, configuration registers are protected at run-time by Config_reg_read_back.

(4) **Support of system level functional redundancy**

More than one channel implements the input capture function as well as PWM output generation and they can be used redundantly to support system level requirements. It is important to note that channel operations can be synchronized through SW (for example, start of different channels), and this can facilitate the comparison of redundant signals. The suggested safety mechanisms for this module along with the external Watch Dog are considered to be sufficient to mitigate possible dependencies.

3.7.2 Asynchronous General Purpose Timer (AGT)

This module can be used a general purpose timer.

(1) **Recommended settings**

Table 3.111 to Table 3.113 describe the recommended settings for the Buses module.

Table 3.111 Recommended settings/usage for the AGT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.112 Recommended settings/usage for interference minimization for the AGT module

ID	Description	Limitation/ Comment
GPT_settl_1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRD.MSTPD2 (AGT1) and MSTPCRD.MSTPD3 (AGT0)	NA

Table 3.113 Recommended settings/usage for protection against systematic System/SW faults for the AGT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.114 describes the safety mechanisms to protect the safety application from permanent failures affecting communication involving the Buses.

Table 3.114 Recommended protection from permanent failure of AGT

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
AGT_prot_1a	Medium	AGT_INT_generation, AGT_TriggerADC, AGT_TriggerDMAC, AGT_TriggerDTC, AGT_prot_1b	Self-test	To achieve the suggested level of coverage, both AGT_prot_1a and AGT_prot_1b protections have to be adopted. If compare match interrupts are not used, AGT_INT_generation can be omitted with no impact on the coverage. If the timer is not used in connection with the ADC, AGT_TriggerADC can be omitted with no impact on the coverage. If the timer is not used in connection with the DMACA and/or DTC, AGT_TriggerDMAC and/or AGT_TriggerDTC can be omitted with no impact on the coverage.
AGT_prot_1b	Medium	GPT_loop_back_AGT, Config_reg_read_back	Run-time	If the timer is not used to generate PWM signals, then GPT_loop_back_AGT can be omitted with no impact on the coverage
AGT_prot_2a	Medium	AGT_sync_chan, AGT_GPT_freq_check,	Self-test	To achieve the suggested level of coverage, both AGT_prot_2a

	AGT_INT_generation, AGT_TriggerADC, AGT_TriggerDMAC, AGT_TriggerDTC, AGT_prot_2b	and AGT_prot_1b protections have to be adopted. If compare match interrupts are not used, GPT_INT_generation can be omitted with no impact on the coverage. If the timer is not used in connection with the ADC, AGT_TriggerADC can be omitted with no impact on the coverage. If the timer is not used in connection with the DMACA and/or DTC, AGT_TriggerDMAC and/or AGT_TriggerDTC can be omitted with no impact on the coverage.

(3) **Consideration for transient faults**

With the exception of the impact on configuration registers, no further mitigation is considered to be required for transient failures because their effects on the timer operations are confined in one period and impact on output signals at system level is expected to be negligible.

As for other modules, configuration registers are protected at run-time by Config_reg_read_back.

(4) **Support of system level functional redundancy**

Two units are present, offering the same functionalities. These can be used redundantly to support system level requirements. The suggested safety mechanisms for this module along with the external Watch Dog are considered to be sufficient to mitigate possible dependencies.

3.7.3 Realtime Clock (RTC)

This module can be used to support calendar functions.

This module is classified as non-safety relevant and only the information related to interference minimization is recommended and reported in Table 3.115.

Table 3.115 Recommended settings/usage for interference minimization for the RTC module

ID	Description	Limitation/ Comment
RTC_sett_I1	If the module is not used, it is recommended to follow the procedure described in section 25.6.7 of [1]. In addition, if the module is in use but the following features are not used, it is recommended to disable the output RTCOUT (by setting RCR2.RTCOE = 0b).	N/A

3.8 Data management module(s)

3.8.1 DMA Controller (DMAC)

This module can be used by the safety application to reduce CPU load in transferring data between memory locations.

(1) **Recommended settings**

Table 3.116 Recommended settings/usage for the DMAC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.117 Recommended settings/usage for interference minimization for the DMAC module

ID	Description	Limitation/ Comment
DMAC_sett_I1	Enable Module-Stop state when the module is not used by setting MSTPCRA.MSTPA22 = 1b	NA

Table 3.118 Recommended settings/usage for protection against systematic System/SW faults for the DMAC module

ID	Description	Limitation/ Comment
DMAC_recc_S1	DMAC programming has to be done by the SW component with the highest safety integrity level. In addition, write access to the DMAC registers shall be protected by the MPU.	N/A

(2) **Recommended protection against permanent faults**

Table 3.119 Recommended protection from permanent failure of DMAC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
DMAC_prot_1a	Medium	DMAC_Crc, DMAC_TriggerISR, DMAC_ExtTrigger, DMAC_prot_1b	Self-test	To achieve the suggested level of coverage, both DMAC_prot_1a and DMAC_prot_1b protections have to be adopted. If an external trigger is not used to kick off the DMAC transfer, then DMAC_ExtTrigger can be omitted with no impact on coverage.
DMAC_prot_1b	Medium	Config_reg_read_back	Run-time	Application level measures reported in other sections of the document, such as the E2E mechanisms and external WDT, also contribute to the protection of this module and have been considered in the coverage estimation. Particularly, E2E mechanisms will cover DMAC faults in transferring the message payload or in overwriting the message payload transferred in memory by the CPU. In addition, overwriting memory locations not configured for the DMAC transfer will potentially result in triggering the external WDT if this corrupts the program flow.

(3) **Consideration for transient faults**

It is recommended to protect the operation of the DMAC from transient failures. In order to do this, see the following information.

When protection DMAC_prot_1b is used:

a. Overall coverage is kept at a medium level, since application level measures reported in other sections of the document, such as the E2E mechanisms and external WDT, also contribute to the protection of this module, and have been considered in the coverage estimation.

(4) **Support of system level functional redundancy**

None.

3.8.2 Data Transfer Controller (DTC)

This module is used by the safety application to support data transfer functionalities.

(1) **Recommended settings**

Table 3.120 to Table 3.122 describe the recommended settings for the Data Transfer Controller module.

Table 3.120 Recommended settings/usage for the DTC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.121 Recommended settings/usage for interference minimization for the DTC module

ID	Description	Limitation/ Comment
DTC_sett_I1	Enable Module-Stop state when the module (/channel) is not used by setting MSTPCRA.MSTPA22 = 1b	N/A
DTC_sett_I2	Define Transfer Data Read setting to skip data read when vector numbers match (DTCCR.RRS=1). This helps to minimize interference towards potential RAM faults that can affect the DTC configuration.	N/A

Table 3.122 Recommended settings/usage for protection against systematic System/SW faults for the DTC module

ID	Description	Limitation/ Comment
DTC_recc_S1	DTC programming has to be done by the SW component with the highest safety integrity level. In addition, write access to the DTC registers shall be protected by the MPU.	N/A

(2) **Recommended protection against permanent faults**

Table 3.123 describes the safety mechanisms to protect the safety application from permanent failure of the Data Transfer Controller module.

Table 3.123 Recommended protection from permanent failure of DTC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
DTC_prot_1a	Medium	DTC_Crc, DTC_TriggerISR, DTC_prot_1b	Self-test	To achieve the suggested level of coverage, both DTC_prot_1a and DTC_prot_1b protections have to be adopted
DTC_prot_1b	Medium	Config_reg_read_back	Run-time	Application level measures reported in other sections of the document, such as the E2E mechanisms and external WDT, also contribute to protection of this module and have been considered in the estimation of the coverage. Particularly, E2E mechanisms will cover DTC faults in transferring the message payload or in overwriting the message payload transferred to memory by the CPU. In addition, overwriting memory locations not configured for the DTC transfer will potentially result in the firing of the external WDT if this corrupts the program flow.

(3) **Consideration for transient faults**

It is also recommended to protect the DTC from transient failures. In order to do this, see the following information.

When protection DTC_prot_1b is used:

Overall coverage is kept at a medium level since application level measures reported in other sections of the document, such as the E2E mechanisms and external WDT, also contribute to the protection of this module and have been considered in the estimation of the coverage.

(4) **Support of system level functional redundancy**

None.

3.8.3 CRC Calculator (CRC)

This module is used by the safety application to support CRC elaboration used for data control functionalities.

(1) **Recommended settings**

Table 3.124 to Table 3.126 describe the recommended settings for the CRC Calculator module.

Table 3.124 Recommended settings/usage for the CRC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.125 Recommended settings/usage for interference minimization for the CRC module

ID	Description	Limitation/ Comment
CRC_sett_I1	Enable Module-Stop state when the module(/channel) is not used by setting MSTPCRC.MSTPC1 = 1b	N/A

Table 3.126 Recommended settings/usage for protection against systematic System/SW faults for the CRC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.127 describes the safety mechanisms to protect the safety application from permanent failures of the CRC Calculator module.

Table 3.127 Recommended protection from permanent failure of CRC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
CRC_prot_1	Medium	CRC_SwCrc, Write_verify	Self-test	N/A

(3) **Consideration for transient faults**

No specific recommendations are provided to protect the CRC from transient failures since run-time failures will most likely result in a CRC error and are then handled by the application in the same way as data faults.

(4) **Support of system level functional redundancy**

None.

3.8.4 Data Operation Circuit (DOC)

This module can be used by the safety application as a co-processor for 16-bit addition, subtraction, and comparison.

(1) **Recommended settings**

Table 3.128 to Table 3.130 describe the recommended settings for the DOC.

Table 3.128 Recommended settings/usage for the DOC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.129 Recommended settings/usage for interference minimization for the DOC module

ID	Description	Limitation/ Comment
DOC_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRC.MSTPC13=1b	N/A

Table 3.130 Recommended settings/usage for protection against systematic System/SW faults for the DOC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.131 describes the safety mechanisms to protect the safety application from permanent failures of the DOC module.

Table 3.131 Recommended protection from permanent failure of DOC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
DOC_prot_1	Medium	Config_reg_read_back, DOC_check	Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the DOC from transient failures. In order to do this, see the following information.

When protection DOC_prot_1 is used:

a. Overall coverage is kept to Medium.

(4) **Support of system level functional redundancy**

None.

3.9 Memory

3.9.1 SRAM

This section deals with the on-chip high-speed SRAM that is the main volatile memory used for program execution.

S3 has two SRAM areas - SRAM0 (addresses from 20000000h to 2001FFFFh) and SRAM1 (addresses from 20020000h to 2002FFFFh).

The first 16 KB of SRAM0 (addresses from 20000000h to 20003FFFh) are protected by Error Correction Code (ECC), whereas the rest of SRAM0 and SRAM1 are protected by parity check.

(1) **Recommended settings**

Table 3.132 to Table 3.134 describe the recommended settings for the SRAM module.

Table 3.132 Recommended settings/usage for the SRAM module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.133 Recommended settings/usage for interference minimization for the SRAM module

ID	Description	Limitation/ Comment
SRAM_sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRA.MSTPA0 = 1b (SRAM0) MSTPCRA.MSTPA1 = 1b (SRAM1) MSTPCRA.MSTPA6 = 1b (ECC SRAM)	An area in SRAM0 provides error correction capability using ECC so that settings of the MSTPA0 bit and MSTPA6 (ECC SRAM Module Stop) bit must be the same

Table 3.134 Recommended settings/usage for protection against systematic System/SW faults for the SRAM module

ID	Description	Limitation/ Comment
RAM_recc_S1	If both NSR and SR functions are present, partition the memory into two areas and use the MPU to protect the SR data (see section 3.4.1 for more information)	N/A

(2) **Recommended protection against permanent faults**

Table 3.135 describes the safety mechanisms to protect the safety application from permanent failures of the RAM module.

Table 3.135 Recommended protection from permanent failure of SRAMs

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
SRAM_prot_1	Medium	SRAM_SWTest [3]	Self-test or Run-time	Application level measures reported in other section of the document, such as the E2E mechanisms and external WDT, also contribute to protection of this module. Particularly, E2E mechanisms will cover corruption of memory locations used to store message payloads. Also, corruption of memory locations used to support SW control flow (for example, stacks, and semaphores) will result in triggering the external WDT.
SRAM_prot_2	Medium	Parity check (see section 3.3.6)	Run-time	In case of fault accumulation, multi-bit corruption can be signaled as single-bit corruption and wrongly fixed
SRAM_prot_3	High	ECC	Run-time	Only partial coverage of SRAM0 is present (see [1] for details)

(3) **Consideration for transient faults**

It is recommended to protect the RAM from transient failures. In order to do this, see the following information.

When protection SRAM_prot_1/2/3 are used:

a. Overall coverage is kept in line with SRAM_prot_1 or SRAM_prot_2 depending on the memory location accessed.

To increase coverage, E2E mechanisms can cover corruption of memory caused by transient failures when locations are used to store message payloads. In addition, the use of multiple operation processing also contributes to the memory protection.

(4) **Support of system level functional redundancy**

None.

3.9.2 Flash Memory

This module is used by the safety application to store SW code and data.

(1) **Recommended settings**

Table 3.136 to Table 3.138 describe the recommended settings for the Flash Memory module.

Table 3.136 Recommended settings/usage for the Flash Memory module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.137 Recommended settings/usage for interference minimization for the Flash Memory module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.138 Recommended settings/usage for protection against systematic System/SW faults for the Flash Memory module

ID	Description	Limitation/ Comment
FlashMem_sett_S1	It is possible to configure an access window start/end block address to P/E command as allowed (AWS.FAWS, AWS.FAWE, and AWSC.FSPR)	These settings can be done only at compile time

(2) **Recommended protection against permanent faults**

Table 3.139 describes the safety mechanisms to protect the safety application from permanent failures of the Flash Memory module.

Table 3.139 Recommended protection from permanent failure of Flash Memory

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
FlashMem_prot_1	Medium	FlashMemory_Crc, ExtWDT (see Section 2.1), Config_reg_read_back	Run-time	Failures related to access error detection interrupt request generation are not covered. Corruption of memory sections storing Look-Up Table data are not generally detected by the external WDT. It is suggested to create dedicated CRCs for such data and then adopt the FlashMemory_Crc mechanism as protection.

(3) **Consideration for transient faults**

It is recommended to protect the Flash Memory from transient failures. In order to do this, see the following information.

When protection FlashMem_prot_1 is used:

a. Overall coverage is kept at a medium level because of the effect of the Watch Dog that compensates for the decreased effectiveness of the FlashMemory_Crc mechanism, which is limited by the test frequency, and considered set at low level, so as to not impact the application performance.

(4) **Support of system level functional redundancy**

None.

3.9.3 Memory Mirror Function (MMF)

This module is used by the safety application to mirror the desired application image load address in code flash memory to the application image link address.

(1) **Recommended settings**

Table 3.140 to Table 3.142 describe the recommended settings for the MMF.

Table 3.140 Recommended settings/usage for the MMF module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.141 Recommended settings/usage for interference minimization for the MMF module

ID	Description	Limitation/ Comment
MMF_Sett_I1	Disabling Memory Mirror Function when it is not used by setting MMEN.EN=0b	N/A

Table 3.142 Recommended settings/usage for protection against systematic System/SW faults for the MMF module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.143 describes the safety mechanisms to protect the safety application from permanent failures of the MMF module.

Table 3.143 Recommended protection from permanent failure of MMF

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
MMF_prot_1	Medium	MMF_Crc, ExtWDT (see section 2.1), Config_reg_read_back	Run-time	Failures related to access error detection interrupt request generation are not covered. Corruption of memory sections storing Look-Up Table data are not generally detected by the external WDT and it is suggested to create dedicated CRCs for such data and then adopt as protection the MMF_Crc mechanism.

(3) **Consideration for transient faults**

It is recommended to protect the Mirror Memory from transient failures. In order to do this, see the following information.

When protection MMF_prot_1 is used:

a. Overall coverage is kept at a medium level because of the effect of the Watch Dog that compensates for the decreased effectiveness of the MMF_Crc mechanism, which is limited by the test frequency, and considered set at a low level, so as to not impact application performance.

(4) **Support of system level functional redundancy**

None.

3.10 System Support Module(s)

3.10.1 Resets

This module is used by the safety application to enable the MCU reset state handling.

(1) **Recommended settings**

No specific setting is suggested for this module. Note that the settings related to the use of the internal HW WDT are described in the internal WDT related sections 3.4.2 and 3.3.4.

(2) **Recommended protection against permanent faults**

Table 3.144 Recommended protection from permanent failure of Resets

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
Resets_prot_1	Medium	SW_reset_check, Write_verify	Self-test	Failure modes related to an inconsistent status of the reset flags (with the exception of the SW reset case) are not covered

(3) **Consideration for transient faults**

No specific recommendation is given.

(4) **Support of system level functional redundancy**

None.

3.10.2 Option-Setting Memory and Information Memory

This module is used by the safety application to support MCU configuration.

(1) **Recommended settings**

Table 3.145 to Table 3.147 describe the recommended settings for the Option-Setting Memory module.

Table 3.145 Recommended settings/usage for the OSM module

ID	Description	Limitation/ Comment
OSM_sett_U1	The auto-start mode shall be enabled for IWDT and WDT to minimize the impact of the faults interfering with the WDT programming (see the Option Function Select Register 0 (OFS0))	N/A
OSM_sett_U2	Low Voltage Detection monitoring shall be enabled to make available this protection mechanism (see the Option Function Select Register 1 (OFS1))	N/A

Table 3.146 Recommended settings/usage for interference minimization for the OSM module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.147 Recommended settings/usage for protection against systematic System/SW faults for the OSM module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.148 describes the safety mechanisms to protect the safety application from permanent failures of the Option-Setting Memory module.

Table 3.148 Recommended protection from permanent failure of OSM

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
OSM_prot_1	High	Config_reg_read_back	Self-test	N/A

(3) **Consideration for transient faults**

The OSM registers are used only after reset and no further mitigation is considered to be required for transient failures.

(4) **Support of system level functional redundancy**

None.

3.10.3 Clock Generation Circuit

This module is used by the safety application for clock generation.

(1) **Recommended settings**

Table 3.149 to Table 3.151 describe the recommended settings for the Clock Generation Circuit module.

Table 3.149 Recommended settings/usage for the CGC module

ID	Description	Limitation/ Comment
CGC_sett_U1	It is recommended to generate the system clock through the PLL and Main clock Oscillator	N/A

Table 3.150 Recommended settings/usage for interference minimization for the CGC module

ID	Description	Limitation/ Comment
CGC_sett_I1	If the external bus and/or external SDRAM are not in use by the application, it shall be disabled by the corresponding clocks by setting EBCKOCR.EBCKOEN = 0b	N/A

Table 3.151 Recommended settings/usage for protection against systematic System/SW faults for the CGC module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.152 describes the safety mechanisms to protect the safety application from permanent failure of the Clock Generation Circuit module.

Table 3.152 Recommended protection from permanent failure of CGC

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
CGC_prot_1	Medium	CGC_clk_monitor, IWDT_Clock_Monitoring	Self-test	None
CGC_prot_2	Medium	CGC_ClkStop, CGC_clk_monitor_run_time EndToEnd, Config_reg_read_back	Run-time	The most critical clocks for the application, based on the recommendation provided in Table 3.149, are the Main clock oscillator, the PLL, and clock for the CPU (ICLK). The CAC Configuration Software provided by Renesas for the CGC_clk_monitor safety mechanism and the IWDT, provides protection for all these clocks.

(3) **Consideration for transient faults**

It is also recommended to protect the CGC from transient failures (these can lead to extra or missing clock pulses as well as a full clock stop, if affecting configuration registers). In order to do this, see the following information.

When protection CGC_prot_2 is used:

a. Overall coverage is decreased to a low level because:
   - effectiveness of Config_reg_read_back is decreased so as to keep the impact of the test frequency at a low level so as to not impact the application performance
   - effectiveness of CGC_clk_monitor_run_time is decreased because the temporary decrease (rise) followed by rise (decrease) of the clock frequency can be masked and then not detected.

   To increase coverage, the usage of multiple operation processing can mitigate the effect of temporary failures in the generation of clock pulses (extra or missing clock pulses).

(4) **Support of system level functional redundancy**

None.

3.10.4 Low Power Mode

This module is used by the safety application to support low power mode handling.

(1) **Recommended settings**

Table 3.153 to Table 3.155 describe the recommended settings for the Low Power Mode module.

Table 3.153 Recommended settings/usage for the LPM module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.154 Recommended settings/usage for interference minimization for the LPM module

ID	Description	Limitation/ Comment
LPM_sett_I1	The Output Port Enable shall be disabled (SBYCR.OPE=0b)	In Software Standby mode or Deep Software Standby mode, the address bus and bus control signals are set to a high-impedance state. This allows for minimizing interference toward the external memory control. In absence of a proper SW control, external drivers (for example, pull-up/pull down) shall drive the memory signals.
LPM_sett_I2	To minimize interference, a module that is temporarily unused, or never used by the application, can be stopped by using the Module Clock Stop function, through the following registers: Module Stop Control Register A (MSTPCRA), Module Stop Control Register B (MSTPCRB), Module Stop Control Register C (MSTPCRC), Module Stop Control Register D (MSTPCRD)	N/A
LPM_sett_I3	To minimize interference toward the CGC_ClkStop mechanism, the MCU shall not enter Subosc-speed mode, where this function is not available (see Operating Power Control Register (OPCCR) and Sub Operating Power Control Register (SOPCCR)).	Entering Subosc-speed mode (see Operating Power Control Register (OPCCR) register and Sub Operating Power Control Register (SOPCCR)) might affect real-time constraints of the application because of the limitations on the maximum frequency. Check the impact on the application before using it.
LPM_sett_I4	To minimize interference with the LVD_Monitoring mechanism, the interrupt generation for the voltage monitoring 1 (DPSIER2.DLVD1IE) and voltage monitoring 2 (DPSIER2.DLVD2IE) shall be kept enabled	N/A

Table 3.155 Recommended settings/usage for protection against systematic System/SW faults for the LPM module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.156 describes the safety mechanisms to protect the safety application from permanent failure of the Low Power Mode module.

Table 3.156 Recommended protection from permanent failure of LPM

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
LPM_prot_1	Medium	ExtWDT (see section 2.1)	Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the LPM from transient failures. In order to do this, see the following information.

When protection LPC_prot_1 is used:

a. Overall coverage is kept at a medium level.

(4) **Support of system level functional redundancy**

None.

3.10.5 Register Write Protection

This module is used by the safety application to enable register write protection.

This module is primarily intended for MCU protection against SW faults.

(1) **Recommended settings**

Table 3.157 to Table 3.159 describe the recommended settings for the Register Write Protection Function module.

Table 3.157 Recommended settings/usage for the RWP module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.158 Recommended settings/usage for interference minimization for the RWP module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.159 Recommended settings/usage for protection against systematic System/SW faults for the RWP module

ID	Description	Limitation/ Comment
RWP_sett_S1	The register protect function shall be used	The Register Write Protection function protects important registers from being overwritten in case a program runs out of control. Remember to disable writing to registers that have the possibility to be protected, after they are set.

(2) **Recommended protection against permanent faults**

Table 3.160 describes the safety mechanisms to protect the safety application from permanent failure of the Register Write Protection Function module.

Table 3.160 Recommended protection from permanent failure of RWP

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
RWP_prot_1	High	Write_verify	Run-time	N/A

(3) **Consideration for transient faults**

It is recommended to protect the Register Write Protection Function module from transient failures. In order to do this, see the following information.

When protection RWP_prot_1 is used consider the following.

a. Overall coverage is kept high.

(4) **Support of system level functional redundancy**

None.

3.10.6 Boundary Scan

This module is primarily intended for debug support.

Note that this module is classified as non-safety relevant and only the information related to interference minimization is recommended and reported in Table 3.161.

Table 3.161 Recommended settings/usage for interference minimization for the Boundary Scan

ID	Description	Limitation/ Comment
BSCAN_sett_I1	Boundary scan must be executed when the RES# pin is driven LOW	

3.10.7 Linear Regulator (LDO)

The MCU includes a linear regulator that supplies voltage to the internal circuit and memory except for I/O, analog, USB, and battery backup power domain.

(1) **Recommended settings**

Table 3.162 to Table 3.164 describe the recommended settings for the Linear Regulator module.

Table 3.162 Recommended settings/usage for the LDO module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.163 Recommended settings/usage for interference minimization for the LDO module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.164 Recommended settings/usage for protection against systematic System/SW faults for the LDO module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.165 describes the safety mechanisms to protect the safety application from permanent failure of the Linear Regulator module.

Table 3.165 Recommended protection from permanent failure of LDO

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
LDO_prot_1	Medium	ExtWDT (see section 2.1)	Run-time	Since the linear regulator is supplying the internal circuit and memory, its failure is very likely to imply a failure of the CPU, and the SW will not be able to correctly refresh the external watchdog
LDO_prot_2	Medium	LDO_ext_monitoring	Self-test or Run-time	This safety mechanism implies an interaction with external HW

(3) **Consideration for transient faults**

It is recommended to protect the Linear Regulator module from transient failures. In order to do this, see the following information.

When protection LDO_prot_1 is used, consider the following:

a. Overall coverage is kept at a medium level.

(4) **Support of system level functional redundancy**

None.

3.11 Human Machine Interface

3.11.1 Key Interrupt Function (KINT)

This module is used by the safety application to generate a key interrupt.

(1) **Recommended settings**

Table 3.166 to Table 3.168 describe the recommended settings for the Key Interrupt Function module.

