AN4437
Application note

L9678 user configurable airbag

Introduction

This document explains the features and benefits of the L9678 device, target for entry level airbag system, with flexible configuration for power supply and management.

The configuration of the device depends on the specific application.

Guidelines for different operating modes of the device are provided.

Meaning features are the flexible configuration, availability of different voltage regulators, two PSI-5 sensor interfaces, four DC sensors interface, two GPOs, high or low level diagnostic test, arming managed following both internal or external safing engine, deployment profile selectable, 32 bit SPI communication.

Note: The following information is given as a hint for the implementation of the device only and shall not be regarded as a description or warranty of a certain functionality, condition or quality of the device.

Contents

List of tables

List of figures

1 Description

The L9678 IC is a system chip solution targeted for emerging market applications. Base system designs can be completed with the L9678, SPC560Px microcontroller and an on-board acceleration sensor or PSI5 sensor.

1.1 Main features

Main features are:

- Energy reserve voltage power supply
  - High frequency boost regulator, 1.875 MHz
  - Output voltage user selectable, 23 V or 33 V ±5%
- User configurable linear power supplies
  - 5.0 V and 7.2 V ±4% output voltages
  - External pass transistor
- Fully integrated 3.3 V ±4% linear regulator
- Battery voltage monitor and shutdown control with wake-up control
- System voltage diagnostics with integrated ADC
- Crossover switch
  - Crossover performance, max 3 Ω, 250 mA max.
- Squib deployment drivers
  - 4 channel HSD/LSD
  - 25 V maximum deployment voltage
  - 1.2 A @ 2 ms and 1.75 A @ 0.5/0.7 ms deployment profiles
  - Integrated safing FET linear regulator, 20 V nominal
  - Current monitoring
  - Rmeasure, STB, STG and leakage diagnostics
  - High and low side driver FET tests
  - Safing FET test
- User customizable safing logic
- Two channel PSI-5 remote sensor interface with SPI selectable switched/regulated output voltage (asynchronous mode)
- Four channel hall-effect, resistive or switch sensor interface
- ISO9141 transceiver
- Dual channel configurable high-side/low-side LED driver
- Watchdog timer
- Two integrated oscillators: 7.5/16 MHz
- COVRACT function to connect externally of IC VIN to reserve capacitor
- Temperature sensor
- 32 bit SPI communications
- Minimum operating voltage = 6 V
- Operating temperature, -40 °C to 95 °C
- Packaging - 64 pin

1.2 Application overview

Different configurations for the device are possible. The device configurations are selected via SPI.

Two different ways for the diagnostics, high level and low level, can be chosen.

In low level diagnostic an external logic takes care of all the required set-up necessary for the requested measurement.

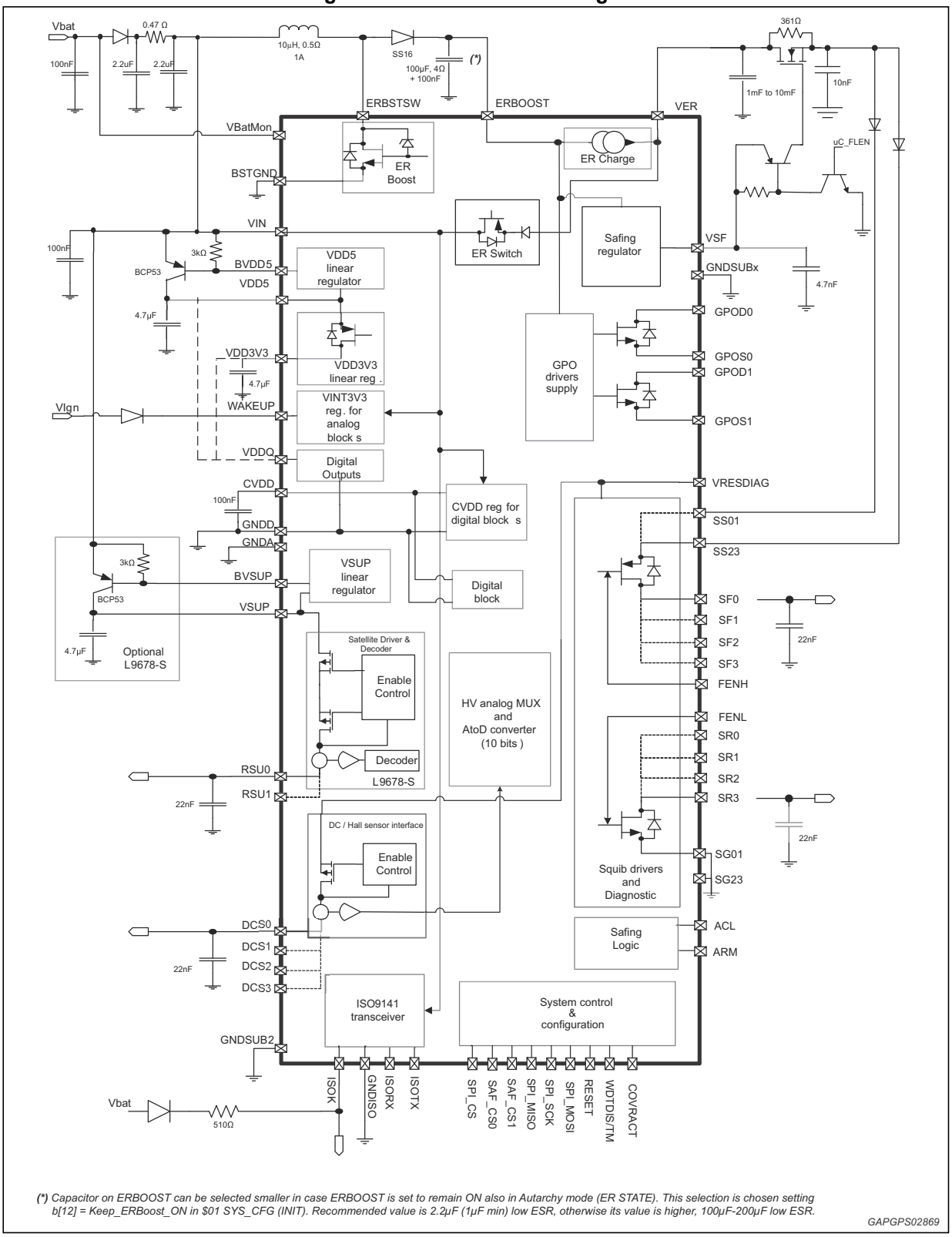
In high level diagnostic, the set-up for the measurement request is managed by the device itself.

The choice of high level or low level diagnostic is done via SPI.

The IC status during its running can be partitioned in 4 phases POWER OFF / SLEEP MODE / ACTIVE MODE / PASSIVE MODE.

This document explains in which way the device can be used.

Figure 1. Functional block diagram



(*) Capacitor on ERBOOST can be selected smaller in case ERBOOST is set to remain ON also in Autarchy mode (ER STATE). This selection is chosen setting b[12] = Keep_ERBoost_ON in $01 SYS_CFG (INIT). Recommended value is 2.2µF (1µF min) low ESR, otherwise its value is higher, 100µF-200µF low ESR.

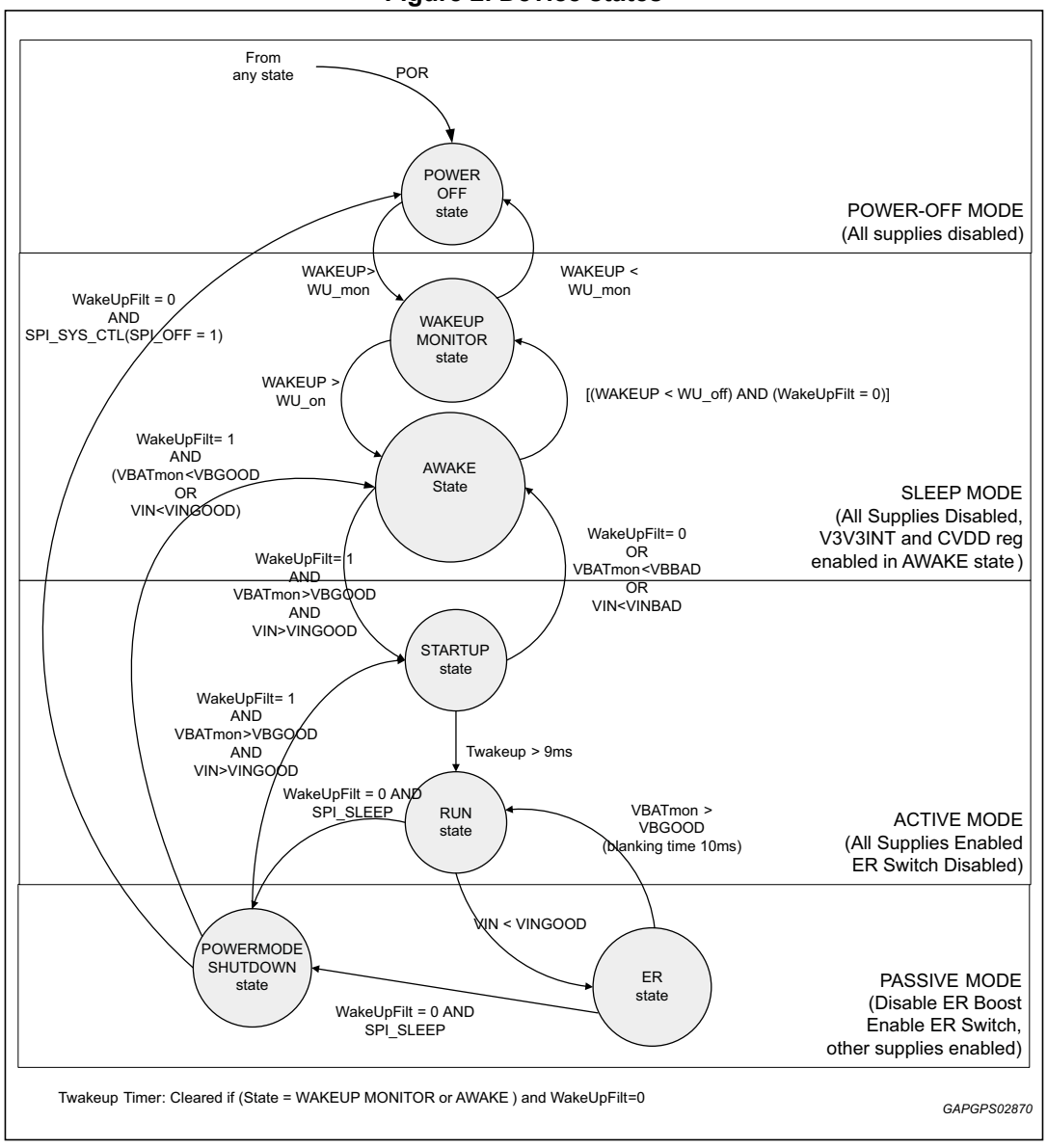
GAPGPS02869

2 Power up

Power off / sleep mode / active mode / passive mode

To pass from one mode to the others, see the flow here below:

Figure 2. Device states



Status of power control is readable via SPI:

$04 SYS_STATE	
POWER_CTL_STATE, bit [2:0]	000 = AWAKE 001 = START UP 010 = RUN 011 = ER 100 = POWER SHUT-DOWN 101, 110, 111 unused

2.1 Power off

This state represents the case of low power supply and POR.

No voltage regulators are enabled.

2.1.1 From power off to sleep mode

Condition to enter into the next phase (sleep mode):

WAKEUP>WU_mon WU_mon (1.5V max)

Power supplies are not necessary at their correct value.

2.2 Sleep mode

Supposing the external power supplies are increasing and WAKEUP pin connected to external power supply, WAKEUP pin raises too.

As WAKEUP pin is over the second threshold WU_on (4V÷5V), the two internal 3.3V voltage regulators run:

3V3INT is not accessible as it is internal only;

CVDD is available on pin 9

WakeUpFlt=1 means that WAKEUP pin is high at least 1 ms.

2.2.1 From sleep to active mode

Condition to enter into the next phase (Active mode):

(WakeUpFlt=1) & (VBATmon>VBGOOD) & (VIN>VINGOOD)

That means: the IC waits until WAKEPU is steadily high (at least 1ms) and the external power supplies have reached their correct values, VBGOOD for VBATmon pin and VINGOOD for VIN pin.

VBGOOD / VBBAD and VINGOOD/VINBAD are fixed via SPI (so through the microcontroller), writing SYS_CTL register at any time and the default value:

b12=0, VINGOOD = [5V:5.5V], VINBAD = [4.5V:5V]

b11, b10=00, VBGOOD = [5.5V:6V], VBBAD = [5V:5.5V]

$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
VIN_TH_SEL, bit 12	0 = 5.5V, VINGOOD = [5V:5.5V], VINBAD = [4.5V:5V] 1 = 7.5V, VINGOOD = [7V:7.5V], VINBAD = [6.5V:7V]
VBATMON_TH_SEL, bit 11, 10	00 = 6V, VBGOOD = [5.5V:6V], VBBAD = [5V:5.5V] 01 = 6.8V, VBGOOD = [6.3V:6.8V], VBBAD = [5.5V:6.3V] 10 = 8V, VBGOOD = [7.5V:8V], VBBAD = [7V:7.5V] 11 = 8.8V, VBGOOD = [8.3V:8.8V], VBBAD = [7.8V:8.3V]

2.2.2 From active back to sleep mode

Condition to turn back into the previous phase (POWER-OFF MODE):

WAKEUP<WU_mon WU_mon (1.5V max)

That means that WAKEUP turns low.

2.3 Active mode

Active mode corresponds to the normal IC operation.

Until POR is not released, ERBOOST follows VBATT; the two internal voltage regulators, VINT3V3 and CVDD are running.

As POR is released (see *Section 5.8: RESET*), ERBOOST starts running, followed by VDD5 and then by VDD3V3 (VDD3V3 is supplied by VDD5).

Once VDD5 and VDD3V3 are at their correct value, the microcontroller is supplied. IC RESET pin is released and the microcontroller starts running too.

Microcontroller first checks

In this phase the microcontroller shall service the watchdog routine, (see *Section 4.1: Initialization - Watchdog function*.

The microcontroller has to verify the correct connection of the external reserve capacitor on VER pin, enabling ER charge block

$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
ER_CUR_EN, bit 7	0 = ER current source OFF requested 1 = ER current source ON requested

and verify that VER voltage increases linearly (constant current).

VER voltage is readable through ADC. Registers involved in this operation are the four DIAGCTRL_x

$3x DIAGCTRL_x → x = A, B, C, D

Case x = A

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished
ADCREQ_A, bit [6:0] request (MOSI)	$26 = VER
ADCREQ_A, bit [16:10] readout (through MISO),	$26 = VER
ADCRES_A, bit [9:0]	10bit ADC result correspondent to the ADCREQ_A, bit [9:0]

Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio of the ADC. In case of VER, it is 15:1.

Table 1. VER measurement of the value ratio ADC

Measurements	Divider ratio				
	15:1	10:1	7:1	4:1	1:1
VER	√				

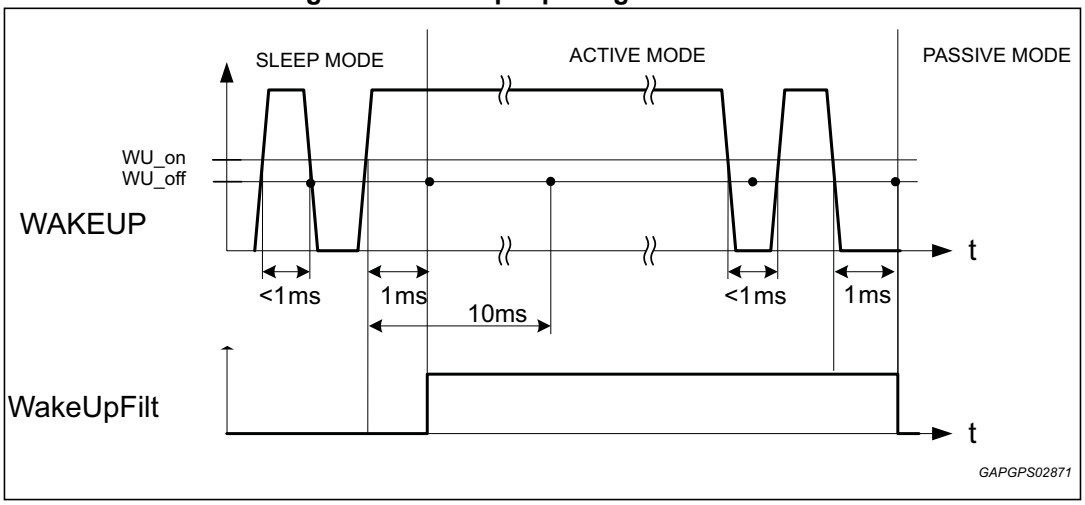
In case of problem, the microcontroller has to switch off the ER charge.

Note: ER charge is limited at 20mA, so any load connected to VER (VRESDIAG for example) subtracts current to the external reserve capacitor charge.

As shown in *Figure 3*, a 1 ms filter time is implemented on the WAKEUP input signal.

After that 9 ms has been elapsed with no WAKEUP de-assertions, WAKEUP active condition is latched.

Figure 3. Wake-up input signal behaviour



2.3.1 From active to passive mode

Conditions to enter into the next phase (PASSIVE MODE):

1 - (WakeUpFlt=0) & SPI_SLEEP cmd

2 - VIN<VBGOOD VBGOOD (5.5 V÷6 V)

The first case represents the normal IC shut-down, required by the microcontroller. The microcontroller puts the WAKEUP pin low for at least 1ms and sends the dedicated SPI command:

$03 SPI_SLEEP	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
SLEEP_MODE, [15:0]	$3C95

The second case represents the event of low battery, because, for example, battery connection has been lost.

2.3.2 From active back to sleep mode

Condition to enter into the previous phases (SLEEP MODE):

(WakeUpFlt=0) OR (VBATmon < VBBAD) OR (VIN<VINBAD)

That means that WAKEUP is not steadily high for at least 1ms in the first 9ms, or the external power supplies are turned at low value:

VBATmon<VBBAD or VIN<VINBAD (4.5 V÷5 V)

being VBBAD = 5 V÷5.5 V and VINBAD = 4.5 V÷5 V.

