Document Number: MC33907_8SMUG
Rev. 1.0, 3/2014

Safety Manual for MC33907 and MC33908

freescale

Table of Contents

1 Preface

This document discusses requirements for the use of the MC33907_8 System Basis Chip (SBC) in functional safety relevant applications requiring high functional safety integrity levels.

It is intended to support system and software engineers using the MC33907_8 available features, as well as achieving additional diagnostic coverage by software measures.

Several measures are prescribed as safety requirements whereby the measure described was assumed to be in place when analyzing the functional safety of this System Basis Chip. In this sense, requirements in the Safety Manual (SM) are driven by assumptions concerning the functional safety of the system that integrates the MC33907_8.

- **Assumption**: An assumption being relevant for functional safety in the specific application under consideration (condition of use). It is assumed that the user fulfills an assumption in his design.
- **Assumption under certain preconditions:** An assumption being relevant under certain preconditions. It is assumed that the user fulfills an assumption in his design, if the associated precondition is met.

Example: **Assumption:** It is assumed that the recommended operating conditions given in the MC33907_8 data sheet are maintained.

Example: **Assumption under certain preconditions:** If an output in high-impedance is not considered safe at system level, it is assumed that countermeasures are placed to bring the safety-critical outputs to their Safe state.

NOTE

Assumptions (or assumptions under certain preconditions) are marked by a tag of the form “SM_*nnn*” at the beginning of the assumption, and are terminated with an “end”. Both of these tags are enclosed within square brackets for easy recognition. These tags could be used to allow importing the assumptions into safety traceability management tools.

For the use of the System Basis Chip, means that if a specific safety manual assumption is not fulfilled, it has to be rationalized that an alternative implementation is at least similarly efficient concerning the functional safety requirement in question (for example, provides same coverage, reduces the likelihood of Common Mode Failure (CMF) similarly well, and so on), or the estimation of an increased failure rate (λ_{SPF}, λ_{RF}, λ_{MPF}, λ_{DU} …) and reduced metrics (SFF: Safe Failure Fraction, SPFM: Single-point Fault Metrics, LFM: Latent Fault Metric) due to the deviation has be specified.

This document also contains guidelines on how to configure and operate the MC33907_8 for functional safety relevant applications requiring high functional safety integrity levels. These guidelines are preceded by one of the following text statements:

- **Recommendation**: A recommendation is either a proposal for the implementation of an assumption, or a reasonable measure which is recommended to be applied, if there is no assumption in place. The user has the choice whether to follow the recommendation.
- **Rationale**: The motivation for a specific assumption and/or recommendation.
- **Implementation hint**: An implementation hint gives specific hints on the implementation of an assumption and/or recommendation on the MC33907_8. The user has the choice whether to follow the implementation hint.

These guidelines are considered to be useful approaches for the specific topics under discussion. The user will need to use discretion in deciding whether these measures are appropriate for their applications.

This document is valid only under the assumption that the System Basis Chip is used in functional safety applications requiring a fail-silent or a fail-indicate System Basis Chip. A fail-operational mode of the MC33907_8 is not described.

This document is targeting high functional safety integrity levels. For functional safety goals which do not require high functional safety integrity levels, system integrators need to tailor the requirements for their specific application.

It is assumed, the user of this document is generally familiar with the MC33907_8 device, ISO 26262, and IEC 61508 standards.

1.1 Related Documents

This sections lists all the documentation mentioned in this Safety Manual:

Document Number	Document Type	Description
IEC 61508	Standard	IEC 61508 Functional safety of electrical/electronic/programmable electronic safety-related systems, international standard, ed. 2.0, April 2010
ISO 26262	Standard	ISO 26262 Road vehicles - Functional safety, November 2011
MC33907_8	Data Sheet	Data Sheet for MC33907_8
FCTNLSFTYWP	White paper	Addressing the Challenges of Functional Safety in the Automotive and Industrial Markets, White Paper, October 2011

1.2 Vocabulary

For the purposes of this document, the vocabulary defined in ISO 26262-1 and IEC 61508-4 apply to this document.

Specifically, the following terms apply.

- **System:** functional safety-related system, both implements the required functional safety goals necessary to achieve or maintain a Safe $state_{system}$ for the equipment under control (control system), and is intended to achieve on its own or with other Electrical/Electronic/Programmable Electronic functional safety-related systems, and other risk reduction measures, the necessary functional safety integrity for the required safety functions.
- **System integrator:** the person who is responsible for the system integration.
- **Element:** part of a subsystem comprising of a single component or any group of components (for example, hardware, software, hardware parts, software units) performing one or more element safety functions (functional safety requirements).

Trip time: the maximum time of operation of the SBC without switching to power down state.

2 General Information

The MC33907_8 is designed to be used in automotive or industrial applications which need to fulfill functional safety requirements, as defined by functional safety integrity levels (for example, ASIL D of ISO 26262 or SIL 3 of IEC 61508).

The following devices are supported by this Safety Manual:

- MC33907
- MC33908

2.1 Assumed Conditions of Operation

Assumption: It is assumed that the recommended operating conditions given in the MC33907_8 Data Sheet are maintained.

Assumption: It is assumed that all field failures of the devices are reported to silicon supplier.

Rationale: To cover the ISO 26262-7 (6.5.4) and ISO 26262-7 (6.4.2.1).

Assumption: It is assumed that the latest device errata is taken into account during system design, implementation, and maintenance. For a functional safety-related device such as MC33907_8, this also concerns functional safety-related activities such as system level functional safety concept development.

2.2 Safety Function

Given the application independent nature of the MC33907_8, no general safety function can be specified. Therefore, this document specifies a safety function being application independent for the majority of applications. This application independent safety function would have to be integrated into a complete (application dependent) system.

2.3 Safe State

A Safe state of the system is named Safe $state_{system}$ whereas a Safe state of the MC33907_8 is named Safe $state_{SBC}$. A Safe $state_{system}$ of a system is an operating mode without an unreasonable probability of occurrence of physical injury or damage to the health of persons. A Safe $state_{system}$ may be the intended operating mode or a mode where it has been disabled.

Likewise, a Safe $state_{SBC}$ of the MC33907_8 is by definition one of following operation modes (see Figure 1):

a) Operating correctly
— Outputs depend on application.
b) Explicitly indicating an error (RESET and/or fail-safe output)
— RESET and fail-safe output are in a state indicating an error (active-low)
c) Reset
— MCU connected is under RESET condition.
d) Completely unpowered

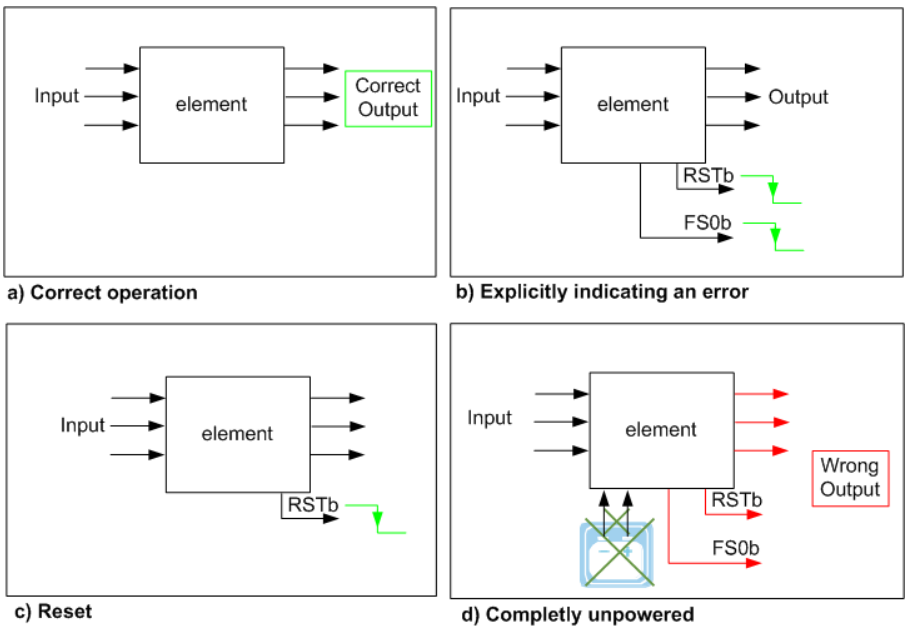

Figure 1. Safe state$_{SBC}$ of the MC33907_8

Assumption: It is assumed that the system transitions itself to a Safe state$_{system}$ when the MC33907_8 explicitly indicates an error via its fail-safe outputs (Reset and FS0b).

Assumption: It is assumed that the system transitions itself to a Safe state$_{system}$ when the MC33907_8 is in reset state.

Assumption: It is assumed that the system transitions itself to a Safe state$_{system}$ when the MC33907_8 is completely unpowered.

2.4 Single-point Fault Tolerant Time Interval and Process Safety Time

The single-point Fault Tolerant Time Interval (FTTI)/Process Safety Time (PST) is the time span between a failure that has the potential to give rise to a hazardous event and the time by which counteraction has to be completed to prevent the hazardous event occurring. It is used to define the sum of worst case fault indication time and time for execution of corresponding countermeasures (reaction). Figure 2 shows the FTTI for a single-point fault occurring in the SBC (Figure 2a) with an appropriate functional safety mechanism to handle the fault (Figure 2b). Without any suitable functional safety mechanism, a hazard may appear after the FTTI elapsed (Figure 2c).

PST in IEC 61508 is the equivalent of FTTI in ISO 26262. Whenever single-point fault tolerant time interval or FTTI is mentioned in this document, it shall be read as PST for IEC 61508 applications.

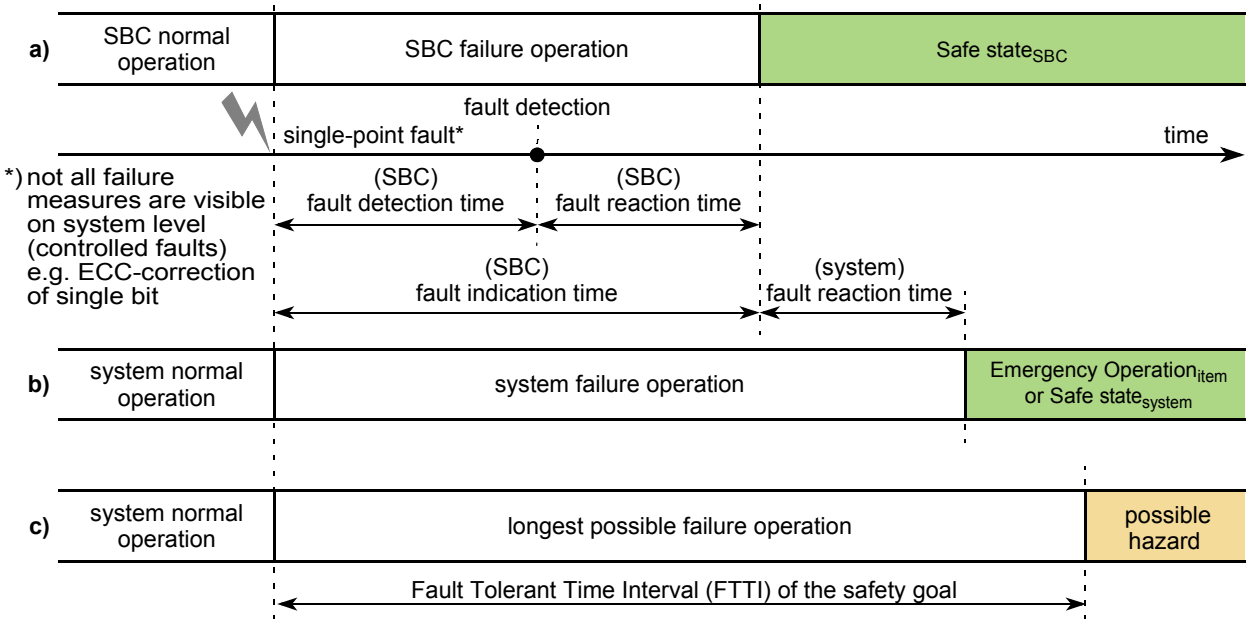

Figure 2. Fault Tolerant Time Interval for Single-point Faults

Fault indication time is the time it takes from the occurrence of a fault to switching into Safe state$_{SBC}$ (for example, indication of that failure by asserting the fail-safe output pins).