Table 3.166 Recommended settings/usage for the KINT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

Table 3.167 Recommended settings/usage for interference minimization for the KINT module

ID	Description	Limitation/ Comment
KINT_Sett_I1	Disabling usage of key interrupt flags when it is not used, by setting KRCTL.KRMD=0b	When KRCTL.KRMD = 0, setting KRF.KRFn = 1b (n=0,…,7) is prohibited

Table 3.168 Recommended settings/usage for protection against systematic System/SW faults for the KINT module

ID	Description	Limitation/ Comment
N/A	N/A	N/A

(2) **Recommended protection against permanent faults**

Table 3.169 describes the safety mechanisms to protect the safety application from permanent failure of the Key Interrupt Function module.

Table 3.169 Recommended protection from permanent failure of KINT

ID	Estimated Coverage Level	Suggested Safety Mechanisms	Suggested Protection Type	Limitation/Comment
KINT_prot_1	Medium	KINT_triggerISR, Config_reg_read_back	Run-time	Take care that the use of KINT_triggerISR implies the use of the external digital input signal to trigger the KEY_INTKR interrupt. The input signal and corresponding input pin shall be chosen according to application constraints. Coverage is a function of how many interrupts are covered. If all are tested, then the coverage is high, and otherwise it is decreased to medium.

(3) **Consideration for transient faults**

It is recommended to protect the Key Interrupt Function from transient failures. In order to do this, see the following information.

When protection KINT_prot_1 is used:

a. Overall coverage is kept medium.

(4) **Support of system level functional redundancy**

None.

3.11.2 Segment LCD Controller/Driver (SLCDC)

This module can be used to provide a controller for LCD display and display pins.

This module is classified as non-safety relevant and only the information related to interference minimization is recommended and shown in Table 3.170.

Table 3.170 Recommended settings/usage for interference minimization for the SLCDC module

ID	Description	Limitation/ Comment
SLCDC_Sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRC.MSTPC4=1b	N/A

3.11.3 Capacitive Touch Sensing Unit (CTSU)

This module can be used to measure the electrostatic capacitance of the touch sensor.

This module is classified as non-safety relevant and only the information related to interference minimization is recommended and shown in Table 3.171.

Table 3.171 Recommended settings/usage for interference minimization for the CTSU module

ID	Description	Limitation/ Comment
CTSU_Sett_I1	Enable Module-Stop state when the peripheral is not used by setting MSTPCRC.MSTPC3=1b	N/A

4. Safety Mechanisms Description

The safety mechanisms described in this section recommend procedures to achieve fault detection. Suitable actions for fault reaction/handling shall be considered by the system integrator.

Many procedures adopt an interrupt-based design strategy. If this is not the generally adopted SW design technique, variations of the same mechanism using a polling-based design can also be adopted.

This section is structured into four sub-sections.

Section 4.1 describes the safety mechanisms that protect functions influencing several MCU modules, and are characterized by the presence of HW features specifically designed for protection. These are called Chip-level safety mechanisms.

Section 4.2 describes the safety mechanisms that protect functions of specific MCU modules, and are characterized by the presence of HW features specifically designed for protection. These are called Module-level safety mechanisms.

Section 4.3 describes the safety mechanisms that protect functions of specific MCU modules, but do not exploit specific HW features, and rely on a full SW implementation. These are called SW-level safety mechanisms.

Section 4.4 describes the safety mechanisms that rely on connections/elements external to the MCU. These are called System-level safety mechanisms.

4.1 Chip level Safety Mechanisms

4.1.1 CAC_self_test

This mechanism can be used to detect faults affecting the CAC peripheral.

The concept is to use the CAC Configuration Software [3] provided by Renesas to configure the CAC HW peripheral to test the main clock oscillator and PLL against the sub-clock oscillator, providing a wrong target frequency, that is, a wrong PCLKB frequency. The test is passed if a frequency error interrupt is generated.

The SW test procedure is shown in Figure 4.1.

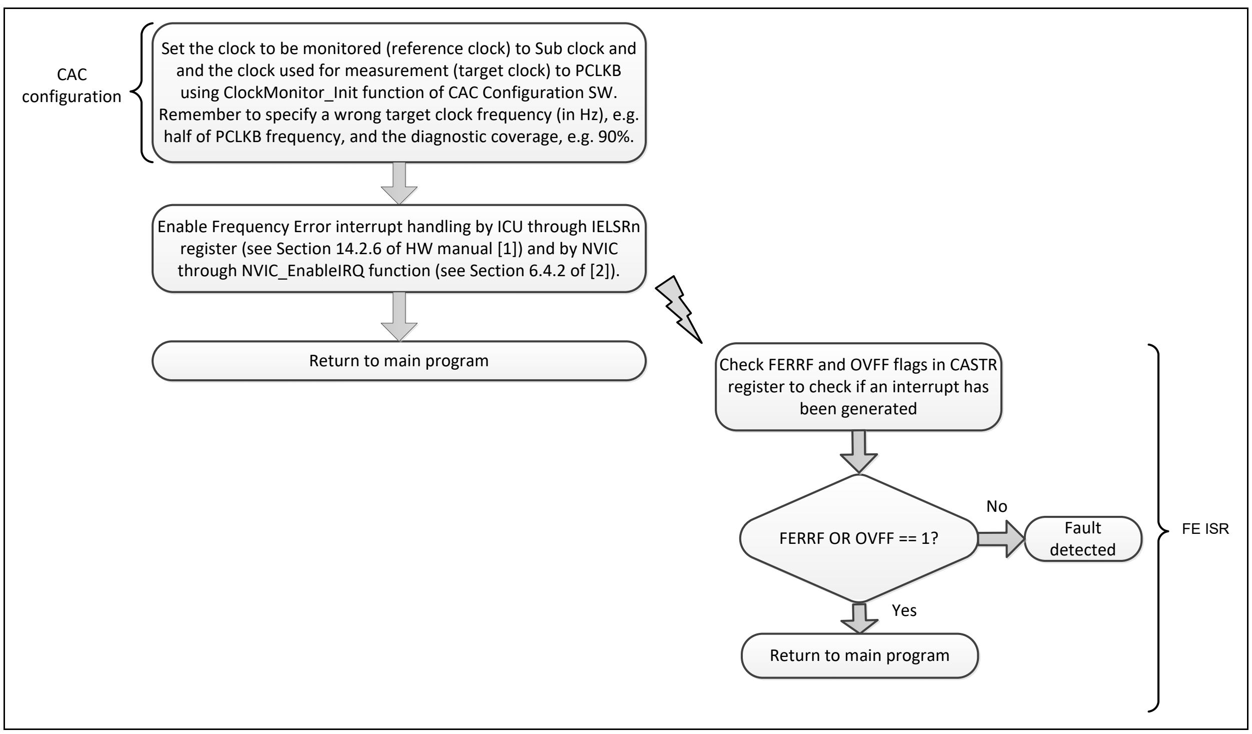

Figure 4.1 SW procedure for CAC_self_test using CAC Configuration Software

4.1.2 CGC_clk_monitor

This mechanism can be used to detect faults causing an alteration of the clocks generated from sources such as the main clock, sub clock, LOCO, HOCO, MOCO, IWDTCLK (IWDT-Dedicated Clock), and Peripheral module clock B (PCLKB).

The fault detection principle is that the distance between two edges of the signal under monitoring is measured using a counter clocked with a different clock source. To be in line with this principle, the following constraints shall be respected:

a. Setting in CACR1.FMCS has to be different from the one in CACR2.RSCS.
b. When PCLKB is selected using CACR1.FMCS (CACR2.RSCS), the clock selected by CACR2.RSCS (CACR1.FMCS) shall not be the one used as the source to derive PCLKB in the Clock Generation Circuit module.

Figure 4.2 shows a general configuration procedure to monitor one of the possible clock sources.

The configuration procedure shall be repeated to stop the monitoring of the current clock signal and start the monitoring of a different one up to the point when all the required clocks are tested in a sequence.

See the details about the SW procedure in section 10.3 of [1].

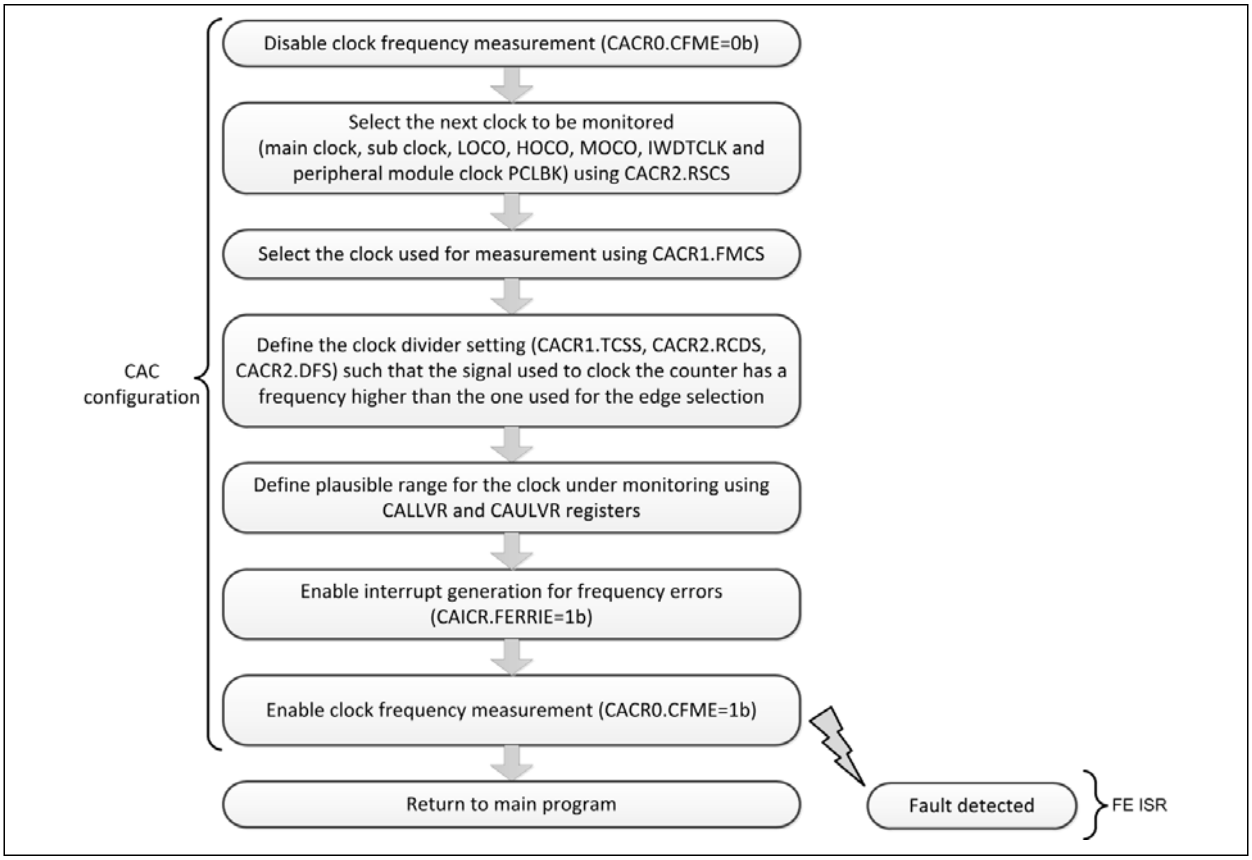

Figure 4.2 SW procedure for CAC_clk_monitor to support monitoring of different clock sources

The CAC Configuration Software [3] from Renesas can be used to aid the testing of the main clock oscillator and PLL against the sub-clock oscillator.

The procedure when using the CAC Configuration software is reported in Figure 4.3.

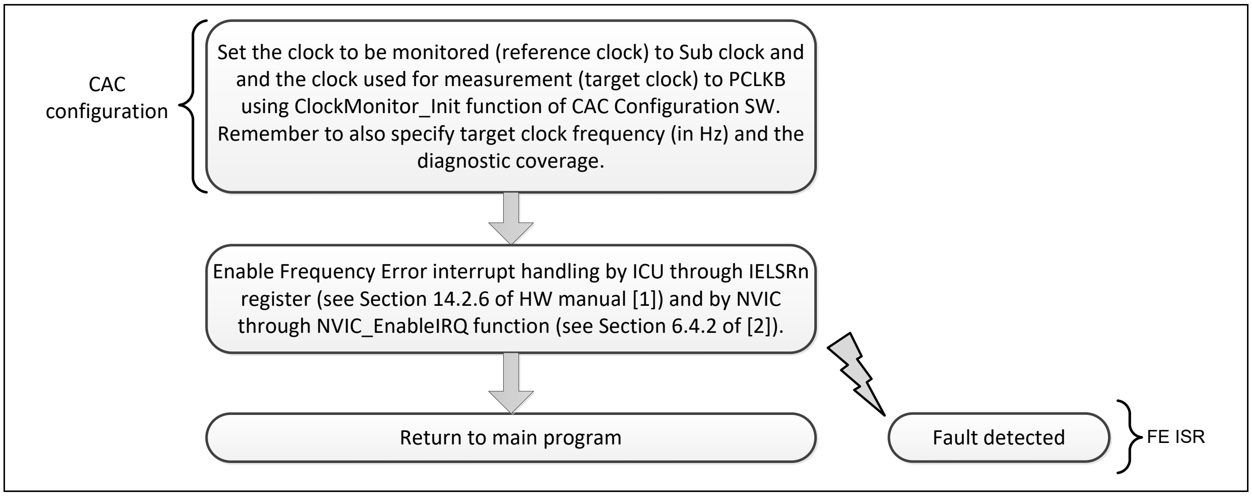

Figure 4.3 SW procedure for CAC_clk_monitor using CAC Configuration Software

Note that the CAC Configuration software also enables the oscillation stop detection peripheral. See [3] for more information.

4.1.3 CGC_clk_monitor_run_time

This mechanism is to be used at run-time to detect faults affecting the Clock Generation Circuit module.

The recommendation is to detect faults affecting the main clock oscillator and PLL, which are the most critical clocks for the application, based on the recommendation provided in Table 3.149.

For this purpose, the recommendation is to use the CAC Configuration Software provided by Renesas [3], as described in section 4.1.2

Alternatively, if this mechanism is to be applied to different clock sources with respect to the main clock oscillator and PLL, the tailoring summarized in Table 4.1 is to be applied to the complete procedure reported in Figure 4.2.

Table 4.1 Tailoring for CGC_clk_monitor_run_time

PCLK Setting	Tailoring of the procedure reported in Figure 4.2				
	Main clock test	**LOCO test**	**Sub-clock test**	**MOCO test**	**HOCO test**
Derived from HOCO	Suitable	Suitable	Suitable	Suitable	Skip
Derived from LOCO	Suitable	Skip	Suitable	Suitable	Skip
Derived from MOCO	Suitable	Skip	Suitable	Skip	Skip
Derived from main clock	Skip	Skip	Suitable	Skip	Skip

4.1.4 LVD_Monitoring

This mechanism can detect faults affecting the propagation of the external Vcc supply. This safety mechanism complements the protection Ext_3 reported in section 2.

Fault detection is primarily done by the HW and complemented by a SW procedure, where the configuration of the LVD module is a key step.

The following strategy is recommended for the LVD configuration:

- Configure a reset generation only for voltage drops down to Vdet0 level, since this is the most severe condition (lowest voltage threshold), and a SW recovery through ISR cannot be guaranteed
- Configure Vdet1 and Vdet2 for interrupt generation in the case of voltage drop and rise conditions - one condition for Vdet1 (for example, drop) and opposite condition (for example, rise) for Vdet2
- Within the ISR, perform required actions to move to a safe state
- In order to minimize the impact of false positives caused by an interrupt generated without a real voltage drop condition, check within the ISR that the proper voltage change flags are set in the Voltage Monitoring 1/2 Circuit Status Registers (LVD1SR/LVD2SR)
- In order to avoid missing fault detection due to faults on interrupt generation, the SW should check periodically, with a polling strategy, the consistency of the status of the voltage change flags in the Voltage Monitoring 1(2) Circuit Status Registers (LVD1SR/LVD2SR).

The procedure for LVD configuration and SW test is reported in Figure 4.4. Note that in the case of voltage drops, misbehaviour of the MCU is possible and the execution of SW code (for example, ISR) cannot be guaranteed.

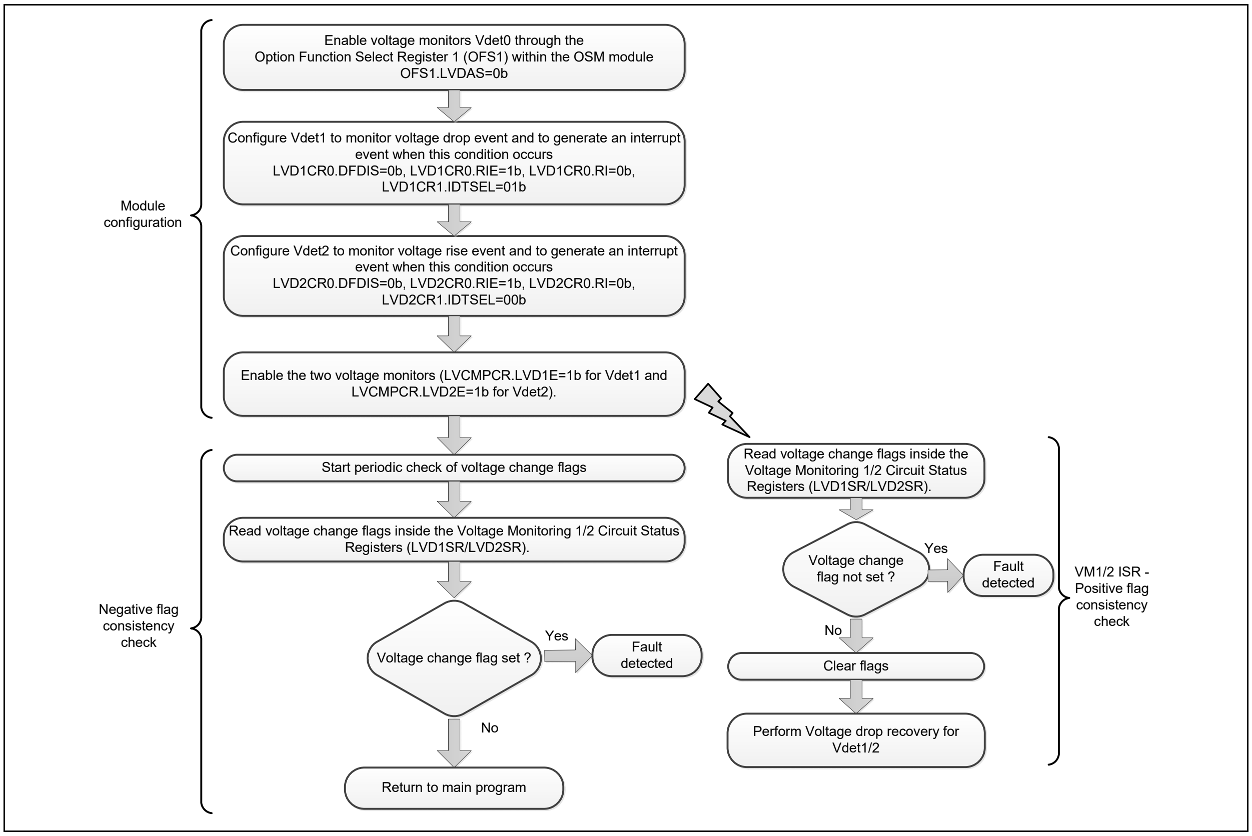

Figure 4.4 SW procedure for LVD_Monitoring

As part of the VM1/2 ISR processing, consider to treat as dangerous, the condition where the voltage drops below Vdet1 without rising again above Vdet2[5], with a voltage drift not so severe as to drop below Vdet0.

When Renesas LVD Configuration SW is used, Voltage Detection Circuit 1 is set to generate an NMI interrupt when Vcc drops below a specified level. See [3] for more details.

[5] for this MCU Vdet1=Vdet2

4.1.5 CGC_ClkStop

This mechanism can be used to detect oscillation stop conditions of the main clock.

Fault detection does not require any SW support besides setting OSTDCR.OSTDE = 1b.

Note that an Oscillation Stop Detection Interrupt can also be configured by setting OSTDCR.OSTDIE = 1b.

See section 9.5 of [1] for more details about the procedure to use this mechanism.

In order to minimize occurrence of false positive, it is recommended to adopt this mechanism while also using a configuration to monitor a clock derived from the main clock oscillator. If the main oscillator is not working, the mechanism should notify a failure, and if this is not the case, a false positive is present.

4.1.6 PWM_short_monitor

This mechanism can be used to detect faults causing overlapping of active phases of the PWM outputs that are supplied to the POEG module.

Fault detection is done by the HW asserting a flag when a fault is detected (see the Detection Flag for GPT or ACMPHS Output-Disable Request POEGGn.IOCF in Section 22.2.1 of HW Manual [1]).

Figure 4.5 shows the SW configuration and the relation to ISR.

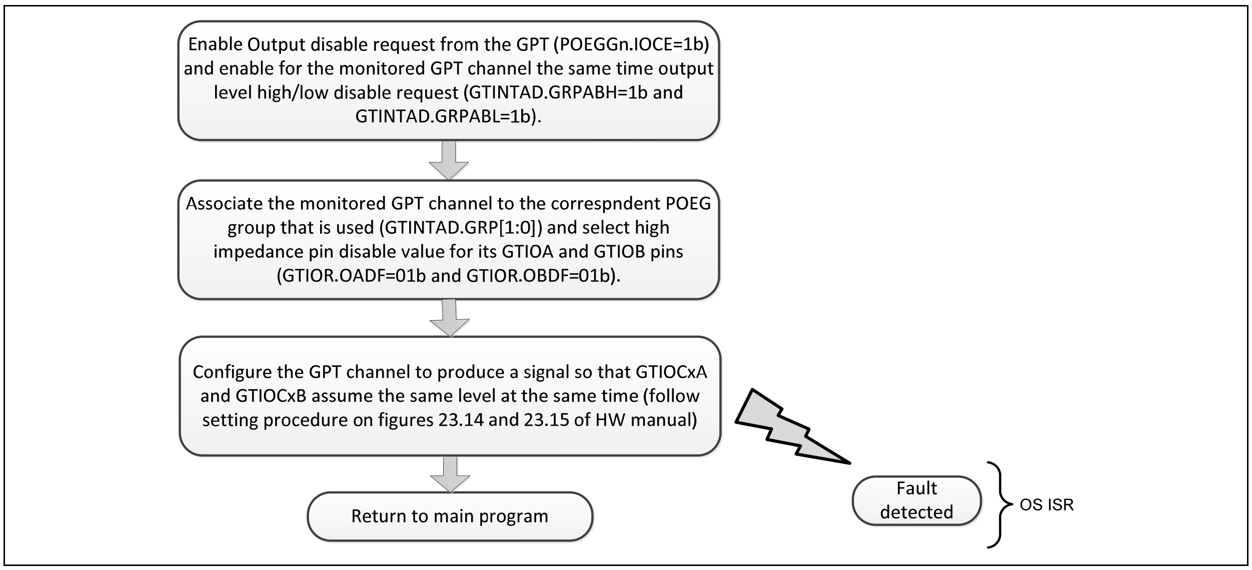

Figure 4.5 SW procedure for PWM_short_monitor

4.1.7 MSPMPU_monitor

This mechanism can be used for underflows/overflows of the main stack pointer.

The concept is based on creating an underflow/overflow of the main stack pointer and checking if a non-maskable interrupt is generated by the MPU. Figure 4.6 shows the test of an underflow of the main stack pointer, and Figure 4.7 shows the overflow test.