2.4 Passive mode

In Passive mode (ER state) ERBOOST can be disabled or not; it depends on KEEP_ERBST_ON bit, SYS_CFG register:

KEEP_ERBST_ON=0 → ERBOOST disabled in ER state

KEEP_ERBST_ON=1 → ERBOOST not disabled in ER state

In PASSIVE MODE "ER charge" is disabled to decouple ERBOOST from VER.

There are two cases:

1. The first case is a driven switch off through an SPI_ SLEEP command and WAKEUP is low.

   The driven switch off can be received by the IC also during a low VIN voltage (second case). In this condition the system survival is guaranteed by the energy stored in the external reserve capacitor.

2. The second case is related to a low VIN voltage value. ER switch is automatically turned on to connect VIN to VER.

   The energy requirement, which may include the firing too, is taken from the external reserve capacitor, until the capacitor is depleted and POR happens.

   In this condition, current from the external reserve capacitor is limited by "ER switch" at 300 mA min, regardless VER value (22 V or 33 V).

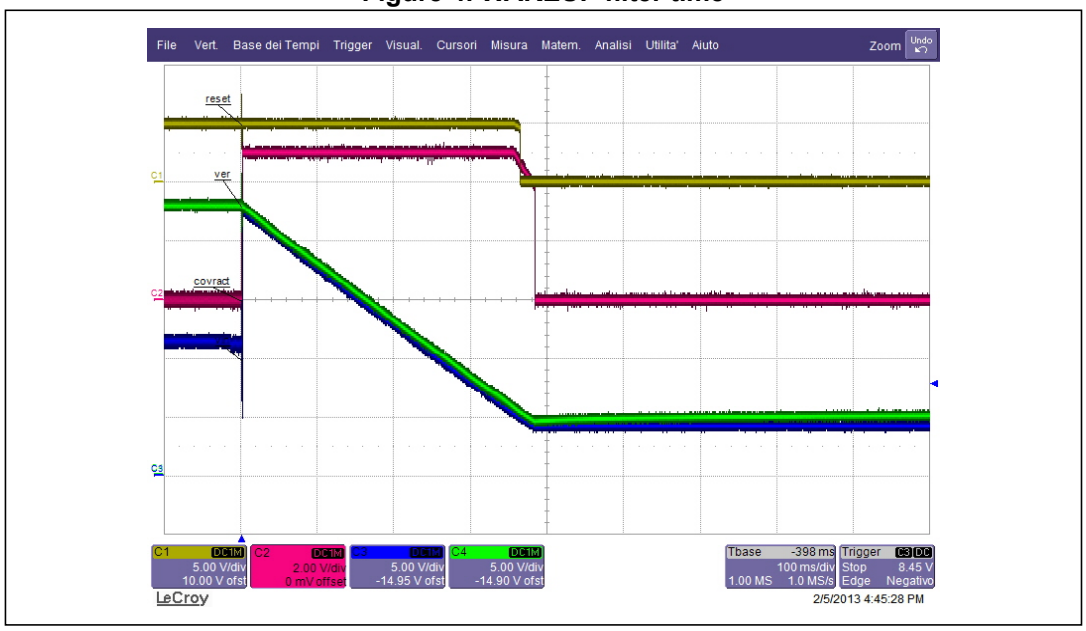
   The external reserve capacitor has to be chosen taking into account the highest energy requirement.

Note: If the capacitor on VER pin is not fully charged and VIN goes low, the external reserve capacitor could not be able to supply the system with the energy required.

2.4.1 COVRACT

When the IC moves in PASSIVE MODE, COVRACT signal is asserted.

Figure 4. WAKEUP filter time

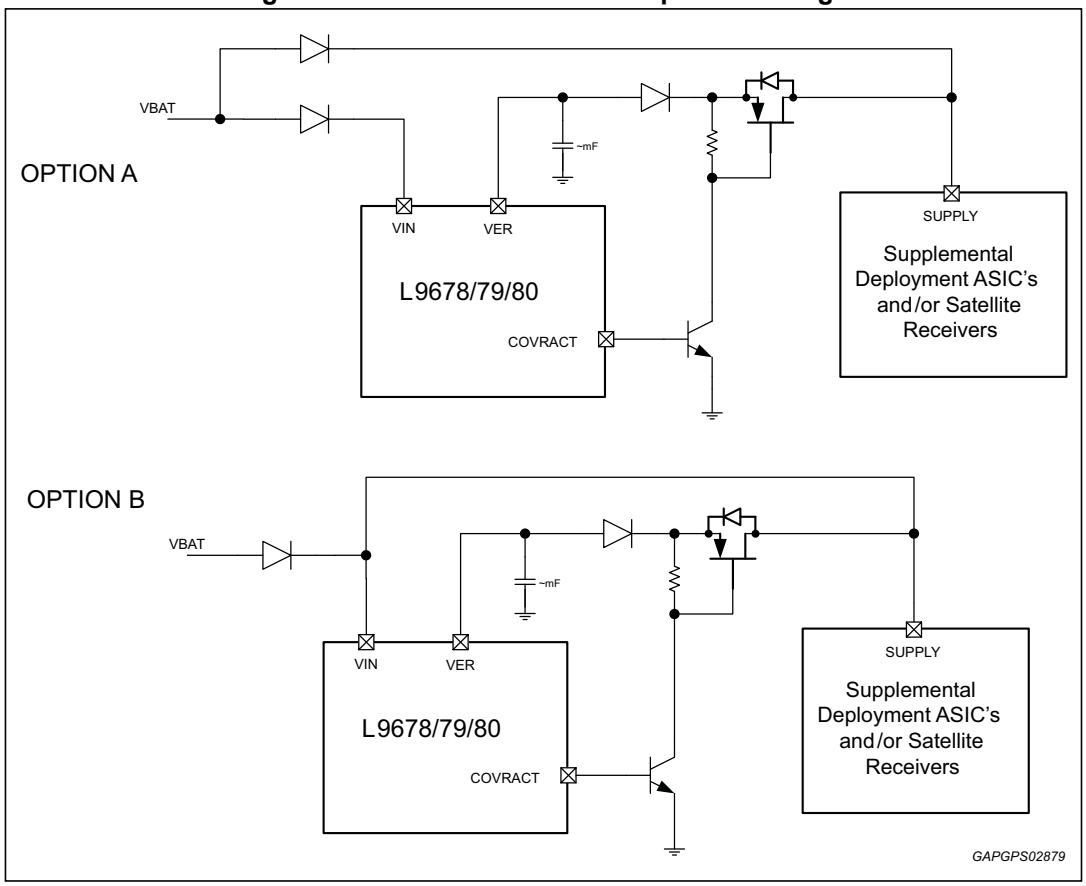


In *Figure 4* are reported signals when the IC moves in PASSIVE MODE because of battery lost: ch1 = RESET, ch2 = COVRACT, ch3 = VIN, ch4=VER.

In the next *Figure 5* two possible usages of COVRACT function are shown in case of supplementary deployment ASICs and/or satellite receivers to be supplied during Autarchy mode (battery loss).

Being ER SWITCH current limited (300-500 mA), in the above mentioned cases a supplementary cross over switch may be needed and the simplified external circuitry to be used is shown below.

Figure 5. COVRACT external components usage



The difference between options A and B is just the current that flows through the diode (active or passive) placed between battery line (VBAT) and VIN pin.

If the current consumption of additional ASICs is too high to permit the L9678 to work at low battery level, the option A is mandatory, otherwise option B is OK too.

2.4.2 From passive back to active mode

Condition to turn into the previous phases (ACTIVE MODE):

If the IC is in PASSIVE MODE due to low battery, as VBATmon turns at correct value (before POR event) the IC turns back into ACTIVE MODE

VBATmon >VBGOOD

If the IC has been passed in PASSIVE MODE due to a SPI request, to turn back in ACTIVE MODE it is necessary:

(WakeUpFilt=1) & (VBATmon >VBGOOD) & (VIN> VINGOOD)

2.4.3 From passive back to sleep mode

Condition to turn into the previous phases (SLEEP MODE):

If the IC is in PASSIVE MODE - power shutdown state, to turn back in SLEEP MODE it is necessary

(WakeUpFilt=1) & [(VBATmon <VBGOOD) OR (VIN< VINGOOD)]

2.4.4 From passive back to power off mode

Condition to turn into POWER OFF MODE:

If the IC is in PASSIVE MODE - power shutdown state, to turn back in POWER OFF MODE it is necessary a second SPI command

WakeUpFlt=0) & SPI_POWER OFF cmd

$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
SPI_OFF, bit 4	0 = no effect 1 = transition to POWER OFF

If this second SPI request is not received by the IC, that is in PASSIVE MODE due to a SPI SLEEP mode request (with correct value of power supply), the IC remains in PASSIVE MODE.

In the table here below disabled blocks in each IC state are shown:

Table 2. Blocks disabled in each IC state

	SLEEP MODE			**ACTIVE MODE**		**PASSIVE MODE**	
	POWER OFF	**WAKEUP**	**AWAKE**	**STARTUP**	**RUN**	**POWERMODE SHUTDOWN**	**ER**
Wakeup Detector	Disabled						
Intern reg.	Disabled	Disabled					

Table 2. Blocks disabled in each IC state (continued)

	SLEEP MODE			ACTIVE MODE		PASSIVE MODE	
	POWER OFF	WAKEUP	AWAKE	STARTUP	RUN	POWERMODE SHUTDOWN	ER
ERBOOST Reg.	Disabled	Disabled	Disabled			Disabled	Disabled
VSUP reg. (only L9678-S)	Disabled	Disabled	Disabled				
ER CAP current charge	Disabled	Disabled	Disabled			Disabled	Disabled
ER switch	Disabled	Disabled	Disabled	Disabled	Disabled		
VDD5 reg.	Disabled	Disabled	Disabled				
VDD3V3 reg.	Disabled	Disabled	Disabled				
Deployment drivers	Disabled	Disabled	Disabled				
VSF safing FET reg.	Disabled	Disabled	Disabled				
Remote Sensor Interf.	Disabled	Disabled	Disabled				
Watchdog	Disabled	Disabled	Disabled				
Diagnostic	Disabled	Disabled	Disabled				
DC sensor interf.	Disabled	Disabled	Disabled				
GPO drivers	Disabled	Disabled	Disabled				
Safing logic	Disabled	Disabled	Disabled				
ISO 9141	Disabled	Disabled	Disabled				

3 Power up and power down

3.1 Power up sequence

The aim of the procedure is to reach WakeUpFilt=1, VBATmon>VBGOOD, VIN>VINGOOD. Different cases are possible as:

WAKEUP rises after that VBAT is GOOD, *Figure 6*;

WAKEUP rises while VBAT is rising, *Figure 7*.

Figure 6. Example of power up sequence - VBAT then WAKEUP

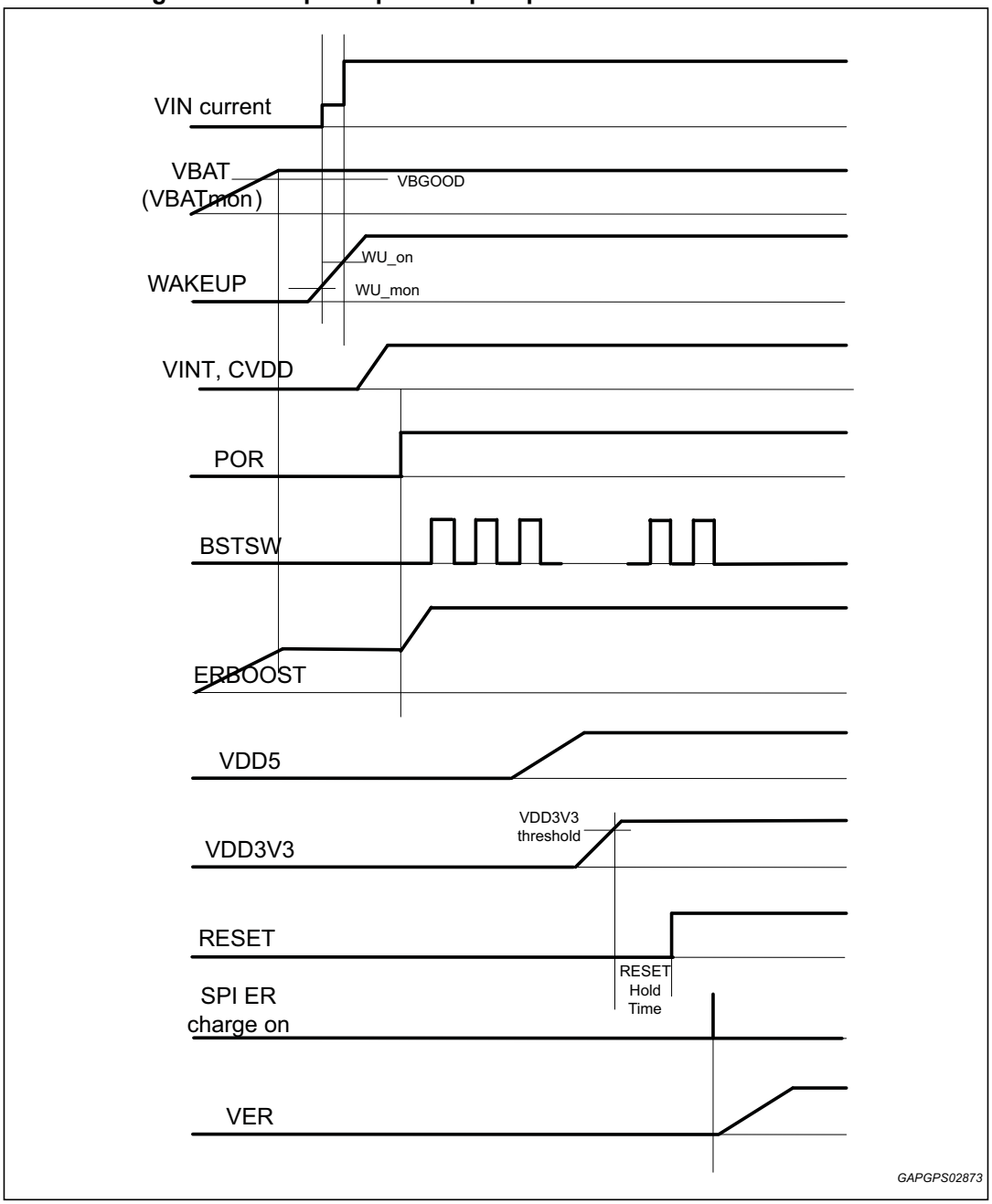
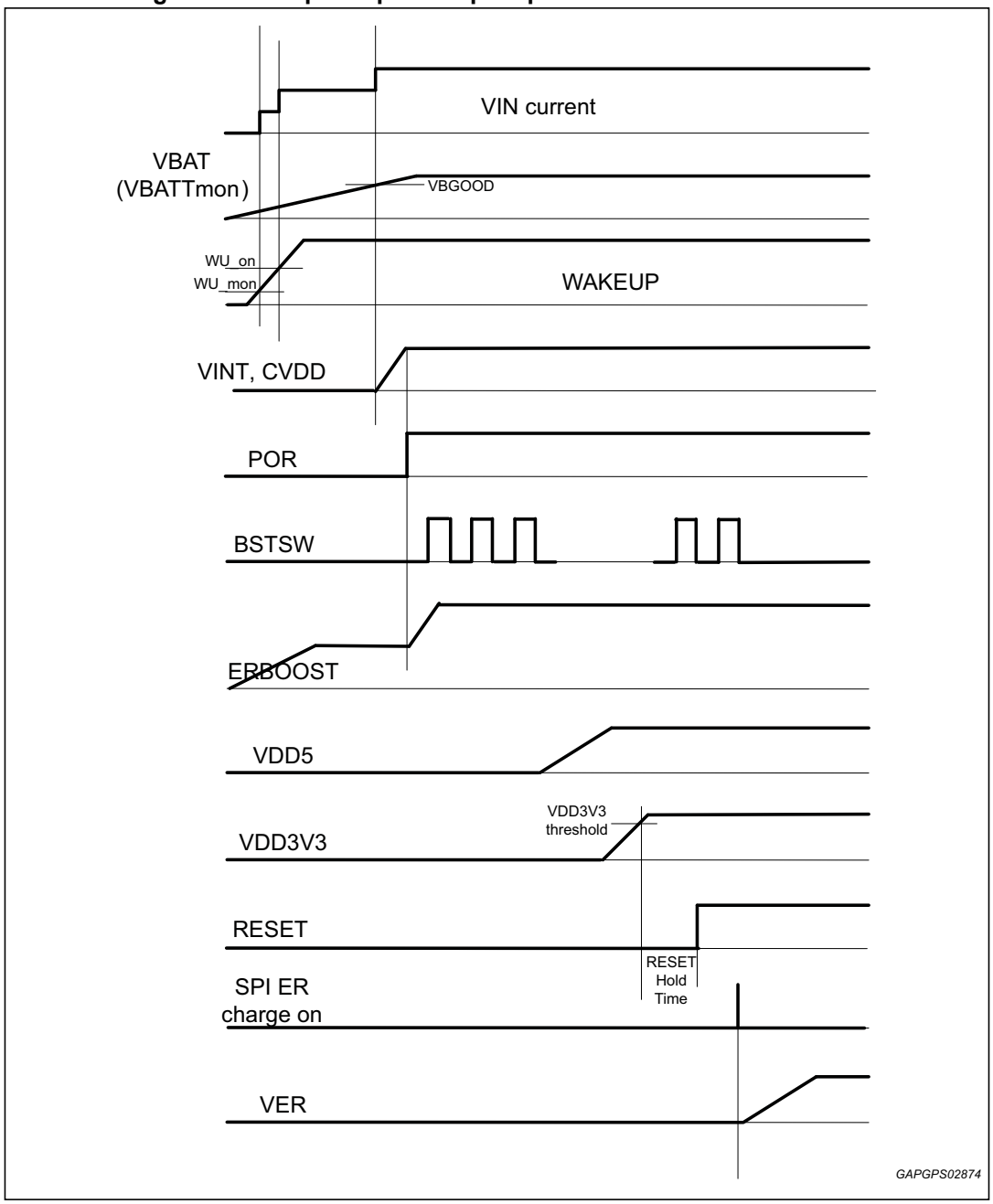


Figure 7. Example of power up sequence - WAKEUP then VBAT



At the end of the power up sequence the device is in ACTIVE MODE.