Fault indication time of the SBC has three components:

Fault indication time = **Recognition time** + **Internal processing time** + **External indication time**.

Each component of fault indication time is described as follows:

- **Fault detection time** is the maximum time for detection of a fault and consists of:
  - **Recognition time** is the maximum of the recognition time of all involved functional safety mechanisms. The three mechanisms with the longest time are:
    - Recognition time of an overvoltage on regulators takes 234 µs maximum (corresponding to filtering time)
    - Acknowledgement counter used for external IC monitoring is maximum 9.7 ms
  - **Fault reaction time** is the maximum of the reaction time of all involved functional safety mechanisms consisting of internal processing time and external indication time:
  - **Internal processing time** is not the same depending of the origin but the longest time is about 80 µs for an overvoltage detection. All others are below 80 µs.
  - **External indication time** to notify an observer about the failure external to the SBC. This time is 3.0 µs for RSTB and 22 µs for FS0B. Time needed to activate fail-safe outputs when the internal command is sent from digital and activates the analog drivers.

The sum of the SBC fault indication time and system fault reaction time shall be less than the FTTI of the functional safety goal.

2.5 Failure Handling

Failure handling can be split into two categories:

- Handling of failures before enabling the system level safety function (for example, during/following the MCU initialization). These errors are required to be handled before the system enables the safety function, or in a time shorter than the respective FTTI after enabling the safety function.
- Handling of failures during runtime with repetitive supervision while the safety function is enabled. These errors are to be handled in a time shorter than the respective FTTI.

Assumption: It is assumed that single-point and latent fault diagnostic measures complete operations (including fault reaction) in a time shorter than the respective FTTI when the safety function is enabled.

Recommendation: It is recommended to identify startup failures before enabling system level safety functions.

A typical failure reaction regarding power-up/start-up diagnostic measures is not to initialize and start the safety function and instead provide failure indication to the operator/user.

3 Failure Rates and FMEDA

3.1 Mission Profile

Table 1 shows the parameters of Mission profile for typical applications. This document is based on these mission profiles although use of MC33907_8 is not limited to these values. Mission profile is a typical automotive profile.

To prevent latent faults to accumulate during a very long time of operation, additional diagnostic measures need to be executed in continuous operation within the multiple-point fault detection interval.

Table 1. Mission profiles

Mission Parameters	Mission profile
Trip time (t_{TRIP})	12 hours
FTTI	10 ms
Lifetime (t_{LIFE})	20 years
Total operating hours	12000 hours

Table 2 shows temperature profiles.

Table 2. Temperature profile for Mission profile

Device type	Temperature range (°C)	Operation time (h)
Packaged device	125	120
	120	960
	76	7800
	23	2400
	-40	720

3.2 Overview

According to ISO 26262-4, chapter 7.4.3.1 and IEC 61508, Table B.6 a functional safety/failure analysis on hardware design shall be applied to identify the causes of failures and the effects of faults. A typical inductive analysis method is FMEDA (Failure Modes Effects and Diagnostic Analysis).

The FMEDA enables selection of functional safety mechanisms planed to be implemented in a specific application. Enabling or disabling the usage of functional safety mechanisms within an application is possible within the sheets.

The complete FMEDA is available upon request when covered by a Freescale Semiconductor NDA (please contact your Freescale Semiconductor representative).

3.3 High Level Safety Function Decomposition and Dynamic FMEDA

To ensure good use of dynamic Failure Modes, Effects & Diagnostic Analysis (FMEDA) delivered for the MC33907_8, this chapter describes which blocks are assigned to individual function. It shows a high level of decomposition and is linked to blocks described in the document. It is important to note that for a given system, the combination of all the safety functions delivered by the MC33907_8 make sense to reach highest level of ASIL.

VCORE function decomposition:

- V_{PRE} + Hardware configuration - buck/boost
- VCORE
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- SPI
- Interruption

VCCA function decomposition:

- VPRE + Hardware configuration - buck/boost
- VCORE
- V_{CCA} + Hardware configuration - V_{CCA_PNP} detection + Hardware configuration - select
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- SPI
- Interruption

NOTE

V_{CORE} is considered here because the SPI and interruption make sense only if MCU is connected and communicate with MC33907_8 for configuration and diagnostics.

VAUX function decomposition:

- V_{PRE} + Hardware configuration - buck/boost
- VCORE
- V_{AUX} + Hardware configuration + Hardware configuration - select
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- SPI
- Interruption

NOTE

V_{CORE} is considered here because the SPI and interruption make sense only if MCU is connected and communicate with MC33907_8 for configuration and diagnostic.

Watchdog function decomposition:

- V_{PRE} + Hardware configuration - buck/boost
- VCORE
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- SPI
- Watchdog
- Interruption

NOTE

V_{CORE} is considered here because the MC33907_8 watchdog needs to be refreshed by MCU, otherwise the device will enter in Deep Fail-safe (Device OFF) due to watchdog timeout answer.

IOs - MCU error monitoring, function decomposition

- V_{PRE} + Hardware configuration - buck/boost
- VCORE
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- IOs - MCU error monitoring
- SPI
- Interruption

NOTE

V_{CORE} is considered here because the MCU needs to be supplied to deliver error monitoring signal to MC33907_8.

IO_0 & IO_1 - IC external error handling, function decomposition

- V_{PRE} + Hardware configuration - buck/boost
- VCORE
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- IO_0 & IO_1 - IC external error handling
- SPI
- Interruption

NOTE

V_{CORE} is considered here because the MCU needs to acknowledge the signal coming from external IC and informs MC33907_8.

AMUX, function decomposition

- V_{PRE} + Hardware configuration - buck/boost
- VCORE
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- AMUX
- SPI
- Interruption

NOTE

V_{CORE} is considered here because the MCU needs to be supplied to send a SPI command and configure AMUX to deliver right signal on its output (Temp sensor, IO_0, IO_1, V_{SENSE}, internal 2.5 V ref voltage).

CAN, function decomposition

- V_{PRE} + Hardware configuration - buck/boost
- VCORE
- Voltage Supervisor and FS bias + Fail-safe outputs - RSTB + Fail-safe outputs - FS0B
- CAN
- SPI
- Interruption

NOTE

V_{CORE} is considered here because the MCU needs to be supplied to send SPI command and configure CAN if needed.

DEBUG, function decomposition

- V_{PRE} + Hardware configuration - buck/boost
- DEBUG

4 Functional Safety Concept

Failures are the main impairment to functional safety:

- A **systematic failure** is manifested in a deterministic way to a certain cause (systematic fault), that can only be eliminated by a change of the design process, manufacturing process, operational procedures, documentation, or other relevant factors. Thus, measures against systematic faults are reductions of systematic faults, for example, implementing and following adequate processes.
- A **random hardware failure** can occur unpredictably during the lifetime of a hardware element and follows a probability distribution. Thus, measures reducing the likelihood of random hardware faults are either the detection and control of the faults during the lifetime, or reduction of failure rates. A random hardware failure is caused by either a permanent fault (for example, physical damage), an intermittent fault, or a transient fault. Permanent faults are unrecoverable. Intermittent faults are for example, faults linked to specific operating conditions or noise. Transient faults are for example, EMI-radiation. An affected configuration register can be recovered by setting the desired value or by a power cycle. Due to a transient fault an element may be switched into a self destructive state (for example, single event latch up), and therefore may cause permanent destruction.

4.1 Faults

The following random faults may generate failures, which may lead to the violation of a functional safety goal. Citations are according to ISO 26262-1. Random hardware faults occur at a random time, which results from one or more of the possible degradation mechanisms in the hardware.

- **Single-point Fault (SPF):**
  An SPF is “a fault in an element that is not covered by a safety mechanism” and that results to a single-point failure “which leads directly to the violation of a safety goal”. Figure 3a shows an SPF inside an element, which generates a wrong output. The equivalent in IEC 61508 to Single-Point Fault is named **Random Fault**. Whenever an SPF is mentioned in this document, it is to be read as a random fault for IEC 61508 applications.
- **Latent Fault (LF):**
  An LF is a “multiple-point fault whose presence is not detected by a safety mechanism nor perceived by the driver”. An LF is a fault which does not violate the functional safety goal(s) itself, but it leads in combination with at least one additional independent fault to a dual- or multiple-point failure, which then leads directly to the violation of a functional safety goal. Figure 3b shows an LF inside an element, which still generates a correct output. No equivalent in IEC 61508 to LF is named.
- **Residual Fault (RF):**
  An RF is a “portion of a fault that by itself leads to the violation of a safety goal”, “where the portion of the fault is not covered by a functional safety mechanism”. Figure 3c) shows an RF inside an element, which - although a functional safety mechanism is set in place - generates a wrong output, as this particular fault is not covered by the functional safety mechanism.
- **Dual-point fault (DPF):**
  A DPF is an “individual fault that, in combination with another independent fault, leads to a dual-point failure” which leads directly to the violation to a goal. Figure 3d shows two LF inside an element, which generates a wrong output.
- **Multiple-point fault (MPF):**
  An MPF is an “individual fault that, in combination with other independent faults, leads to a multiple-point failure” which leads directly to the violation of a functional safety goal. Unless otherwise stated multiple-point faults are considered as safe faults and are not covered in functional safety concept of the MC33907_8.
- **Safe Fault (SF):**
  An SF is a “fault whose occurrence will not significantly increase the probability of violation of a safety goal”. Safe faults are not covered in this document. Single-point faults, residual faults or dual-point faults are not safe faults.

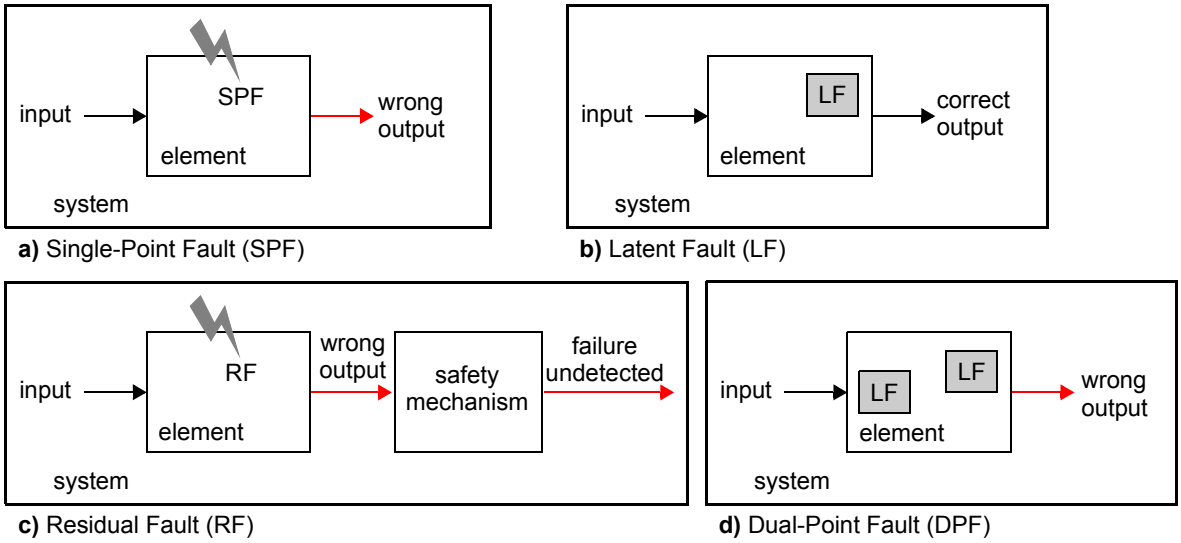

Figure 3. Faults

SPFs shall be detected within the FTTI. Latent Faults (dual-point faults) shall be detected within the L-FTTI. In automotive applications L-FTTI is in general accepted to be once per typical automotive trip time (t_{TRIP}) by test routines (for example, BIST after power-up). This reduces the accumulation time of latent faults from lifetime of the product t_{LIFE} to t_{TRIP}.

Chapter 3, "Failure Rates and FMEDA" lists a profile with a typical trip time for automotive applications and an alternative profile for continuous operation.