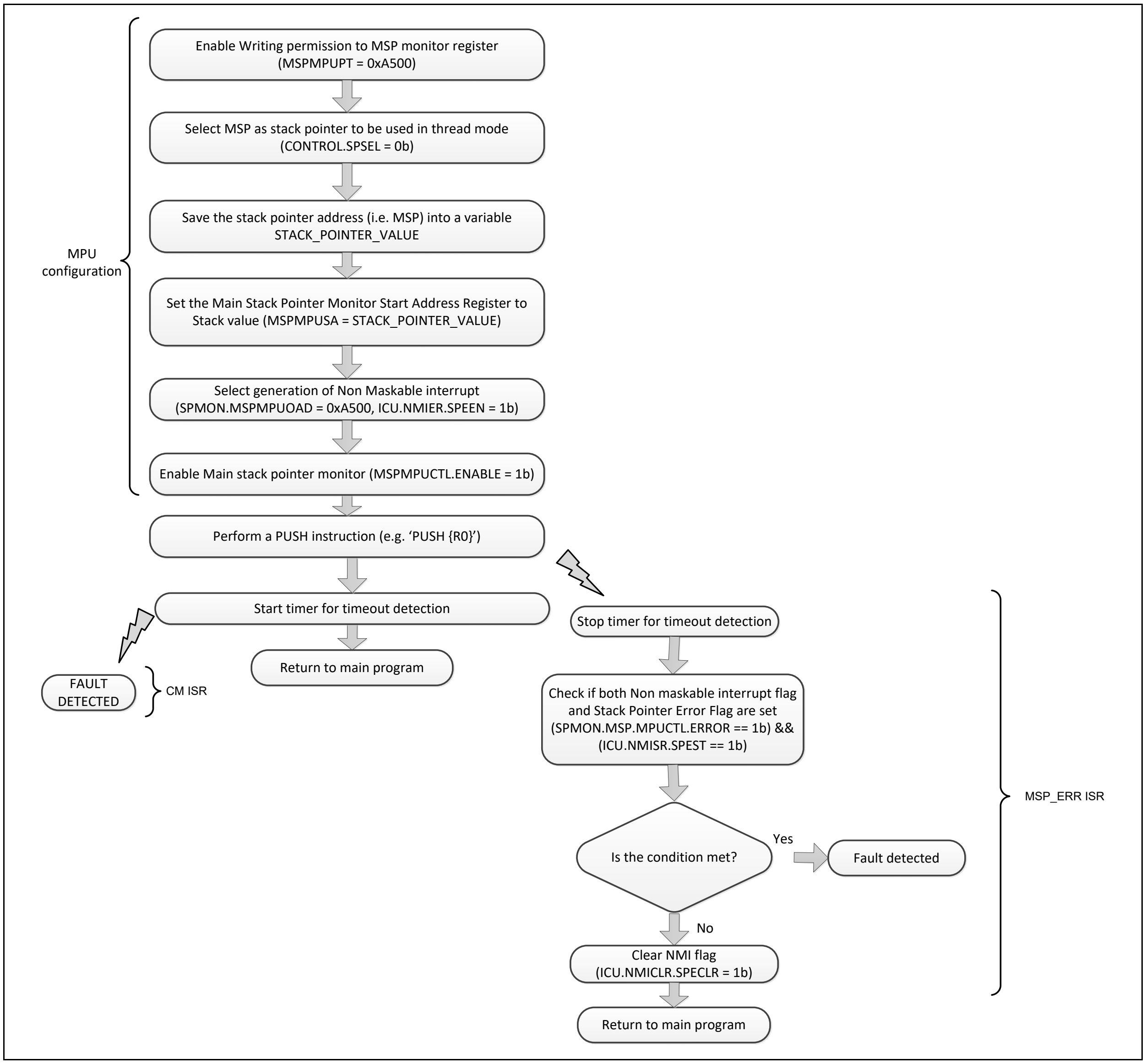

Figure 4.6 SW procedure for MSPMPU_monitor – underflow

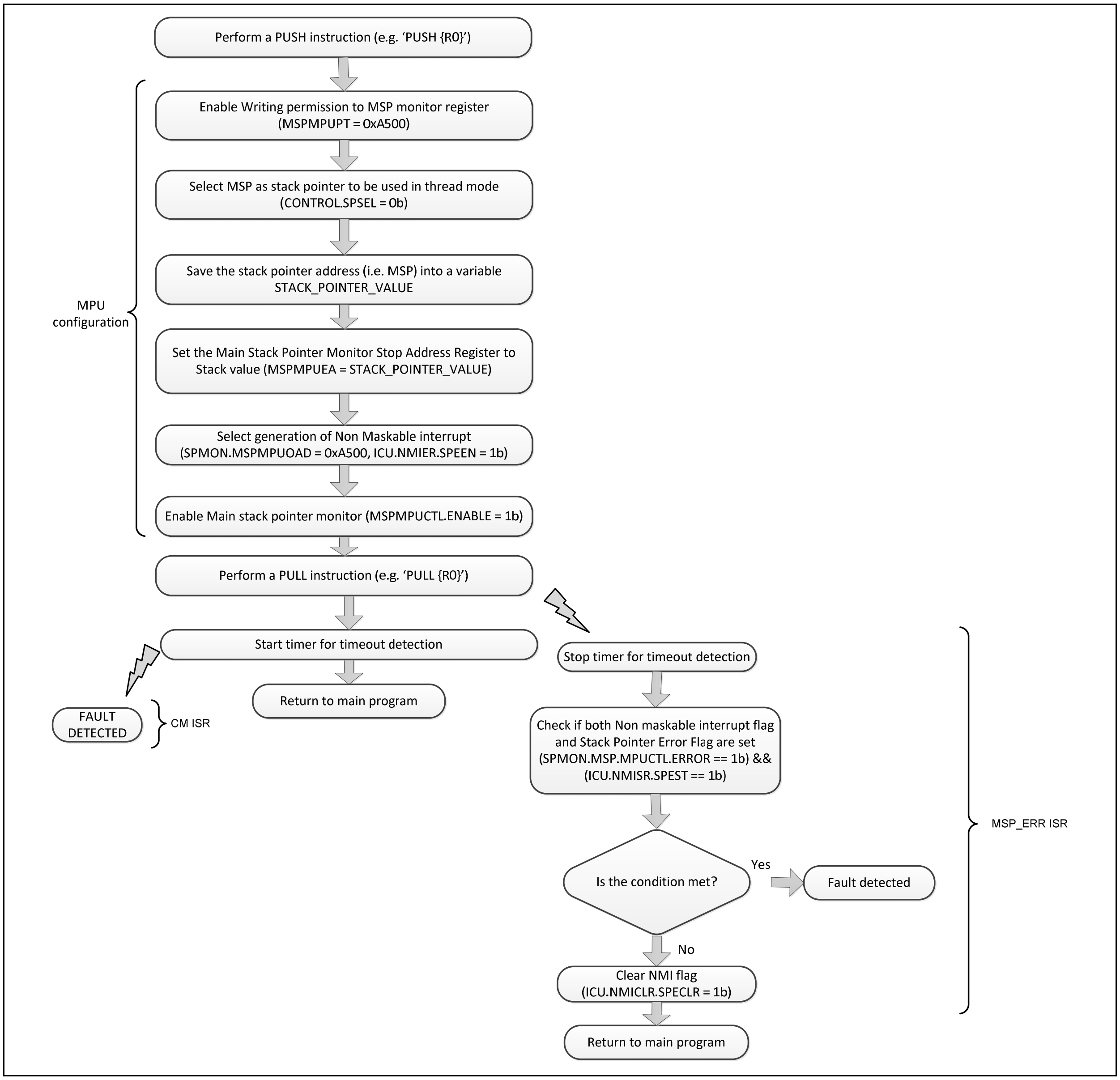

Figure 4.7 SW procedure for MSPMPU_monitor – overflow

4.1.8 PSPMPU_monitor

This mechanism can be used to detect underflows/overflows of the processor stack pointer.

The safety concept is the same as that of the MSPMPU_monitor, but is applied with the processor stack pointer as the object of the test (CONTROL.SPSEL = 1b).

See section 4.1.7 for more information.

4.2 Module level Safety Mechanisms

4.2.1 BSC_Monitoring

This mechanism can be used to detect faults affecting the internal buses.

Fault detection does not require SW support except for enabling the setting of the Ignore Error Responses bit in the Master Bus Control Register (BUSCMNT<master>.IERES = 0b, see section 15.3.7 of HW manual [1]).

It is also recommended to enable an interrupt request generation in order to notify when a bus error occurs, as shown in Figure 4.8.

Note: The execution of the error ISR is not guaranteed because faults on the buses might prevent a proper SW execution. However, this would be detected by the WDT (internal or external).

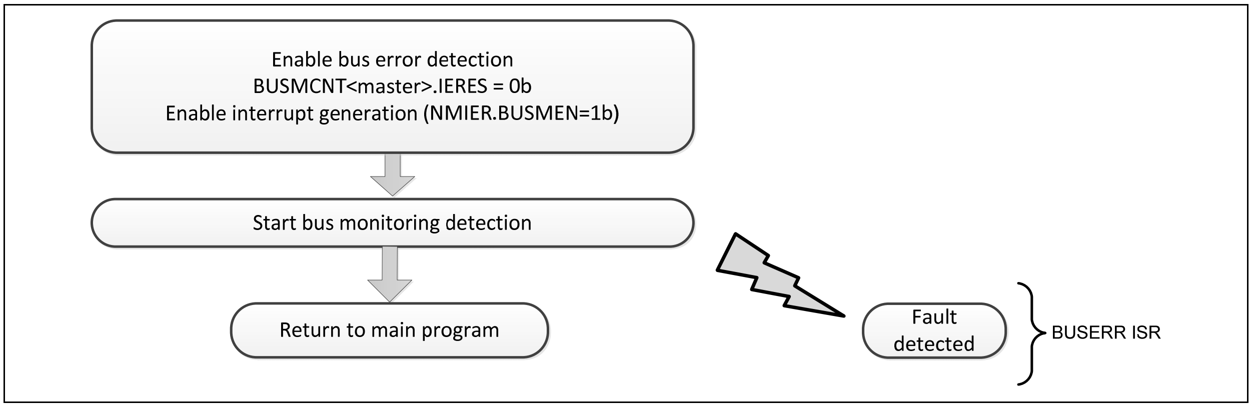

Figure 4.8 SW procedure for BSC_Monitoring

4.3 SW level Safety Mechanisms

4.3.1 Config_reg_read_back

This mechanism can be used to detect faults affecting the content of the configuration registers.

SW should periodically read back content of the configuration registers and check for possible corruption of their content.

A fault is detected when the actual content does not match the expected one.

SW procedure is reported in Figure 4.9.

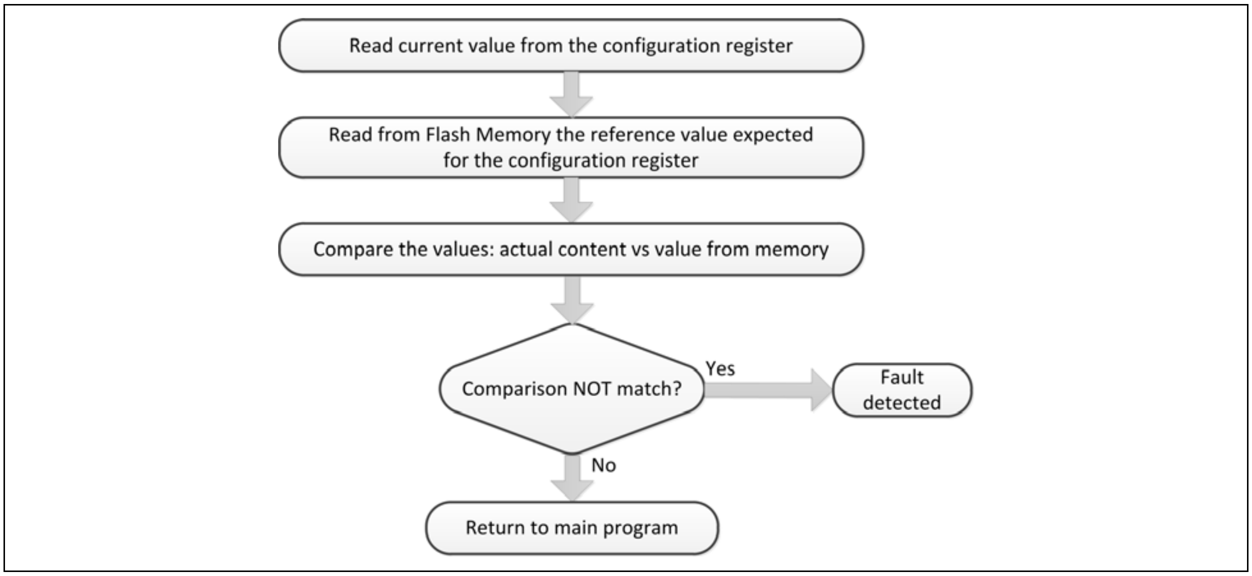

Figure 4.9 SW procedure for Config_reg_read_back

4.3.2 Write_verify

This mechanism can be used to detect faults affecting the update of the configuration registers/memory location from the SW.

When updating the content of the register/memory location, SW should read back the content of configuration registers/memory location and check for a possible corruption during the preceding write action.

Fault is detected when read back content does not match the expected content.

Note: This should not be considered as a standalone SW procedure but as a recommended practice to be included as part of the register configuration procedure.

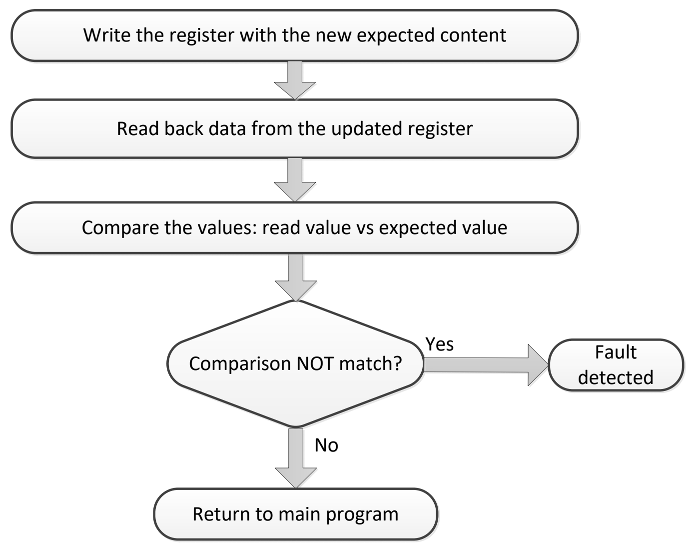

Figure 4.10 SW procedure for Write_verify

4.3.3 ICU_int_source_check

This mechanism can be used to detect faults leading to wrong interrupts being serviced.

Note: The flow provided in Figure 4.11 is recommended as a best practice to be embedded as part of each ISR procedure rather than a standalone mechanism.

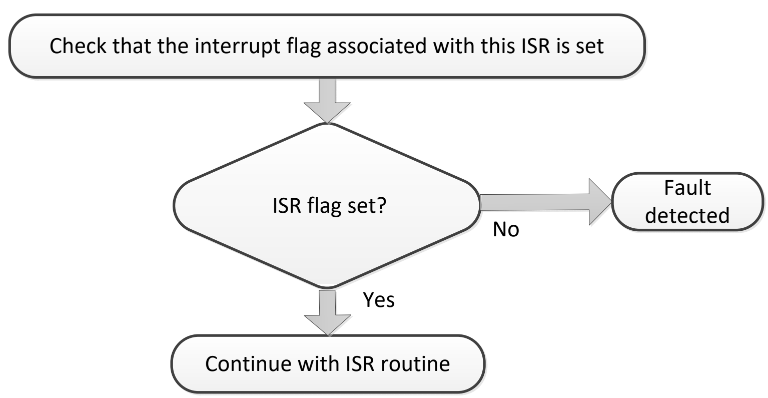

Figure 4.11 SW procedure for ICU_int_source_check

4.3.4 ICU_int_count

This mechanism can be used to detect faults affecting periodic interrupt requests.

Note: This not a standalone SW procedure, but should be used as part of the ISRs. The concept is based on the monitoring of periodic interrupt requests. Within a predefined application dependent period, a particular number of requests is expected, and if the actual number of requests is outside a plausible range, then a failure is detected.

For a given interrupt request to be monitored, the following SW actions are necessary:

a. Start detection of a timeout that is longer than one period of the periodic interrupt request.
b. Inside ISR, update the request counter.
c. Implement fault detection in the timeout expiration routine as reported in Figure 4.12 (regardless of whether the timeout is ISR based or not).

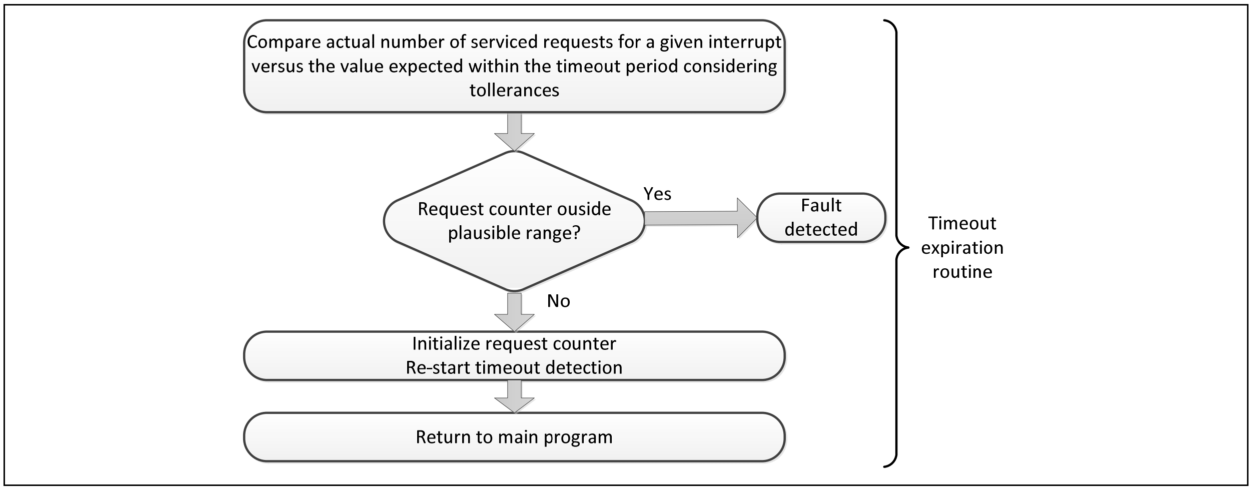

Figure 4.12 SW procedure for ICU_int_count

4.3.5 WDT_Counter_Monitoring

This mechanism can be used to detect faults affecting the WDT counter.

The fault detection principle is based on the cross checking of the WDT counter against another counter.

The SW test procedure to be performed is reported in Figure 4.13.

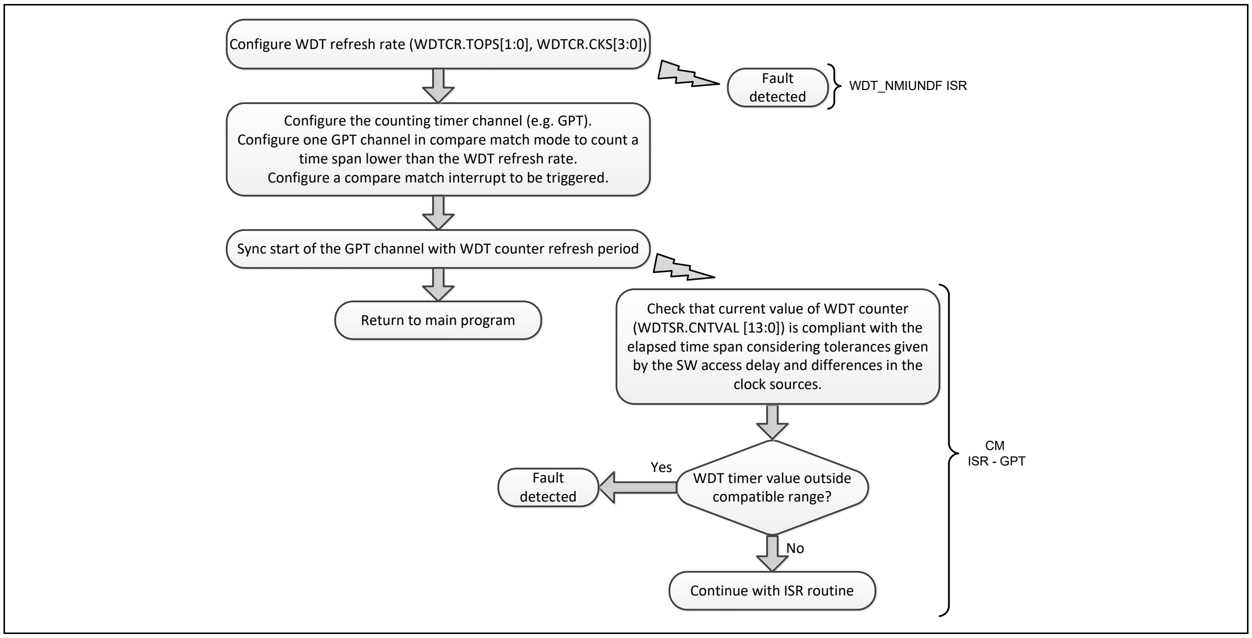

Figure 4.13 SW procedure for WDT_Counter_Monitoring

4.3.6 WDT_Expire

This mechanism can be used to detect faults affecting the MCU reset capability of the WDT.

This mechanism configures the WDT such that an MCU reset is triggered and monitors if the reset procedure is executed.

SW procedure is reported in Figure 4.14.

Note: It is necessary to memorize in Data Flash a variable such as MARK_WDT_TEST, which stores information to be used by the SW in the start-up procedure, to understand whether the reset signal was triggered by this mechanism or by other conditions.

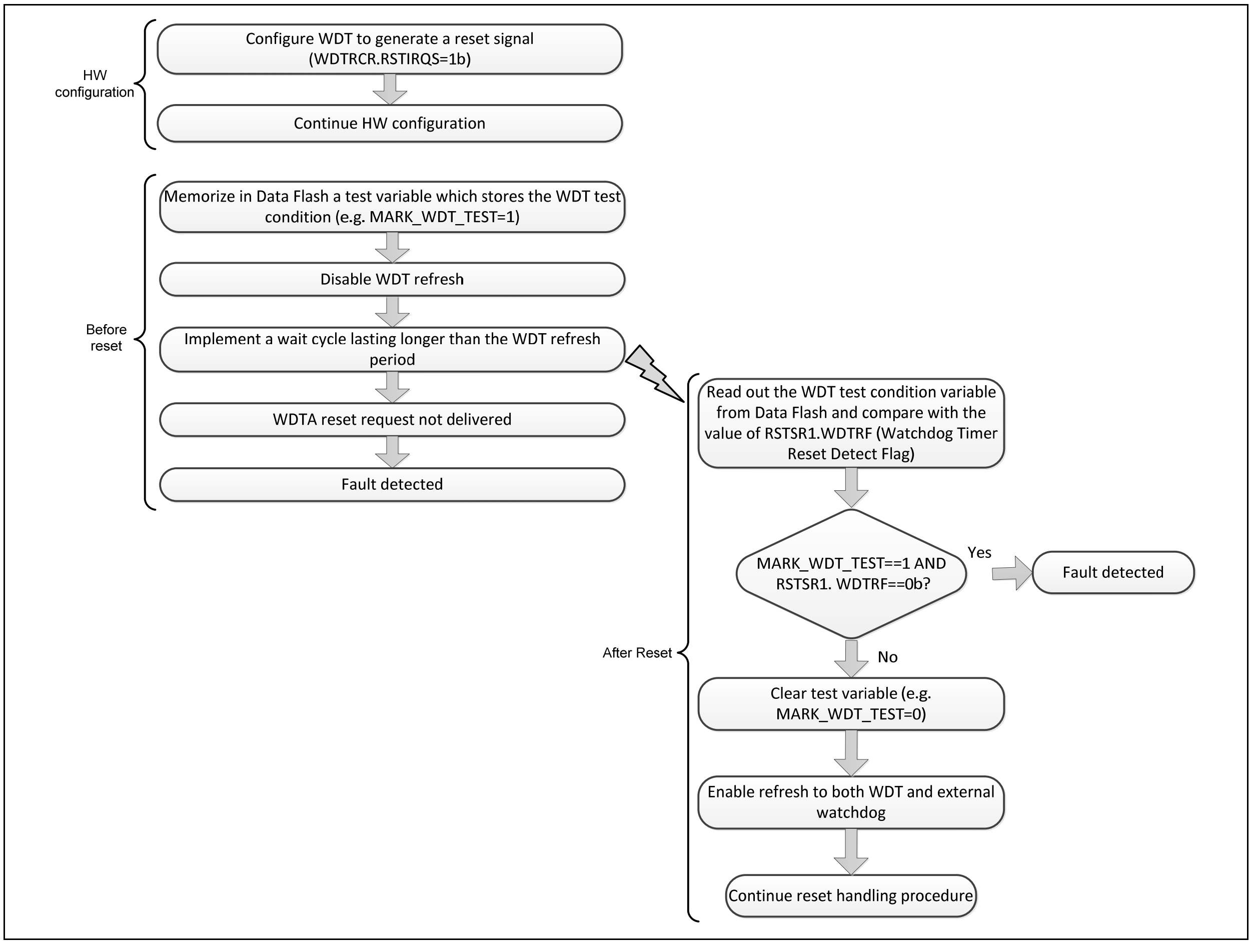

Figure 4.14 SW procedure for WDT_Expire

4.3.7 IWDT_Clock_Monitoring

This mechanism can be used to detect faults affecting the IWDT counter and/or the IWDT clock frequency.

The fault detection principle is based on the cross checking of the IWDT counter against another counter provided by a channel of an available timer unit.

The SW test procedure to be performed is reported in Figure 4.15.