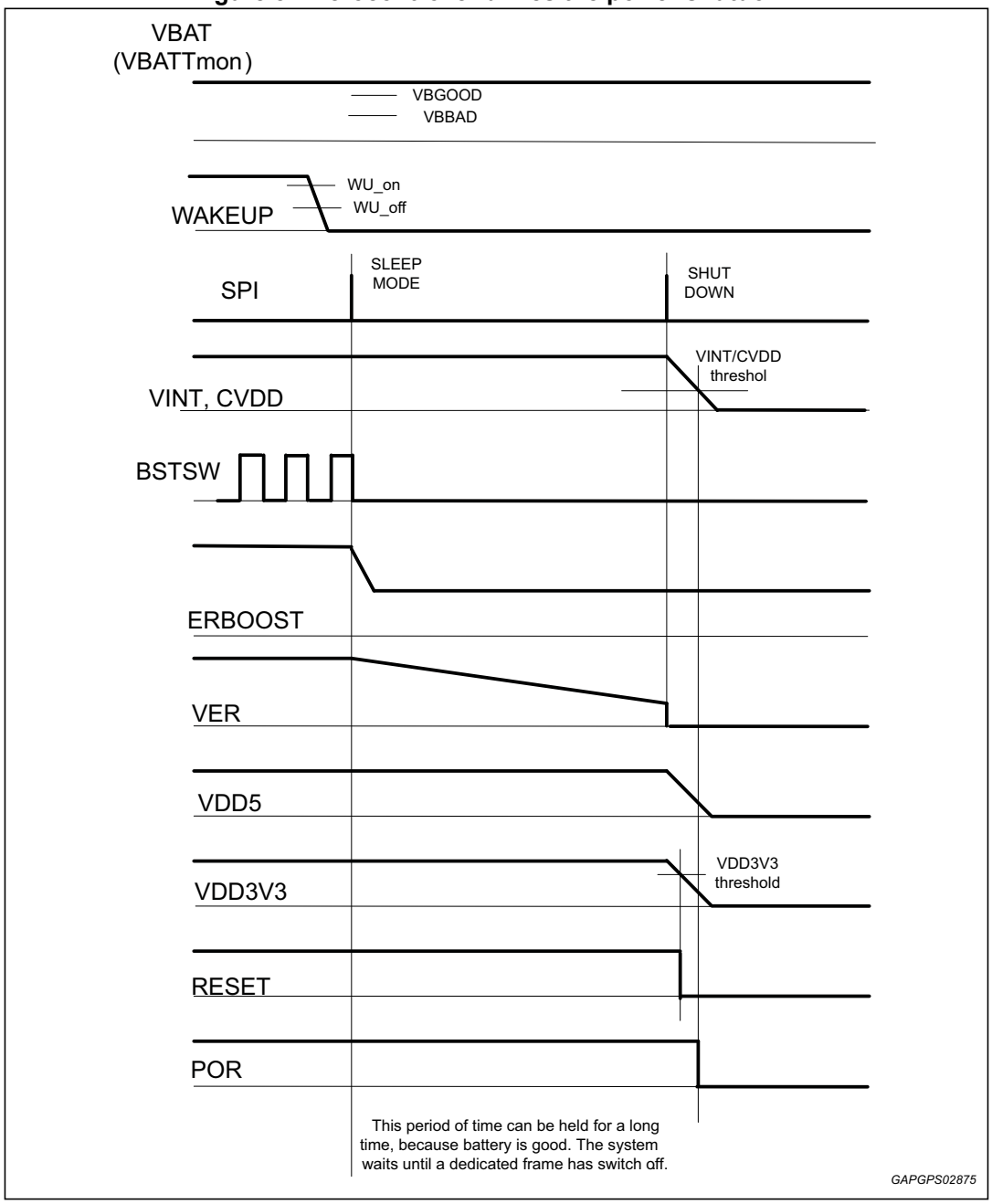
3.2 Power down sequence

Different possibilities are available.

Normal power shut down driven by microcontroller: first WAKEUP is driven low and then an appropriate command is sent (SLEEP MODE), *Figure 8*:

It could happen that the IC receives the first command, indicated as SLEEP MODE in the figure, but not the second. Being VIN at its correct value, the IC remains in that state. Microcontroller should recognize the bad situation and manage it.

Figure 8. Microcontroller drives the power shutdown

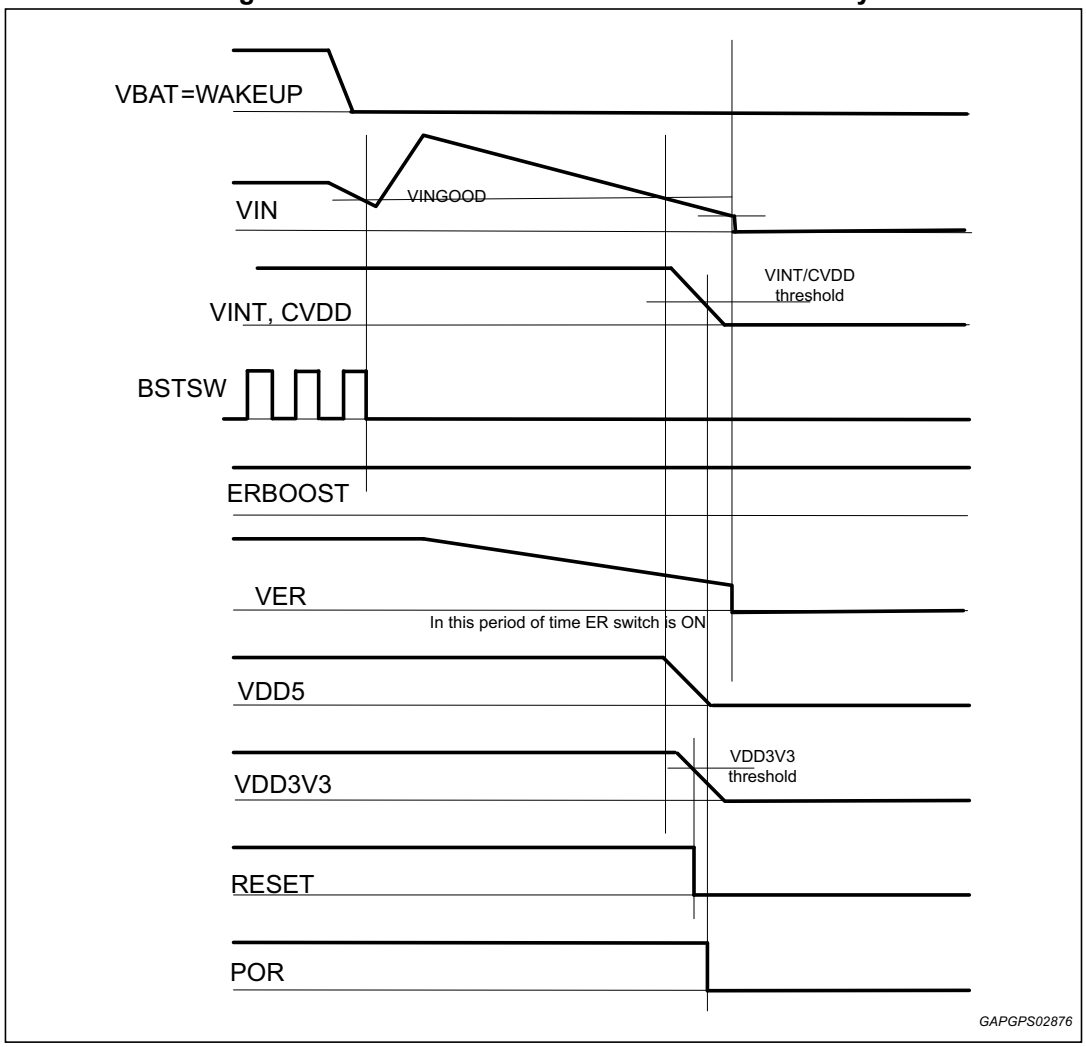


Another shut down is related to low battery. Shut down is related to low energy available, until POR happens, see *Figure 9*.

In this case "ER switch" is automatically closed. VIN is thus connected to VER whose voltage is close to ERBOOST. As consequence, the external reserve capacitor depletes, VER and VIN voltages are consequently reduced.

CVDD/VINT and VDD5/VDD3V3 switch off depends on their load. The higher is the current they have to furnish, the shorter is the time they are on.

Figure 9. Power shutdown due to low or lost battery



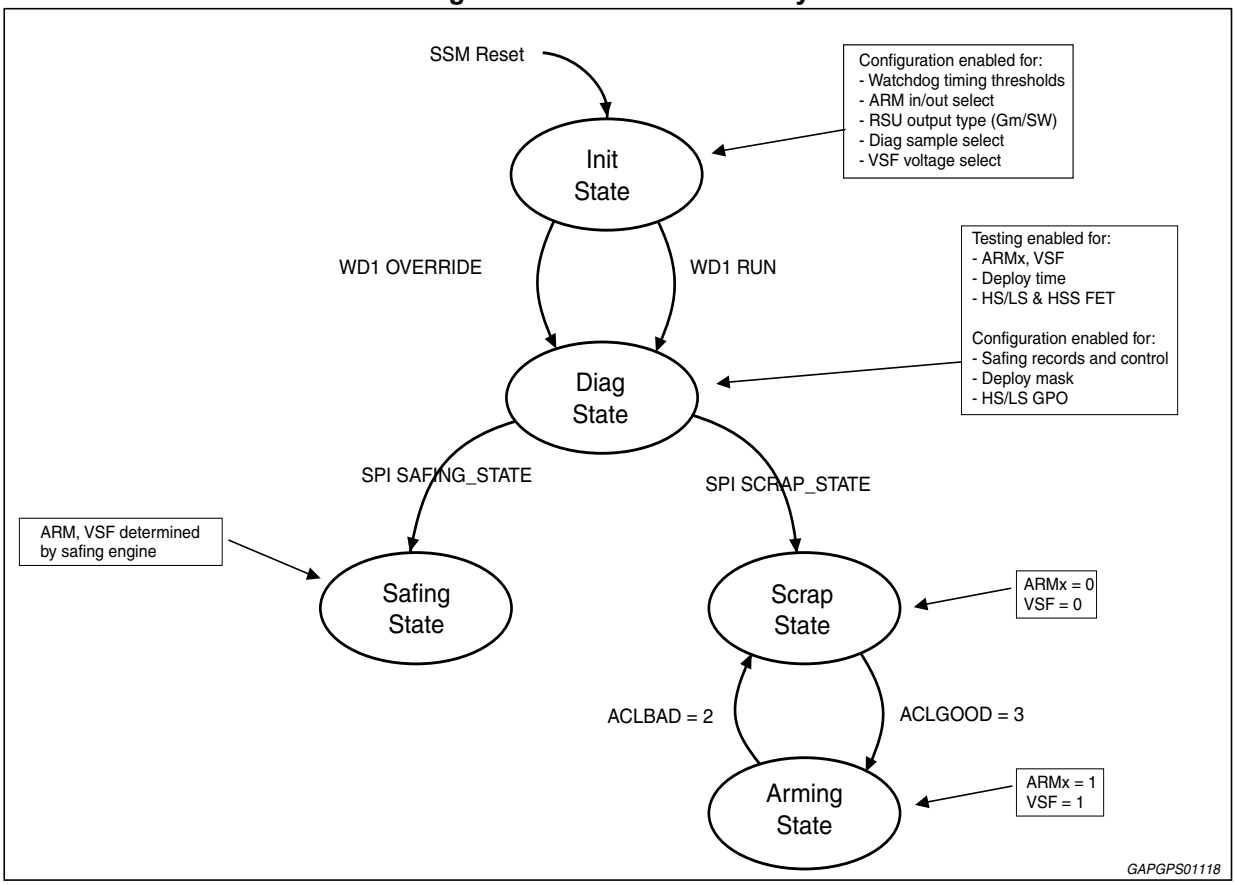
4 Operative state

As the device has been turned on, all voltage regulators run. Being VDD5, VDD3V3 present, the microcontroller is supplied, but still in RESET. As VDD5 and VDD3V3 are at their correct value and no fails are present (see *Section 5.8: RESET*), RESET pin is released and the microcontroller starts working.

The following states are distinguishable (see *Figure 10*):

Initialization / diagnostic / safing / scrap / arming

Figure 10. Device functionality



IC states are readable through SPI:

$04 SYS_STATE	
OPER_CTL_STATE, bit [10:8]	000 = INIT 001 = DIAG 010 = SAFING 011 = SCRAP 100 = ARMING 101, 110, 111 unused

4.1 Initialization - Watchdog function

The first state of the IC is when the microcontroller starts is the INITIALIZATION state.

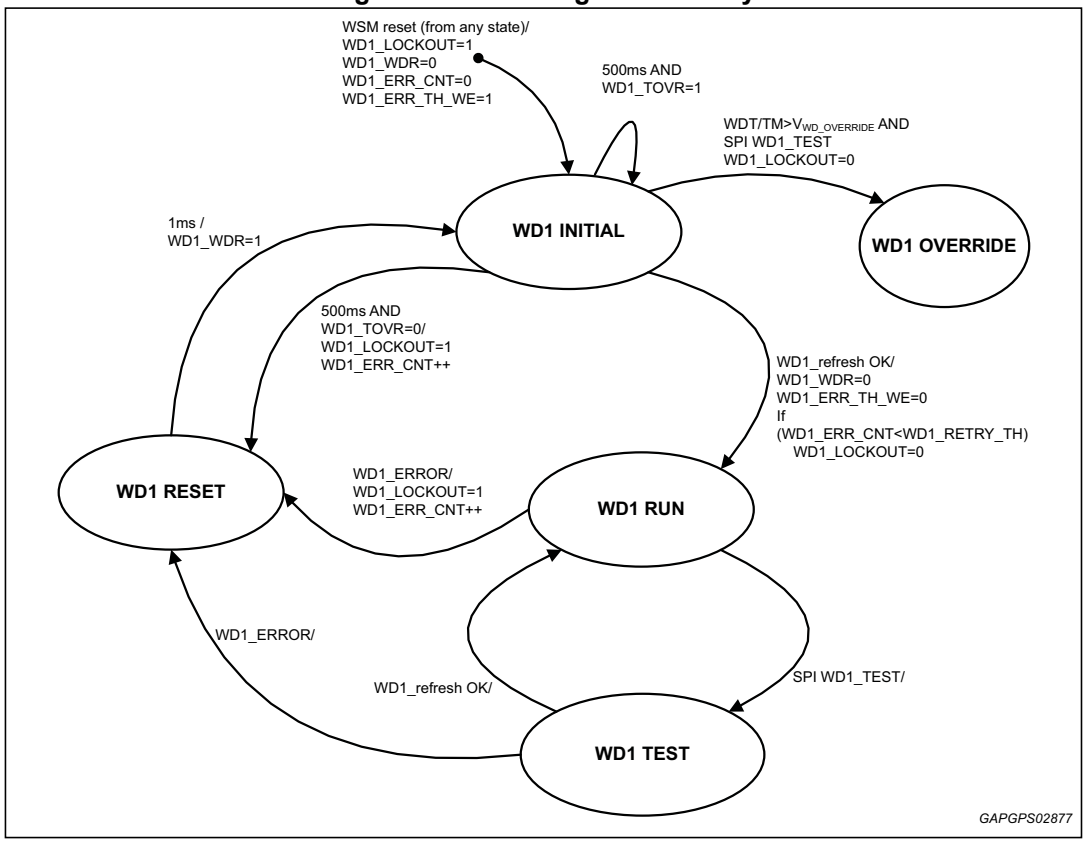
The state in which the IC is, is readable via SPI:

$04 SYS_STATE	
OPER_CTL_STATE bit [10:8]	000 = IC in INIT state

In this phase the following configurations are managed:

- Set-up of watchdog parameters, before the watchdog is asserted
- Safing engine parameters
- General system configuration
- VSF voltage

Figure 11. Watchdog functionality



Watchdog status is readable via SPI:

$2C WD_STATE	
WD1_STATE bit [10:8]	000 = INIT 001 = RUN 010 = TEST 011 = RESET 100 = OVERRIDE

4.1.1 WATCHDOG INITIAL

As WSM is released (see *Section 5.8: RESET*), watchdog is in its initial state, WD1 INITIAL. In WD1 INITIAL state all arming signals are disabled to prevent deployment. As entered in WD1 INITIAL, the counter of a first 500 ms time window is started.

In this phase, through WDTCR register, it is also defined the time window to service WD1. As the IC passes in DIAGNOSTIC state, WD1 parameters can't be modified any more.

WD_RETRY_CONF register $28 has also to be programmed during this phase in order to set the WD1_RETRY_TH bit: the meaning of such a threshold is explained in *Section 4.1.2: WD1 INITIAL - WD1 RESET*.