4.2 Failures

- **Common Cause Failure (CCF):**
  CCF is a coincidence of random failure states of two or more elements in separate channels of a redundancy element, leading to the defined element failing to perform its intended safety function, resulting from a single event or root cause (chance cause, non-assignable cause, noise, Natural pattern, …). Common Cause Failure causes the probability of multiple channels (N) having a failure rate to be larger than $\lambda_{single\ channel}^{N}$ ($\lambda_{redundant\ element} > \lambda_{single\ channel}^{N}$).

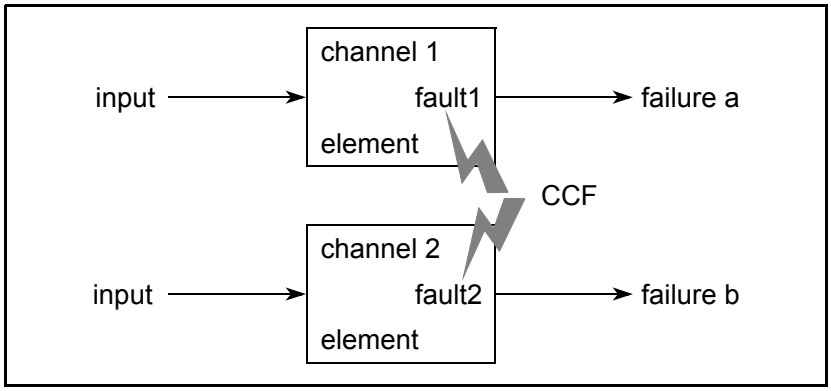

Figure 4. Common Cause Failures

- **Common Mode Failure (CMF):**
  CMF is a subset of CCF. A single root cause leads to similar coincidental erroneous behavior (with respect to the safety function) of two or more (not necessarily identical) elements in redundant channels, resulting in the inability to detect the failures.
  Figure 5 shows three elements within two redundant channels. One single root cause (CMF A or CMF B) leads to undetected failures in the primary channel and in one of the elements of the redundant channel.

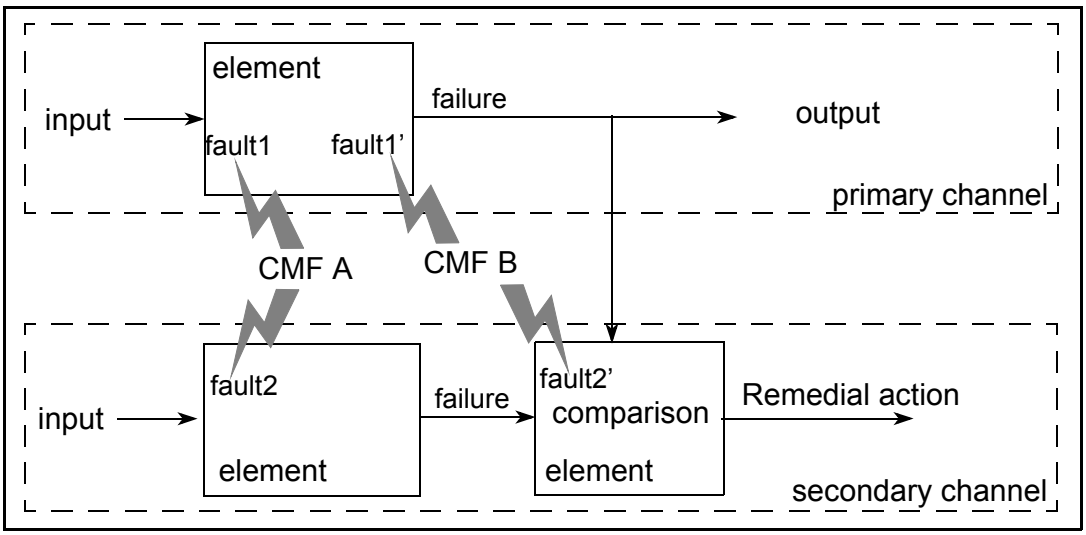

Figure 5. Common Mode Failures

- **Cascading Failure (CF):**
  CFs occur when local faults of an element in a system ripple through interconnected elements causing another element or elements of the same system and within the same channel to fail. Cascading failures are dependent failures that are not common cause failures. Figure 6 shows two elements within a single channel, to which a single root cause leads to a fault (fault 1) in one element resulting in a failure (failure a) causing a second fault (fault 2) within the second element (failure b).

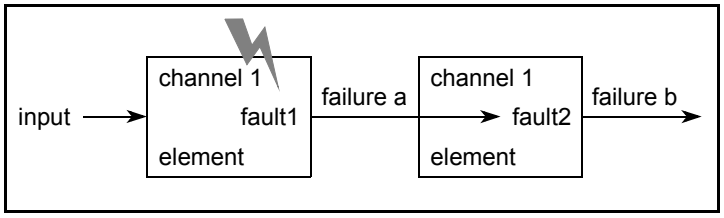

Figure 6. Cascading Failures

4.3 General Functional Safety Concept

Figure 7 shows the block diagram of the MC33907_8.

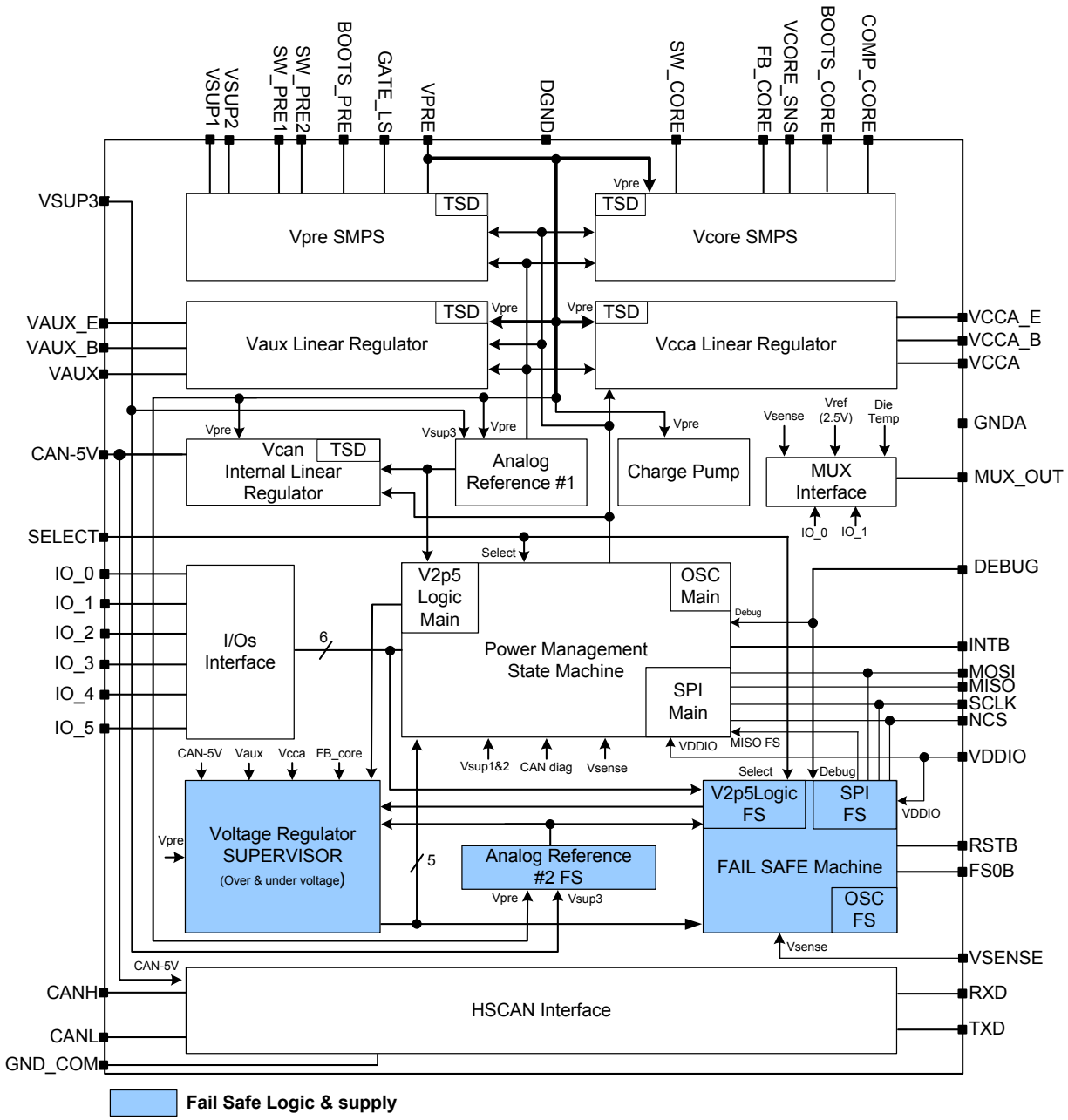

Figure 7. MC33907_8 Block diagram

Functional Safety integrity measures are as follows:

- Replication of supply pins: three VSUP pins (VSUP1, VSUP2, and VSUP3) are used to supply the product. The loss of VSUP1 or VSUP2 shows the MC33907_8 is able to continue to work properly thanks to supply redundancy.
- Fail-safe Machine is powered by VSUP3 and a redundant supply connection is available with VSENSE
- Fail-safe machine is electrically independent from the rest of the circuit with its own oscillator, own reference voltages (Analog Reference #2FS, V2P5LogicFS), and own SPI register configurations.

- Fail-safe machine internal voltage references are monitored against overvoltage.
- Error correction or detection, or both, to reduce the effect of transient faults and permanent faults is implemented.
- Internal supplies and clock are supervised by dedicated monitors.
- Monitoring of the external voltages (FB_{CORE}, V_{CCA}, V_{AUX}) are provided through the voltage supervisor. Also internal reference voltages like V2P5 Main analog (Analog reference#1), V2P5 LogicMain, V2P5 FS analog (Analog Reference #2), V2P5LogicFS are monitored.
- Built-in self tests (ABIST and LBIST) are implemented in hardware to detect in general latent faults only and therefore reduce the risk of coincident latent faults (multiple-point faults).
- The MC33907_8 can react to failure notifications coming from the Freescale Microcontroller, using FCCU (Fault Collection and Control Unit) or external error IC monitoring.
- Risk of CMFs are reduced by a set of measures for both control and reduction of CMFs spanning system level approaches (such as temperature and non-functional signal monitoring), physical separation, or diversity.
- Use of internal (and external) watchdogs or timeout measures.
- A dedicated mechanism is provided to check the functionality of safety path (such as by an application)

5 Hardware Requirements on System Level

This section lists necessary or recommended measures on the system level for the MC33907_8 to achieve the functional safety goal(s).

The MC33907_8 offers an integrated functional safety architecture, a variety of replicated function blocks, self-test unit, and other items to detect faults. By these means, single point failures and latent failure can be detected with a high diagnostic coverage.

However, not all failure mode may be detected on a complete system by the MC33907_8. So it is assumed a separate circuitry is used to bring the system into Safe $state_{system}$ (MCU) in such cases.

Figure 8 depicts the functional safety related elements of the MC33907_8.