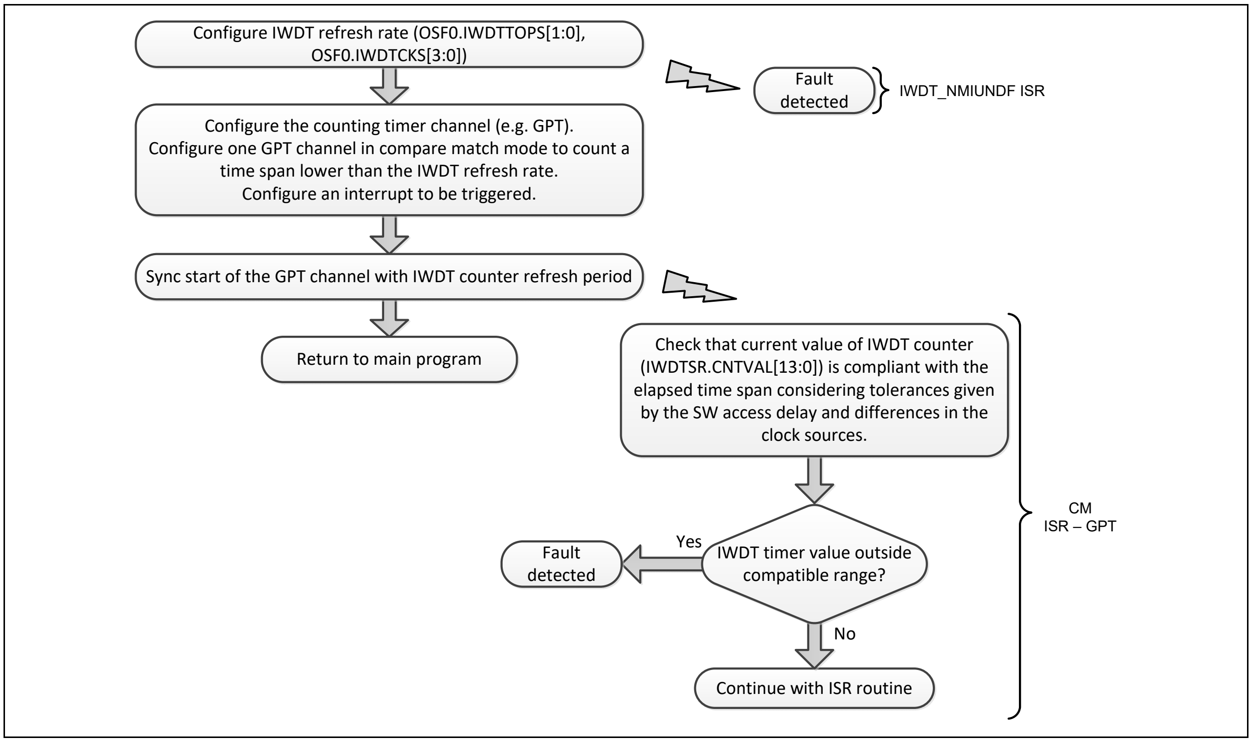

Figure 4.15 SW procedure for IWDT_Clock_Monitoring

4.3.8 IWDT_Expire

This mechanism can be used to detect faults affecting the MCU reset capability of the IWDT.

The concept of the safety mechanism is the same as WDT_Expire but applied on iWDT. See section 4.3.6 for more information.

The iWDT Management Software [3] from Renesas can be used for enabling the iWDT in order to perform this self-test. In addition, the SW can also be used to check whether the iWDT performed a reset. The SW procedure changes as shown in Figure 4.16.

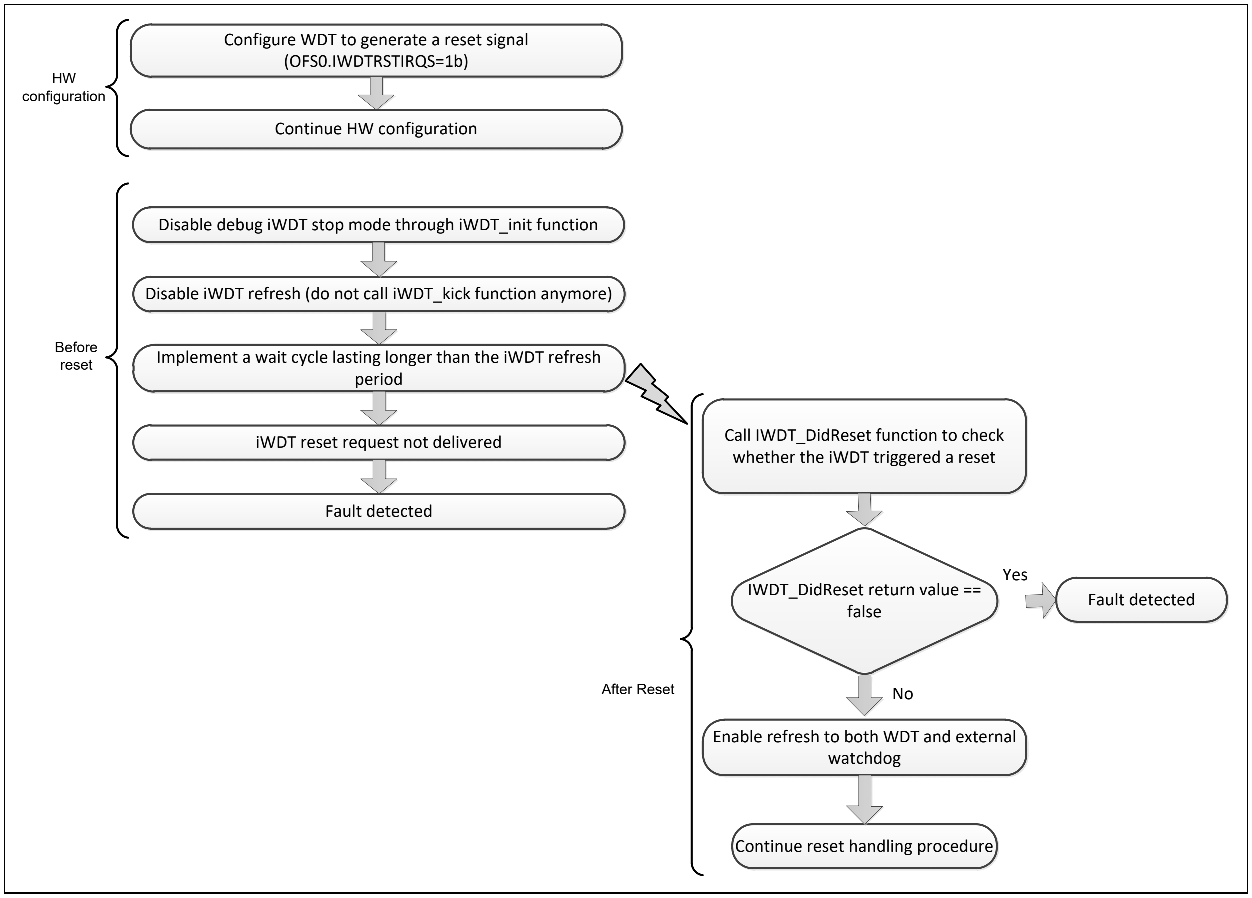

Figure 4.16 SW procedure for iWDT_Expire using iWDT Management Software

4.3.9 CPU_SW_Test

This mechanism can be used to detect faults affecting the CPU core.

This is an instruction-based CPU self-test, and a fault is detected if any of the tested CPU instructions provide a result different from the expected one, or if the communication with an external WDT is not correctly executed.

For more information on the CPU SW test, see [3] and for additional considerations on WDT usage, see section 2.1.

4.3.10 SRAM_SWTest

This mechanism can be used to detect faults in the on-chip high-speed SRAM.

The SW test applies fault detection algorithms (Extended March C- and WALPAT) on the memory content and returns a PASS/FAIL test result.

For more information on the SRAM test, see RAM SW Test in [3].

4.3.11 FlashMemory_Crc

This mechanism can be used to detect faults of the code/data Flash.

The SW test is CRC based. The CRC of the content of a given set of memory locations is calculated and compared against a reference value evaluated at compile time and is stored in memory during flashing of the device.

For more information on the Flash Memory test, see ROM SW Test in [3].

4.3.12 MMF_Crc

This mechanism can be used to detect faults of the code/data Memory Mirror Function (MMF).

The safety concept is the same as that of FlashMemory_Crcbut, applied to the Memory Mirror Function.

See section 4.3.11 for details.

4.3.13 SW_reset_check

This mechanism can be used to detect faults causing a SW reset request to be generated without a CPU write action. In other words, it checks if the last SW reset was issued in response to a SW action.

Note: The information provided in Figure 4.18 can be considered as a best practice to be embedded in the usual reset handling SW procedure rather than as a standalone mechanism.

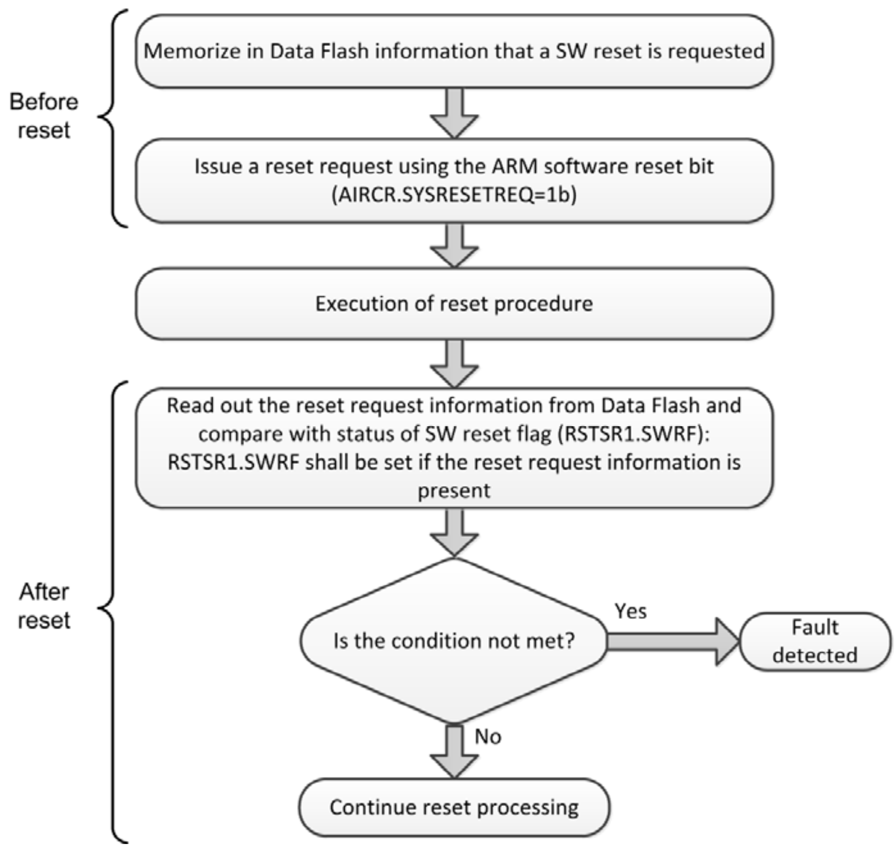

Figure 4.17 SW procedure for SW_reset_check

4.3.14 CRC_SwCrc

This mechanism can be used to detect faults affecting the CRC Calculator module.

The test concept is based on the comparison of the HW module elaboration against a SW implemented version, as shown in Figure 4.18.

Apply the procedure for the polynomial used in the application. If more than one polynomial is used (among the five supported by the HW module), repeat the procedure for all the polynomials used in the application.

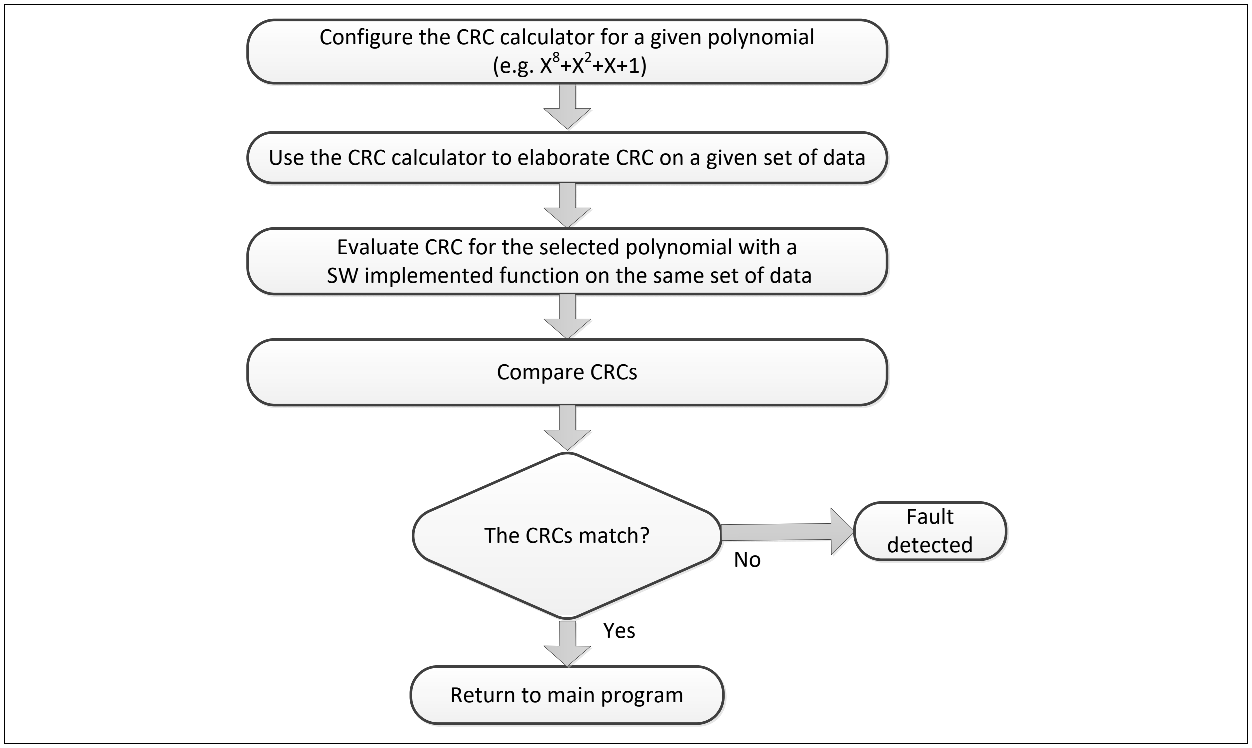

Figure 4.18 SW procedure for CRC_SwCrc

4.3.15 ADC14_TriggerDMA

This mechanism can be used to detect faults affecting the interaction between the ADC14 converter and the DMA (either DMAC or DTC).

The safety concept of this mechanism relies on the conversion of the internal reference voltage to trigger a DMA transfer of the converted value.

The SW procedure is shown in Figure **4.19**.

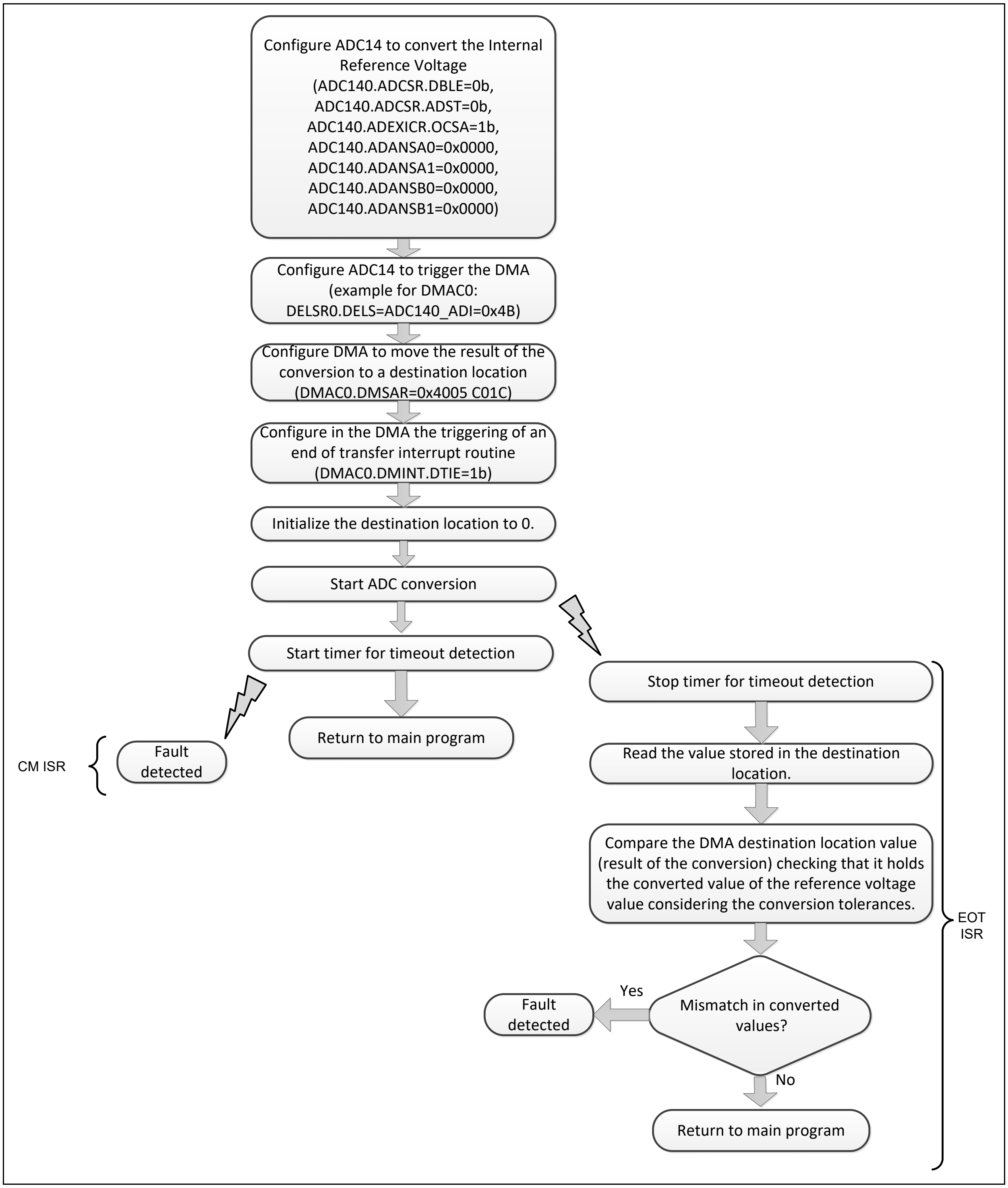

Figure 4.19 **SW procedure for ADC14_TriggerDMA**

4.3.16 ADC14_Comparator_SW

This mechanism can be used to detect faults affecting the embedded analog comparator of the analog to digital converter unit of the ADC14 module.

The safety concept of this mechanism is based on the possibility of converting an internal reference voltage, providing a known value through a dedicated channel of the ADC14.

The SW procedure is shown in Figure 4.20.

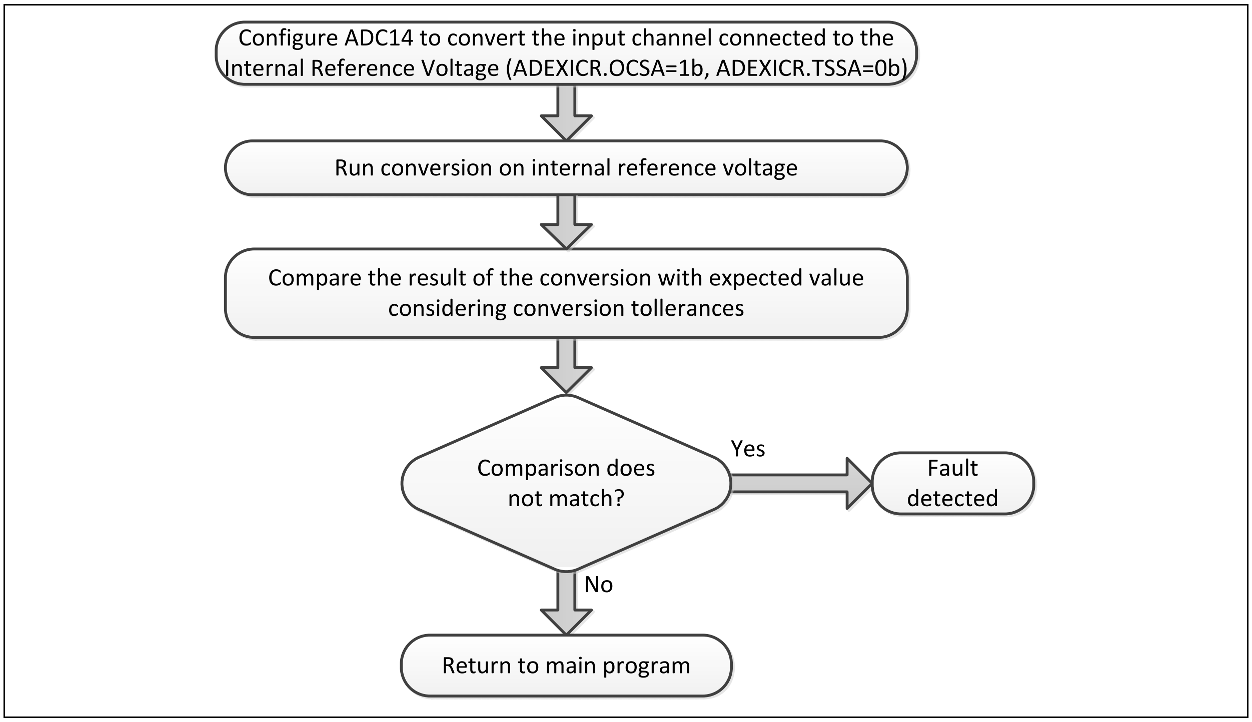

Figure 4.20 SW procedure for ADC14_SelfIntVolt

When Renesas ADC Comparator Software [3] is used, the SW procedure is as shown in Figure 4.21 depending on whether the test is intended at run-time or is a self-test.

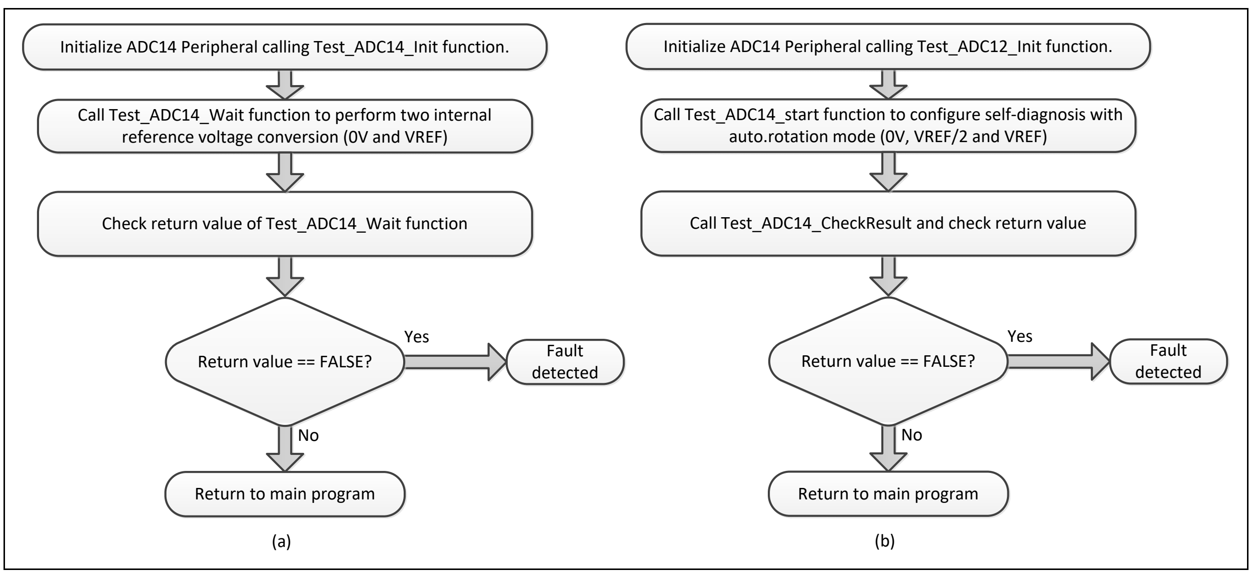

Figure 4.21 SW procedure for ADC14_Comparator_SW using ADC Comparator Software from Renesas: (a) self-test; (b) run-time

4.3.17 ADC14_TempSens

This mechanism can be used to protect the MCU against possible systematic failures such as operating temperature outside nominal range caused by environmental stress.

The safety concept of this mechanism is based on a range plausibility check. Depending on the application, a valid range of temperatures can be identified, and if the converted temperature value from the ADC14 is outside this range, then, a sensor fault is detected. This SW procedure is shown in Figure 4.22.

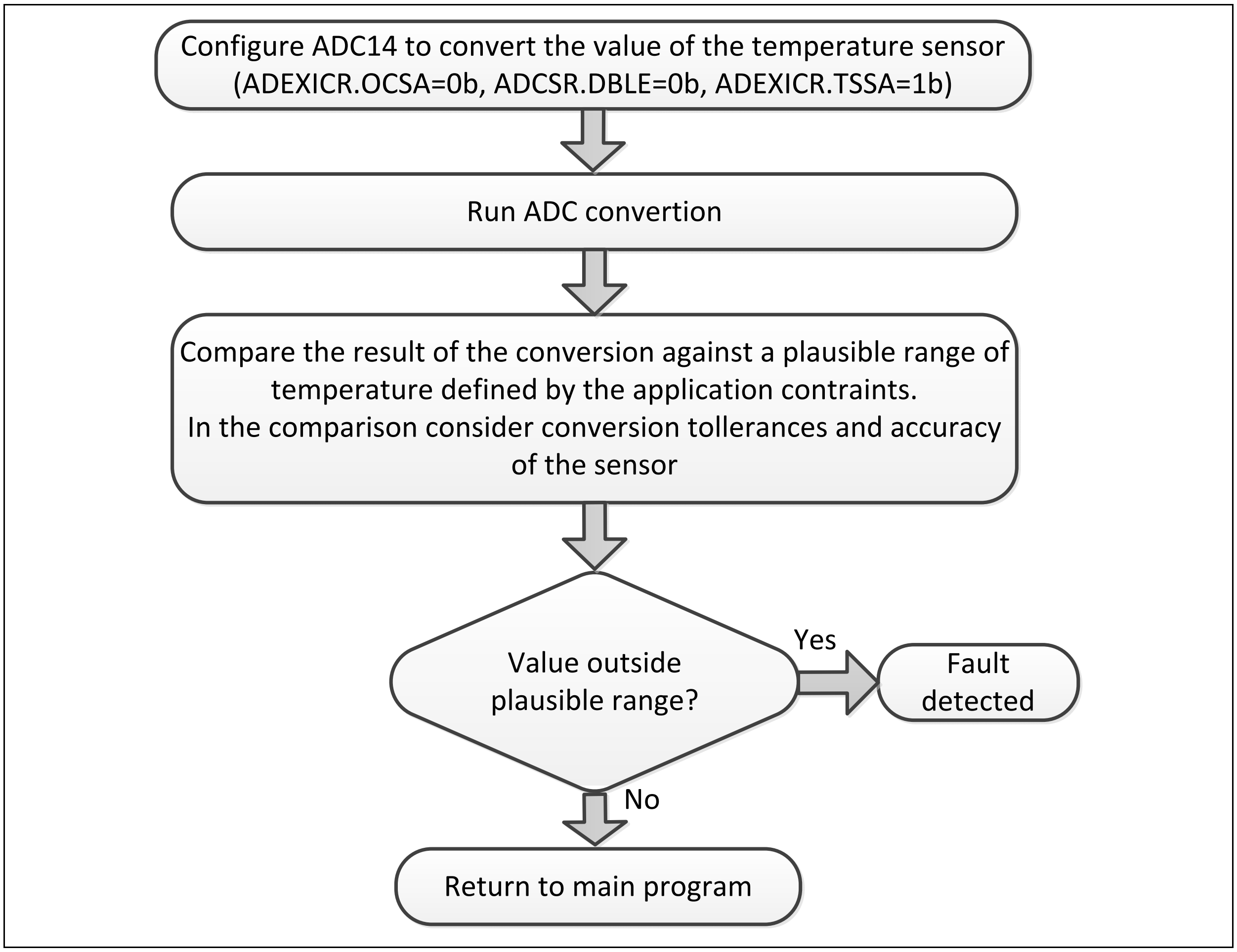

Figure 4.22 SW procedure for ADC14_TempSens

If TSN Management Software from Renesas [3] is used, the SW procedure is as shown in Figure 4.23.