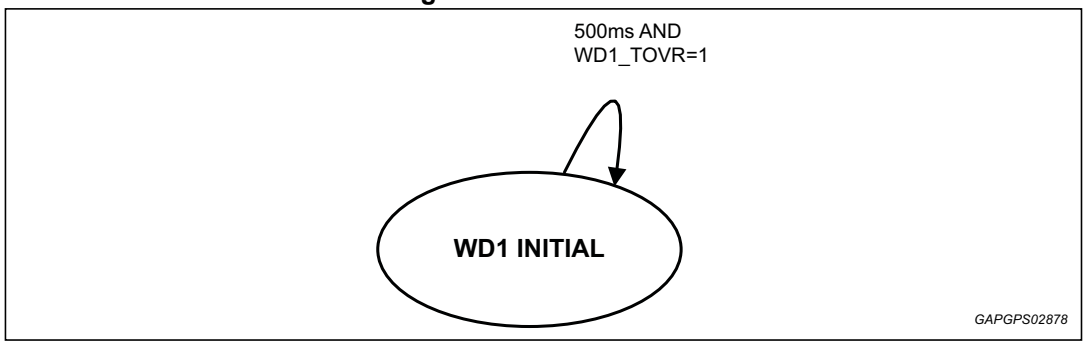
$2A WDTCR	Config. in INIT only state
WD1_MODE, bit 14	0 = fast WD1, 8 µs timer resol. (2 ms max value) 1 = slow WD1, 64 µs timer resol. (16.3 ms max value)
WDTMIN bit [13:7]	WD1 min time window. In according to WD1_MODE, $32 = 400 µs in fast mode. Update by WSM_RESET or dedicated SPI write
WDTDELTA bit [6:0]	WD1 delta value (WDTMAX=WDTMIN+MDTDELTA), in according to WD1_MODE, $19 = 200 µs in fast mode Update by WSM_RESET or dedicated SPI write

$28 WD_RETRY_CONF	Config. in INIT only state
WD1_RETRY_TH bit [2:0]	WD1 ERR_CNT threshold before setting definitively WD1_LOCKOUT bit.

Once the ASIC leaves the WD1INITIAL state, the register $28 cannot be anymore written (internal signal WD1_ERR_TH_WE is no more asserted and register is locked for writing access).

The watchdog block can remain indefinitely in INITIAL state, without any watchdog service, in case WD1_TOVR is set via SPI. In this case, RESET toggling (1ms RESET active each 500ms) is disabled allowing some very initial operations (i.e. microcontroller FW flash)

Figure 12. WD1 INITIAL



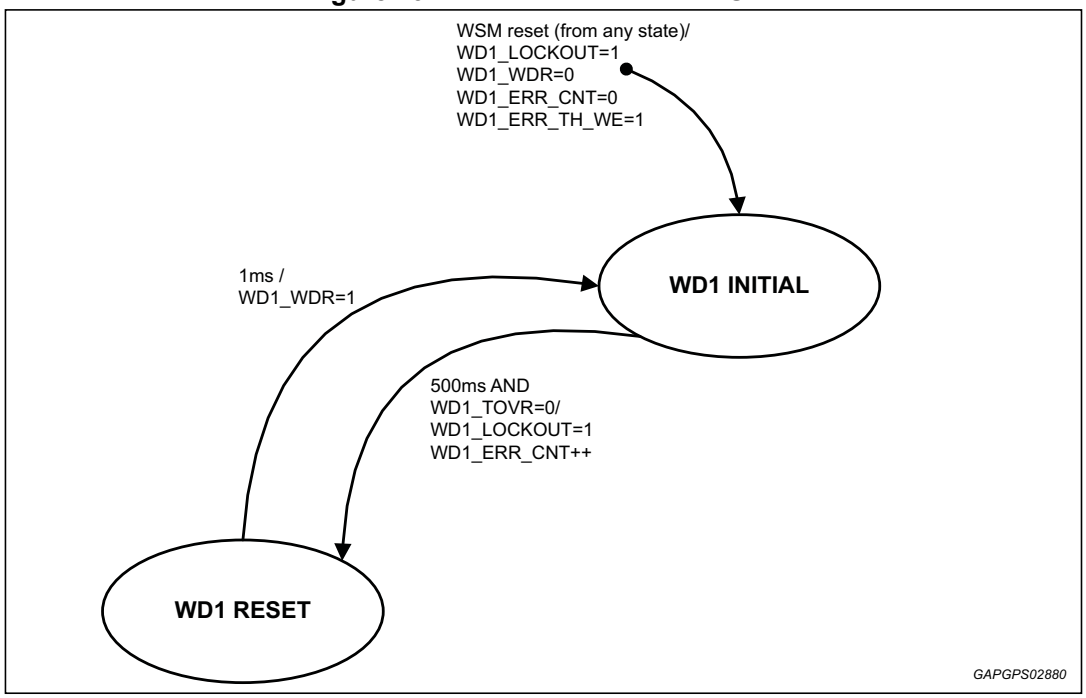
$01 SYS_CFG	Config. in INIT only state
WD1_TOVR, bit 0	0 = timeout is active 1 = timeout is disabled

Note: Disabling initial RESET toggling does not depend on the state of WDT/TM pin: SPI command is effective even if WDT/TM is grounded.

To pass to another WD1 state, from WD1 INITIAL, there are the following possibilities:

4.1.2 WD1 INITIAL - WD1 RESET

Figure 13. WD1 INITIAL - WD1 RESET



If neither WD1_TOVR is set in the first 500 ms nor WDOG service occurs, WD1_LOCKOUT bit is set and an error counter, WD1_ERR_CNT, is incremented. Then WATCHDOG block moves in reset, WD1 RESET.

WD1_ERR_CNT is readable via SPI:

$2C WD_STATE	
WD1_ERR_CNT bit [3:0]	Range 0:7

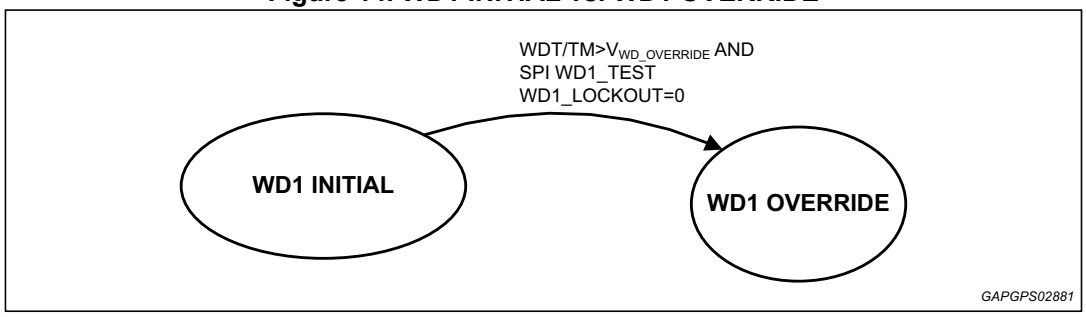
When WD1_LOCKOUT is set, all arming signals are disabled and deployment is inhibited.

WD1_LOCKOUT is readable on SPI:

$00 FLTSR	
WD1_LO, bit 7	0 = WD1_LOCKOUT is inactive 1 = WD1_LOCKOUT is active

4.1.3 WD1 INITIAL - WD1 OVERRIDE

Figure 14. WD1 INITIAL vs. WD1 OVERRIDE



The watchdog transition from WD1_INITIAL state to WD1_OVERRIDE state corresponds to the transition from INIT to DIAGNOSTIC mode of the IC, see *Figure 10*.

In order to enter this state, the pin WDT/TM must be biased to the voltage VWD_OVERRIDE and a proper SPI frame must be sent, accessing to the register

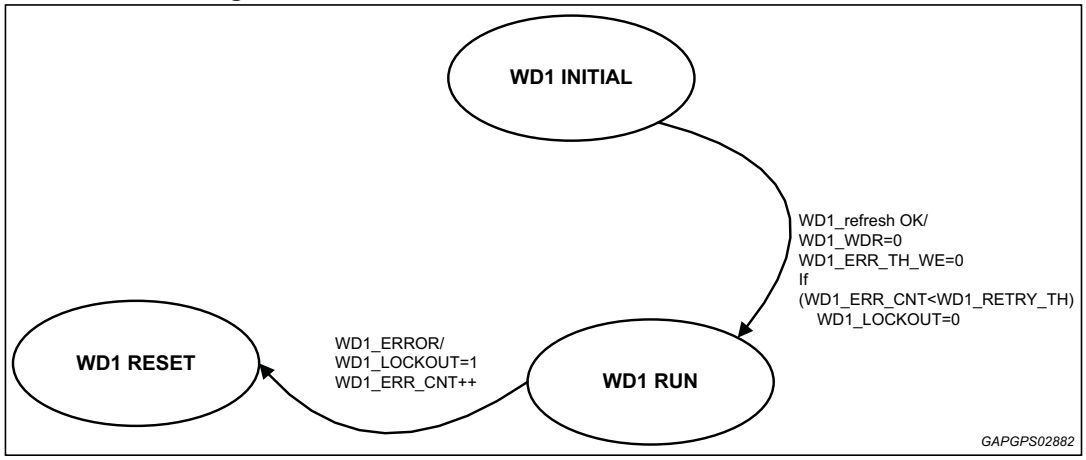
$35 WD_TEST	Config. in INIT, DIAG, SAFING state
Bit [15:8]	$3C to enter in WD_TEST

VWD_OVERRIDE = 10 V ÷ 14 V

4.1.4 WD1 INITIAL - WD1 RUN

When the watchdog routine is correctly serviced, the watchdog block enters in RUN MODE (*Figure 15*).

Figure 15. WD1 INITIAL vs. WD1 RUN vs. WD1 RESET



In the transition WD1_INITIAL state to WD1_RUN, WD1_ERR_CNT holds its value.

If WD1_ERR_CNT has reached its threshold, WD1_RETRY_TH, WD1_LOCKOUT bit remains set; lockout is automatically removed otherwise.

The number of allowed error cycle before permanently asserting the lockout is defined via SPI into the WD1_RETRY_TH.

$28 WD_RETRY_CONF	Config. in INIT
WD1_RETRY_TH bit [2:0]	WD1 ERR_CNT threshold before setting definitively WD1_LOCKOUT bit.

The watchdog transition from WD1_INITIAL state to WD1_RUN state corresponds to the transition from INIT to DIAGNOSTIC mode of the IC, see *Figure 10*.

Watchdog service is defined into register WD1T:

$2B WD1T	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
WD1CTL, bit[1:0]	00 OR 11 NOP 01 = code A 10 = code B

Correct watchdog service is controlled through a programmable time resolution counter, WD1_TIMER, whose parameters are defined in WDTCR register $2A in INIT state, (see *Section 4.1.1: WATCHDOG INITIAL*).

$2B WD1T	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
WD1_TIMER, bit[15:8]	8 bit wd counter

The counter WD1_TIMER, is reset every time a valid WD1CTL (watchdog service) is received.

If WD1CTL are serviced too early or too late with respect to the time window, the watchdog error counter, WD1_ERR_CNT, is incremented. WD1_LOCKOUT is set and watchdog status passes in WD1 RESET state, (see *Section 4.1.2: WD1 INITIAL - WD1 RESET*) for the management.

RESET pin is asserted with t_{wdrst} time:

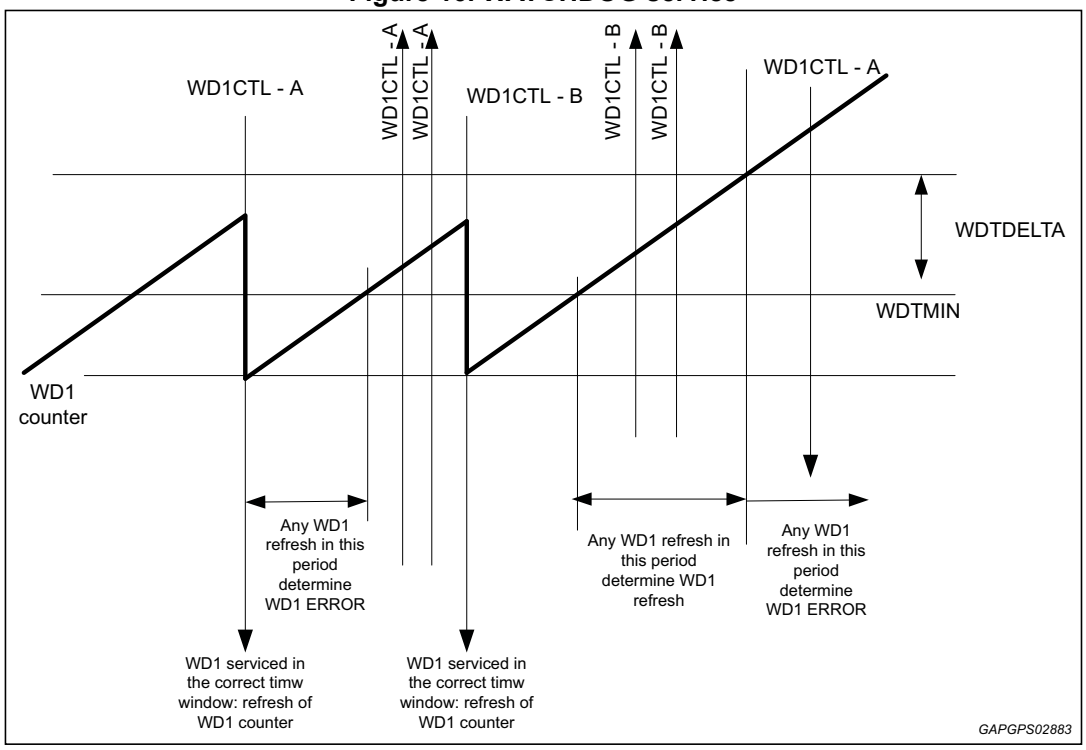
0.9ms < twdrst <1.1ms

Watchdog error, WD1 WDR, is readable via SPI:

$00 FLTSR	
WD1 WDR, bit5	0: WD1_WDR signal = 0, WD1 correctly serviced 1: WD1_WDR signal = 1 being WD1 not correctly serviced

In the follow *Figure 16* is sketched the WD1 service.

Figure 16. WATCHDOG service



If more than one WD1 with the same key value is received (for example A instead of B, see *Figure 16*), the WD1 counter is not refreshed until the correct key value is received in the defined time window (WDTMIN, WDTDELTA) and no error signals are asserted.

If more than one WD1 with the same key value (….A A A instead of …A and than B) is received no error signals are asserted even if the WD1 refresh command arrived before the counter has reached WDTMIN programmed value.

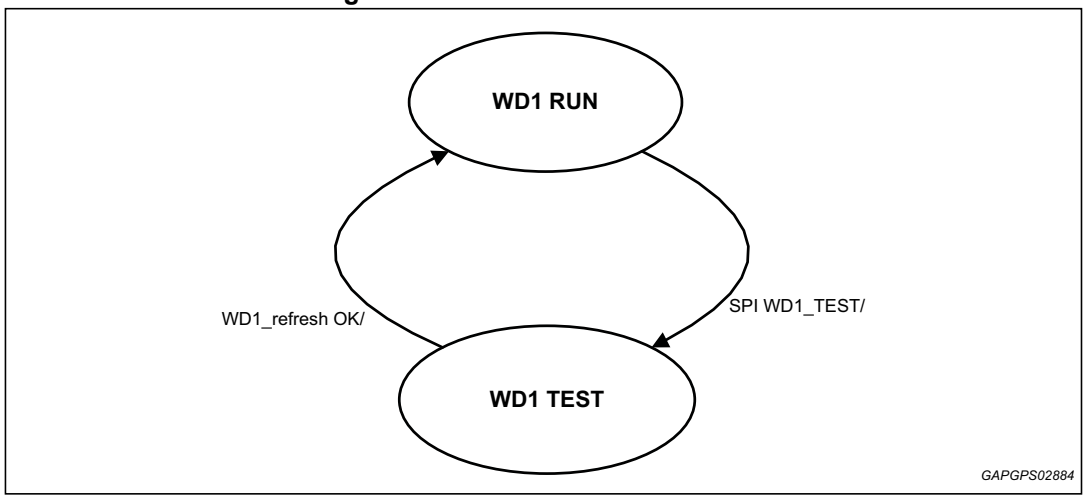
4.1.5 WD1 RUN - WD1 TEST

From WD1 RUN, through a SPI command, WD1 passes in WD1 TEST:

$35 WD_TEST	Config. in INIT, DIAG, SAFING state
bit [15:8]	$3C to enter in WD_TEST

This state generates a WD1_ERROR, without asserting WD1_LOCKOUT=1.

Figure 17. WD1 RESET - WD1 TEST



This is used to test the WD refresh. Typically it is implemented once per power up, even if there are no restrictions to accede to this mode in other moments.

In this state, deployments are not enabled.

Servicing WD1, watchdog turns back into WD1 RUN.

4.2 Diagnostic

Once WD1 is in WD1 RUN state or WD1 OVERRIDE state, the IC passes in DIAGNOSTIC state, see *Figure 10*.

The state in which the IC is, is readable via SPI:

$04 SYS_STATE	
OPER_CTL_STATE bit [10:8]	001 = IC in DIAG state

In DIAGNOSTIC state, the remaining IC configuration is allowed:

Safing records;

Deployment masks.

In DIAGNOSTIC state, several tests of the device are allowed:

HS LS switch tests of the squib drivers (ignored out of this phase);

HS safing FET (ignored out of this phase);

Arming output to check for non stuck-at conditions on the pin and for configured firing time.

Note: ARM output and VSF test are mutually exclusive.

Diagnostic foresees two different modes, high level and low level diagnostic.

In high level diagnostic, set-up of each requested test is managed by the IC itself.

In low level diagnostic, set-up of each requested test is managed by the external logic (microcontroller).

The selection between high and low level diagnostic is done writing the appropriate value of DIAG_LEVEL (bit 15) $38 LPDIAGREQ register:

$38 LPDIAGREQ	Config. in DIAG, SAFING, SCRAP, ARMING state
DIAG_LEVEL, bit 15	0 = low level mode 1 = high level mode

Once in DIAGNOSTIC state there is the possibility to pass in SAFING state or in SCRAP state, through dedicated commands.

4.3 SAFING

To pass in SAFING from DIAG state it is necessary a dedicated SPI frame:

$31 SAFING_STATE	Config. in DIAG state only
bit [15:0]	$ACAC to enter in SAFING state

The state in which the IC is, is readable via SPI:

$04 SYS_STATE	
OPER_CTL_STATE bit [10:8]	010 = IC in SAFING state

Logic performs safing function through safing engine management.

Once in SAFING state, the IC returns to INITIALIZATION state only through a SSM_RESET.

4.4 SCRAP

To pass in SCRAP from DIAG state it is necessary a dedicated SPI frame:

$30 SCRAP_STATE	Config. in DIAG state only
bit [15:0]	$3535 to enter in SCRAP state

The state in which the IC is, is readable via SPI:

$04 SYS_STATE	
OPER_CTL_STATE bit [10:8]	011 = IC in SCRAP state

Once in SCRAP state, the IC returns to INITIALIZATION state only through a SSM_RESET.