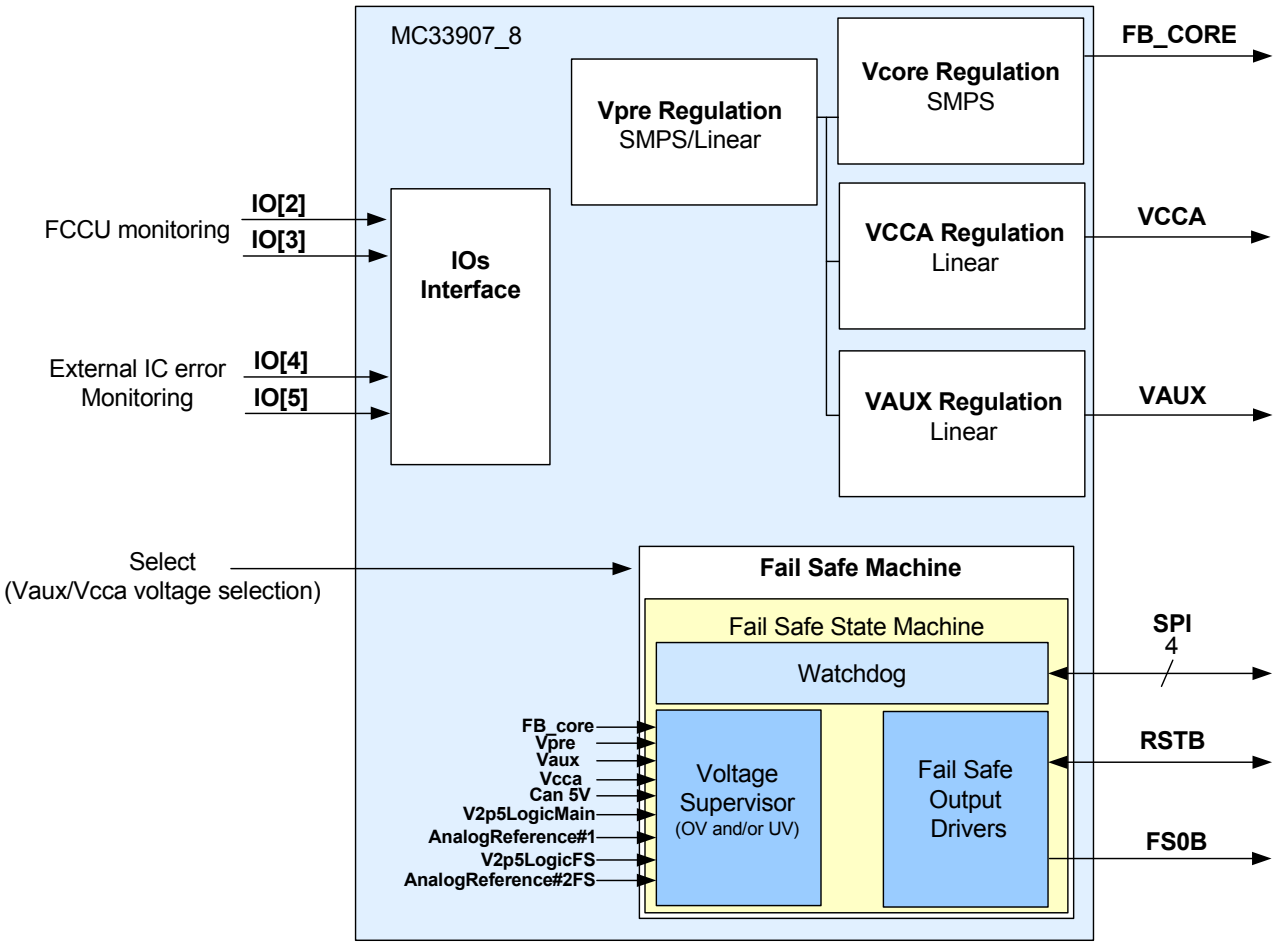

Figure 8. MC33907_8 functional safety blocks

- V_{PRE} is a voltage mode SMPS regulator supplying several block inside the MC33907_8 like internal reference voltage, linear, and SMPS regulators.
- V_{CORE} is a voltage mode SMPS regulator. it is dedicated to the MCU core supply (1.2 V, or 3.3 V), configurable thru external resistor bridge.
- V_{CCA} is a 5.0 V/3.3 V linear voltage regulator dedicated usually to the MCU ADC reference voltage.
- V_{AUX} is a linear voltage regulator dedicated usually to auxiliary functions (3.3 V/5.0 V) or sensor supply (tracking of V_{CCA}).
- Select pin is the pin to configure the voltage level of the V_{CCA} and V_{AUX} regulators.

- Based on safety requirements, the IOs can be used to monitor external error signals coming from the MCU or from other integrated circuits in the system.
- The Fail-safe Machine (FSM) is part of the safety system partitioning. This FSM is made of three main blocks which are:
  - Voltage supervisor (VS)
  - Fail-safe Output Drivers (FSO)
  - Watchdog (WD)

Figure 9 depicts a simplified application schematic for a functional safety relevant application in conjunction with an MCU (only functional safety-related elements shown). The MC33907_8 supply the MCU with the required supply voltages (1.2 V, or 3.3 V). Although for most applications the 1.2 V for digital core supply is generated by an external ballast transistor from 3.3 V supply. Voltages generated by the MC33907_8 are monitored for overvoltage by the embedded voltage supervision.

The MC33907_8 also monitors the state of the error out signals FCCU_F[n] (error monitor) using bi-stable protocol only.

Via the SPI communication interface, the MC33907_8 repetitively triggers the watchdog from the MCU with a valid answer. A dedicated fail-safe state machine is implemented to bring and maintain the application in Safe $state_{system}$. In case of a failure (e.g. watchdog not serviced correctly), RSTB is asserted LOW to reset the MCU. A fail-safe output (FS0B) is available to control or deactivate any fail-safe circuitry (e.g. power switch). MC33907_8 includes Built-in-self-tests.

An interrupt output (INTB) for error information is connected to the NMI input of the MCU.

By a connection of the signal MUX_OUT to an ADC-input of MCU further diagnostic measures are possible (e.g. reading temperature or measuring V_{BATT}). Digital inputs (IO_0, IO_1, IO_4, IO_5) may be used for monitoring error signal handling of other devices. Additionally, MC33907_8 may act as a physical interface to connect the MCU directly with a CAN bus.

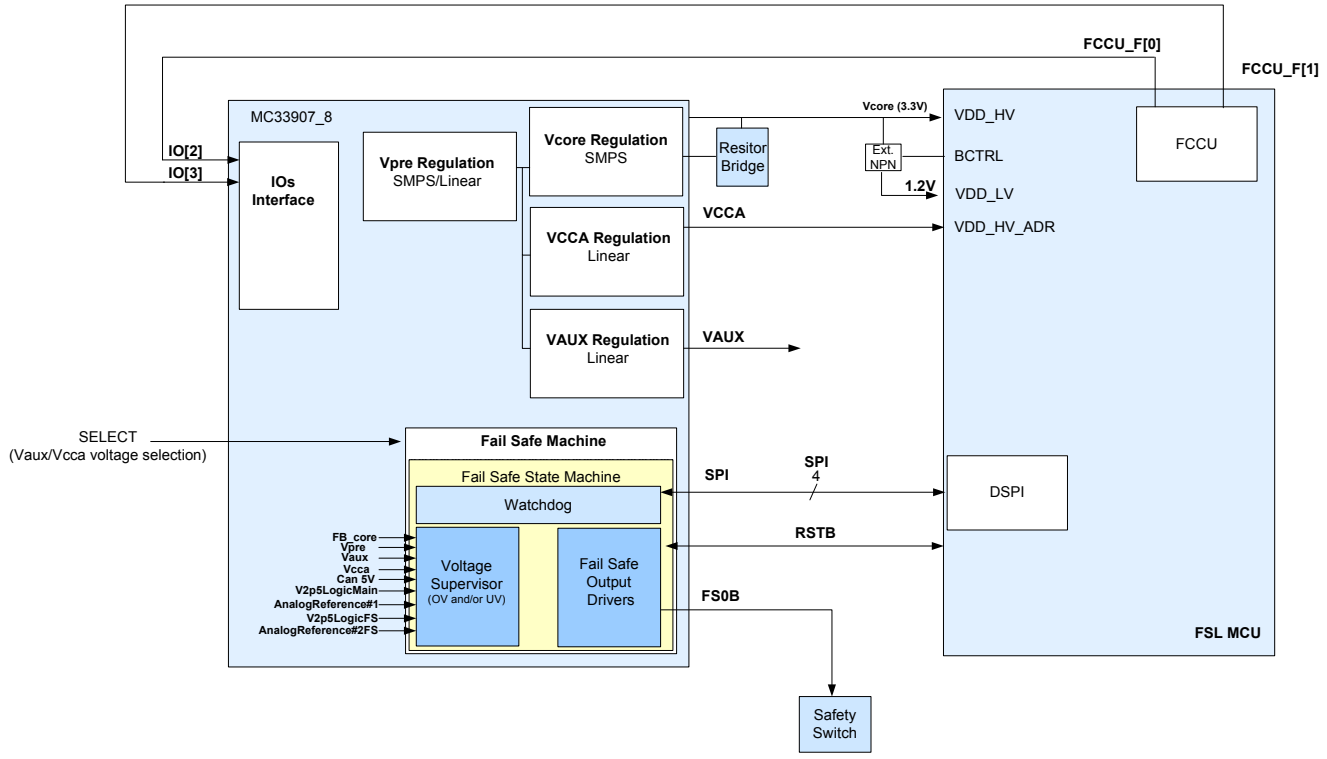

Figure 9. Functional Safety-related Connection to the MCU

6 Safety Interoperation with Separate Circuitry (MCU)

This section describes safety inter operation with MC33907_8 and external circuitry like MCU for application requiring high functional safety integrity levels.

Failure rates of external devices have to be included in the system FMEDA by the system integrator.

6.1 Power Supply

6.1.1 V_{CORE}

The MC33907_8 provides a dedicated voltage supply rail for the main input voltage of the MCU or directly for the core of the MCU (i.e. V_{CORE}).

The voltage level of V_{CORE} supply is configurable thru external resistor bridge. The accuracy of V_{CORE} is ±2.0% without taking account the accuracy of the external resistor bridge.

It will be mandatory to select appropriate resistor tolerance for the external resistor bridge (inferior or equal to ±1.0%).

Assumption: It is assumed the right resistor values are well connected between V_{CORE_SNS} and ground, with the middle point connected to $FB_{\_CORE}$ to configure the right voltage to MCU.

Rationale: To ensure overall operation of the MCU according to its datasheet

Figure 10 shows how to configure external resistor bridge for two ranges of supply level (3.3 V and 1.2 V).

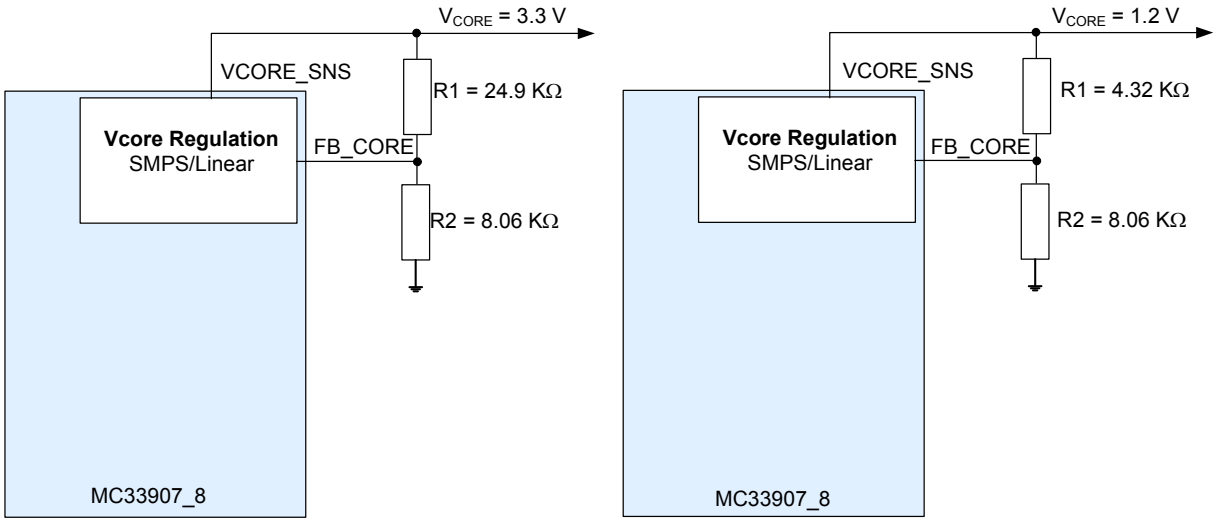

Figure 10. V_{CORE} External Resistor Bridge Configuration

This supply voltage must be in the specified operating range of the MCU because an overvoltage might cause permanent damage to the MCU even if the MCU is kept in reset. Therefore it is either required to de-energize the MCU or to decommission/replace MCU after an overvoltage event.

An undervoltage might leads to an unexpected behavior of the MCU.

Recommendation: It is recommended at system level to avoid V_{CORE} overvoltage and/or undervoltage or to permanently disable (Safe $state_{system}$) the system in case of overvoltage/undervoltage event.

Rationale: To ensure overall operation of the MCU according to its datasheet.