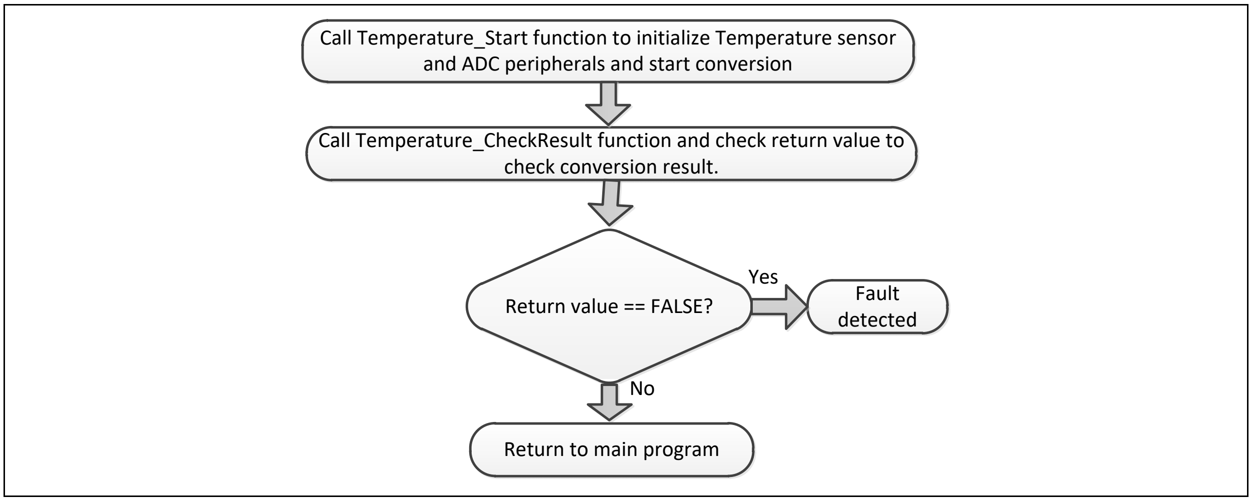

Figure 4.23 SW procedure for ADC14_TempSens with TSN Management Software from Renesas

4.3.18 GPT_DMAC_connection

This mechanism can be used to detect faults affecting the interaction between the General PWM Timer (GPT) and the DMAC.

The safety concept of this mechanism is based on activating the DMAC operation through any interrupt source (such as GPT32m.GTCCRA (m=0 to 9), compare matching the interrupt of each interested channel, and transferring known data.

The SW procedure is shown in Figure 4.24.

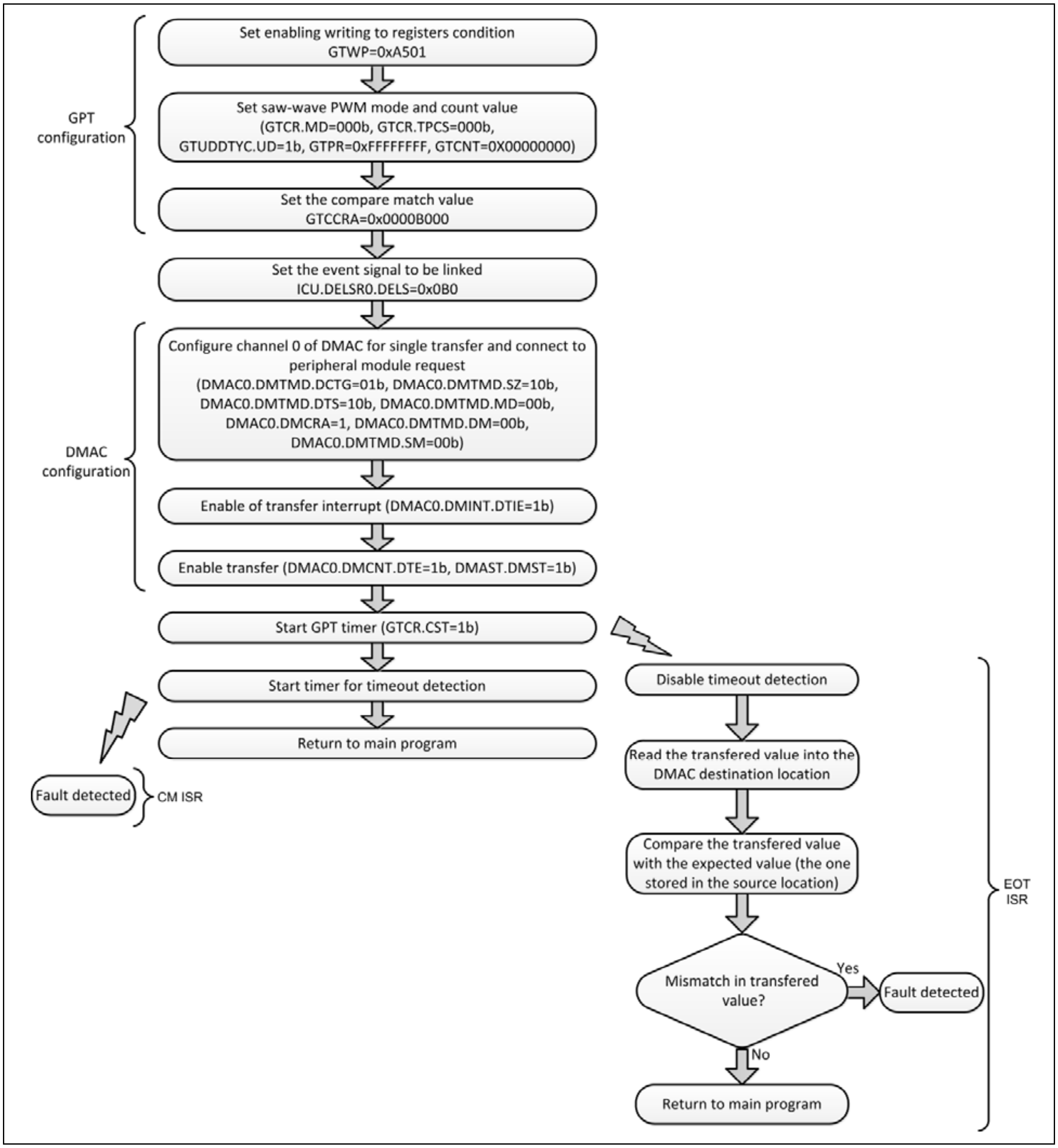

Figure 4.24 SW procedure for GPT_DMAC_connection

4.3.19 GPT_DTC_connection

This mechanism can be used to detect faults affecting the interaction between the General PWM Timer (GPT) and the Data Transfer Controller (DTC).

The safety concept is the same as GPT_DMAC_connection, but applied on the DTC connection. The activation of DTC operation is performed through any interrupt source (such as GPT32m_CCMPA (m = 0 to 9).

See section 4.3.19 for more information.

4.3.20 GPT_INT_generation

This mechanism can be used to detect faults affecting the generation of interrupts from the GPT.

The safety concept of this mechanism is to trigger a compare match interrupt by configuring one GPT channel and implementing a simple control flow monitoring as shown in Figure 4.25.

A timeout feature is also proposed to cope with faults on the interrupt logic, preventing the ISR from being correctly executed.

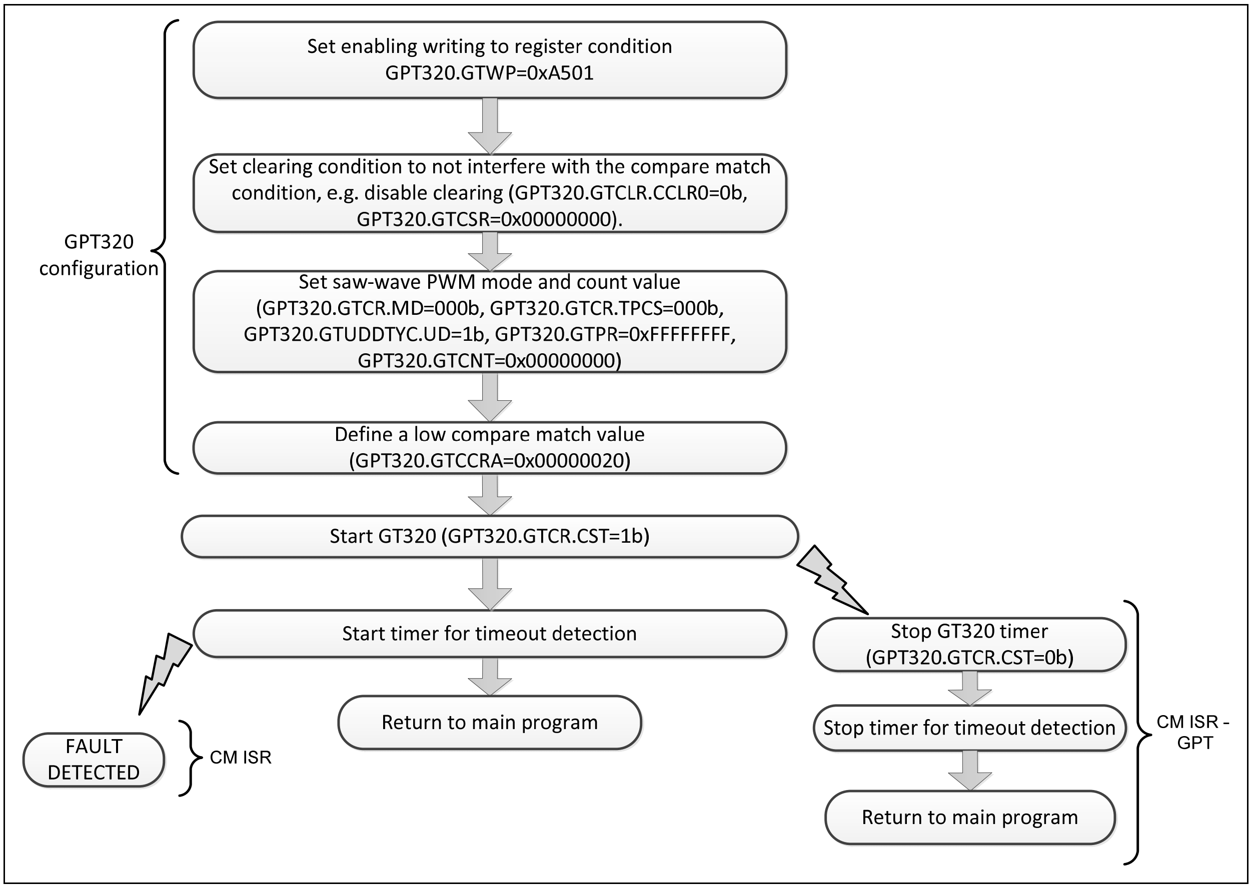

Figure 4.25 SW procedure for GPT_INT_generation

4.3.21 GPT_sync_chan

This mechanism can be used to detect faults affecting the GPT counting logic.

The safety concept is described in Figure 4.26 and Figure 4.27. Two strategies are proposed to compare the values of the counters of different channels that are started synchronously.

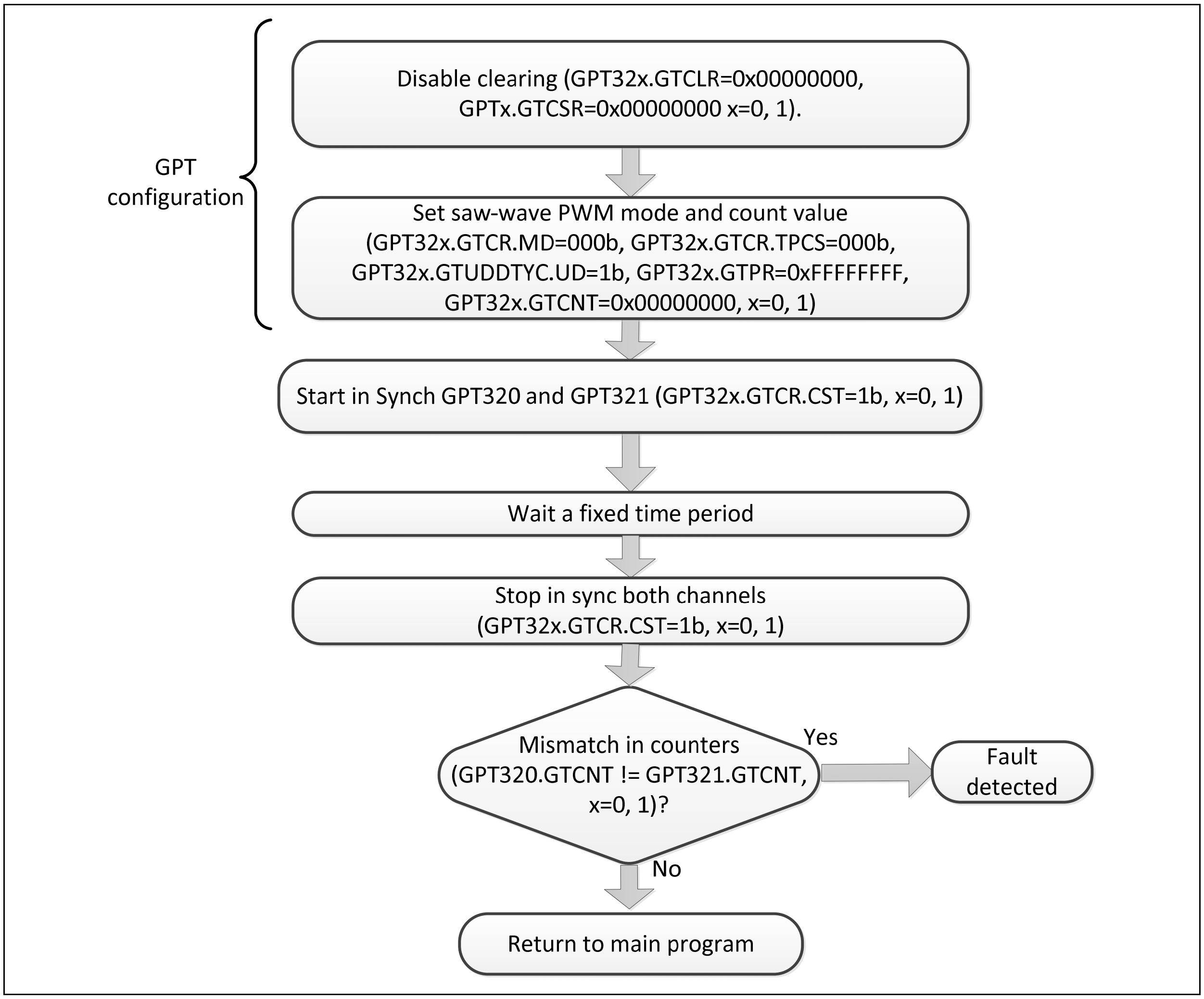

Figure 4.26 SW procedure 1 for GPT_sync_chan

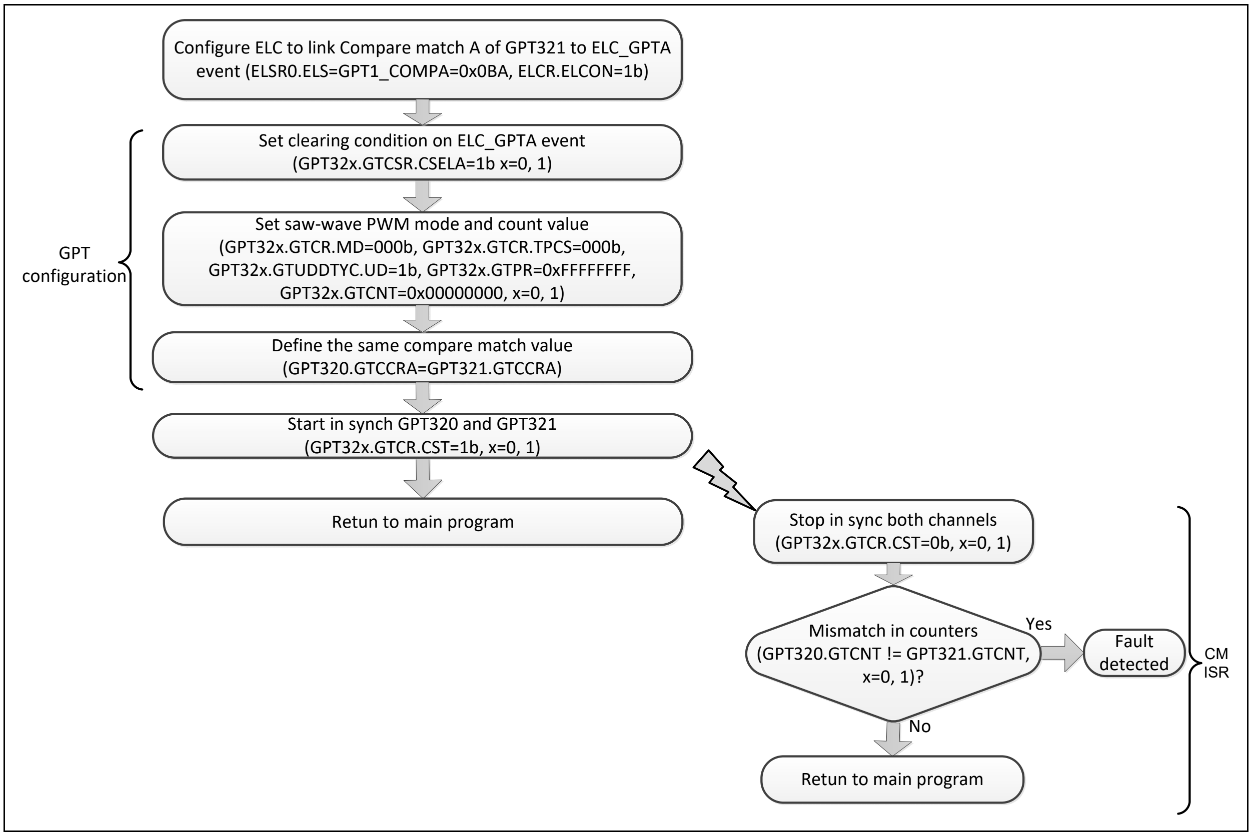

Figure 4.27 SW procedure 2 for GPT_sync_chan

4.3.22 AGT_GPT_freq_check

This mechanism can be used to detect faults of the timer count logic for either the 16-bit Asynchronous General Purpose Timer (AGT) or the General PWM Timer (GPT).

The safety concept of this mechanism is to use one timer to cross check the counting operation of the other one.

The SW procedure is described in Figure 4.28.

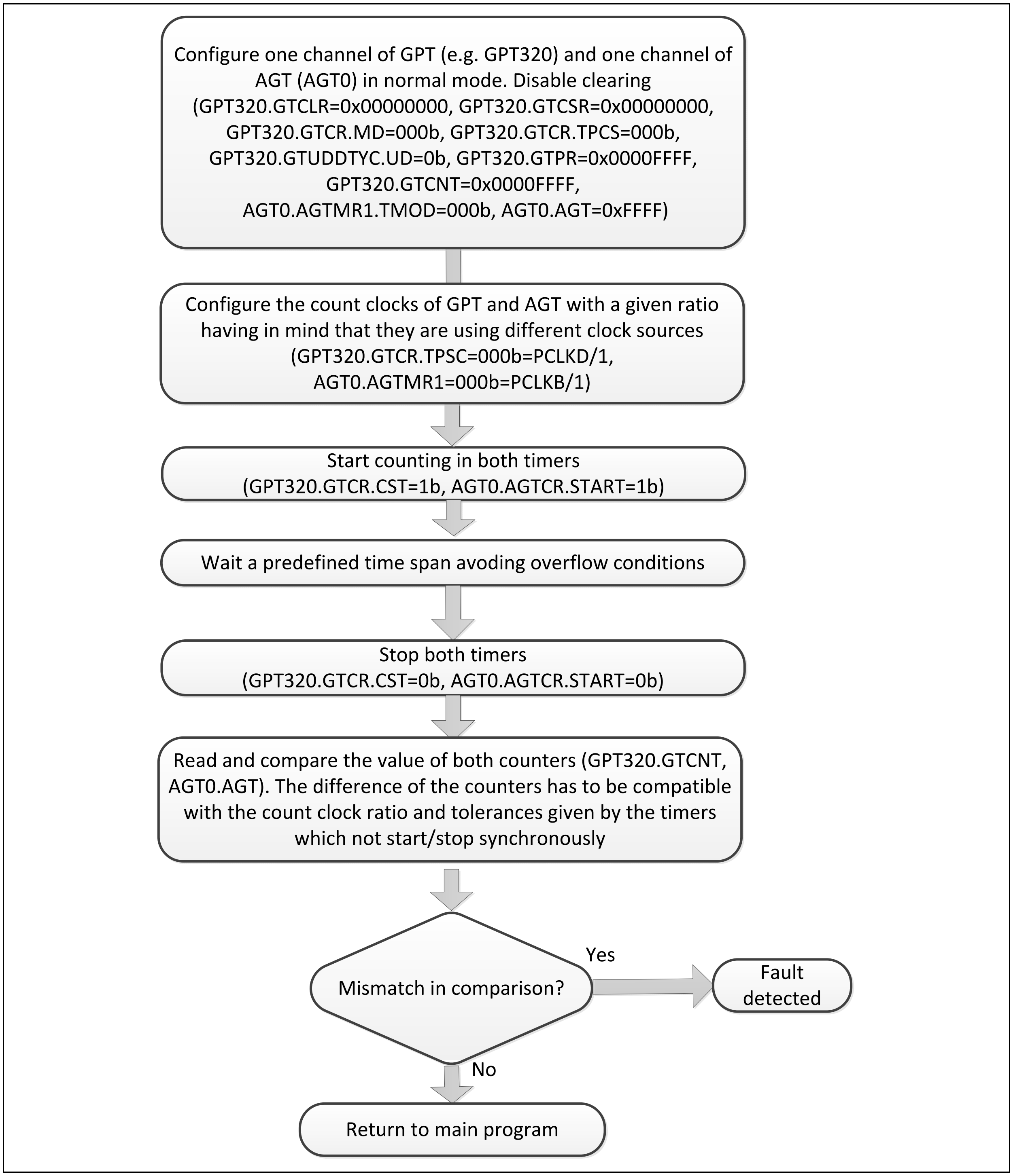

Figure 4.28 SW procedure for AGT_GPT_freq_check

4.3.23 GPT_redund_chan_input_capt

This mechanism can be used to detect faults of the input capture function of the General PWM Timer (GPT).

Safety concept of this mechanism is based on the usage of two redundant channels of the GPT configured in input capture mode and connected to the same source of external information to be monitored.

The test concept is the same as AGT_GPT_freq_check, but it is applied on two channels of GPT.

See section 4.3.24 for more information.

4.3.24 AGT_TriggerDTC

This mechanism can be used to detect faults affecting the interaction between the 16-bit Asynchronous General Purpose Timer (AGT) and Data Transfer Controller (DTC).

The safety concept is the same as that of the GPT_DMAC_connection, but applied with the AGT as the source trigger for the DMA connection, and the DTC as DMA.

See section 4.3.19 for more information.

4.3.25 AGT_TriggerDMAC

This mechanism can be used to detect faults affecting the interaction between the 16-bit Asynchronous General Purpose Timer (AGT) and DMA Controller (DMAC).

The safety concept is the same as that of the GPT_DMAC_connection, but applied with the AGT as the source trigger for the DMA connection.

See section 4.3.19 for more information.

4.3.26 AGT_TriggerADC

This mechanism can be used to detect faults affecting the interaction between the AGT and the ADC14 (14-bit ADC), particularly as it deals with the ADC activation triggered by the timer.