In SCRAP state the microcontroller drives the transition to ARMING state and monitors Remote Sensor SPI interface (L9678-S). Safing logic is disabled.

4.5 ARMING

This phase is the last enabling for the deployment.

Special care has to be taken to control the entry into and exit from ARMING state. The mechanism is managed through ACL (Additional Communication Line).

In ARMING state the arming output is asserted.

Exit from this state occurs when a timeout happens without a correct ACL signal or SSM_RESET is asserted.

5 Voltage regulators

2 internal linear voltage regulators:

- 3V3INT for internal analog blocks
- CVDD for internal digital blocks. It requires an external filter capacitor (100 nF) on pin CVDD.

1 switching regulator, ERBOOST 23 V or 3 3V programmable via SPI

1 linear regulator VDD5 5 V

1 linear regulator VDD3V3 3.3 V

1 optional linear regulator VSUP 7 V (L9678-S)

1 linear regulator VSF 20 V, 25 V programmable via SPI

5.1 Internal voltage regulators

In SLEEP MODE, the two internal regulators, 3V3INT and CVDD, are switched on, so that the device is ready for the full activation.

The other voltage regulators are enabled in ACTIVE MODE.

Requirement: VIN>VINGOOD1max = 5.5 V

5.2 ERBOOST

Features

- Boost enable via SPI (default active)
- Boost works at 1.875 MHz, 2.13 MHz, 2 MHz, selectable via SPI.
- ERBOOST is 23 V as default value and it is configurable, via SPI, at 33 V.

SPI configuration:

$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
ER_BST_V, bit 9	0 = 23V selected (22.6v÷25V) 1 = 33V selected (31.65V÷35V)
ER_BST_EN, bit 6	0 = ER_BOOST OFF 1 = ER_BOOST ON - default

$2D CLK_CONF	Config. in INIT state only
ERBST_SEL, bit [1,0]	00 = 1.88Mhz 01 = 2.13Mhz 10 = 11 = 2Mhz

Activation

Requirement: VINGOOD1max = 5.5 V ≤ VIN ≤ 18 V

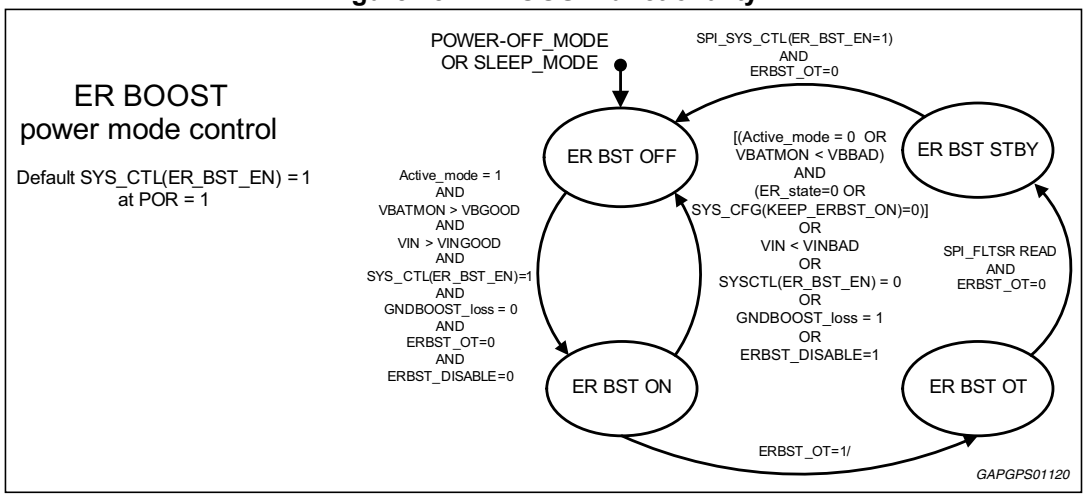
IC in ACTIVE mode

ERBOOST pin follows battery until POR is released. As POR is released, BOOST runs being as default active, and ERBOOST voltage increases up to 23 V (default) or 33 V, if selected, (see *Section 3.1: Power up sequence*).

Normal function

ERBOOST supplies VSF safing regulator, the GPO drivers and the internal current generator "ER charge" 30 mA typ. ER charge then charges the external reserve capacitor on VER pin.

Figure 18. ERBOOST functionality



Diagnostic

Diagnostic read out via SPI:

- ERBOOST status (ON or OFF)
- OVER/UNDERVOLTAGE

$05 POWER STATE	
ER_BST_NOK, bit 12	1 = V_ERBOOST<ERBOOST_OK 0 = V_ERBOOST>ERBOOST_OK
ER_BST_ON, bit8	1 = ERBOOST ON 0 = ERBOOST OFF or (OVERTEMPERATURE or STANDBY)

ERBOOST_OK = 18 V ÷ 22 V if ERBOOST = 23 V

ERBOOST_OK = 26 V ÷ 30 V if ERBOOST = 33 V

Protection

Short to battery determines an over-temperature event (readable via SPI) that switches off the regulator.

Overtemperature

$00 FLTSR	
ER_BST_OT, bit 17	0 = NO FAULT 1 = FAULT

The over-temperature event is latched in the register until it is read through the SPI.

To switch on again ERBOOST, it is necessary to enable again ERBOOST via SPI:

$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
ER_BST_EN, bit 6	0 = ER_BOOST OFF 1 = ER_BOOST ON

Ground loss determines the regulator's switch off. As soon as ground connection is restored, the regulator restarts automatically, without the ‘SWITCH ON’ SPI command.

After an under/over voltage detection, as soon as voltage turns to its correct value, the regulator restarts automatically, without the ‘SWITCH ON’ SPI command.

Threshold to deactivate ERBOOST:

ERBST_DISABLETH = VIN-ERBOOST

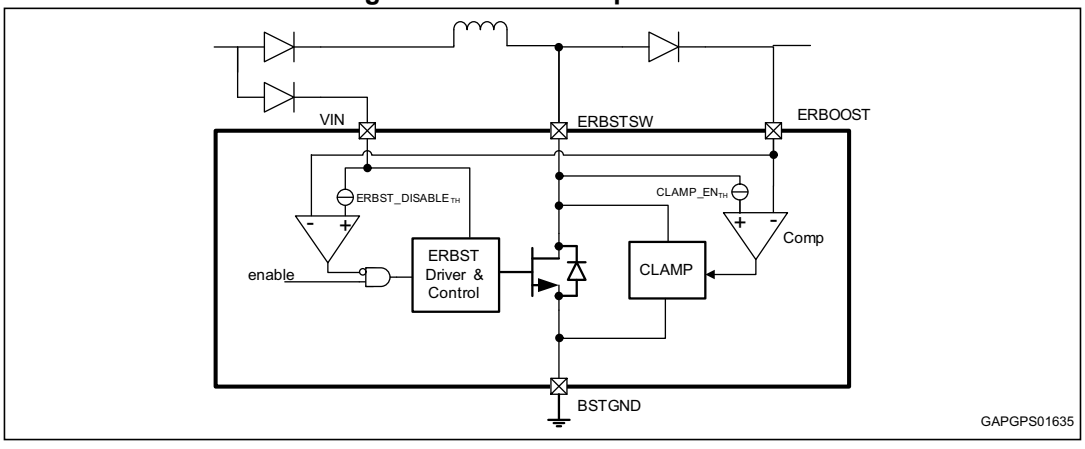
$$1.6\ V < ERBST\_DISABLE_{TH} < 2.5\ V$$

Threshold to activate the ER Boost CLAMP:

$CLAMP\_EN_{TH}$ = (ERBSTSW - ERBOOST)

$$1.6\ V < CLAMP\_EN_{TH} < 2.5\ V$$

Figure 19. ERBOOST protection



Deactivation

ERBOOST can be disabled via SPI:

$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
ER_BST_EN, bit 6	0 = ER_BOOST OFF 1 = ER_BOOST ON

In case of low or lost battery, the IC passes in PASSIVE MODE / ER state, the regulator can be switched off or not, depending on KEEP_ERBST_ON bit, SYS_CFG register:

$01 SYS_CFG	Config. in INIT
KEEP_ERBST_ON bit 12	0 = ERBOOST disabled in ER state 1 = ERBOOST not disabled in ER state

5.3 ER CHARGE - ER SWITCH

ER CHARGE

Features

- IER_charge 20 mA ÷ 40 mA
- RDS(on) max = 20 Ω

Activation

- Requirement: VINGOOD1max = 5.5 V ≤ VIN ≤ 35 V
- ERBOOST ≤ 8 V
- IC in ACTIVE mode

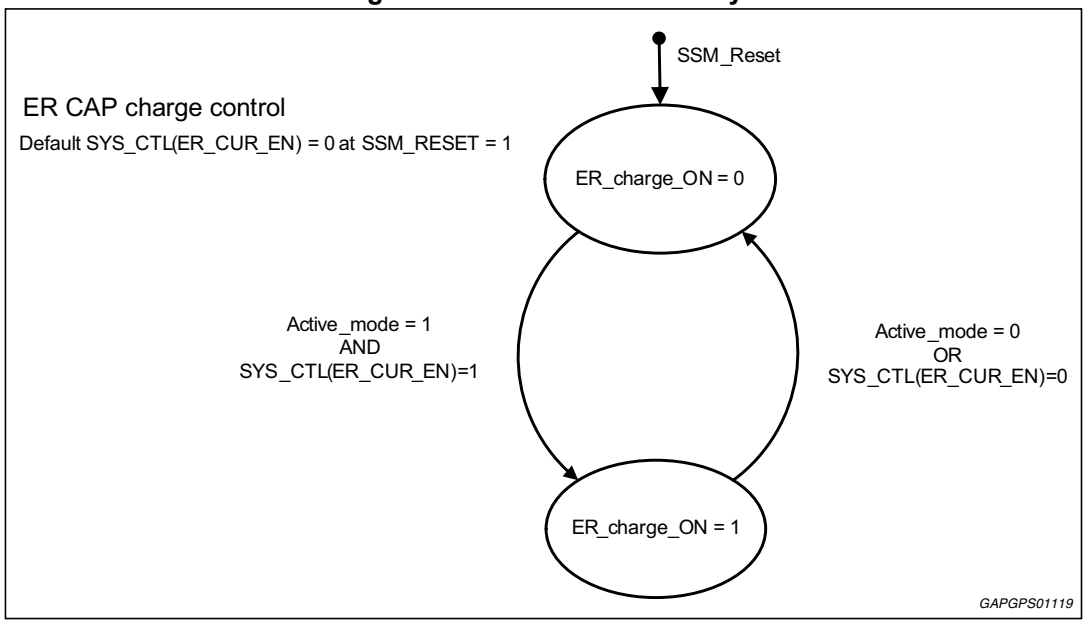
SPI command:

$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
ER_CUR_EN, bit 7	0 = ER current source OFF requested 1 = ER current source ON requested

Normal function

ER CHARGE charges the external reserve capacitor connected to VER pin.

Figure 20. ER CAP functionality



Note: ER CHARGE is able to deliver 20 mA (worse case) to charge the external capacitor. Any connections to VER pin determines an additive contribution in terms of current consumption. Due to the max current limitation of ER CHARGE block, each additive load on the pin lowers the current inside the external reserve capacitor increasing the time needed to fully charge it.

For example, connecting VRESDIAG to VER, during diagnostic or at the DCS switch on, the current required by VRESDIAG is higher than 20mA (the capacitor provides the extra current).

Diagnostic

ER charge status is readable on SPI:

$05 POWER STATE	
ER_CHRG_ON, bit 7	0 = ER charge ON 1 = ER charge OFF

Deactivation

Out of ACTIVE mode ER charge is switched off to decouple ERBOOST pin from VER

ER SWITCH

Features

- RDS(on) = 0.5 Ω ÷ 3 Ω
- current limitation at his activation (300 mA min / 500 mA max)

Activation

Requirement: VINGOOD1max = 5.5 V ≤ VIN ≤ 35 V

IC in PASSIVE mode.

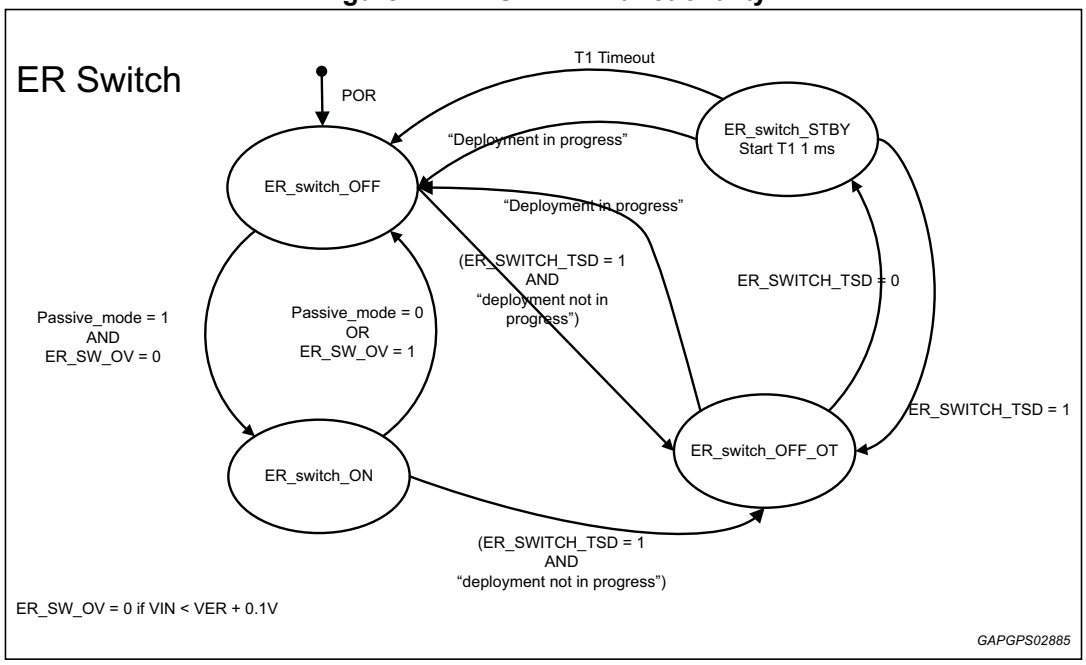
Normal function

When ER switch is on, VIN is supplied through ER switch by the external reserve capacitor.

The energy stored in the external reserve capacitor guarantees the energy supply for the device until POR occurs, including the deployment (if required) in case of battery low or lost too.

ER switch implements a current limitation (300mA:500mA) to avoid in-rush current as ER switch is enabled or in case of a short to ground.

Figure 21. ER SWITCH functionality



Protection

Dedicated thermal sensor, active when no deploy is running and monitored through ER_SW_TSD comparator.

In thermal shutdown, with no deployment running, ER_SWITCH is turned OFF so the energy taken from the external reserve capacitor is not available. In this condition the regulators are not any more supplied and POR occurs.

Comparator thresholds are:

150°C < TJSD_ERSW < 190°C

5°C < THYS_TSDERSW <15°C

ER switch status is readable on SPI:

$05 POWER STATE	
ER_SW_ON, bit 4	0 = ER SWITCH OFF 1 = ER SWITCH ON

To avoid continuously on and off cycling, there is a timeout T1, 1ms typ.

ER switch is protected against reverse biasing to avoid back-feeding of battery to the external reserve capacitor.

Deactivation

It is deactivated when the device exits from PASSIVE mode.

5.4 VDD5 linear regulator

Features

- 5 V linear regulator, slope controlled, derived directly from battery line through an external PNP (h_{FE} min 50).
- VDD5 output voltage: 4.85 V ÷ 5.15 V
- VDD5 load current 0.5 mA ÷ 200 mA
- VDD5 stability requires an external small capacitor 3 µF ÷ 31 µF.
- Current limitation is guaranteed controlling the output current on BVDD5 pin.

Activation

Requirement: VINGOOD1max = 5.5 V ≤ VIN ≤ 35 V

It starts in ACTIVE mode and continues to work in PASSIVE mode too, being supplied through ER switch by the external reserve capacitor.

As the regulator starts running, the under-voltage that occurs before the regulator has reached its correct value, it is not considered a fault (under-voltage fault) in the first 3ms after the regulator is switched on.

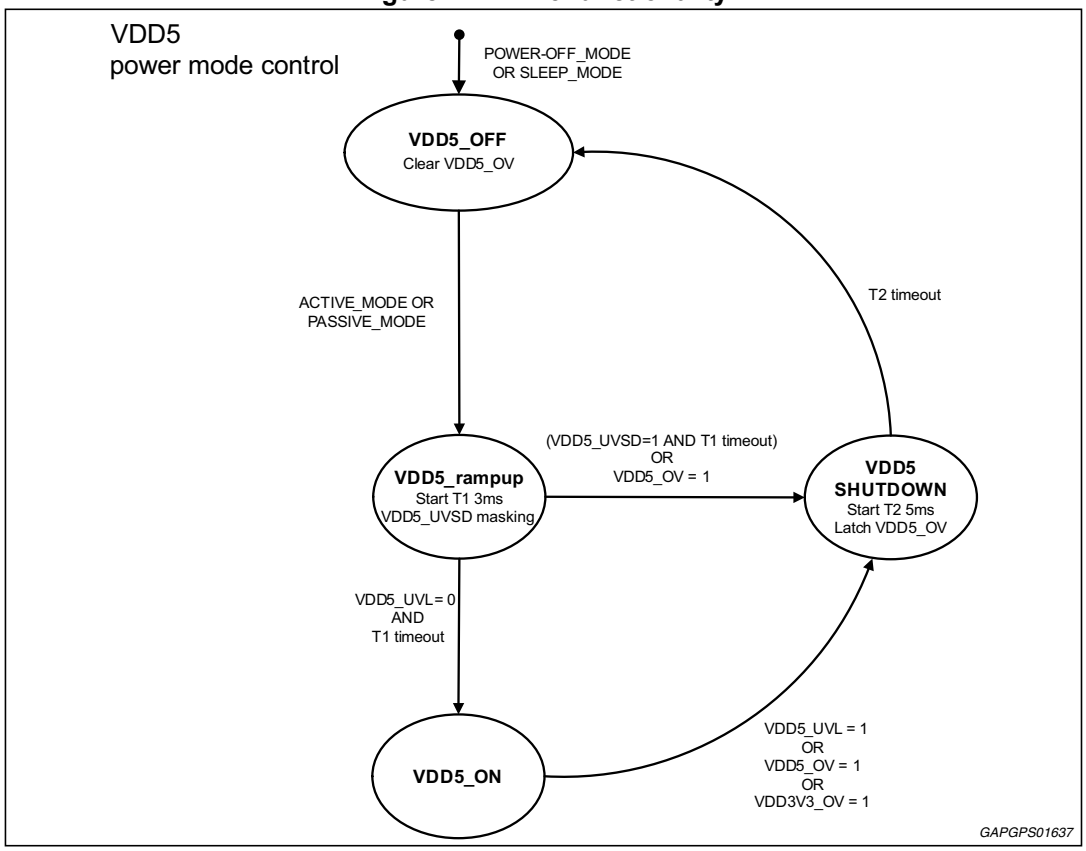
Normal function

VDD5 is used to supply eventual 5 V compatible devices on board.