Implementation hint: The MC33907_8 provide an overvoltage/undervoltage monitoring of the $FB_{\_CORE}$. V_{CORE} voltage is set with external resistor bridge. If the $FB_{\_CORE}$ is above or under the value specified in the MC33907_8 datasheet, the MCU is kept powerless by switching off $FB_{\_CORE}$, and the SBC switches the system to a Safe $state_{system}$ within the FTTI and maintains Safe $state_{system}$ thru fail-safe outputs (FS0B, RSTB)

- Internal register configuration:

  In the MC33907_8 a register can be configured during initialization phase to manage the impact at the system level of such overvoltage/undervoltage on V_{CORE}.

  INIT SUPERVISOR 1 register - **Vcore_FS1:0** bits can be configured to perform actions on Fail-safe outputs if there is overvoltage and/or undervoltage on V_{CORE}.

  By default, both V_{CORE_OV} and V_{CORE_UV} do have an impact on RSTB and FS0B.

  The values of overvoltage and undervoltage are specified in the MC33907_8 Data Sheet as well as the filtering time to avoid any sporadic detection.

  Voltage supervisor is able also to detect any spikes, oscillation, or drift of the $FB_{\_CORE}$ voltage if the defined spikes, oscillation or drift are in the range of the detection capability (filtering time and voltage threshold specified on $FB_{\_CORE}$).

Table 3. INIT SUPERVISOR 1- Vcore_FS1:0

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	0	0	1	P	Vcore_FS1	Vcore_FS0	VCCA_FS1	VCCA_FS0	Secure_3	Secure_2	Secure_1	Secure_0

MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	Vcore_FS1	Vcore_FS0	VCCA_FS1	VCCA_FS0

Vcore_FS1:0	Description	V_{CORE} Safety Input
	00	No effect on V_{CORE_OV} and V_{CORE_UV} on RSTB and FS0B
	01	V_{CORE_OV} DOES HAVE an impact on RSTB and FS0B. V_{CORE_UV} DOES HAVE an impact on RSTB
	10	V_{CORE_OV} DOES HAVE an impact on RSTB and FS0B. No effect of V_{CORE_UV} on RSTB and FS0B
	11	Both V_{CORE_OV} and V_{CORE_UV} DO HAVE an impact on RSTB and FS0B
	Reset Condition	Power On Reset

6.1.2 V_{CCA}

The MC33907_8 provides a dedicated voltage supply rail for the Analog to Digital converter of an MCU or for local ECU supply.

The supply voltage must stays in its specified operating range because an overvoltage might cause permanent damage to the MCU (if connected as reference voltage of an Analog to Digital converter for example) even if the MCU is kept in reset.

An undervoltage might lead to an unexpected behavior of the external circuitry, for example where V_{CCA} is the main supply or create bad conversion result in case V_{CCA} is used as a reference voltage for Analog to Digital converter of an MCU.

Assumption: it is assumed that measures at system level maintain the Safe $state_{system}$ during and after V_{CCA} supply voltage above or under the specified operational range.

Rationale: To ensure overall operation of the analog to digital converter of the MCU or the external circuitry where V_{CCA} is connected.

Implementation hint: The MC33907_8 provide an overvoltage/undervoltage monitoring of the V_{CCA}. If V_{CCA} is above or under the value specified in the MC33907_8 Data Sheet and according to its nominal value (5.0 V or 3.3 V), the MCU or external circuitry is kept powerless, and the SBC switches the system to a Safe $state_{system}$ within the FTTI and maintains Safe $state_{system}$ thru Fail-safe outputs (FS0B, RSTB)

- Internal register configuration:

  In the MC33907_8 a register can be configured during initialization phase to manage the impact at the system level of such overvoltage/undervoltage on V_{CCA}.

  INIT_SUPERVISOR 1 register - **Vcca_FS1:0** bits must be configured to perform actions on Fail-safe Outputs if there is overvoltage and/or undervoltage on V_{CCA}.

  By default, both V_{CCA_OV} and V_{CCA_UV} do have an impact on RSTB and FS0B.

  The values of overvoltage and undervoltage are specified in the MC33907_8 Data Sheet as well as the filtering time to avoid any sporadic detection.

  Voltage supervisor is able also to detect any spikes, oscillation, or drift on the V_{CCA} voltage if the defined spikes, oscillation, or drift are in the range of the detection capability (filtering time and voltage threshold specified on V_{CCA})

Table 4. INIT SUPERVISOR 1 - VCCA_FS1:0

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	0	0	1	P	Vcore_FS1	Vcore_FS0	VCCA_FS1	VCCA_FS0	Secure_3	Secure_2	Secure_1	Secure_0
MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	Vcore_FS1	Vcore_FS0	VCCA_FS1	VCCA_FS0

Vcca_FS1:0	Description	Vcca Safety Input
	00	No effect on V_{CCA_OV} and V_{CCA_UV} on RSTB and FS0B
	01	V_{CCA_OV} DOES HAVE an impact on RSTB and FS0B. V_{CCA_UV} DOES HAVE an impact on RSTB
	10	V_{CCA_OV} DOES HAVE an impact on RSTB and FS0B. No effect of V_{CCA_UV} on RSTB and FS0B
	11	Both V_{CCA_OV} and V_{CCA_UV} DO HAVE an impact on RSTB and FS0B
	Reset Condition	Power On Reset

6.1.3 V_{AUX}

The MC33907_8 provides a dedicated voltage supply rail for the IOs of a MCU. It can be configurable to supply external sensors.

This supply voltage must stays in its specified operating range because an overvoltage might cause permanent damage to the MCU, sensors or local ECU supply

An undervoltage might leads to an unexpected behavior of the external circuitry for example where V_{AUX} is the main supply.

Assumption: it is assumed that measures at system level maintain the Safe $state_{system}$ during and after V_{AUX} supply voltage above or under the specified operational range.

Rationale: To ensure overall operation of the MCU or external circuitry (sensors) where V_{AUX} is connected.

Implementation hint: The MC33907_8 provide an overvoltage/under voltage monitoring of the V_{AUX}. If V_{AUX} is above or under the value specified in the MC33907_8 Data Sheet and according to its nominal value (5.0 V or 3.3 V), the MCU or external circuitry is kept powerless and the SBC switches the system to a Safe $state_{system}$ within the FTTI and maintains Safe $state_{system}$ thru Fail-safe outputs (FS0B, RSTB)

- Internal register configuration:

  In the MC33907_8 a register can be configured during initialization phase to manage the impact at the system level of such overvoltage/undervoltage on V_{AUX}.

  INIT_SUPERVISOR 2 register - **Vaux_FS1:0** bits must be configured to perform actions on Fail-safe Outputs if there is overvoltage and/or undervoltage on V_{AUX}.

  By default, both V_{AUX_OV} and V_{AUX_UV} do have an impact on RSTB and FS0B.

  The values of overvoltage and undervoltage are specified in the MC33907_8 Data Sheet as well as the filtering time to avoid any sporadic detection.

  Voltage supervisor is able also to detect any spikes, oscillation, or drift on the V_{AUX} voltage if the defined spikes, oscillation or drift are in the range of the detection capability (filtering time and voltage threshold specified on V_{AUX})

Table 5. INIT SUPERVISOR 2 - VAUX_FS1:0

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	0	1	0	P	Vaux_FS1	Vaux_FS0	0	DIS_8s	Secure_3	Secure_2	Secure_1	Secure_0
MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	0	DIS_8s	Vaux_FS1	Vaux_FS0

Vaux_FS1:0	Description	V_{AUX} Safety Input
	00	No effect on V_{AUX_OV} and V_{AUX_UV} on RSTB and FS0B
	01	V_{AUX_OV} DOES HAVE an impact on RSTB and FS0B. V_{AUX_UV} DOES HAVE an impact on RSTB
	10	V_{AUX_OV} DOES HAVE an impact on RSTB and FS0B. No effect of V_{AUX_UV} on RSTB and FS0B
	11	Both V_{AUX_OV} and V_{AUX_UV} DO HAVE an impact on RSTb and FS0b
	Reset Condition	Power On Reset

6.1.4 V_{AUX} - Sensor Supply

V_{AUX} can be used as sensor supply in a system. To ensure ratiometric conversion between sensors supplied by the V_{AUX} and the analog to digital converter supplied by V_{CCA} the V_{AUX} must be configured as a tracker of V_{CCA}.

Assumption: it is assumed the V_{CCA} linear regulator is used as reference voltage of analog to digital converter of the MCU.

Rationale: to ensure ratiometric conversion on sensors data output powered by V_{AUX}.

Implementation hint: During initialization phase the **INIT_VREG2** register must be configured to activate the Vaux_trk_EN bit. V_{AUX} will be then tracker of V_{CCA} with a tracking accuracy of ±15 mV. By default the V_{AUX} tracker is not activated.

Table 6. INIT Vreg 2 - VAUX_trk_EN

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	0	0	0	0	1	0	P	0	Tcca_lim_off	Icca_lim	0	Reserved	Taux_lim_off	Vaux_trk_EN	reserved
MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	0	Tcca_lim_off	Icca_lim	0	Reserved	Taux_lim_off	Vaux_trk_EN	reserved

Vaux_trk_EN	Description	Configure V_{AUX} regulator as a tracker
	0	No tracking.
	1	Tracking enabled
	Reset Condition	Power On Reset

6.2 Safety Inputs - IOs

6.2.1 IO[2] & IO[3] MCU Error Monitoring - FCCU

The MC33907_8 measures internal errors coming from Freescale MCU. In case the MCU signals an internal failure via its error out pins (FCCU[0] and FCCU[1]), the system may no longer rely on integrity of the device's outputs for safety functions. If an error out is indicated, the system switches and remains in Safe $state_{system}$. The SBC switches the system to a Safe $state_{system}$ within the FTTI and maintains Safe $state_{system}$ thru Fail-safe outputs (FS0B, RSTB) as a reaction to the indicated error out.

Only the "Bi-stable" protocol is covered on the MC33907_8 to use with FCCU pins coming from Freescale Microcontroler. Refer to respective Freescale MCU datasheet.

Assumption: It is assumed the bi-stable protocol has been configured in the Freescale MCU for FCCU protocol.

Rationale: To monitor the Freescale MCU error out signals for correct functionality of the device. The system (for example ECU) may not rely on any I/O other than FCCU_F[0] and FCCU_F[1], when those signals indicate an error.

Implementation hint: Connect MCU FCCU_F[0] error output to MC33907_8 IO[2] input and MCU FCCU_F[1] error output to MC33907_8 IO[3] input. A pull-down must be connected to FCCU_F[0]/IO[2] and a pull-up must be connected to FCCU_F[1]/IO[3].

Figure 11 shows the connections of MU FCCU and MC33907_8 IOs.