The safety concept is similar to the one described for GPT_ADC_connection, but using AGT and ADC14.

For more information, see section 4.3.19.

4.3.27 AGT_sync_chan

This mechanism can be used to detect faults affecting AGT counting logic.

The safety concept is similar to the one described for GPT_sync_chan, but applied to the two AGT channels instead.

See section 4.3.23 for more details.

4.3.28 AGT_INT_generation

This mechanism can be used to detect faults affecting the generation of interrupts by the AGT.

The safety concept of this mechanism is to trigger a compare match interrupt by configuring one AGT channel and implementing a simple control flow monitoring similar to what is described in GPT_INT_generation with AGT in the place of GPT.

See section 4.3.22 for more information.

4.3.29 DMAC_Crc

This mechanism can be used to detect faults involving transfers managed by the DMAC module.

The safety concept of this mechanism is based on the transfer of a set of data. The comparison between the original data and its transferred copy is done while compressing the information with a CRC.

The SW procedure to be performed is shown in Figure 4.29.

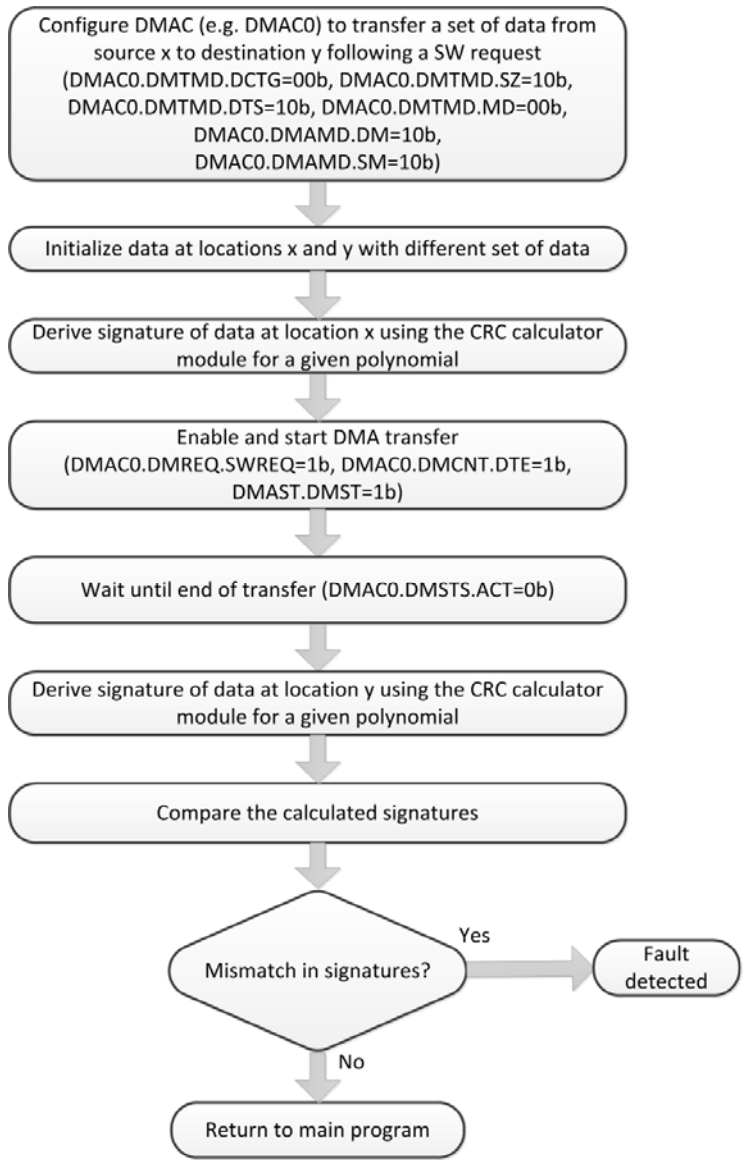

Figure 4.29 SW procedure for DMAC_Crc

(1) **Extension for "End-of-transfer interrupt generation"**

The safety concept can be extended to cover the logic related to the end-of-transfer interrupt generation.

Evaluation of signatures and their comparison is moved inside ISR, and a timeout feature is also added as shown in Figure 4.30.

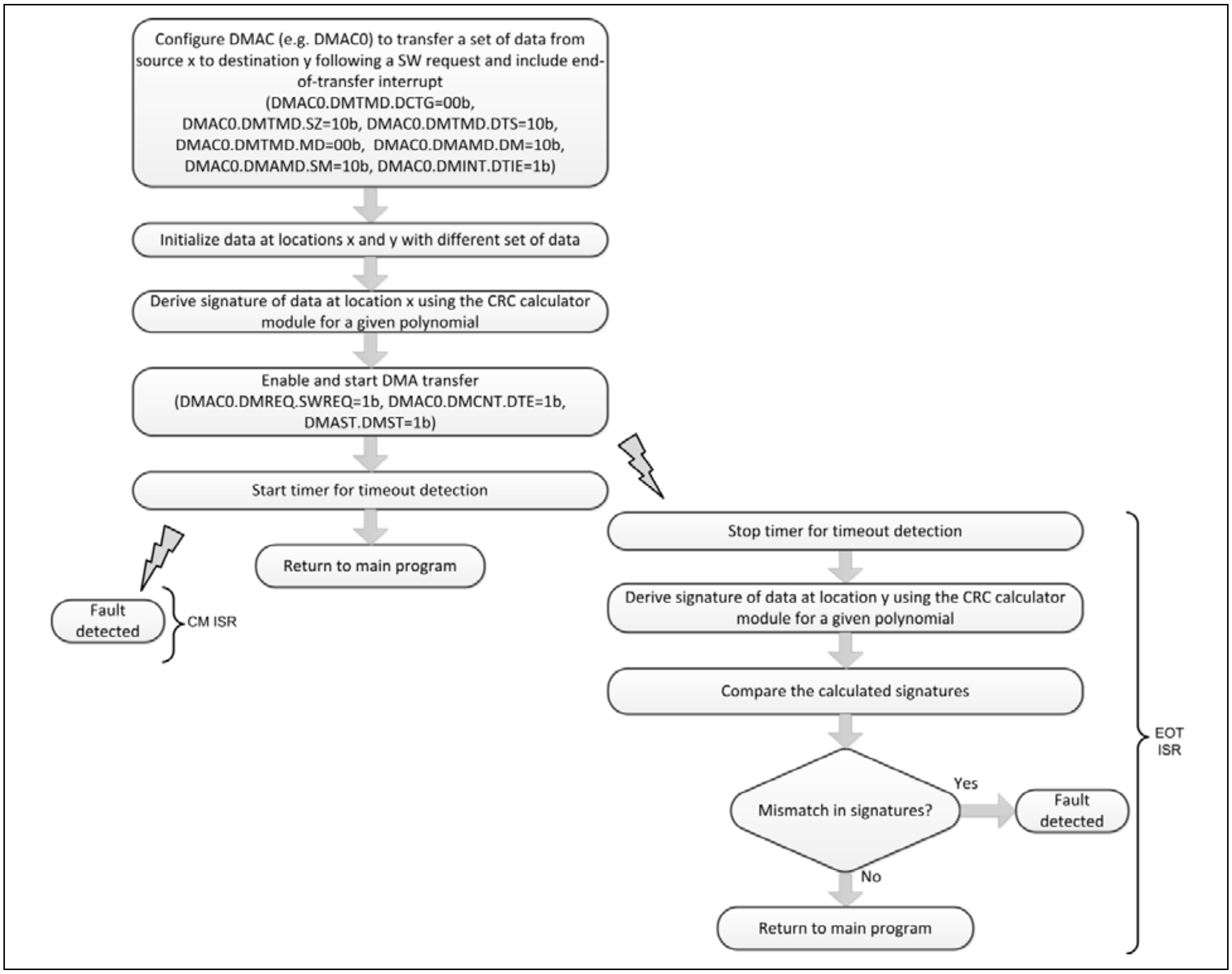

Figure 4.30 SW procedure for DMAC_Crc – Extension for EOT interrupt

4.3.30 DMAC_TriggerISR

This mechanism can be used to detect faults affecting the interaction between the DMAC and the interrupt controller ICU, particularly, the possibility for the ICU to trigger a DMAC transfer.

The safety concept of this mechanism is based on a simple control flow monitoring principle such that the execution of the proper order of SW routines is monitored as shown in Figure 4.31 through a timeout. The timeout will not expire (CM ISR not be entered) if the EOT ISR is correctly executed.

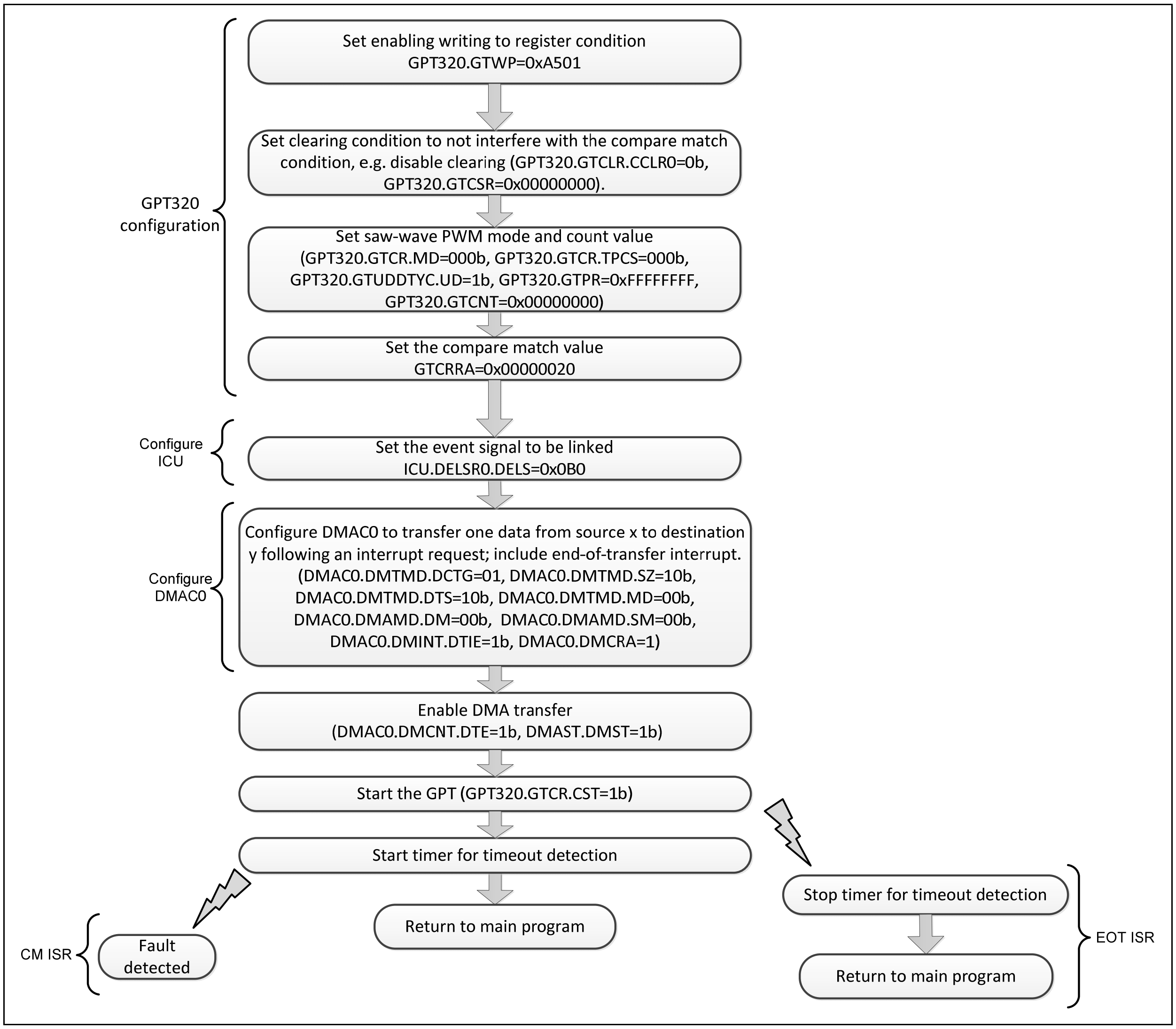

Figure 4.31 SW procedure for DMACA_TriggerISR

See section 17.3.4 of [1] for more information about DMACA activation through the interrupt controller.

4.3.31 DTC_Crc

This mechanism can be used to detect faults causing misbehavior of the DTC, while transferring data.

The safety concept of this mechanism is the same as that of the DMAC_Crc, but as applied to the DTC module.

See section 4.3.31 for more information.

Note: For the SW activation of the transfer, Event Link Software Event Generation Register n (ELSEGRn) (n = 0, 1) must be properly set up in combination with the association of the corresponding ELC event to a dedicated IELSRn register, where DTC activation is enabled (for example, ICU.IELSR0.IELS = 0x098 = ELC_SWEVT0 or ICU.IELSR0.IELS = 0x098 = ELC_SWEVT1 and ICU.IELSR0.DTCE = 1b).

4.3.32 DTC_TriggerISR

This mechanism can be used to detect faults effecting the interaction between the DTC and the interrupt controller ICU, particularly, the possibility for the ICU to trigger a DTC transfer.

The safety concept of this mechanism is based on the simple control flow monitoring, following the same principle described in section 4.3.32 for the DMAC_TriggerISR.

See section 18.3 of [1] for more information about DTC activation through the interrupt controller.

4.3.33 MPU_region_check

This mechanism can be used to detect faults affecting the MPU.

The safety concept is based on a simple control flow monitoring. The procedure creates the conditions to trigger an access exception, and the execution of the exception handler is recorded by the means of a SW variable.

The SW procedure to check the protection of a generic region n is shown in Figure 4.32. The procedure has to be repeated for all regions from n = 0 to n = 7 (if used by the application).

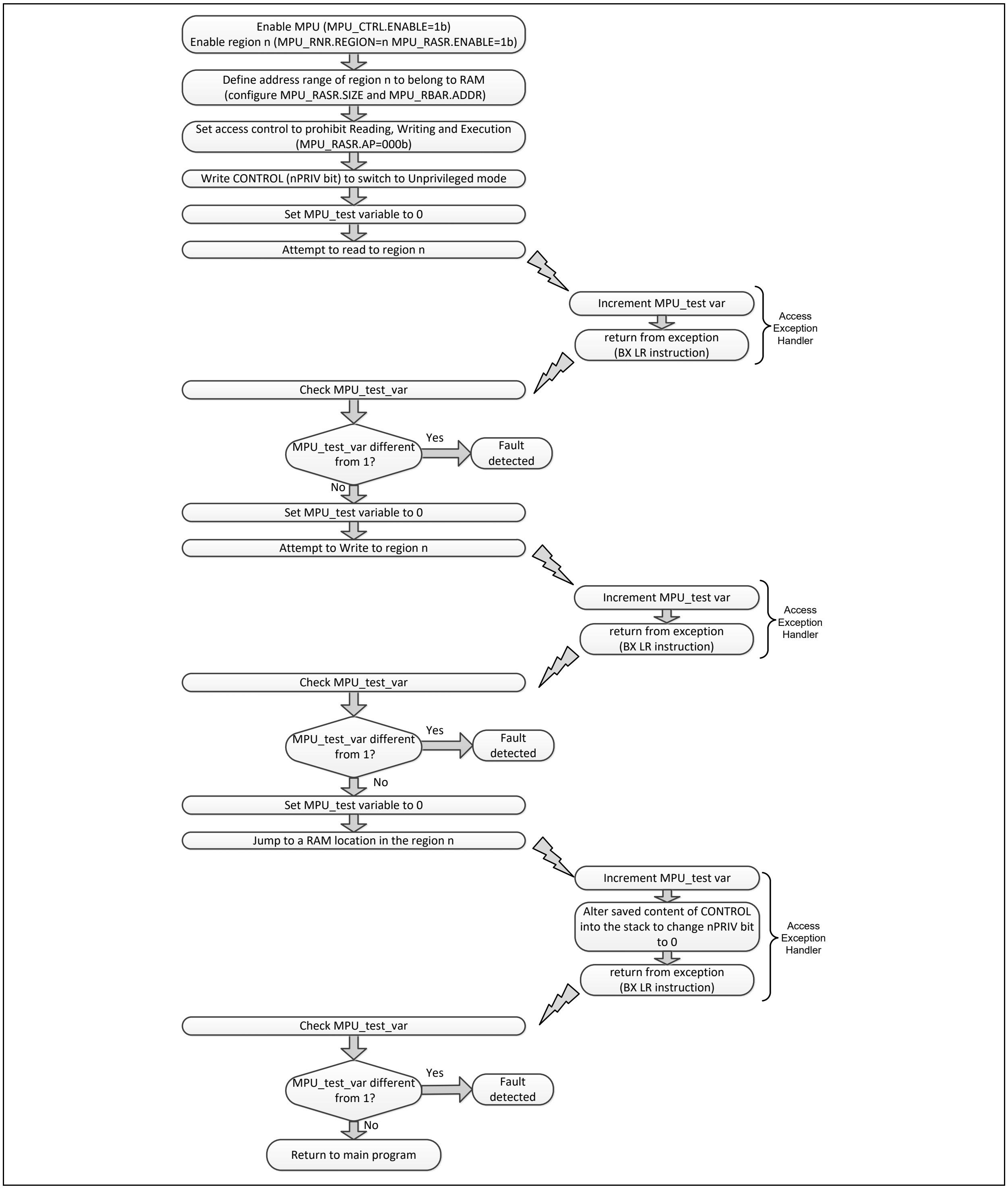

Figure 4.32 SW procedure for MPU_region_check for region n, n=0 to 7

Note: Misbehaviors of the MPU resulting in blocking valid accesses and preventing the normal SW execution are not addressed by this procedure. In such cases, the external WDT is effective and provides additional protection.

4.3.34 KINT_triggerISR

This mechanism can be used to detect faults affecting the Key Interrupt Function.

The safety concept is based on triggering a KEY_INTKR interrupt and checking if the interrupt is effectively produced.

The SW procedure is shown in Figure 4.33.

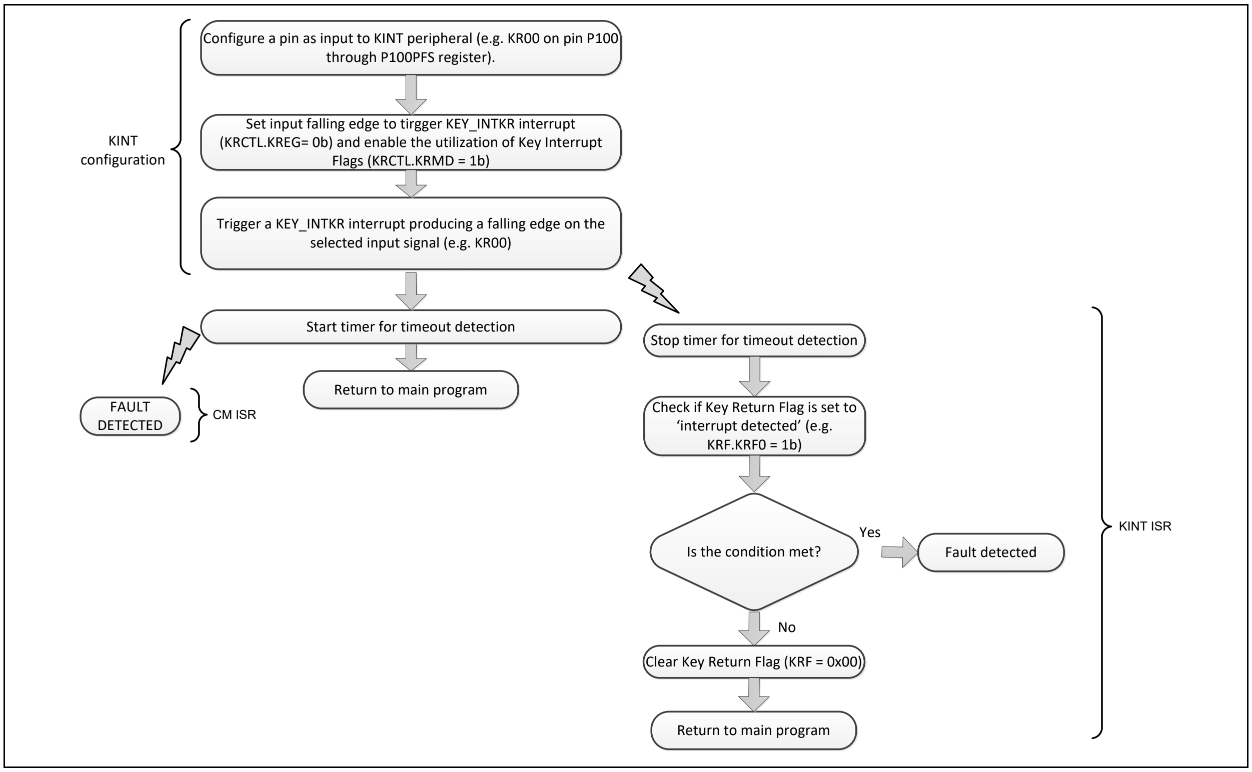

Figure 4.33 SW procedure for KINT_triggerISRv

Note: This procedure can be applied to any Key Interrupt Function input signal, that is, from KR00 to KRM07.

4.3.35 SDHI_DMAC_triggerISR

This mechanism can be used to detect faults affecting the SD/MMC Host Interface (SDHI).

The safety concept is based on defining two double word variables, SDHI_TEST_WRITE_VARIABLE and SDHI_TEST_READ_VARIABLE, and initializing SDHI_TEST_WRITE_VARIABLE with a predefined value. Then, SDHI_TEST_WRITE_VARIABLE is written to the SD/MMC card, exploiting the DMAC and SDHI peripherals. The variable is then read back from SD/MMC and the read value is stored in the SDHI_TEST_READ_VARIABLE. Finally, the signatures of these two variables are evaluated through the CRC peripheral and compared to check for mismatches.

Note: This test requires an unused 32-bit memory allocation of the SD/MMC card to store the SDHI_TEST_WRITE_VARIABLE variable.

The SW procedure is shown in Figure 4.34.

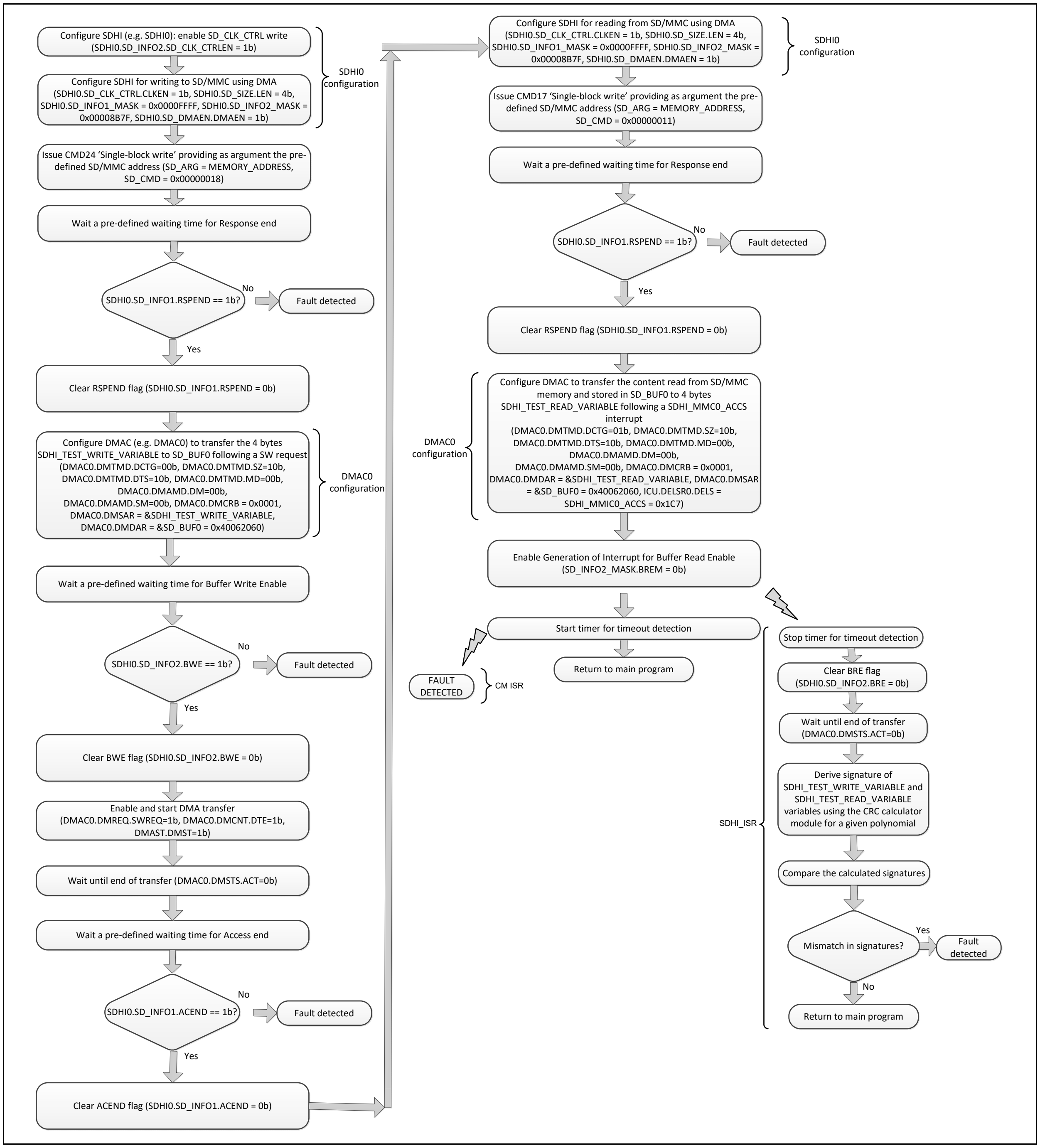

Figure 4.34 SW procedure for SDHI_DMAC_triggerISR

4.3.36 ACMPHS_triggerISR

This mechanism can be used to detect faults affecting the High-Speed Analog Comparator (ACMPHS).