VDDQ pin defines the voltage level on I/O pin:

- if VDDQ is connected to VDD5 it means that digital I/O pins 5V are compliant.
- if VDDQ is connected to VDD3V3 it means that digital I/O pins 3.3V are compliant.

Figure 22. VDD5 functionality



VDD5UVSD = VDD5UVL = 1.8 V ÷ 2.2 V

Diagnostic:

OVERVOLTAGE	VDD5_OV = 5.2V ÷ 5.5V, latched
UNDERVOLTAGE	VDD5_UV = 4.5V ÷ 4.8V
UNDERVOLTAGE LOW LEVEL	VDD5_UVL = 1.8V ÷ 2.2V
VDD5 state active VDD5_ACT	

Diagnostic read out through SPI:

$05 POWER STATE	
VDD5_UV, bit 11	0 = VDD5>VDD5_UV 1 = VDD5>VDD5_UV
VDD5_OV, bit 10	0 = VDD5<VDD5_OV 1 = VDD5>VDD5_OV
VDD5_ACT, bit 3	0 = VDD5 in OFF or SHUTDOWN state 1 = VDD5 in RAMPUP or ON

VDD5_ACT bit is used by the microcontroller to monitor the VDD5 status, ON or OFF.

Protection

Current limitation is guaranteed by means of controlling output current on BVDD5 pin (4 mA to 10 mA).

Deactivation

Deactivation is determined by all the conditions that bring back the IC in SLEEP mode.

In case of VDD5_UV, or VDD5_OV or VDD3V3_OV the regulator is switched off.

5.5 VDD3V3 linear regulator

Features

3.3 V full integrated linear regulator slope controlled derived from VDD5.

VDD3V3: 3.2 V ÷ 3.4 V

VDD3V3 load current 0.5 mA ÷ 125 mA

VDD3V3 load current limitation: 150 mA min

VDD3V3 stability requires an external small capacitor 3 µF ÷ 31 µF.

Activation

Requirement: VDD5min = 4.85 V ≤ VDD5.

It starts in ACTIVE mode and continues to work in PASSIVE mode too, being supplied by VDD5 and VDD5 is supplied, via ER switch, by the external reserve capacitor.

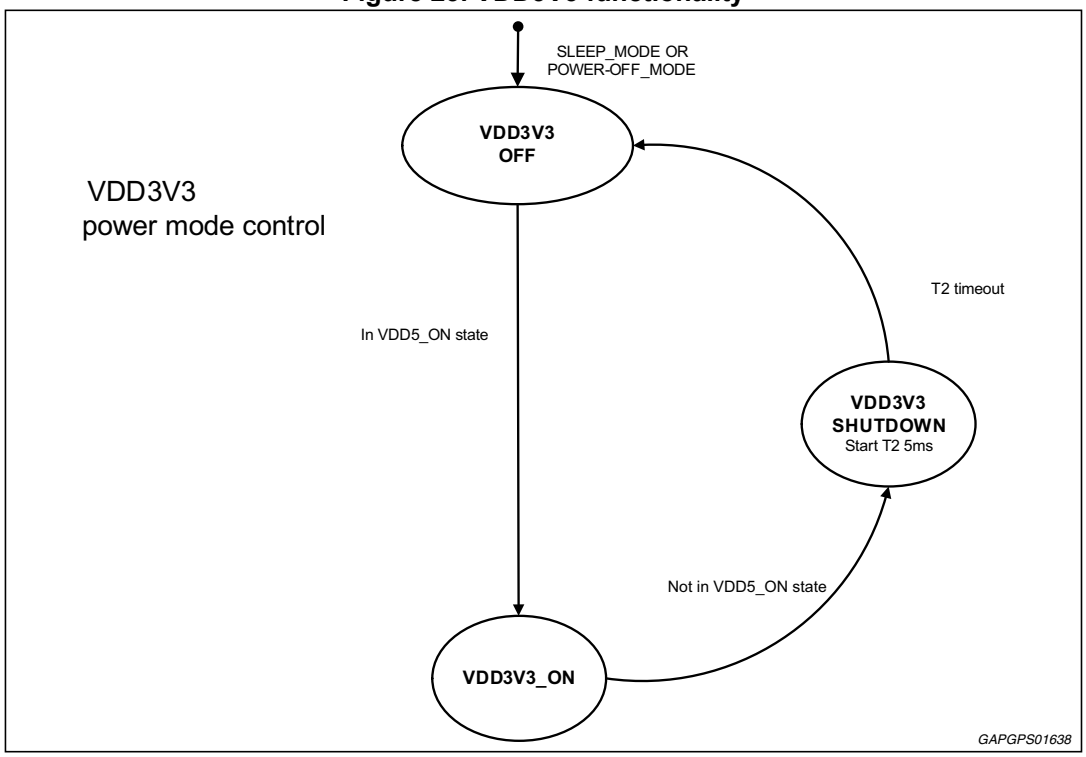
Normal function

VDD3V3 is used to supply 3.3 V devices, as for example, the microcontroller if it has been chosen one whose I/O pins are 3.3 V compatible.

VDDQ pin defines the voltage level on I/O pin:

- if VDDQ is connected to VDD5 it means that digital I/O pins 5 V are compliant.
- if VDDQ is connected to VDD3V3 it means that digital I/O pins 3.3 V are compliant.

Figure 23. VDD3V3 functionality



Diagnostic

OVERVOLTAGE VDD3V3OV = 3.43V ÷ 3.6V

UNDERVOLTAGE VDD3V3UV = 3.0V ÷ 3.17V

VDD3V3 active state

Diagnostic read out through SPI:

$05 POWER STATE	
VDD3V3_UV, bit 14	0 = VDD3V3>VDD3V3_UV 1 = VDD3V3>VDD3V3_UV
VDD3V3_OV, bit 13	0 = VDD3V3<VDD3V3_OV 1 = VDD3V3>VDD3V3_OV
VDD3V3_ACT, bit 1	0 = VDD3V3 in OFF or SHUTDOWN state 1 = VDD3V3 is ON

VDD3V3_ACT bit is used by the microcontroller to monitor the VDD3V3 status, ON or OFF.

Protection

Current limitation of VDD3V3 is IO_LIM = 150 mA max.

Note: If the VDDQ pin (digital outputs supply) is connected to the VDD3V3 and any of the digital I/O pins are connected to 5V logic, there is no internal blocking diode to prevent 3.3V supply back feeding.

Deactivation

Deactivation is determined by all the conditions that bring back the IC in SLEEP mode.

It is switched off in case VDD5 is NOT IN on STATE.

5.6 VSUP Linear regulator (available for L9678S version)

Features

- 6.8 V linear regulator slope controlled, derived directly from battery line through an external PNP (h_{FE} min 50)
- VSUP output voltage: 6.5 V ÷ 7.1 V
- VSUP load current 0.5 mA ÷ 200 mA
- VSUP stability requires an external small capacitor 3 µF ÷ 30 µF.
- Current limitation is guaranteed controlling the output current on BVSUP pin.

As the regulator starts running, the under-voltage that occurs before the regulator has reached its correct value, it is not considered a fault (under-voltage fault) in the first 3ms after the regulator is switched on.

Activation

Requirement: VINGOOD2max = 6.8 V ≤ VIN ≤ 35 V

IC in ACTIVE or PASSIVE mode; in PASSIVE mode VSUP is supplied through ER switch by the external reserve capacitor.

SPI command:

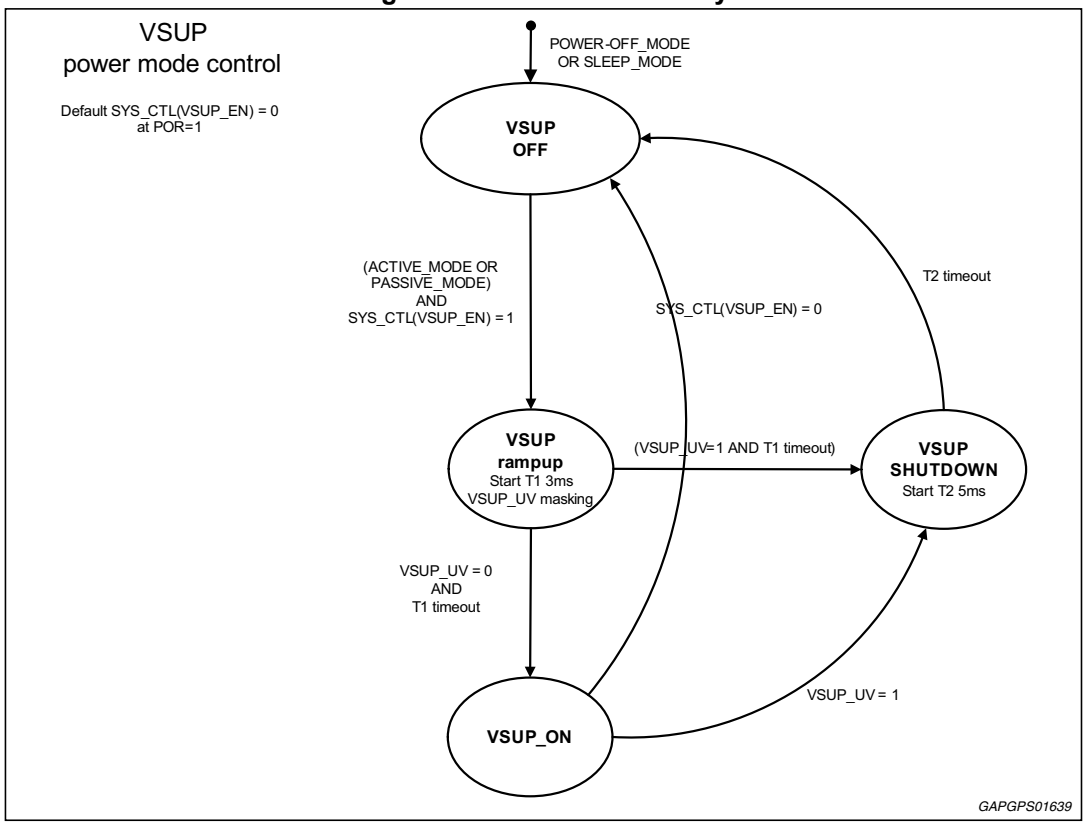
$02 SYS_CTL	Config. in INIT, DIAG, SAFING, SCRAP, ARMING state
VSPU_EN, bit 5	0 = VSUP commanded OFF
	1 = VSUP commanded ON

Normal function

VSUP is used to supply 6.8 V devices as, for example, the PSI-5 sensor.

In case of L9678 version, VSUP pin can be connected to GND.

Figure 24. VSUP functionality



Diagnostic

$05 POWER STATE	
VSUP_NOK, bit 9	0: VSUP is in its correct range VTHLVSUP<VSUP<VTHHVSUP 1: VSUP is out of its correct range (VSUP<VTHLVSUP) OR (VSUP> VTHHVSUP)
VSUP_ACT, bit 2	0 = VSUP in OFF or SHUTDOWN state 1 = VSUP in RAMPUP or ON

VTHHVSUP = 8 V max

VTHLVSUP = 6.5 V min

VSUP_ACT bit is used by the microcontroller to monitor the VSUP status, ON or OFF.

Protection

Current limitation is guaranteed by means of controlling output current on BVSUP pin (4 mA to 10 mA).

Deactivation

Through SPI disable command.

In case of VSUP under-voltage condition that lasts for at least $T_{vsupuvfilt}$ (27 µs ÷ 33 µs) after 5 ms, VSUP is switched off.

5.7 VSF linear regulator

Features

- full integrated linear regulator derived from ERBOOST
- VSF is 20 V as default value and it is configurable at 25 V via SPI

$01 SYS_CFG	Config. in INIT state
VSF_V, bit 2	0 = 20 V selected (18 V ÷ 20 V) 1 = 25 V selected (23 V ÷ 27 V)

VSF stability requires an external small capacitor 2.9 nF ÷ 14 nF.

Turn on time = 100 µs

Activation

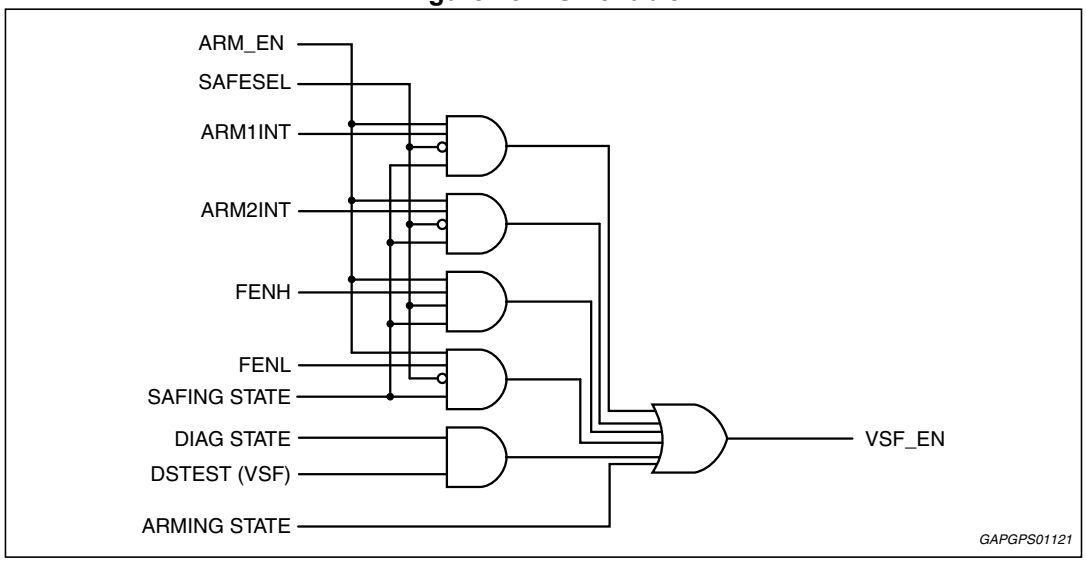
Requirement: VINGOOD1max = 5.5 V ≤ VIN ≤ 35 V

VSF + 2 V ≤ ERBOOST

IC in ACTIVE or PASSIVE mode

Activation of this regulator is so conditioned:

Figure 25. VSF enable



SAFESEL configures (via SPI) the safing logic, internal or external:

$01 SYS_CFG	Config. in INIT state
SAFESEL, bit 3	0 = internal safing engine 1 = external safing engine - default

If the IC is in DIAGNOSTIC state, VSF_EN is linked to the request to activate VSF:

$36 SYSDIAGREQ	Config. in DIAG state
DSTEST, bit [3÷0]	0110 = VSF regulator active

Normal function

VSF is used to supply an external N channel safing FET.

Diagnostic

$05 POWER STATE	
VSF_ACT, bit 0	0: VSF_EN = 0 1: VSF_EN = 1

Protection

Output load current limitation I_{O_LIM} = 7 mA ÷ 13 mA

Output voltage drop-out = (ERBOOST-VSF) = 2 V max

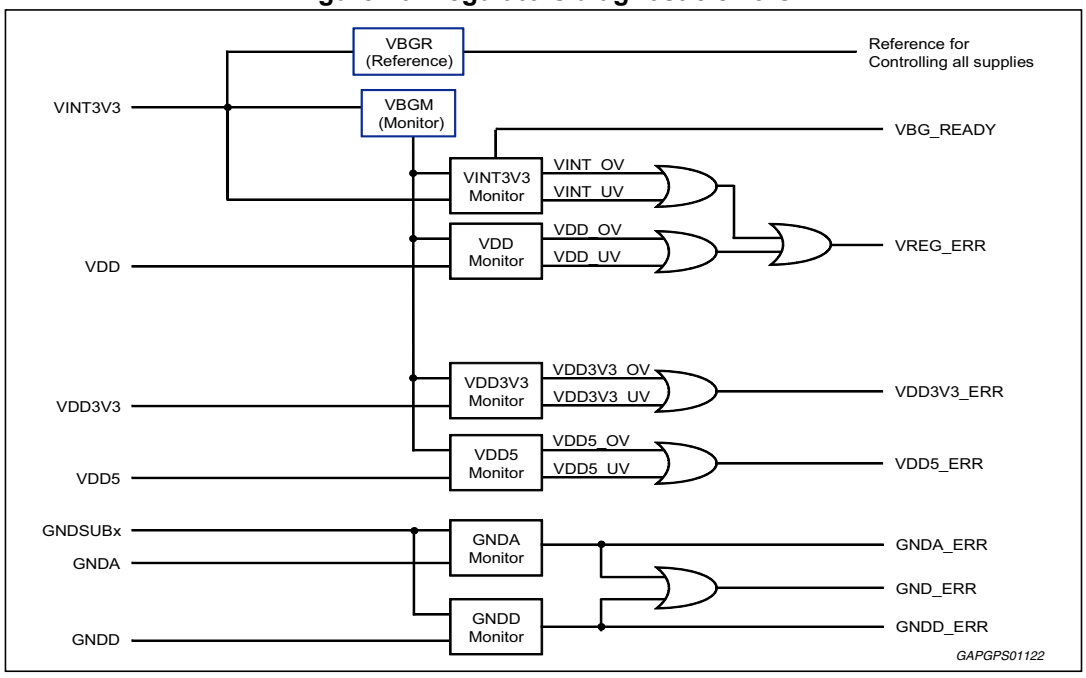
Deactivation

See picture above: any condition that does not satisfy the condition reported.