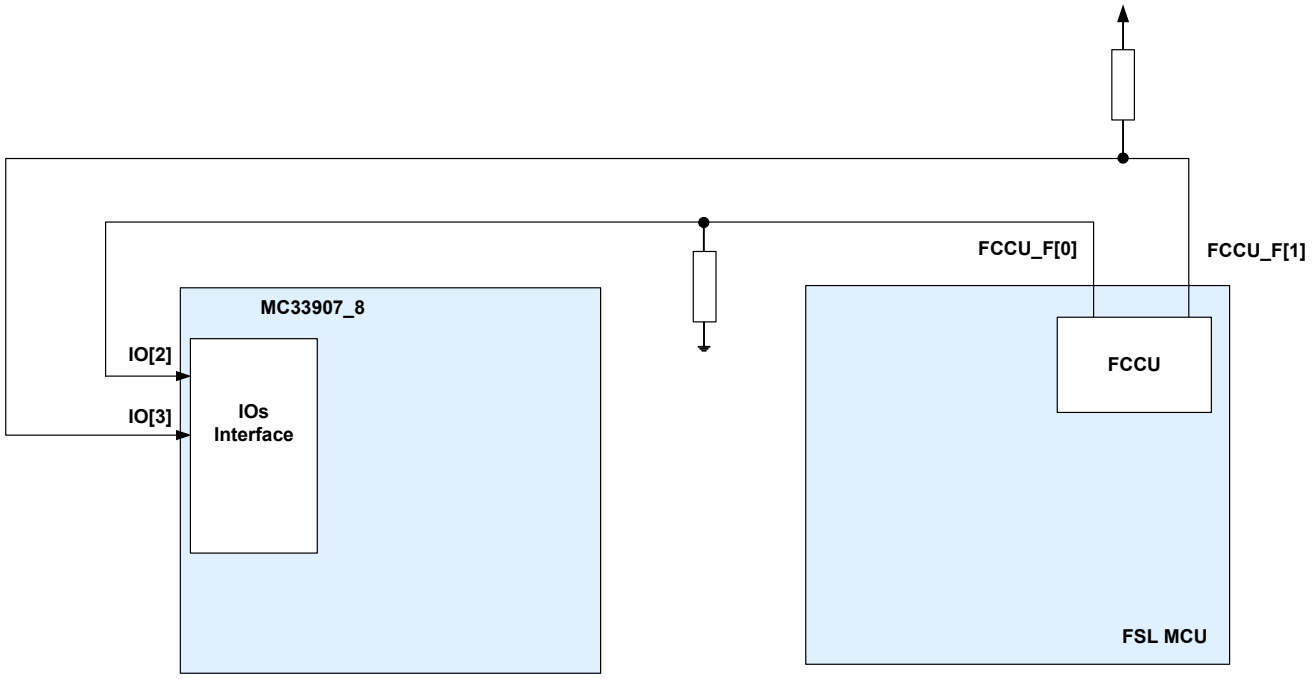

Figure 11. FCCU Connection with MC33907_8 IOs

- Internal register configuration:
  In the MC33907_8, a register must be configured during initialization phase to activate the FCCU error out monitoring.
  INIT_FSSM2 register - **IO_23_FS** bits must be configured as safety inputs
  By default, **IO_23_FS** bit is activated.

Table 7. INIT_FSSM2 - IO_23_FS

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	1	0	1	P	RSTb_err_FS	IO_23_FS	PS	0	Secure_3	Secure_2	Secure_1	Secure_0

MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	RSTb_err_FS	IO_23_FS	PS	0

IO_23_FS	Description	Configure the couple of IO_3:2 as safety inputs for FCCU monitoring
	0	NOT SAFETY
	1	SAFETY CRITICAL
	Reset Condition	Power On Reset

6.2.2 IC Error Signal Monitoring

The MC33907_8, out of the Freescale MCU FCCU monitoring, can monitors error signal coming from external IC. This is possible by using digital inputs (IOs) by pair.

On each pair of digital inputs, one must be dedicated to monitor the output error signal coming from the external circuitry and the other one must be connected to an output of the MCU to listen the acknowledgement of the error by the MCU itself.

When an error from the external circuitry is NOT acknowledged by the MCU within a specific “acknowledgment timing”, the MC33907_8 switches the system to a Safe $state_{system}$ within the FTTI and maintains Safe $state_{system}$ thru Fail-safe outputs (FS0B).

Assumption: it is assumed the error output signal from external IC is well connected to one SBC IO and one MCU IO to ensure MCU is able to listen the fault.

Rationale: Monitor a safety function realized in the ECU, out of the MCU, and bring the system in Safe $state_{system}$ in case of a fault.

Implementation hint: In the MC33907_8 a register can be configured during initialization phase to manage the impact at the system level of such error monitoring using IOs by pair out of IO[2] & IO[3] which are dedicated to FCCU monitoring.

IO[0] & IO[1] AND/OR IO[4] & IO[5] can be used to enable this safety functions.

INIT FSSM1 register - **IO_01_FS** bits must be configured as **safety critical** to perform actions on Fail-safe outputs (FS0B) if there is an error reported by external IC on IO[0] and not acknowledged by MCU on IO[1]. Refer to the MC33907_8 Data Sheet.

By default, IO_01_FS are not configured as safety critical inputs

INIT FSSM1 register - **IO_45_FS** bits must be configured as **safety critical** to perform actions on Fail-safe outputs (FS0B, FS1, RSTB) if there is an error reported by external IC on IO[4] and not acknowledged by MCU on IO[5]. Refer to the MC33907_8 Data Sheet.

By default, IO_45_FS are not configured as safety critical inputs

Table 8. INIT_FSSM1 - IO_01_FS - IO_45_FS

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	1	0	0	P	IO_01_FS	IO_1_FS	IO_45_FS	RSTb_low	Secure_3	Secure_2	Secure_1	Secure_0

MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	IO_01_FS	IO_1_FS	IO_45_FS	RSTb_low

IO_01_FS	Description	Configure the couple of IO_1:0 as safety inputs
	0	NOT SAFETY
	1	SAFETY CRITICAL
	Reset Condition	Power On Reset
IO_45_FS	Description	Configure the couple of IO_5:4 as safety inputs
	0	NOT SAFETY
	1	SAFETY CRITICAL
	Reset Condition	Power On Reset

Figure 12 shows a connection example of an external IC error out on IO[4] and the MCU acknowledgement on IO[5]

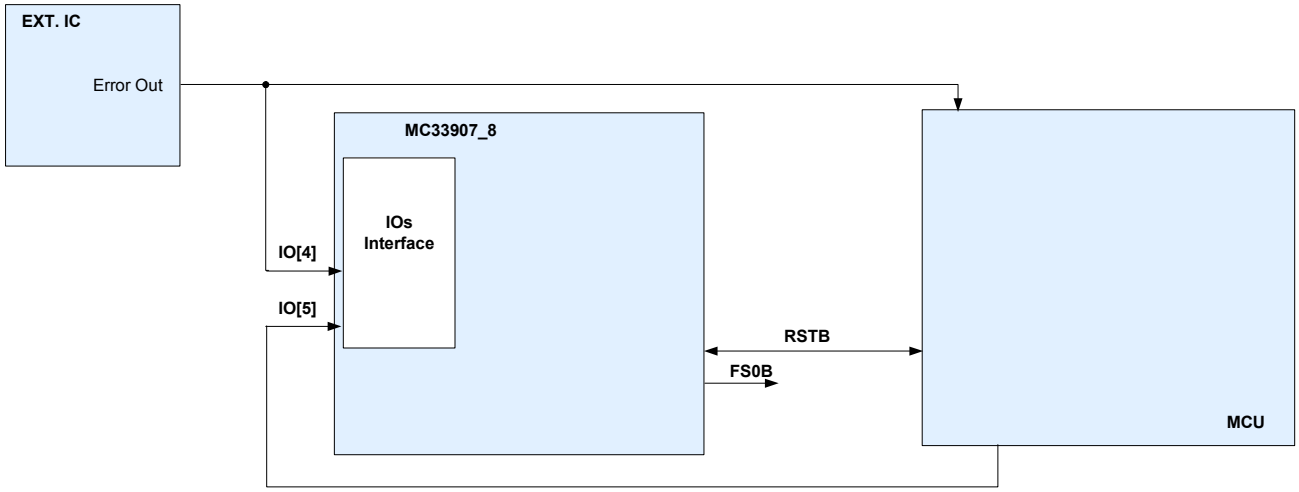

Figure 12. External IC Error Connection

Figure 13 shows the signal in case of an error on the external IC with or without MCU acknowledgement.

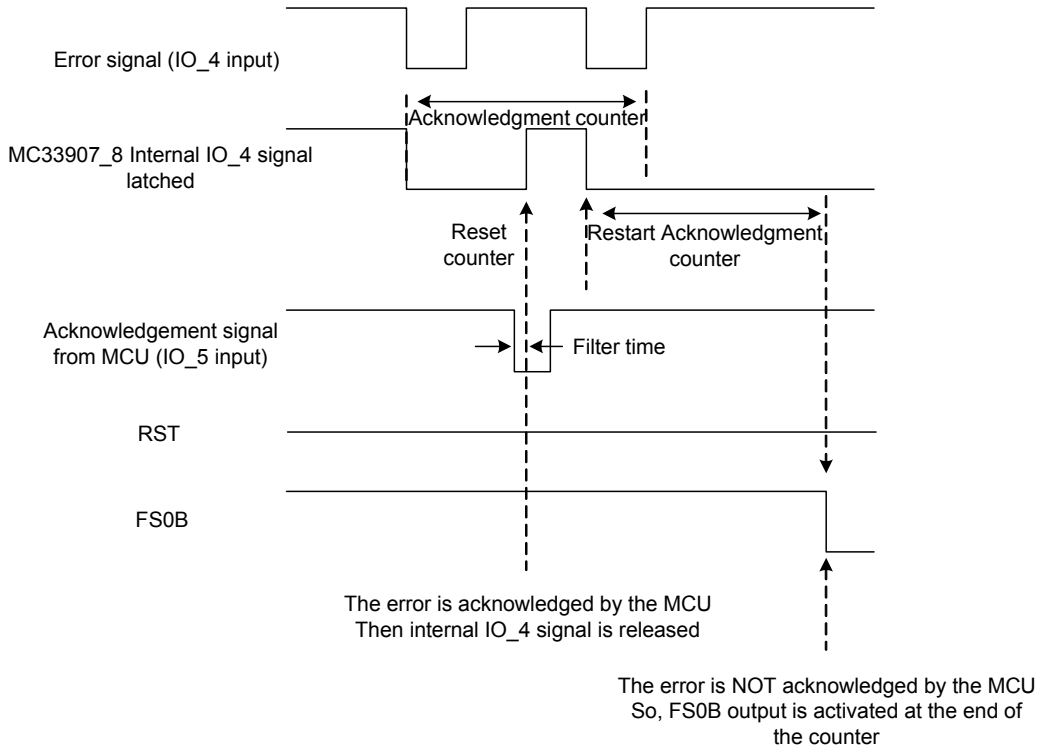

Figure 13. External IC Error Monitoring (Timing)

Refer to the MC33907_8 Data Sheet for filtering time, and counters.

6.2.3 IC Error Signal Monitoring or How to Verify the Safety Path in a System

If an error is reported to IO[4] and if the MCU doesn't acknowledge the fault, the SBC will assert only the FS0B output to bring the system in Safe state$_{system}$. The RSTB is not asserted low in this specific case.

The external IC error out can also be replaced by the MCU itself using a GPIO.

Recommendation: at each startup of the system it is recommended to verify the safety path.

Rationale: to ensure the system is well in Safe state$_{system}$ when the FS0B is asserted low before to start the application.

Implementation hint: connection of one MCU GPIO to IO[4] and a second GPIO to IO[5], and drive the first GPIO (IO[4]) like the error output of an external IC. Bit IO_45_FS in register showed in Table 8 must be configured.

Figure 14 shows the connection for the verification of safety path each startup.

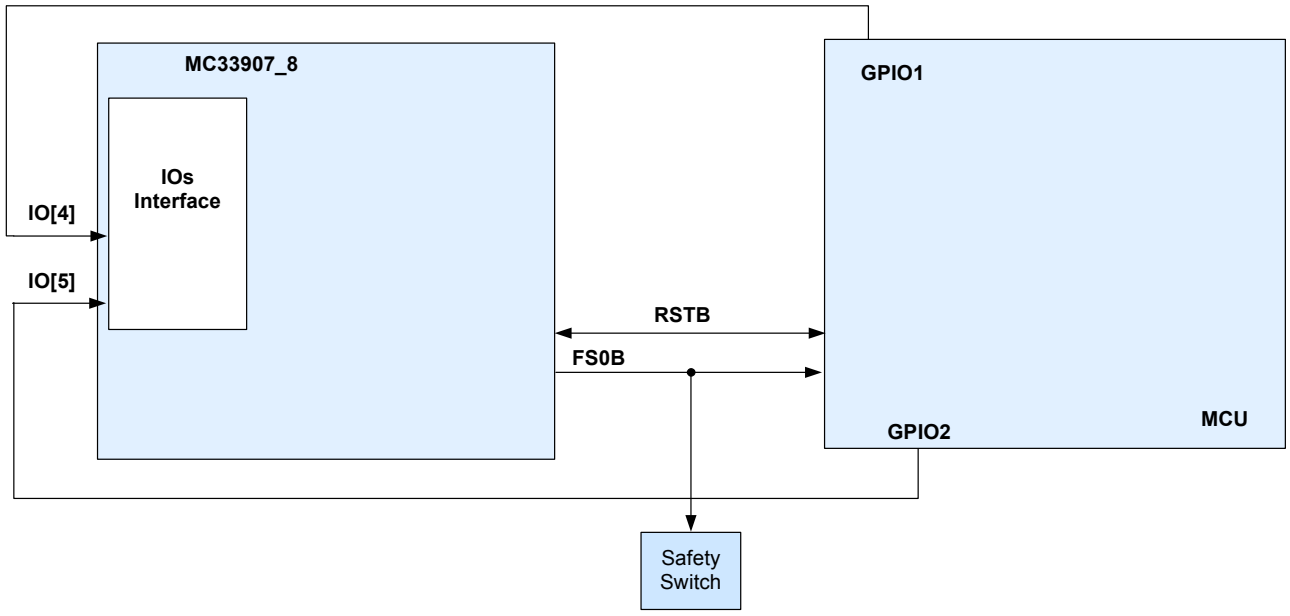

Figure 14. Safety Path Verification

6.3 Watchdog

A common mode failure may lead to a state where an MCU is not able to signal an internal failure via its error out pins (see IO[2] & IO[3] MCU Error Monitoring - FCCU). With the use of an item (system) level timeout (e.g. watchdog) function, the likelihood that common mode failures affect the functional safety of the system can be reduced significantly.