The safety concept is based on providing two voltage outputs through the Digital/Analog Converter (DAC), which are compared with each other. As a first step, the input voltage value provided to ACMPHS (that is, DA1) is lower than the reference value (that is, DA0). In the second step, the input value is higher and an ACMPHS interrupt is generated.

The SW procedure is shown in Figure 4.35.

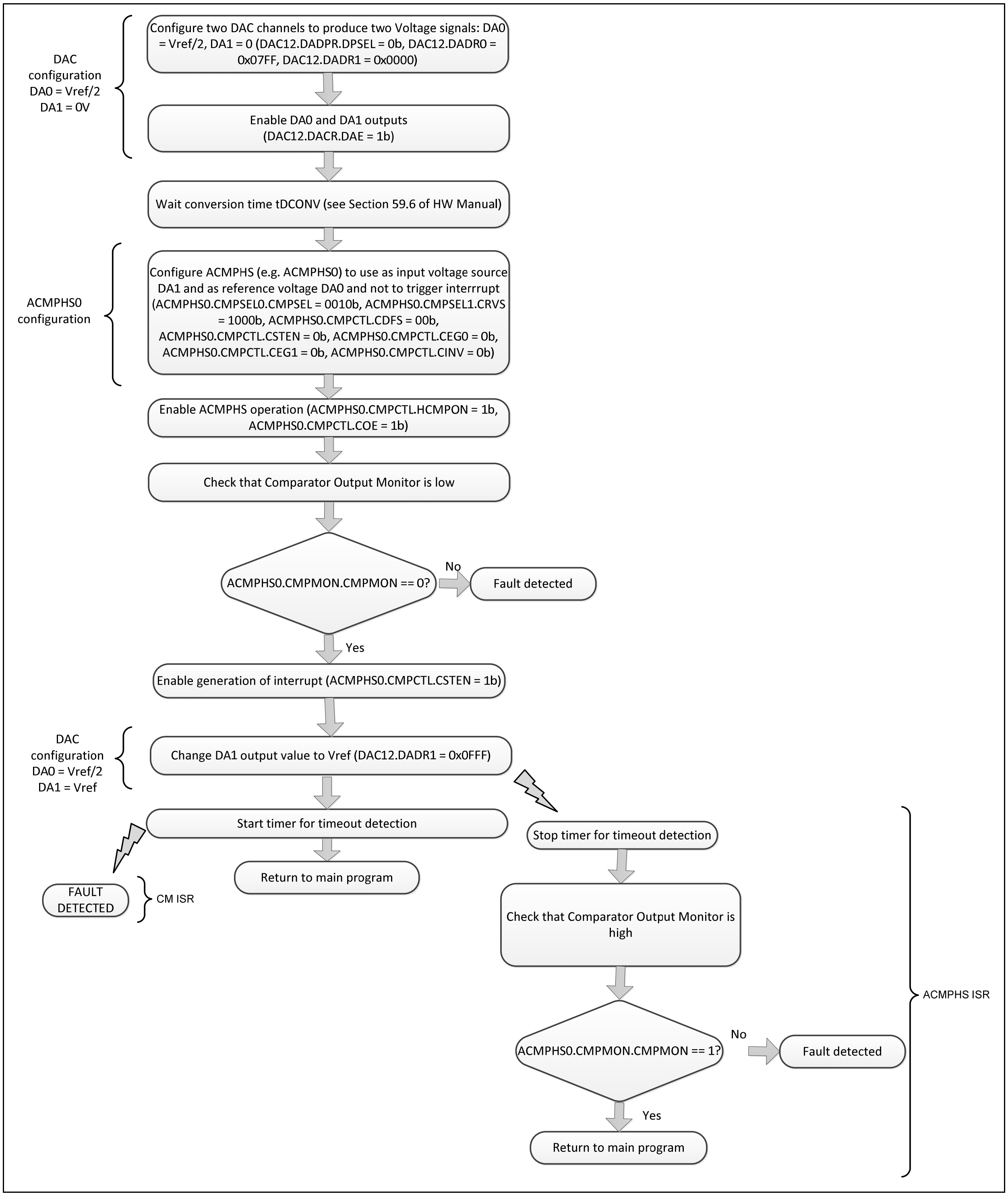

Figure 4.35 SW procedure for ACMPHS_triggerISR

4.3.37 DOC_check

This mechanism can be used to detect faults causing misbehavior of the DOC module.

The safety concept of this mechanism is based on the SW diagnostic of the DOC's three functionalities - data comparison, data addition, and data subtraction. Particularly, the mechanism exercises the DOC with a set of known data and compares the results against the expected ones.

The full procedure has four steps.

The first step checks for the correct execution of data comparison. The procedure, shown in Figure 4.36, has to be repeated for different sets of input data, including the set shown in Table 4.2.

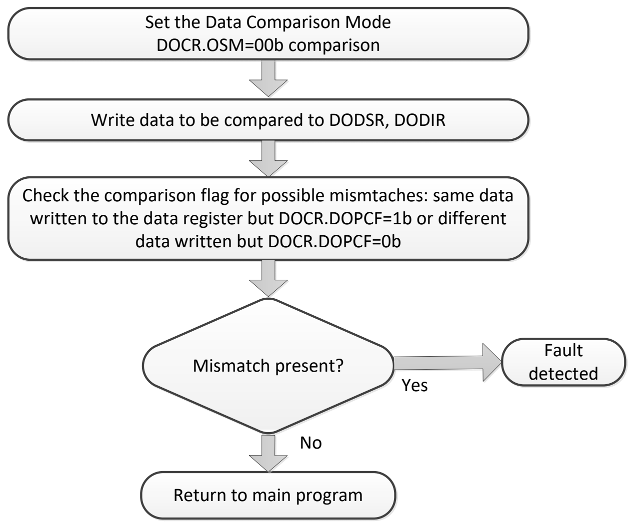

Figure 4.36 Basic procedure for the diagnostic of data comparison mode for DOC

Table 4.2 Minimum set of input data for data comparison mode

DODSR	DODIR	DOCR.DCSEL
AAAA	AAAA	0
5555	5555	0
AAAA	5555	0
5555	AAAA	0
AAAA	AAAA	1
5555	5555	1
AAAA	5555	1
5555	AAAA	1

The second step checks for the correct execution of data addition. The procedure, shown in Figure 4.37, has to be repeated for different sets of input data such that all input bits of both operands and output bits of the result are toggled at least once.

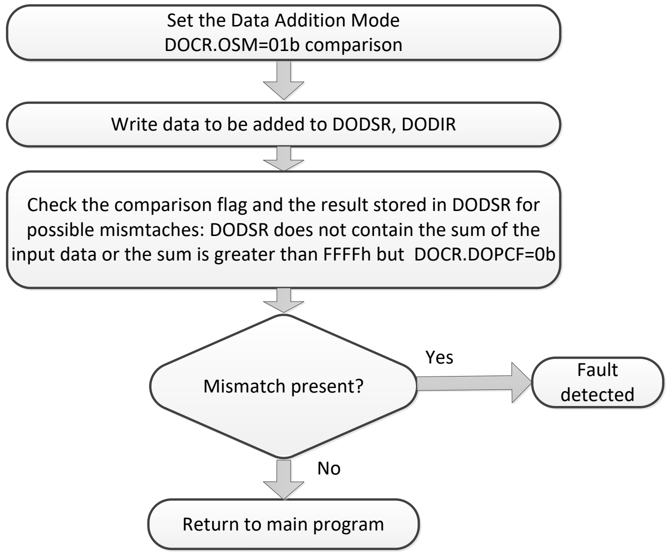

Figure 4.37 Basic procedure for diagnostic of data addition mode for DOC

The third step checks for the correct execution of data subtraction: The procedure, shown in Figure 4.38, has to be repeated for different sets of input data such that all input bits of both operands and output bits of the result are toggled at least once.

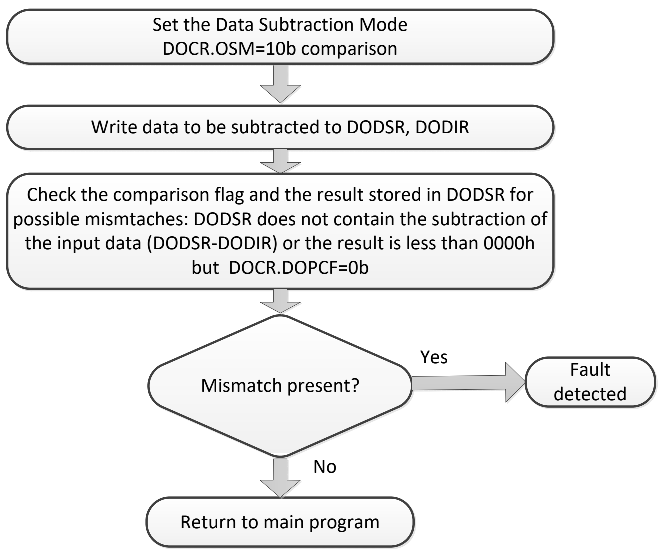

Figure 4.38 Basic procedure for diagnostic of data subtraction mode for DOC

Finally, the last step described in Figure 4.39, provides coverage of the interrupt logic of the DOC with simple control flow monitoring.

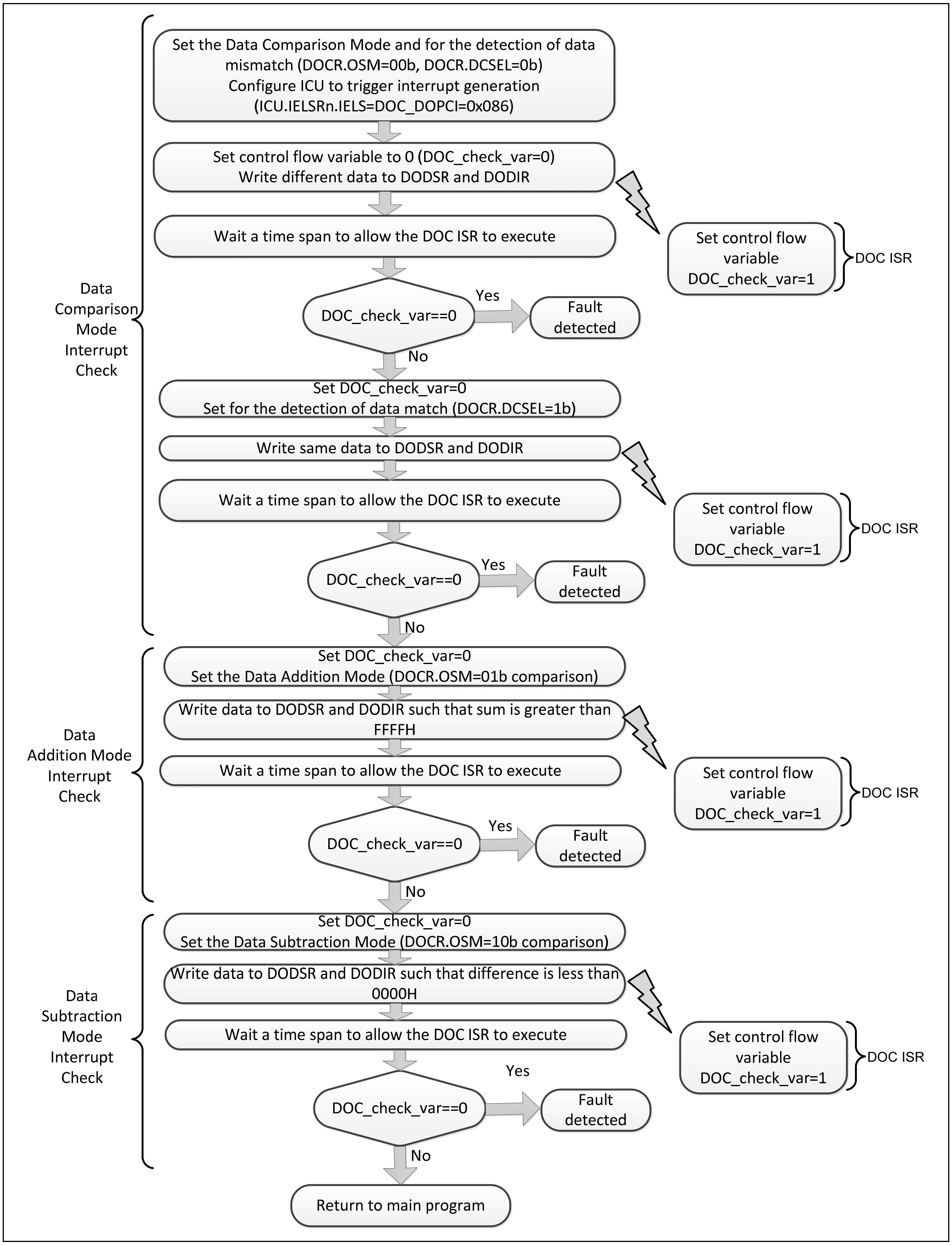

Figure 4.39 Procedure for diagnostic of interrupt generation logic for DOC

4.4 System Level Safety Mechanisms

4.4.1 BatteryBckp_check

This mechanism can be used to detect faults causing misbehavior of the Battery Backup.

Since the battery backup supplies RTC and backup memory, the safety concept is to initially save the RTC in two locations of the backup memory, and then turn the power off in order to trigger the switch between VCC and VBATT. When power is restored, and the power source switches back from VBATT to VCC, it is possible to check the following:

(i) if the two RTC copies stored in backup memory are still identical, and

(ii) if the RTC current value is greater than the one stored in the two copies in backup memory.

4.4.2 GPIO_in_redundancy

This mechanism can be used to detect faults causing the misbehavior of the general I/O pins configured as input.

The input information is provided redundantly on two input pins and the acquired values are then internally compared by SW. A fault is detected if the acquired values are not compatible with each other, which means that they do not have the same information content.

Note: It is recommended to use diverse signals (for example, negated) even if still bringing the same information.

The SW procedure to be performed is shown in Figure 4.40.

Note: This is not a standalone procedure but has to be integrated in to the part of the program when the acquisition of input signal is done.

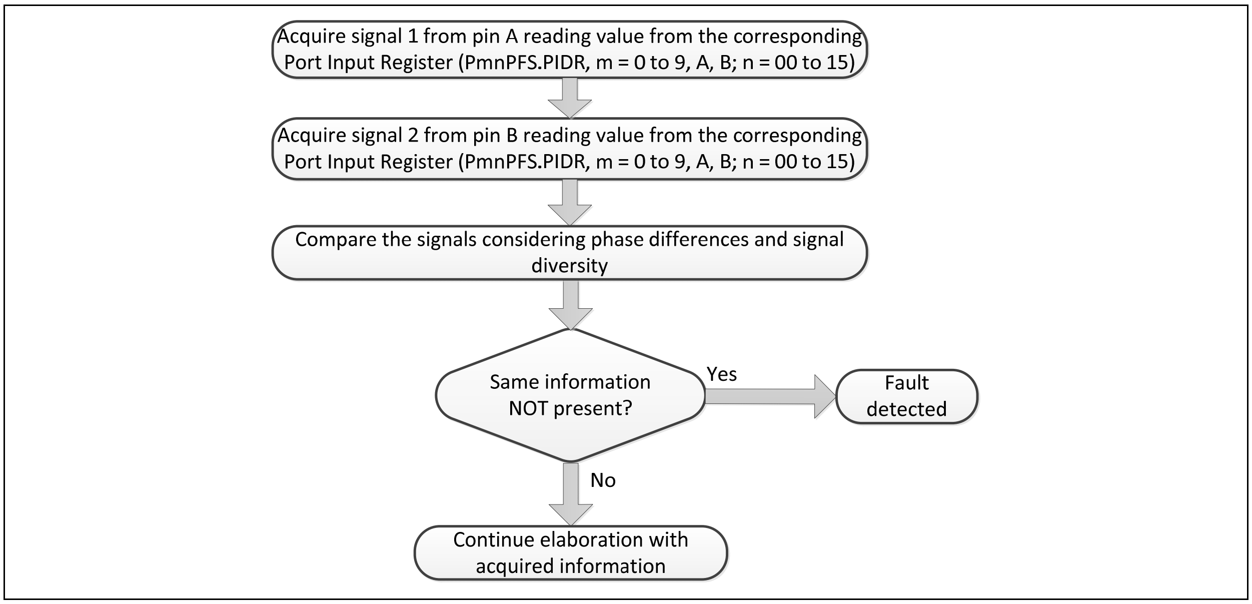

Figure 4.40 SW procedure for GPIO_in_redundancy

In order to mitigate dependencies, it is recommended to select pins belonging to different port groups.

4.4.3 GPIO_out_redundancy

This mechanism can be used to detect faults causing a misbehavior of the general I/O pins configured as output.

The SW drives two output pins redundantly, that is, the same output information is provided to both pins. According to the system-level safety concept, the information can be provided with diversity, that is, the same type of information is provided but with diverse signals (for instance, one signal is the logical negated version of the other one). It is also recommended to drive each pin using different SW routines to mitigate computation failures.

Fault detection happens when the external signals are not consistent and it is up to the system integrator to provide the comparison logic and deliver the resulting information back to the MCU.

In order to mitigate dependencies, it is recommended to select pins belonging to different port groups.

4.4.4 POEG_ExtLoop

This mechanism can be used to detect faults of the protection function provided by the POEG module.

Safety concept of this mechanism is based on an external circuitry configured as follows:

- The monitored input pin (for example, GTETRGA to GTETRGD) is connected and driven by a GPIO configured as output
- The PWM output under test (for example, GTIOCxA and GTIOCxB, where x is the GPT channel number) is externally looped back on a GPIO configured as input
- External pull-up (respectively pull-down) circuitry is connect on the PWM outputs.

SW procedure to be performed is shown in the Figure 4.41.

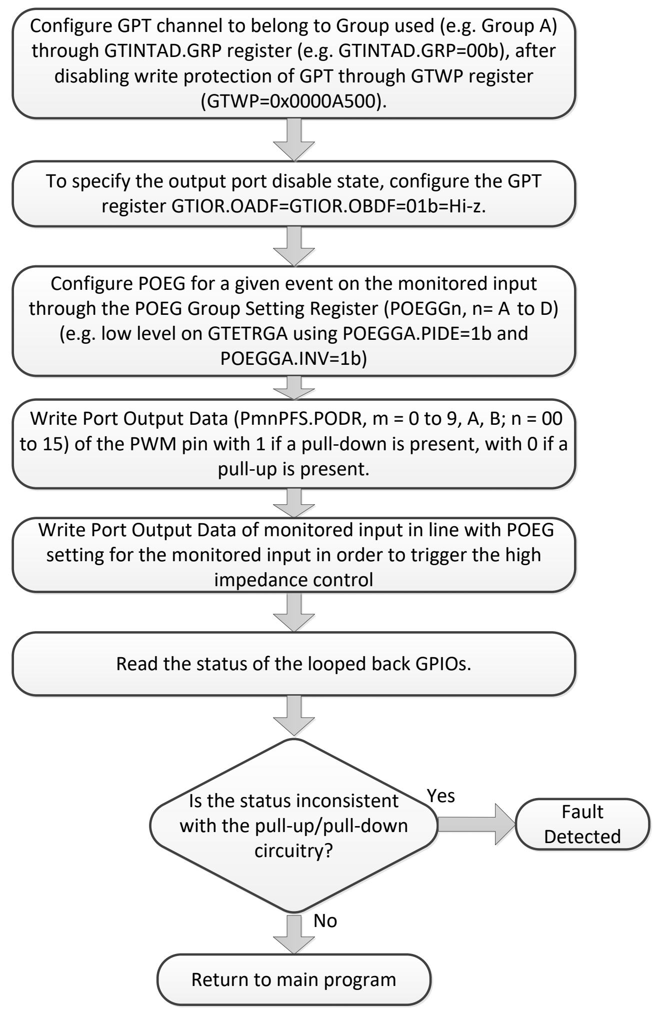

Figure 4.41 SW procedure for POEG_ExtLoop

Generation of an interrupt request can also be configured. In this case, the fault detection can be a part of control flow monitoring integrated within the ISR so that interrupt generation logic is also covered by the test. In this case, a SW timeout also has to be implemented in order to cover the case where the interrupt is not generated even if configured.

(1) **Extension for "Pins placed in high impedance under SW control"**

If the function to enable high impedance control by SW is used, then the following extension of the safety mechanism has to be considered.

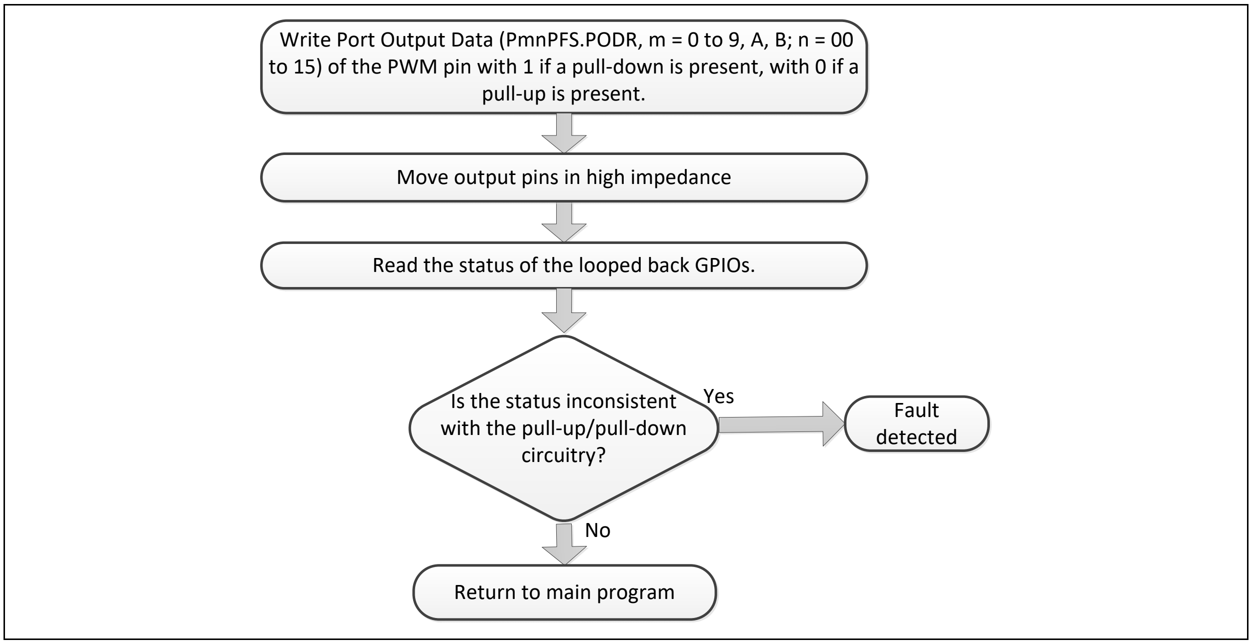

Figure 4.42 Extension of the SW procedure for POEG_ExtLoop

4.4.5 EndToEnd

This mechanism can be used to detect faults affecting the communication, including payload failures on the CAN, Ethernet, USB, SCI, I^2C, and SPI channels.

The end-to-end protection has to be implemented considering the following features:

a. Add CRC on payload.
b. Use of frame identifier to mark a given payload.
c. Establish a precise order in the message exchange and use message counters.
d. Define timeout for each message reception.

Note: For communications over the SCI and I^2C channels, the slaves might be not smart enough to support an end-to-end protocol with the above mentioned features. In such cases, with this MCU generally acting as master of communication, good protection can also be achieved by performing write and read-back operations of the slave registers that are accessible through the SCI/I^2C channel.

4.4.6 DAC12_Loop_ADC14

This mechanism can be used to detect faults of the DAC12.

The safety concept of this mechanism is based on an external closed control loop through ADC14. One analog output channel of the DAC12 is connected to one analog input of the ADC14.

The SW procedure to be performed is shown in Figure **4.43**.

Note: This is not a standalone procedure but has to be integrated in to the procedure executed when a new digital value has to be converted.

With reference to operations described in section 40.3 and Figure 40.2 of [1], the procedure has to be placed before a new update of the data registers (DADR0/1).