5.8 RESET

RESET is active low.

Figure 26. Regulators diagnostic errors

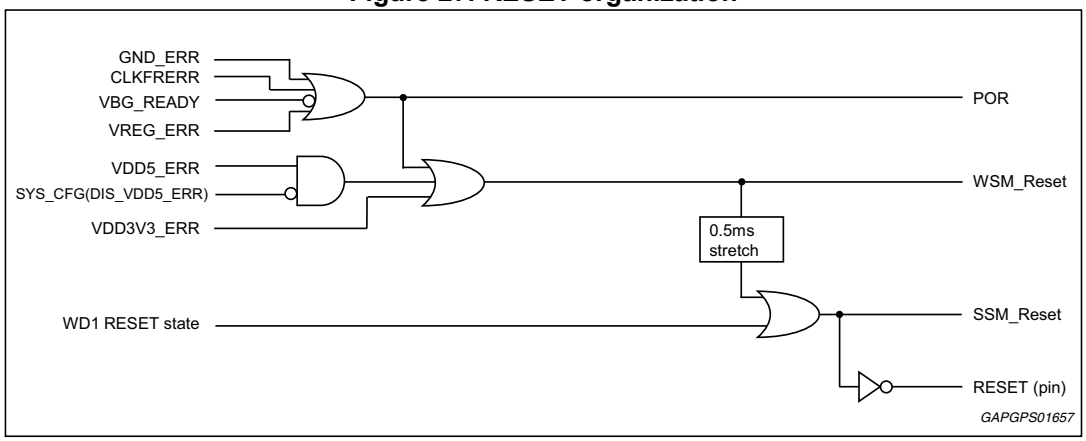


List of the internal reset signals:

VBG_READY	problem on band gap
VREG_ERR	problem on internal voltage regulators
VDD3V3_ERR	problem on VDD3V3 voltage regulator
VDD5_ERR	problem on VDD5 voltage regulator
GNDA_ERR	GNDA shift greater than 300mV respect to GND SUB
GNDD_ERR	GNDD shift greater than 300mV respect to GND SUB
GND_ERR	GNDA_ERR or GNDD_ERR

These signals are grouped into 3 internal bit:

Figure 27. RESET organization



POR (POR bit) takes into account problems on the internal voltage regulators (3V3INT and CVDD), on GND connections, and problems in the clock frequency.

WSM_Reset (WSMRST bit) takes into account the same problem of POR and problems on the external voltage regulators, VDD5 or VDD3V3.

Problems on VDD5 can be ignored depending on DIS_VDD5_ERR bit configuration:

$01 SYS_CFG	Config. in INIT state
DIS_VDD5_ERR, bit 14	0 = VDD5 OV/UV generates reset 1 = VDD5 OV/UV doesn't generate reset

SSM_Reset (SSMRST bit) takes into account the same problem of WSM_Reset and problems on the watchdog service.

$00 FLTSR	
WSMRST, bit 3	0 = WSM RESET has not occurred 1 = WSM RESET has occurred
SSMRST, bit 2	0 = SSM RESET has not occurred 1 = SSM RESET has occurred
POR, bit 0	0 = POR RESET has not occurred 1 = POR RESET has occurred

RESET pin (not SSM_Reset) summarizes all the above mentioned cases.

6 SPI

For configuration, control and read out status of the IC.

Table 3. Global SPI register map

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
0	0	0	0	0	0	0	0	$00	R	FLTSR	Global fault status register					
0	0	0	0	0	0	0	1	$01	R/W	SYS_CFG	Power supply configuration (regulators' output voltage selection, enable internal safing engine)	X				
0	0	0	0	0	0	1	0	$02	R/W	SYS_CTL	Register to control the power management (enable for tests in diag state, enable for power mode control bit)	X	X	X	X	X
0	0	0	0	0	0	1	1	$03	W	SPI_SLEEP	Sleep Mode command	X	X	X	X	X
0	0	0	0	0	1	0	0	$04	R	SYS_STATE	Read register to report in which state the power control state machine is and also in which Operating state we are.					
0	0	0	0	0	1	0	1	$05	R	POWER_STATE	Power state register (feedback on regulators' status and voltage thresholds)					
0	0	0	0	0	1	1	0	$06	R/W	DCR_0	Deployment configuration register		X	X	X	X
0	0	0	0	0	1	1	1	$07	R/W	DCR_1			X	X	X	X
0	0	0	0	1	0	0	0	$08	R/W	DCR_2			X	X	X	X
0	0	0	0	1	0	0	1	$09	R/W	DCR_3			X	X	X	X
0	0	0	0	1	0	1	0	$0A								
0	0	0	0	1	0	1	1	$0B								
0	0	0	0	1	1	0	0	$0C								
0	0	0	0	1	1	0	1	$0D								
0	0	0	0	1	1	1	0	$0E								
0	0	0	0	1	1	1	1	$0F								
0	0	0	1	0	0	0	0	$10								
0	0	0	1	0	0	0	1	$11								

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
0	0	0	1	0	0	1	0	$12	R/W	DEPCOM	Deployment command register			X		X
0	0	0	1	0	0	1	1	$13	R	DSR_0	Deployment status register					
0	0	0	1	0	1	0	0	$14	R	DSR_1						
0	0	0	1	0	1	0	1	$15	R	DSR_2						
0	0	0	1	0	1	1	0	$16	R	DSR_3						
0	0	0	1	0	1	1	1	$17								
0	0	0	1	1	0	0	0	$18								
0	0	0	1	1	0	0	1	$19								
0	0	0	1	1	0	1	0	$1A								
0	0	0	1	1	0	1	1	$1B								
0	0	0	1	1	1	0	0	$1C								
0	0	0	1	1	1	0	1	$1D								
0	0	0	1	1	1	1	0	$1E								
0	0	0	1	1	1	1	1	$1F	R	DCMTS01	Deployment current monitor register					
0	0	1	0	0	0	0	0	$20	R	DCMTS23						
0	0	1	0	0	0	0	1	$21								
0	0	1	0	0	0	1	0	$22								
0	0	1	0	0	0	1	1	$23								
0	0	1	0	0	1	0	0	$24								
0	0	1	0	0	1	0	1	$25	R/W	SPIDEPEN	Lock/Unlock command			X		X
0	0	1	0	0	1	1	0	$26	R	LP_GNDLOSS	Loss of ground fault for squib loops					
0	0	1	0	0	1	1	1	$27	R	VERSION_ID	Device version					
0	0	1	0	1	0	0	0	$28	R/W	WD_RETRY_CONF	Watchdog Retry Configuration	X				
0	0	1	0	1	0	0	1	$29								
0	0	1	0	1	0	1	0	$2A	R/W	WDTCR	Watchdog timer configuration	X				
0	0	1	0	1	0	1	1	$2B	R/W	WD1T	Watchdog key transmission & Test mode	X	X	X	X	X

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
0	0	1	0	1	1	0	0	$2C	R	WD_STATE	Watchdog state					
0	0	1	0	1	1	0	1	$2D	R/W	CLK_CONF	Clock Configuration	X				
0	0	1	0	1	1	1	0	$2E								
0	0	1	0	1	1	1	1	$2F								
0	0	1	1	0	0	0	0	$30	W	SCRAP_STATE	Scrap State command		X			
0	0	1	1	0	0	0	1	$31	W	SAFING_STATE	Safing State command		X			
0	0	1	1	0	0	1	0	$32								
0	0	1	1	0	0	1	1	$33								
0	0	1	1	0	1	0	0	$34								
0	0	1	1	0	1	0	1	$35	W	WD_TEST	Watchdog first and second level test	X	X	X	X	X
0	0	1	1	0	1	1	0	$36	R/W	SYSDIAGREQ	Diagnostic command for system safing		X			
0	0	1	1	0	1	1	1	$37	R	LPDIAGSTAT	Diagnostic results register for deployment loops					
0	0	1	1	1	0	0	0	$38	R/W	LPDIAGREQ	Diagnostic configuration command for deployment loops		X	X	X	X
0	0	1	1	1	0	0	1	$39	R/W	SWCTRL	DC sensor diagnostic configuration		X	X	X	X
0	0	1	1	1	0	1	0	$3A	R/W	DIAGCTRL_A	In WID is AtoD converter control register A. In RID is AtoD result A request.		X	X	X	X
0	0	1	1	1	0	1	1	$3B	R/W	DIAGCTRL_B	In WID is AtoD converter control register B. In RID is AtoD result B request.		X	X	X	X
0	0	1	1	1	1	0	0	$3C	R/W	DIAGCTRL_C	In WID is AtoD converter control register C. In RID is AtoD result C request.		X	X	X	X
0	0	1	1	1	1	0	1	$3D	R/W	DIAGCTRL_D	In WID is AtoD converter control register D. In RID is AtoD result D request.		X	X	X	X
0	0	1	1	1	1	1	0	$3E								
0	0	1	1	1	1	1	1	$3F								
0	1	0	0	0	0	0	0	$40								
0	1	0	0	0	0	0	1	$41								

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State[1]				
												Init	Diag	Ssafing	Scrap	Arming
0	1	0	0	0	0	1	0	$42	R/W	GPOCR	General Purpose Output configuration	X	X			
0	1	0	0	0	0	1	1	$43	R/W	GPOCTRL0	General Purpose Output control register	X	X	X	X	X
0	1	0	0	0	1	0	0	$44	R/W	GPOCTRL1	General Purpose Output control register	X	X	X	X	X
0	1	0	0	0	1	0	1	$45								
0	1	0	0	0	1	1	0	$46	R	GPOFLTSR	General Purpose Output fault status register					
0	1	0	0	0	1	1	1	$47	R	ISOFLTSR	ISO9141 fault status register					
0	1	0	0	1	0	0	0	$48								
0	1	0	0	1	0	0	1	$49								
0	1	0	0	1	0	1	0	$4A	R/W	RSCR1	PSI5 configuration register		X			
0	1	0	0	1	0	1	1	$4B	R/W	RSCR2			X			
0	1	0	0	1	1	0	0	$4C								
0	1	0	0	1	1	0	1	$4D								
0	1	0	0	1	1	1	0	$4E	R/W	RSCTRL	Remote sensor control register		X	X	X	X
0	1	0	0	1	1	1	1	$4F								
0	1	0	1	0	0	0	0	$50	R	RSDR1	Remote sensor data and fault flag registers					
0	1	0	1	0	0	0	1	$51	R	RSDR2						
0	1	0	1	0	0	1	0	$52								
0	1	0	1	0	0	1	1	$53								
0	1	0	1	0	1	0	0	$54								
0	1	0	1	0	1	0	1	$55								
0	1	0	1	0	1	1	0	$56								
0	1	0	1	0	1	1	1	$57								
0	1	0	1	1	0	0	0	$58								
0	1	0	1	1	0	0	1	$59								
0	1	0	1	1	0	1	0	$5A								
0	1	0	1	1	0	1	1	$5B								

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
0	1	0	1	1	1	0	0	$5C								
0	1	0	1	1	1	0	1	$5D								
0	1	0	1	1	1	1	0	$5E								
0	1	0	1	1	1	1	1	$5F								
0	1	1	0	0	0	0	0	$60								
0	1	1	0	0	0	0	1	$61								
0	1	1	0	0	0	1	0	$62								
0	1	1	0	0	0	1	1	$63								
0	1	1	0	0	1	0	0	$64								
0	1	1	0	0	1	0	1	$65								
0	1	1	0	0	1	1	0	$66	R/W	SAF_ALGO_CONF	Safing Algorithm configuration register		X			
0	1	1	0	0	1	1	1	$67								
0	1	1	0	1	0	0	0	$68								
0	1	1	0	1	0	0	1	$69								
0	1	1	0	1	0	1	0	$6A	R	ARM_STATE	Status of internal arming signals FENH, FENL, ARMx					
0	1	1	0	1	0	1	1	$6B								
0	1	1	0	1	1	0	0	$6C								
0	1	1	0	1	1	0	1	$6D								
0	1	1	0	1	1	1	0	$6E	R/W	LOOP_MATRIX_ARM1	Assignment of ARM 1 pin to which LOOPS		X			
0	1	1	0	1	1	1	1	$6F	R/W	LOOP_MATRIX_ARM2	Assignment of ARM 2 pin to which LOOPS		X			
0	1	1	1	0	0	0	0	$70								
0	1	1	1	0	0	0	1	$71								
0	1	1	1	0	0	1	0	$72								
0	1	1	1	0	0	1	1	$73	R	AEPSTS_ARM1	Arming pulse stretch timer value					
0	1	1	1	0	1	0	0	$74	R	AEPSTS_ARM2						

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
0	1	1	1	0	1	0	1	$75								
0	1	1	1	0	1	1	0	$76								
0	1	1	1	0	1	1	1	$77								
0	1	1	1	1	0	0	0	$78								
0	1	1	1	1	0	0	1	$79								
0	1	1	1	1	0	1	0	$7A								
0	1	1	1	1	0	1	1	$7B								
0	1	1	1	1	1	0	0	$7C								
0	1	1	1	1	1	0	1	$7D								
0	1	1	1	1	1	1	0	$7E								
0	1	1	1	1	1	1	1	$7F	R/W	SAF_ENABLE	Safing record enable		X	X	X	X
1	0	0	0	0	0	0	0	$80	R/W	SAF_REQ_MASK_1	Safing record request mask		X			
1	0	0	0	0	0	0	1	$81	R/W	SAF_REQ_MASK_2			X			
1	0	0	0	0	0	1	0	$82	R/W	SAF_REQ_MASK_3			X			
1	0	0	0	0	0	1	1	$83	R/W	SAF_REQ_MASK_4			X			
1	0	0	0	0	1	0	0	$84								
1	0	0	0	0	1	0	1	$85								
1	0	0	0	0	1	1	0	$86								
1	0	0	0	0	1	1	1	$87								
1	0	0	0	1	0	0	0	$88								
1	0	0	0	1	0	0	1	$89								

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
1	0	0	0	1	0	1	0	$8A			Safing record request mask					
1	0	0	0	1	0	1	1	$8B								
1	0	0	0	1	1	0	0	$8C								
1	0	0	0	1	1	0	1	$8D								
1	0	0	0	1	1	1	0	$8E								
1	0	0	0	1	1	1	1	$8F								
1	0	0	1	0	0	0	0	$90								
1	0	0	1	0	0	0	1	$91								
1	0	0	1	0	0	1	0	$92								
1	0	0	1	0	0	1	1	$93	R/W	SAF_REQ_TARGET_1	Safing record request target		X			
1	0	0	1	0	1	0	0	$94	R/W	SAF_REQ_TARGET_2			X			
1	0	0	1	0	1	0	1	$95	R/W	SAF_REQ_TARGET_3			X			
1	0	0	1	0	1	1	0	$96	R/W	SAF_REQ_TARGET_4			X			
1	0	0	1	0	1	1	1	$97								
1	0	0	1	1	0	0	0	$98								
1	0	0	1	1	0	0	1	$99								
1	0	0	1	1	0	1	0	$9A								
1	0	0	1	1	0	1	1	$9B								
1	0	0	1	1	1	0	0	$9C								
1	0	0	1	1	1	0	1	$9D								
1	0	0	1	1	1	1	0	$9E								
1	0	0	1	1	1	1	1	$9F								
1	0	1	0	0	0	0	0	$A0								
1	0	1	0	0	0	0	1	$A1								
1	0	1	0	0	0	1	0	$A2								

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
1	0	1	0	0	0	1	1	$A3			Safing record request target					
1	0	1	0	0	1	0	0	$A4								
1	0	1	0	0	1	0	1	$A5								
1	0	1	0	0	1	1	0	$A6	R/W	SAF_RESP_MASK_1	Safing record response mask		X			
1	0	1	0	0	1	1	1	$A7	R/W	SAF_RESP_MASK_2			X			
1	0	1	0	1	0	0	0	$A8	R/W	SAF_RESP_MASK_3			X			
1	0	1	0	1	0	0	1	$A9	R/W	SAF_RESP_MASK_4			X			
1	0	1	0	1	0	1	0	$AA								
1	0	1	0	1	0	1	1	$AB								
1	0	1	0	1	1	0	0	$AC								
1	0	1	0	1	1	0	1	$AD								
1	0	1	0	1	1	1	0	$AE								
1	0	1	0	1	1	1	1	$AF								
1	0	1	1	0	0	0	0	$B0								
1	0	1	1	0	0	0	1	$B1								
1	0	1	1	0	0	1	0	$B2								
1	0	1	1	0	0	1	1	$B3								
1	0	1	1	0	1	0	0	$B4								
1	0	1	1	0	1	0	1	$B5								
1	0	1	1	0	1	1	0	$B6								
1	0	1	1	0	1	1	1	$B7								
1	0	1	1	1	0	0	0	$B8								
1	0	1	1	1	0	0	1	$B9	R/W	SAF_RESP_TARGET_1	Safing record response target		X			
1	0	1	1	1	0	1	0	$BA	R/W	SAF_RESP_TARGET_2			X			
1	0	1	1	1	0	1	1	$BB	R/W	SAF_RESP_TARGET_3			X			
1	0	1	1	1	1	0	0	$BC	R/W	SAF_RESP_TARGET_4			X			