In general, the external watchdog covers common mode failures which are related to:

- missing/wrong power
- missing/wrong clocks
- missing/wrong resets
- general destruction of internal components (e.g. latch-up at redundant input pads)
- errors in mode change (e.g. test, debug, sleep/wake-up)

Since these errors do not result in subtle output variations of the MCU but typically in a complete failure, a simple watchdog is sufficient.

The watchdog function is required to be sufficiently independent to the SBC (e.g regarding clock generation, power supply, implementation, etc.).

The MC33907_8 acts as a supervisor of the operation and, as a consequence, includes a windowed watchdog that need to be refreshed periodically by the MCU. It means the MC33907_8 watchdog function is in permanent communication with MCU. As soon as there is no correct communication, after repetitive and defined tries, the SBC switches the system to Safe state$_{system}$ within the FTTI. Thus either MCU or SBC can switch the system to Safe state$_{system}$.

Assumption: it is assumed the MCU refresh periodically the MC33907_8 watchdog.

Rationale: to cover situations, when MCU is not able to signal a failure.

Implementation hint: The duration of the watchdog window is configurable to allow different MCU handshake strategies. The duty cycle of the window is fixed and is 50%. Therefore the first half of the window is said “closed” and the second half of the window is said “open”. “Open” window is the window where the watchdog must be refreshed.

- Internal register:

  In the MC33907_8, a register can be configured during initialization phase or in normal operation. Doing the change in normal operation allow the system integrator to configure the watchdog window duration on the fly (the new WD window duration will be taken into account when the previous one is finished).

  WD WINDOW register - **WD_Window_x** bits (where x=0 to 3) can be configured

  By default, a window of 3.0 ms is configured.

Table 9. WD_Window

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	1	1	0	P	WD_Window_3	WD_Window_2	WD_Window_1	WD_Window_0	Secure_3	Secure_2	Secure_1	Secure_0
MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	WD_Window_3	WD_Window_2	WD_Window_1	WD_Window_0

WD_Window_3:0	Description	Configure the couple of IO_1:0 as safety inputs
	0000	Disable
	0001	1.0 ms
	0010	2.0 ms
	0011	3.0 ms
	0100	4.0 ms
	0101	6.0 ms
	0110	8.0 ms
	0111	12 ms
	1000	16 ms
	1001	24 ms
	1010	32 ms
	1011	64 ms
	1100	128 ms
	1101	256 ms
	1110	512 ms
	1111	1024 ms
	Reset Description	Power On Reset

Figure 15 shows the refresh slot allowed during WD refresh

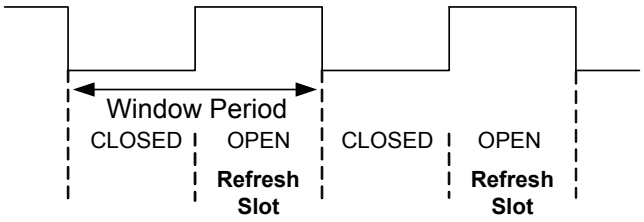

Figure 15. WD Refresh Slot

The windowed watchdog is based on an 8 bits pseudo-random word generated thanks to a Linear Feedback Shift Register implemented in the SBC. A default LFSR value (0xB2) is available in the WD_LFSR register at startup and the MCU, during the SBC initialization phase, can read back the LFSR to start its own calculation and perform then the watchdog answer.

- Internal register:

  In the MC33907_8, a register can be checked during initialization phase or even in normal operation to read back the LFSR value. It is also possible for the MCU to write its own LFSR. the new LFSR will be taken by the SBC to perform then its own calculation.

  WD_LFSR register - **WD_LFSR_x** bits (where x=0 to 7) - Read or Write allowed.

When the MCU read back the LFSR from the MC33907_8, the MCU must start the calculation using simple formula. Refer to the MC33907_8 Data Sheet.

As soon as the result is available and when the window is open, the MCU must send the result to the SBC. The two results (MCU & SBC) will be then compared.

- Internal register:

  In the MC33907_8, a register is available to write the result of the simple calculation based on the LFSR.

  WD ANSWER register - **WD_answer_x** bits (where x=0 to 7) - Read or Write allowed.

Table 10 shows when a Watchdog answer is considered as good or wrong.

Table 10. Watchdog Error Table

		WINDOW	
		CLOSED	OPEN
SPI	BAD Key	WD_NOK	WD_NOK
	Good Key	WD_NOK	WD_OK
	None (Timeout)	No_issue	WD_NOK

Three counters are involved each time a good or wrong watchdog refresh is performed. Refer to the MC33907_8 Data Sheet to understand how they interact each others).

- WD_error counter
- WD_refresh counter
- Reset error counter

NOTE

After consecutive bad watchdog refresh, the MC33907_8 will switch the system in Fail-safe sate when reset error counter reaches intermediate level. Then, any correct watchdog refresh is also monitored to allow the MCU to “get out” from a fail-safe state because the MCU behaves again as expected.

Too much consecutive bad watchdog refresh (if reset error counter reaches final value) will definitively switch the system to deep fail-safe mode. Only a key off/Key on sequence at system level can help to recover the situation. Refer to the MC33907_8 Data Sheet.

6.4 Safety Outputs - FS0B, RSTB

The safety outputs are used to switch the system in Fail-safe state (Safe $state_{system}$).

6.4.1 RSTB

RSTB is a dedicated active low signal integrated in the MC33907_8 to bring the MCU under RESET, in case of an SBC internal fault or a fault reported by the system.

Assumption: an output in high-impedance is not considered safe at system level, it is assumed that external components connected to RSTB are available to bring the safety critical outputs to known level during operation.

Rationale: to bring at anytime the functional safety-critical outputs to a defined voltage level

Implementation: an external pull-down capacitor (filtering) and an external pull-up resistor must be connected to the right voltage rail (5.0 V or 3.3 V).

Figure 16 shows the connection of external components to insure good safety operation of RSTB

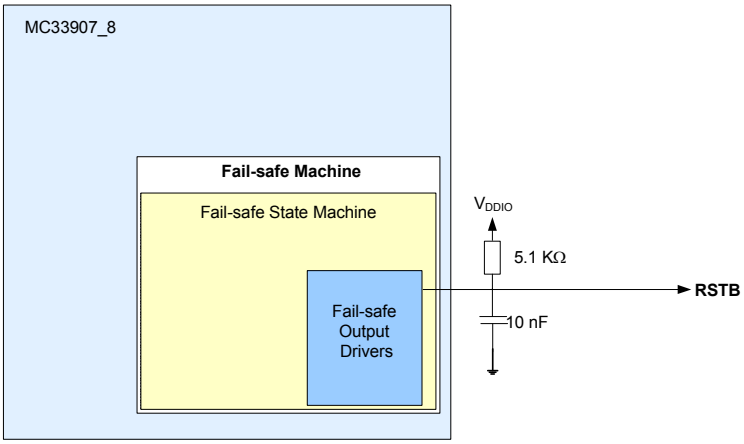

Figure 16. External Components on RSTB

The duration of the reset is configurable during initialization phase of the SBC.

- Internal register:

  In the MC33907_8, a register can be configured only during initialization phase to define the reset duration when it is asserted low.

  INIT FSSM1 register - **RSTB_low** bits (1.0 ms or 10 ms low level duration available).

  By default, the reset low duration time is settled to 10 ms

Table 11. INIT_FSSM1 - RSTB_low

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	1	0	0	P	IO_01_FS	IO_1_FS	IO_45_FS	RSTb_low	Secure_3	Secure_2	Secure_1	Secure_0

MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	IO_01_FS	IO_1_FS	IO_45_FS	RSTb_low

RSTB_Low	Description	Configure the Reset low duration time
	0	10 ms
	1	1.0 ms
	Reset Condition	Power On Reset

An RSTB low pulse can also be requested by SPI if needed. This request will come from the MCU itself and is a software request.

- Internal register:

  In the MC33907_8, a write command can be send by the MCU to request a low reset pulse **RSTB_request** register - **RSTB_request** bit.
  By default, the **RSTB_request** bit is not activated.

Table 12. RSTB_request

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	1	0	1	0	P	0	0	Rstb_request	0	Secure_3	Secure_2	Secure_1	Secure_0

MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	0	0	0	0	0	0	0	0

RSTb_Request	Description	Request a RSTB low pulse
	0	No request
	1	Request a RSTB low pulse
	Reset Condition	Power On Reset / When RSTB is done

The RSTB pin is bi-directional so the MC33907_8 can bring the MCU under RESET and the MCU can maintain the RSTB low even if MC33907_8 is ready to release it. All the resets numbers asserted by the MC33907_8 are populated in the reset error counter.

The reset error counter manages the reset events and count the number of reset occurring in the system. This counter is incremented by 1 each time a reset is generated.

The reset error counter has two outputs values (intermediate and final). The intermediate output value is used to handle the transition from reset (RSTB is asserted low) to reset and fail where RSTB and FS0B are activated. The final value is used to handle the transition from reset and fail to deep reset and fail (Deep fail-safe mode) where all regulators are of, reset and FS0B are asserted low, and a power on reset or a transition on IO[0] is needed to recover.

Rationale: if reset error counter reaches its final value, it means a critical permanent issue is reported at stem level and the SBC will completely switch off the MCU, and will maintain the system in fail-safe state (Safe $state_{system}$).

implementation hint: In the MC33907_8 a register, INIT_FSSM2 can be configured during initialization phase for the intermediate and final values of the reset error counter.

- Internal register:
  INIT_FSSM2 register - **RSTB_err_FS** bit
  By default, the **RSTB_err_FS** bit is configured for an intermediate value = 3 and a final value = 6.

Table 13. INIT_FSSM2 - RSTB_err_FS

Write																
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
MOSI	1	1	0	0	1	0	1	P	RSTb_err_FS	IO_23_FS	PS	0	Secure_3	Secure_2	Secure_1	Secure_0
MISO	SPI_G	WU	CAN_G	Reserved	IO_G	Vpre_G	Vcore_G	Vothers_G	SPI_FS_err	SPI_FS_CLK	SPI_FS_Req	SPI_FS_Parity	RSTb_err_FS	IO_23_FS	PS	0

RSTB_err_FS	Description	Configure the values of reset error counter
	0	Intermadiate=3; final=6
	1	Intermadiate=1; final=2
	Reset Condition	Power On Reset

Conditions that can lead to an incrementation of the RSTB error counter and according to product configuration are

- Watchdog error counter = 6
- Watchdog refresh not OK during INIT phase or watchdog timeout
- IO_23 error detection FCCU
- Undervoltage
- Overvoltage
- FS0B shorted to VDD
- SPI DED
- Reset request by SPI (software request)
- External reset

6.4.1.1 Reset Error Counter at Startup or Resuming from LPOFF Mode

At startup or when resuming from LPOFF mode the reset error counter starts at level 1.

6.4.2 FS0B

FS0B is a dedicated active low signal integrated in the MC33907_8 to bring the system in fail-safe state (Safe $state_{system}$) when needed. This safety output can be used for opening the power supply line, opening a security MOSFET is a series with a Motor/Valve,...).