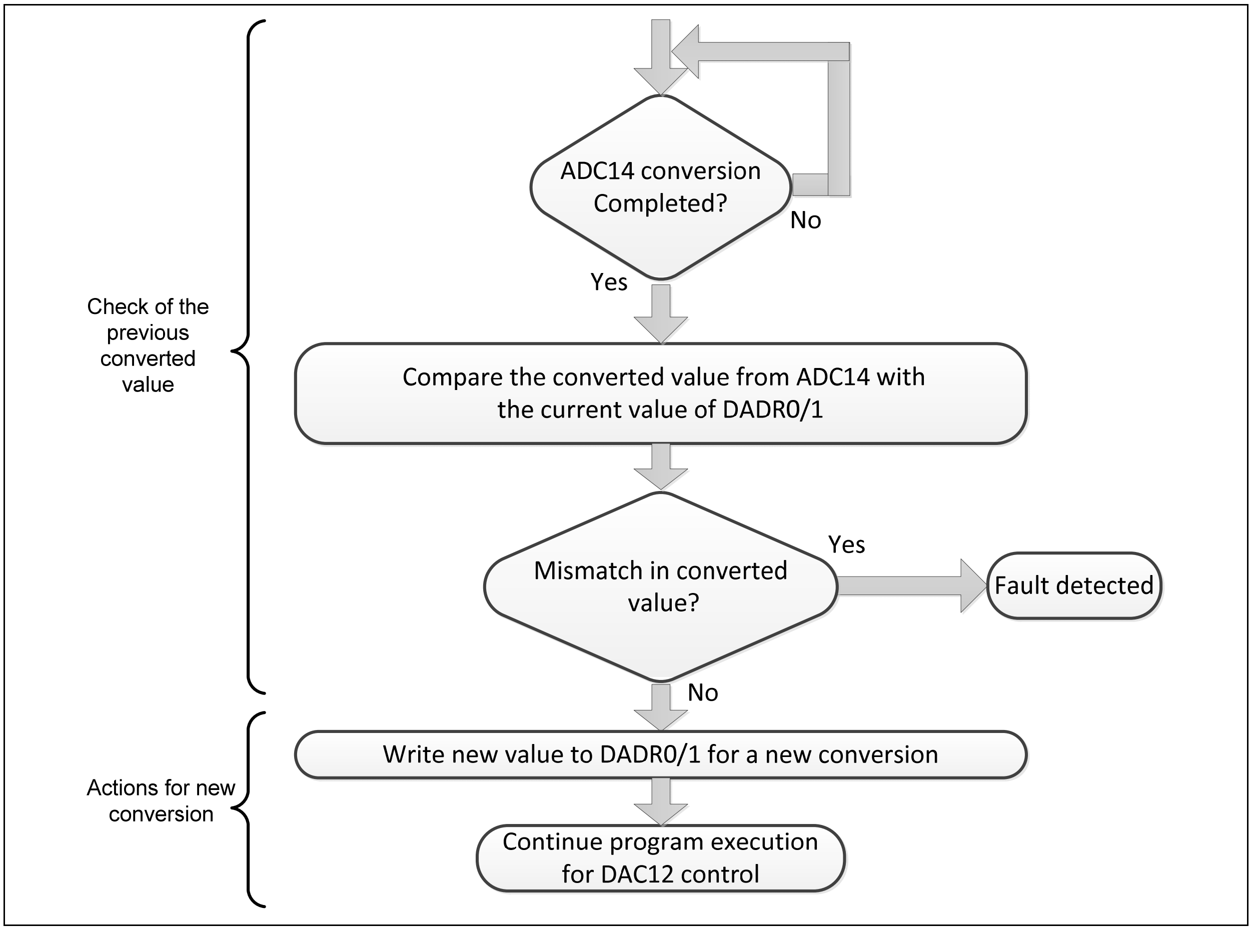

Figure 4.43 **SW procedure for DAC12_Loop_ADC14**

4.4.7 DAC12_system_redundancy

This mechanism can be used to detect faults, causing a misbehavior of the DAC converter (DAC12).

The safety concept of this mechanism is based on the use of two channels of the DAC12 redundantly. The same data is provided to both channels and fault detection can be implemented external to the MCU by comparing the two converted values.

4.4.8 GPT_chan_loop_back

This mechanism can be used to detect faults of the GPT unit in the generation of the PWM signal.

The safety concept of this mechanism is based on looping back the output of one of the channels of the GPT (for example, GPT320) to the compare match input of another channel (for example, GPT321) and measuring the frequency and duty cycle of the PWM signal.

The PWM signal generation on GPT320 is configured according to application requirements, while GPT321 acts as the monitoring channel. The output signal is looped back on both GTIOC1A and GTIOC1B pins and GPT321 is configured to capture the rising and falling edges and to derive from them the frequency and duty cycle of the monitored signal.

The SW procedure for GPT321 is shown in Figure **4.44**.

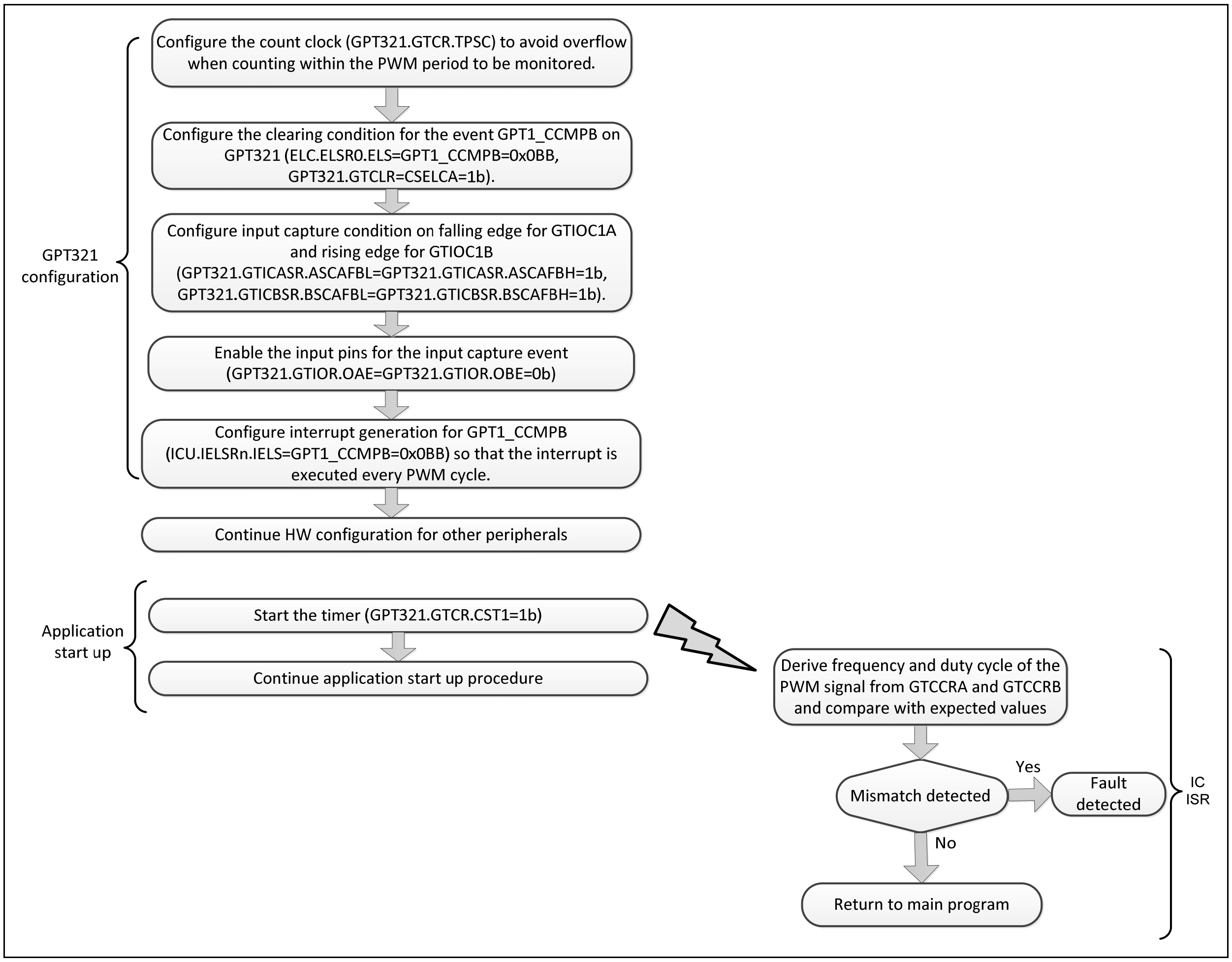

Figure 4.44 **SW procedure for GPT_chan_loop_back**

Note that the clock source for this test is common and the related failures are not covered by the mechanism. In order to mitigate possible issues, the user should periodically monitor the timer counter value to check that it is varying its value.

4.4.9 GPT_loop_back_AGT

This mechanism can be used to detect faults of the AGT unit in the generation of the PWM signal.

This mechanism is similar to GPT_chan_loop_back with the difference that the PWM signal is generated by the AGT unit.

For this test, it is strongly recommended to not use the same clock source or one derived from the same clock source for both GPT and AGT peripherals to mitigate dependencies. See section 4.4.8 for more information.

4.4.10 DMAC_ExtTrigger

This mechanism can be used to detect faults affecting the DMAC behavior in accepting external transfer triggers.

The safety concept is the same as that of DMAC_Crc but with the exception that an external transfer trigger has to be present.

See section 4.3.31 for more information.

4.4.11 Ext_TempSens

This mechanism can be used to detect faults affecting the internal temperature sensor.

The safety concept is based on the usage of an external temperature sensor to cross check the behavior of the internal sensor as shown in Figure **4.45**.

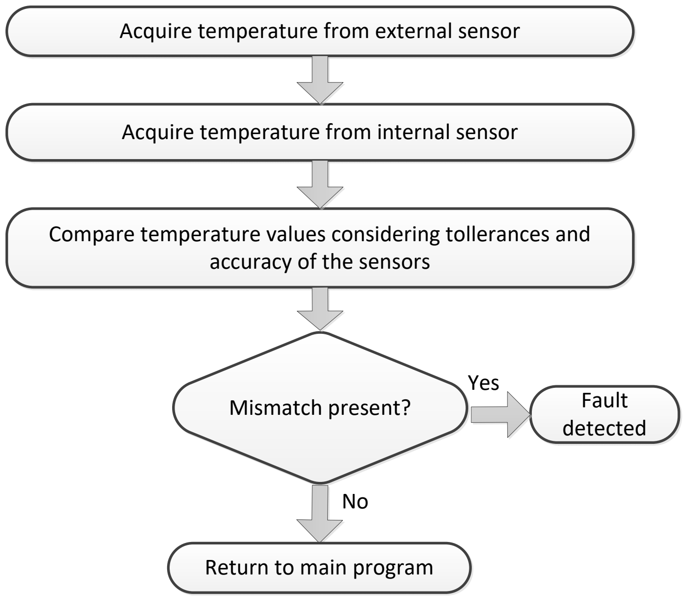

Figure 4.45 **SW procedure for Ext_TempSens**

4.4.12 LVD_check

This mechanism can be used to detect faults affecting the LVD module. It can be used as a self-test to check if voltage monitoring of the LVD works correctly.

The safety concept is based on the usage of external HW that, on reception of a pulse signal from the MCU, starts a voltage drop procedure on the power supply, driving VCC below Vdet0.

The controlled VCC change procedure has to be executed in three steps:

1. Decrease of voltage below Vdet1.
2. Increase of voltage above Vdet2.
3. Decrease of voltage below Vdet0.

The SW procedure to be performed is shown in Figure **4.46**.

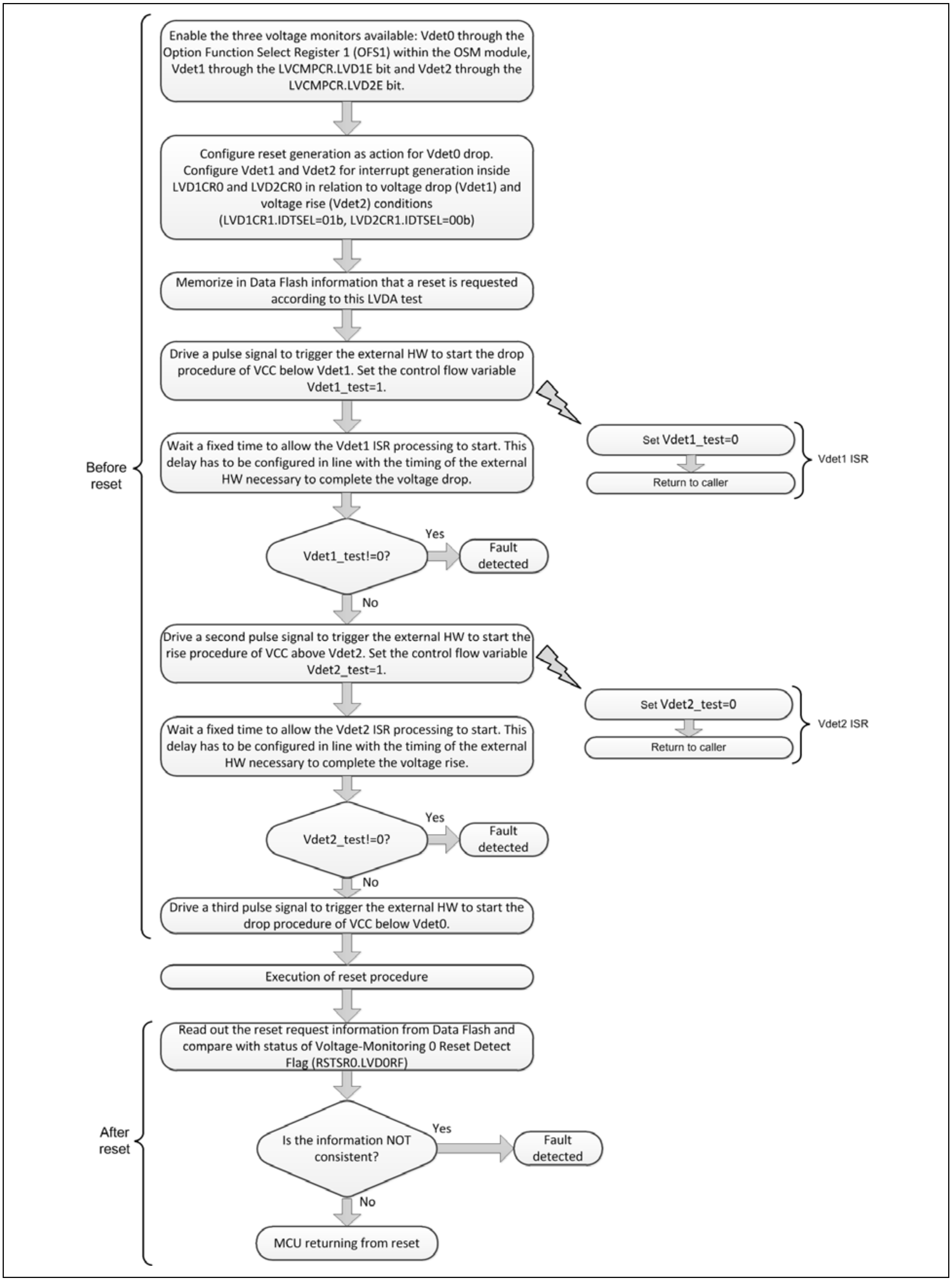

Figure 4.46 **SW procedure for LVD_check**

4.4.13 SSI_check

This mechanism can be used to detect the faults affecting the SSI module. It can be used as a self-test to check if the SSI module works correctly.

The safety concept is based on the usage of external HW, in particular, a digital audio device. At start-up, the SSI shall generate a pre-defined sound message that is sent to the audio device, which will reproduce it. The user can recognize the correct functionality of the SSI by hearing the pre-defined sound message.

4.4.14 SDHI_information_redundancy

This mechanism can be used to detect faults affecting the memory card.

The safety concept is to detect faults through redundant information. There are two different kinds of memory utilization - read only or read and write. In this section, we describe a mechanism for each kind of use. In both cases, it is suggested to add information redundancy that can be checked by SW.

- Read only case: when the memory utilization is read only, it is suggested to add redundant information (for example, CRC code) of the content stored in the memory (or in a given set of its locations) to be compared against a calculated value by SW
- Read/write case: when the memory utilization is read and write, the user has the ability to evaluate some kind of redundant information of the contents to be stored (for example, CRC code) and store it in the memory as well, to allow a further check by the SW.

4.4.15 LDO_ext_monitoring

This mechanism can be used to detect faults affecting the linear regulator.

The safety concept relies on using the external HW to monitor the LDO output voltage. In particular, the LDO output voltage can be monitored by connecting to VCL0 – VCL2 and VCL_F pins. The external monitoring shall check whether the LDO output voltage is within the nominal operating range.

4.4.16 ACMPLP_triggerISR

This mechanism can be used to detect faults affecting the Low Power Analog Comparator (ACMPLP).

The safety concept is based on externally providing two voltage inputs, which are compared with each other. As a first step, the input voltage value provided to ACMPLP (that is, CMPIN0) is lower than the reference value (that is, CMPREF0). In the second step, the input value CMPIN0 is higher and an ACMPLP interrupt is generated.

The SW procedure is shown in Figure 4.47.

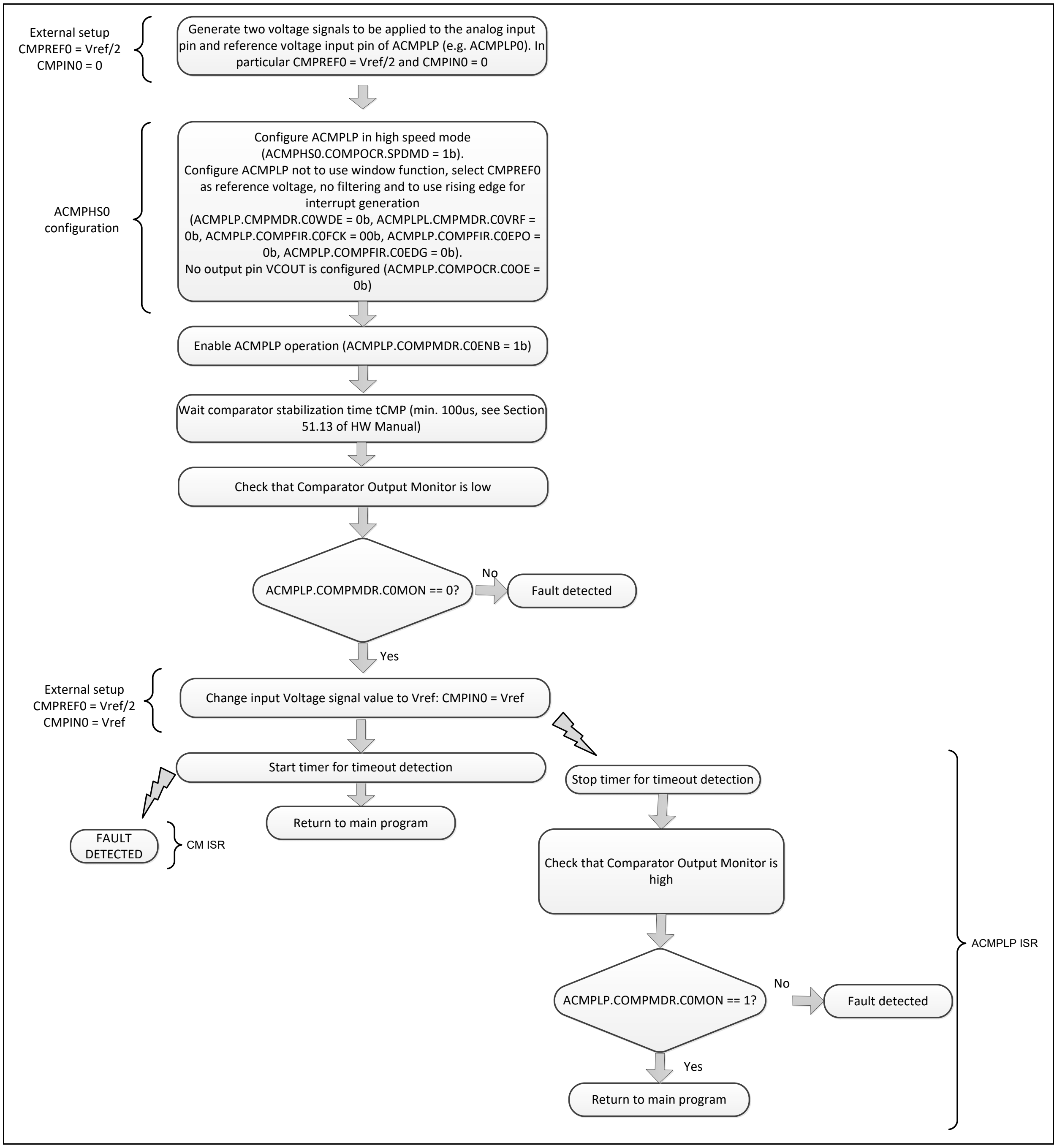

Figure 4.47 SW procedure for ACMPLP_triggerISR

4.4.17 ACMPLP_win_triggerISR

This mechanism can be used to detect faults affecting the Low Power Analog Comparator (ACMPLP).

The safety concept is based on exploiting the window function of ACMPLP. It consists of externally providing a voltage input, which is compared w.r.t. VRFH and VRFL internal reference voltages. As a first step, the input voltage value provided to ACMPLP (that is, CMPIN0) is within the range [VRFL:VRFH]. In the second step, the input value CMPIN0 is lower than VRFL and an ACMPLP interrupt is generated.

The SW procedure is shown in Figure 4.48.

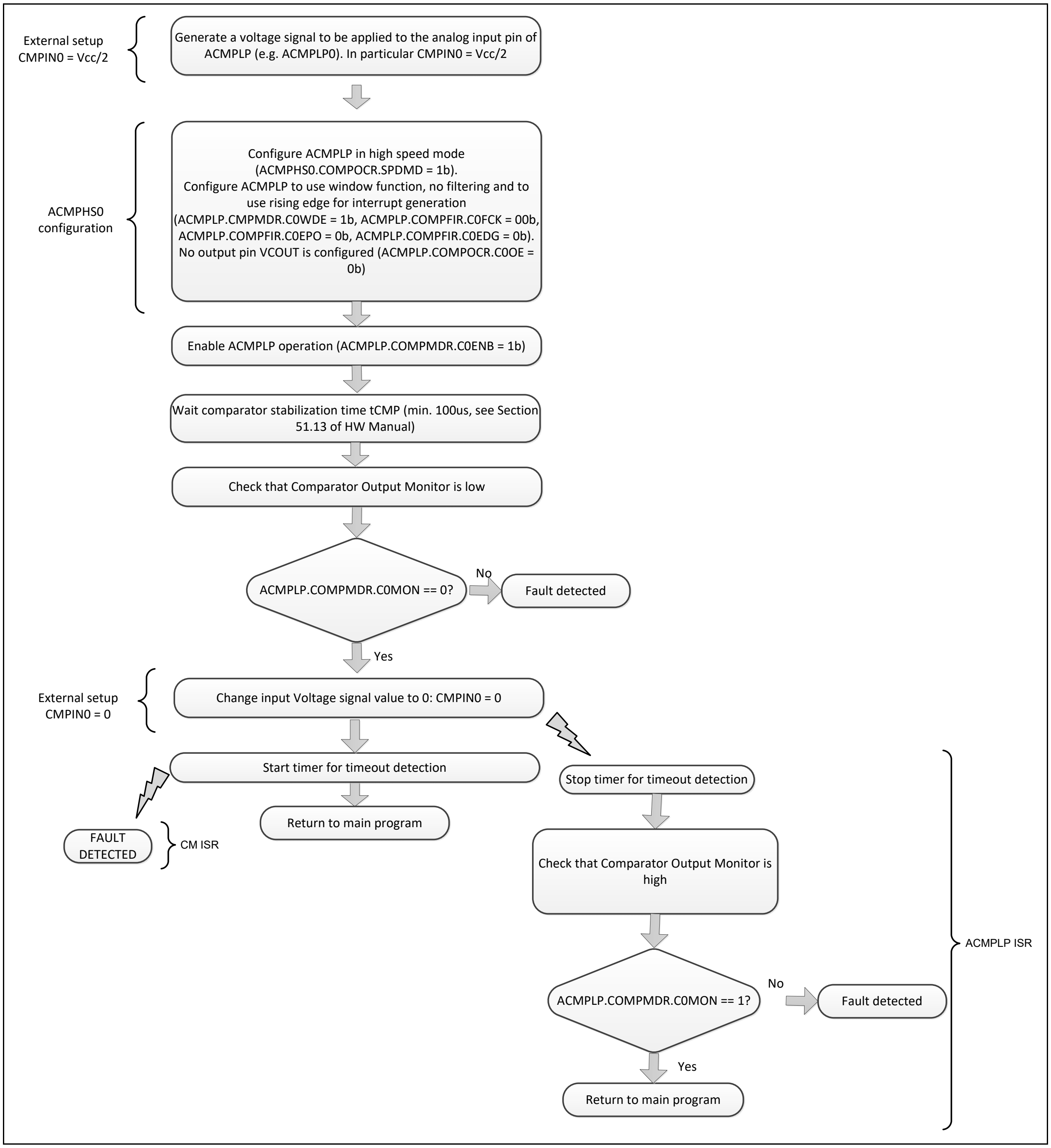

Figure 4.48 SW procedure for ACMPHS_win_triggerISR

Website and Support

Support: https://synergygallery.renesas.com/support

Technical Contact Details:

- America: https://renesas.zendesk.com/anonymous_requests/new
- Europe: https://www.renesas.com/en-eu/support/contact.html
- Japan: https://www.renesas.com/ja-jp/support/contact.html

Revision History

Rev.	Date	Description	
		Page	Summary
0.01	May 29, 2017		First draft
0.02	June 26, 2017	Sec. 3.8.1	Canceled comment on table 3.117.
		Sec. 3.6.2 and 4.4.17	Added cross references.
		Sec. 1.2	Updated references with external codes
		Sec. 3.2.2	Added “ADC14_TriggerDMA” in ADC14_prot_1b.
1.00	June 29, 2017	-	Approval

RENESAS

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 6.0