Assumption: an output in high impedance is not considered safe at system level, it is assumed that external components connected to FS0B are available to bring the safety critical outputs to a known level during operation.

Rationale: in order to bring at anytime the functional safety-critical outputs to a defined voltage level

Implementation: an external pull-up resistor must be connected to the right voltage rail (up to battery voltage).

Assumption: it is assumed resistor in series is well connected to FS0B.

Rationale: FS0B can be connected externally to the system. In that case, it must be robust against automotive transients that can appears on battery line.

Implementation: A resistor of 5.1 kOhm in series must be connected on FS0b.

Figure 17 shows the connection of external components to insure good safety operation of FS0B

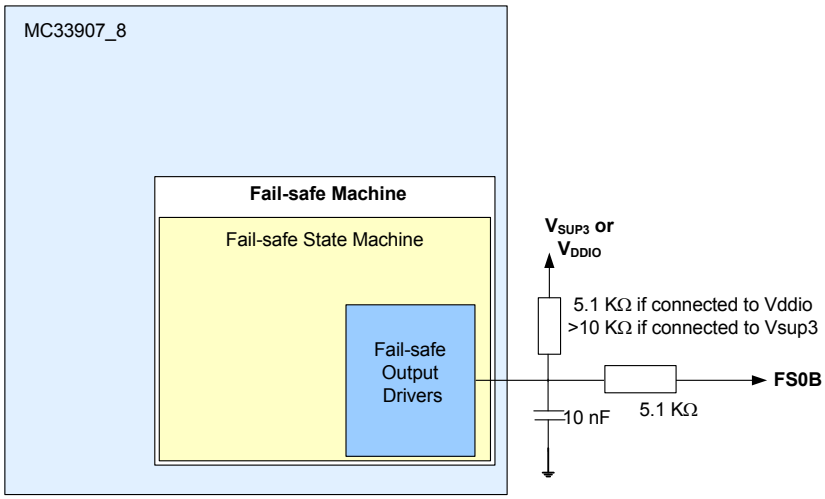

Figure 17. FS0B Connection

Condition leading to a Fail-safe activation are the following, but also depending of product configuration.

Table 14. List of Fail-safe Error Handling

Error Flag	Description	Main action in case of fault	RSTb	FS0b	Comments
Vpre_OV	Overvoltage on V_{PRE}	V_{PRE} switched OFF	LOW	LOW	-
Vcore_OV	Overvoltage on V_{CORE}	V_{CORE} switched OFF	LOW	LOW	Fail-safe impact selectable thru the SPI
Vcore_UV	Undervoltage on V_{CORE}	V_{CORE} kept ON	LOW	LOW	Fail-safe impact selectable thru the SPI
Vcca_OV	Overvoltage on V_{CCA}	V_{CCA} switched OFF	LOW	LOW	Fail-safe impact selectable thru the SPI
Vcca_UV	Undervoltage on V_{CCA}	V_{CCA} kept ON	LOW	LOW	Fail-safe impact selectable thru the SPI
Vaux_OV	Overvoltage on V_{AUX}	V_{AUX} switched OFF	LOW	LOW	Fail-safe impact selectable thru the SPI
Vaux_UV	Undervoltage on V_{AUX}	V_{AUX} kept ON	LOW	LOW	Fail-safe impact selectable thru the SPI

Table 14. List of Fail-safe Error Handling (continued)

IO_0:1	External IC error handling	-	HIGH	LOW	-
IO_2:3	MCU FCCU error handling	-	LOW	LOW	-
IO_4:5	External IC Error Handling	-	HIGH	LOW	-
WD	Watchdog	WD error counter / WD refresh counter	LOW	HIGH	-
RSTb shorted to High	Short circuit	-	HIGH (Externally)	LOW	-
SPI DED	Dual error detection in SPI	-	LOW	LOW	-
V2P5 main analog	Overvoltage on internal reference voltage for main V2P5 analog	-	LOW	LOW	-
V2P5 main digital	Overvoltage on internal reference voltage for main V2P5 digital	-	LOW	LOW	-
V2P5 FS analog	Overvoltage on internal reference voltage for Fail-safe V2P5 analog	-	LOW	LOW	-
V2P5 FS digital	Overvoltage on internal reference voltage for Fail-safe V2P5 digital	-	LOW	LOW	-
LBIST	Logic Built In Self Test	Keep device stuck in reset	LOW	HIGH	-
ABIST	Analog Built In Self Test	Bring system in Fail-safe	LOW	LOW	-

6.5 Built-in Hardware Self Tests (BIST)

Built-in hardware self-test (BIST) is a mechanism that permits circuitry to test itself.

Not every fault expresses itself immediately. For example, a fault may remain unnoticed if a component is not used or the context is not causing an error or the error is masked.

If faults are not detected over a long time (latent faults), they can pile up once they propagate.ISO 26262 requires 90% latent-fault metric for ASIL D, 80% for ASIL C, and 60% for ASIL B. Typically hardware assisted BIST is therefore used as safety integrity measure to detect latent faults.

The MC33907_8 is equipped with a built-in hardware self-test:

- Logic (LBIST, executed at startup, and going out from LPOFF mode)
  - During LBIST the device tests the functional logic of the fail-safe machine against stuck at fault
- Analog (ABIST, executed at start-up, and going out from LPOFF mode)
  - during ABIST the product tests the analog monitoring functions showed in Table 15:

Table 15. ABIST checks

Parameters	ABIST Checks			Comments
	Overvoltage	Undervoltage	OK / NOK	
V_{PRE}	X			
V_{CORE}	X	X		
V_{CCA}	X	X		
V_{AUX}	X	X		
V2P5 Main Digital	X			undervoltage not checked because undervoltage means power on reset state
V2P5 Main Analog	X			undervoltage not checked because undervoltage means power on reset state

Table 15. ABIST checks (continued)

Parameters	ABIST Checks			Comments
V2P5 Fail-safe Digital	X			undervoltage not checked because undervoltage means power on reset state
V2P5 Fail-safeS Analog	X			undervoltage not checked because undervoltage means power on reset state
Osc Fail-safe			X	
RSTb			X	Internal Sense path is checked for high and low level
FS0b			X	Internal Sense path is checked for high and low level

LBIST and ABIST are performed after the occurrence of a Power on reset of the SBC. All startup tests are executed before application software starts because during this time the SBC maintain its own reset and as a consequence keep the MCU in reset.

After approximately 16 ms the RSTB is released and operation can starts. If failed, the MC33907_8 will not leave Safe state$_{SBC}$. The RSTB stays low and the MCU never starts.

7 List of Fail-safe Errors and Potential Cascade Effects

Impacts on Fail-safe activation (RSTB and FS0B) depend also of the product configuration.

Table 16. List of Fail-safe Error Handling and potential cascade effect

Error Flag	Description	Main action in case of fault	RSTb	FS0b	Potential Cascade effect	RSTb	FS0b
Vpre_OC	Overcurrent	V_{PRE} switched OFF	HIGH	HIGH	Undervoltage reported on all regulators	**LOW**	**LOW**
Vpre_Ilim	Current limitation	duty cycle reduction	HIGH	HIGH	Undervoltage reported on all regulators	**LOW**	**LOW**
Vpre_OV	Overvoltage on V_{PRE}	V_{PRE} switched OFF	LOW	LOW	All regulators will be switched off	-	-
Vpre_UV	Undervoltage on V_{PRE}	V_{PRE} kept ON	HIGH	HIGH	Undervoltage reported on all regulators	**LOW**	**LOW**
VPRE_TSD	Thermal shutdown	V_{PRE} switched OFF	HIGH	HIGH	Undervoltage reported on all regulators	**LOW**	**LOW**
Vcore_OV	Overvoltage on V_{CORE}	V_{CORE} switched OFF	LOW	LOW	-	-	-
Vcore_UV	Undervoltage on V_{CORE}	V_{CORE} kept ON	LOW	LOW	-	-	-
Vcore_Ilim	Current limitation	duty cycle reduction	HIGH	HIGH	Undervoltage on V_{CORE}	**LOW**	**LOW**
Vcore_TSD	Thermal shutdown	Vcore switched OFF	HIGH	HIGH	Undervoltage on V_{CORE}	**LOW**	**LOW**
Vcca_OV	Overvoltage on V_{CCA}	V_{CCA} switched OFF	LOW	LOW	-	-	-
Vcca_UV	Undervoltage on V_{CCA}	V_{CCA} kept ON	LOW	LOW	-	-	-
VCCA_ILIM	Current limitation	-	HIGH	HIGH	-	-	-
Vcca_TSD	Thermal shutdown (internal Pmos)	V_{CCA} switched off	HIGH	HIGH	Undervoltage on V_{CCA}	**LOW**	**LOW**
Vaux_OV	Overvoltage on V_{AUX}	V_{AUX} switched OFF	LOW	LOW	-	-	-
Vaux_UV	Undervoltage on V_{AUX}	V_{AUX} kept ON	LOW	LOW	-	-	-
Vaux_Ilim	Current limitation		HIGH	HIGH	-	-	-

Table 16. List of Fail-safe Error Handling and potential cascade effect (continued)

Error Flag	Description	Main action in case of fault	RSTb	FS0b	Potential Cascade effect	RSTb	FS0b
Vaux_TSD	Thermal shutdown (internal reverse transistor)	V_{AUX} switched off	HIGH	HIGH	Undervoltage on V_{AUX}	**LOW**	**LOW**
Vcan _OV	Overvoltage on V_{CAN}	V_{CAN} switched off and CAN Physical layer off	HIGH	HIGH	-	-	-
Vcan _UV	Undervoltage on V_{CAN}	CAN physical OFF	HIGH	HIGH	-	-	-
Vcan_ILIM	Current limitation		HIGH	HIGH	-	-	-
Vcan_TSD	Thermal shutdown	V_{CAN} switched off and Physical layer OFF	HIGH	HIGH	-	-	-
IO_0:1	External IC error handling	-	HIGH	LOW	-	-	-
IO_2:3	MCU FCCU error handling	-	LOW	LOW	-	-	-
IO_4:5	External IC Error Handling	-	HIGH	LOW	-	-	-
WD	Watchdog	WD error counter / WD refresh counter	LOW	HIGH	Fail-safe low depending of intermediate value configuration and level on reset error counter.	-	-
RSTb shorted to High	Short-circuit		HIGH (Externally)	LOW	-	-	-
SPI DED	Dual error detection in SPI		LOW	LOW	-	-	-
LBIST	Logic Built In Self Test	Keep device stuck in reset	LOW	LOW	Deep fail-safe (RSTB = 8.0 s)	-	-
ABIST	Analog Built In Self Test	Bring system in Fail-safe	LOW	LOW	Deep fail-safe (RSTB = 8.0 s)	-	-

8 Acronyms and abbreviations

A short list of acronyms and abbreviations used in this document is summarized for completeness:

Table 17. Acronyms and Abbreviations

Terms	Meanings
ABIST	Analog Built-in Self-test
ADC	Analog-to-Digital Converter
BIST	Built In Self Test
CCF	Common Cause Failure
CF	Cascading Failure
CMF	Common Mode Failure
DPF	Dual-point fault
FCCU	Fault Collection and Control Unit
FMEDA	Failure Modes, Effects & Diagnostic Analysis
FSM	Fail-safe Machine
FSO	Fail-safe Outputs
FSSM	Fail-safe State Machine
FTTI	Single-Point Fault Tolerant Time Interval
GPIO	General Purpose I/O
L-FTTI	Latent - Fault Tolerant Time Interval
LBIST	Logic Built-In-Self-Test
LF	Latent Fault
LFSR	Linear Feedback Shift Register
MCU	Microcontroller Unit
MPF	Multiple-point fault
OV	Overvoltage
PST	Process Safety Time
RF	Residual Fault
SBC	System Basis Chip
SF	Safe Fault
SPF	Single-Point Fault
UV	Undervoltage

9 Document Revision History

Table 18 summarizes revisions to this document.

Table 18. Revision history

Revision	Date	Description of changes
1.0	3/2014	• Initial release

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Information in this document is provided solely to enable system and software implementers to use Freescale products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document.

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer’s technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Document Number: MC33907_8SMUG
Rev. 1.0
3/2014