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
1	0	1	1	1	1	0	1	$BD			Safing record response target					
1	0	1	1	1	1	1	0	$BE								
1	0	1	1	1	1	1	1	$BF								
1	1	0	0	0	0	0	0	$C0								
1	1	0	0	0	0	0	1	$C1								
1	1	0	0	0	0	1	0	$C2								
1	1	0	0	0	0	1	1	$C3								
1	1	0	0	0	1	0	0	$C4								
1	1	0	0	0	1	0	1	$C5								
1	1	0	0	0	1	1	0	$C6								
1	1	0	0	0	1	1	1	$C7								
1	1	0	0	1	0	0	0	$C8								
1	1	0	0	1	0	0	1	$C9								
1	1	0	0	1	0	1	0	$CA								
1	1	0	0	1	0	1	1	$CB								
1	1	0	0	1	1	0	0	$CC	R/W	SAF_DATA_MASK_1	Safing record data mask		X			
1	1	0	0	1	1	0	1	$CD	R/W	SAF_DATA_MASK_2			X			
1	1	0	0	1	1	1	0	$CE	R/W	SAF_DATA_MASK_3			X			
1	1	0	0	1	1	1	1	$CF	R/W	SAF_DATA_MASK_4			X			
1	1	0	1	0	0	0	0	$D0								
1	1	0	1	0	0	0	1	$D1								

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
1	1	0	1	0	0	1	0	$D2			Safing record data mask					
1	1	0	1	0	0	1	1	$D3								
1	1	0	1	0	1	0	0	$D4								
1	1	0	1	0	1	0	1	$D5								
1	1	0	1	0	1	1	0	$D6								
1	1	0	1	0	1	1	1	$D7								
1	1	0	1	1	0	0	0	$D8								
1	1	0	1	1	0	0	1	$D9								
1	1	0	1	1	0	1	0	$DA								
1	1	0	1	1	0	1	1	$DB								
1	1	0	1	1	1	0	0	$DC								
1	1	0	1	1	1	0	1	$DD								
1	1	0	1	1	1	1	0	$DE								
1	1	0	1	1	1	1	1	$DF	R/W	SAF_THRESHOLD_1	Safing record threshold		X			
1	1	1	0	0	0	0	0	$E0	R/W	SAF_THRESHOLD_2			X			
1	1	1	0	0	0	0	1	$E1	R/W	SAF_THRESHOLD_3			X			
1	1	1	0	0	0	1	0	$E2	R/W	SAF_THRESHOLD_4			X			
1	1	1	0	0	0	1	1	$E3								
1	1	1	0	0	1	0	0	$E4								
1	1	1	0	0	1	0	1	$E5								
1	1	1	0	0	1	1	0	$E6								
1	1	1	0	0	1	1	1	$E7								
1	1	1	0	1	0	0	0	$E8								
1	1	1	0	1	0	0	1	$E9								
1	1	1	0	1	0	1	0	$EA								

Table 3. Global SPI register map (continued)

GID	RID / WID							Hex	R/W	Name	Description	Operating State(1)				
												Init	Diag	Ssafing	Scrap	Arming
1	1	1	0	1	0	1	1	$EB			Safing record threshold					
1	1	1	0	1	1	0	0	$EC								
1	1	1	0	1	1	0	1	$ED								
1	1	1	0	1	1	1	0	$EE								
1	1	1	0	1	1	1	1	$EF	R/W	SAF_CONTROL_1	Safing record control		X			
1	1	1	1	0	0	0	0	$F0	R/W	SAF_CONTROL_2			X			
1	1	1	1	0	0	0	1	$F1	R/W	SAF_CONTROL_3			X			
1	1	1	1	0	0	1	0	$F2	R/W	SAF_CONTROL_4			X			
1	1	1	1	0	0	1	1	$F3								
1	1	1	1	0	1	0	0	$F4								
1	1	1	1	0	1	0	1	$F5								
1	1	1	1	0	1	1	0	$F6								
1	1	1	1	0	1	1	1	$F7								
1	1	1	1	1	0	0	0	$F8								
1	1	1	1	1	0	0	1	$F9								
1	1	1	1	1	0	1	0	$FA								
1	1	1	1	1	0	1	1	$FB								
1	1	1	1	1	1	0	0	$FC								
1	1	1	1	1	1	0	1	$FD								
1	1	1	1	1	1	1	0	$FE								
1	1	1	1	1	1	1	1	$FF	R	SAF_CC	Safing record compare complete					

1. A check mark indicates in which operating state a WRITE-command is valid.

SPI_MOSI: for input register. MSB is the first bit received, LSB the last

SPI_MISO: for output register. MSB is the first bit transmitted, LSB the last

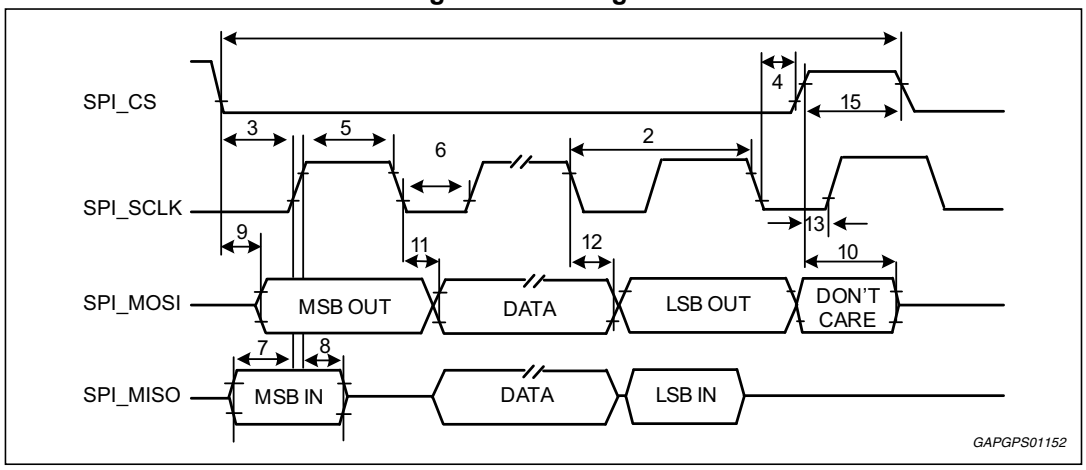
SPI_SCK: clock, slave input data latched on SPI_SCK rising edge and slave output data shifted on falling edge

SPI_CS: chip select CS = H → SPI pins in tri-state

CS = L → SPI communication enabled

There are other two chip selects, SAF_CS0 and SAF_CS1. They are used for a specific communication between sensor interface and IC, see *Section 4.3: SAFING*.

Figure 28. SPI signals



SPI is 32 bit, so organized:

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI_MOSI	GID	RID[6:0]							WID[6:0]							WPAR
SPI_MISO	GSW[10:0]											RPAR	READ[19:16]			

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI_MOSI	WRITE[15:0]															
SPI_MISO	READ[15:0]															

GID	global ID, shared between RID and WID
WID / RID	register definition Write and Read
WPAR / RPAR	odd parity bit Write and Read, calculated on all the 32 bit
GSW	Global Status Word
WRITE	Data input (15:0)
READ	Data output (19:0)

MSB is the first bit transmitted from the master to the IC slave and LSB (MOSI signal) is the last. The same happens on MISO signal.

Global Status Word bit organization:

MISO BIT	31	30	29	28	27	26	25	24	23	22	21
MISO	SPIFLT	DEPOK	RSFLT	WDTDIS_S	ERSTATE	POWERFLT	FLT	CONVRDY2	CONVRDY1	ERR_WID	ERR_RID
GSW BIT	10	9	8	7	6	5	4	3	2	1	0

Due to an SPI_MOSI, the correspondent SPI_MISO is available in the same slot of time of the SPI_MOSI (in frame).

FAULT cases on SPI

GSW	SPIFLT, bit 31 (bit 10 of GSW)
	0 = no fault
	1 = fault

SPI_FLT indicates:

1. number of SPI_SCK different from 32;
2. WPAR bit error in MOSI

Any incorrect access to a register (write/read) is forbidden, but SPIFLT bit is not set.

ERR_WID / ERR_RID

If an SPI write command, with correct WID, is received but the IC is in a status where the writing operation into the addressed register is forbidden (see INIT, DIAG, SAFING, SCRAP, ARMING states), the command itself is discharged and ERR_WID in the next GSW is set.

ERR_WID is not set when a read only register is addressed through WID. The command is ignored.

GSW	ERR_WID, bit 22 (bit 1 of GSW)
	0 = no error
	1 = error

If an SPI command is referred, through a RID address, to an unused register, ERR_RID is set in the current GSW and the command is discharged.

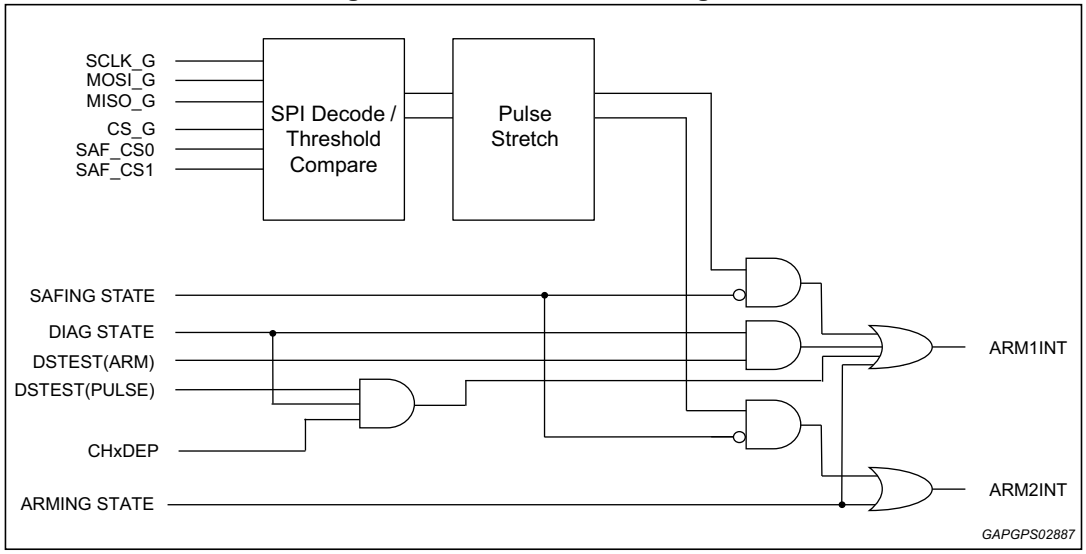
GSW	ERR_RID, bit 21 (bit 1 of GSW)
	0 = no error
	1 = error

7 SAFING

Safing logic is based on

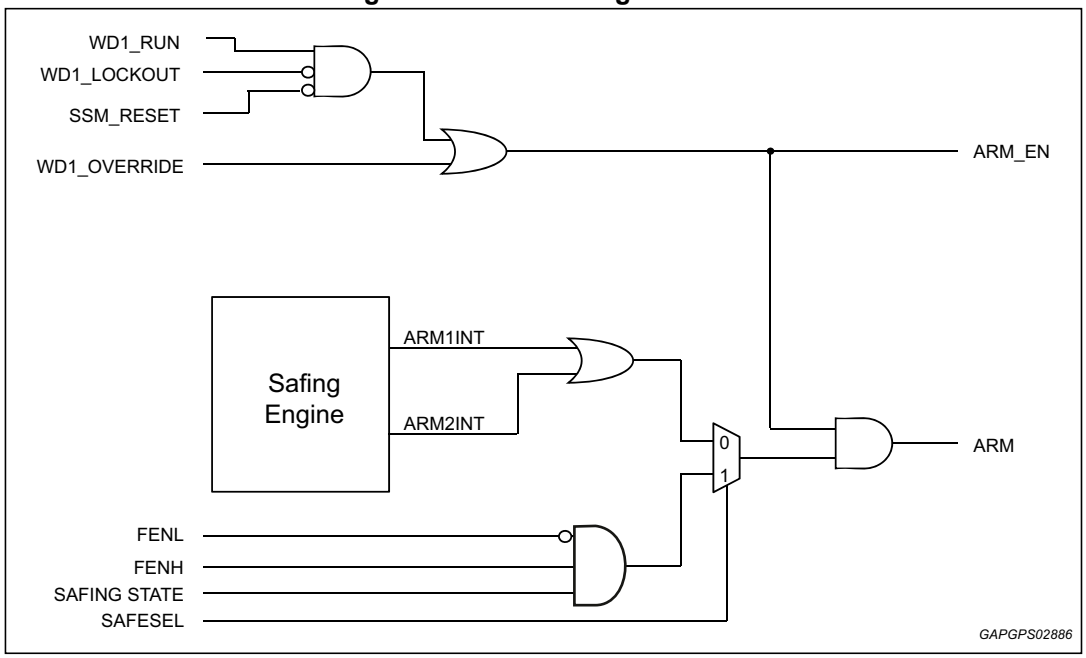
- On board sensors and/or remote sensors (PSI 5)
- ARM signal: two internal ARMiINT (i=1,2) or two external FENH, FENL

Figure 29. Internal ARIMNG signals



ARM pin can be (in according to SAFESEL, bit3 $01 SYS_CFG) the output of the arming signal generated by the integrated safing engine or it can be the combination of FENH/FENL signals that come from external logic:

Figure 30. ARMING organization



7.1 SPI sensor data decoding - Configuration

IC is able to process data coming from both PSI5 remote sensors or on board sensor. Data processed can be used in order to engage the ARMING internal procedure.

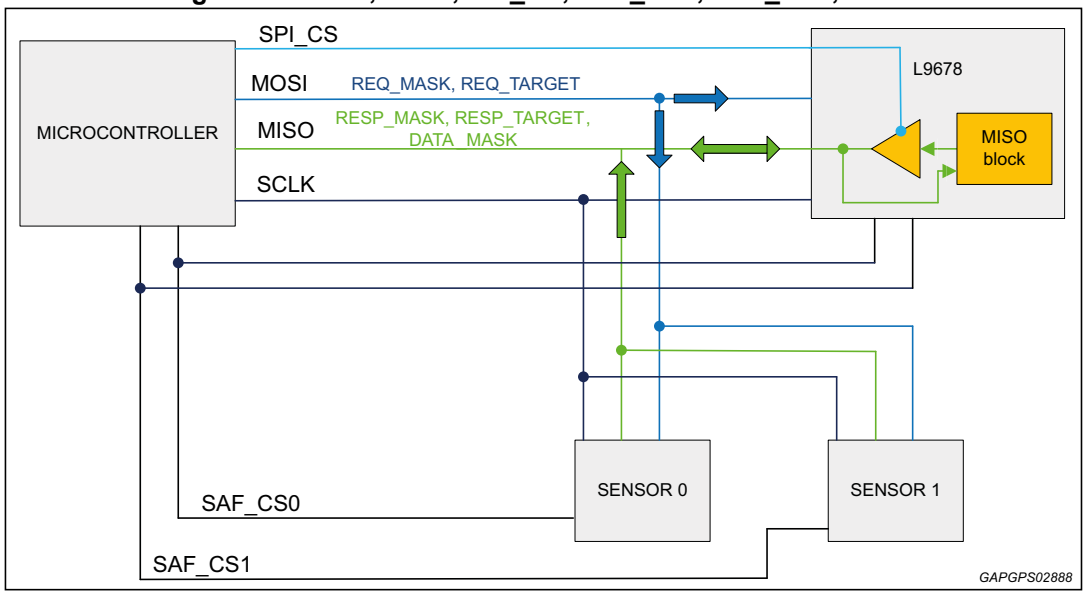
Following pins are involved in this process:

- SPI_CS (active low) is the standard chip select of SPI;
- SAF_CS0, SAF_CS1 (active low) to select up to two on board sensors;
- MOSI is the input data for the IC and MISO is the output data of the IC when CS is low.

MISO is internally read back. Data will be sent out the IC or not basing on SPI_CS.

When SAF_CS0 or SAF_CS1 is asserted, IC can sniff data on MISO pin (MOSI remains an input pin). So the IC knows what microcontroller requires and what sensor answers.

Figure 31. MOSI, MISO, SPI_CS, SAF_CS0, SAF_CS1, SCLK



Note: The external logic guarantees that SAF_CS0, SAF_CS1 and SPI_CS are not asserted simultaneously; if it happens, the frame is discharged.

Safing configuration register is done through SAF_CONTROL_x registers x=1÷4.

$EF → SAF_CONTROL_x → x=1

$F0 → SAF_CONTROL_x → x=2

$F1 → SAF_CONTROL_x → x=3

$F2 → SAF_CONTROL_x → x=4

All safing records are reset by SSM reset.

Case x=1

ARMING considering positive or negative acceleration

It defines if arm is with positive or negative acceleration or both.

$EF SAF_CONTROL_1	Config only in DIAG state
ARMSEL1, bit [15,14]	00, 11 = ARMP or ARMN 01 = ARMP 10 = ARMN

SPI field selection

SPI field selects (for the safing record x) which 16-bit field in the SPI messages will be used in the response on MISO. In case of message having less than 32 bit, this bit doesn't care.

This bit will be relevant in the SAF_RESP_MASK definition, where it is defined which bit, sent from sensor to microcontroller, contain the data, the first 16 or the second 16 bit.

$EF SAF_CONTROL _1	Config. only in DIAG state
SPIFLDSEL1, bit 13	0 = first 16 bit of SPI_MISO frame 1 = second 16 bit of SPI_MISO frame

LIMIT SELECTION

It defines the out-of-range threshold, considering PSI 5 sensor at 8 bit or 10 bit.

This bit is taken into account if LIM_ENx is set.

$EF SAF_CONTROL _1	Config. only in DIAG state
LIM_EN1, bit 12	0 = 8 bit data range |data|>120d, data not recognized valid 1 = 10 bit data range |data|>480d, data not recognized valid

LIMIT ENABLE

To enable or not the out of range control.

$EF SAF_CONTROL _1	Config. only in DIAG state
LIM_EN1, bit 11	0 = data range limit disabled 1 = data range limit enabled

COMB

This command allows, performing mathematic elaboration as sum or difference between accelerations read from different sensors, to elaborate the direction of the acceleration detected from x-y axis to have an on-axis response. The direction depends on the sensor orientation inside the system.

$EF SAF_CONTROL_1	Config. only in DIAG state
COMBx, bit [10]	0 = combine function disabled 1 = combine function enabled

When the “combine function” is enabled, the elaboration, being x=1, 3, is:

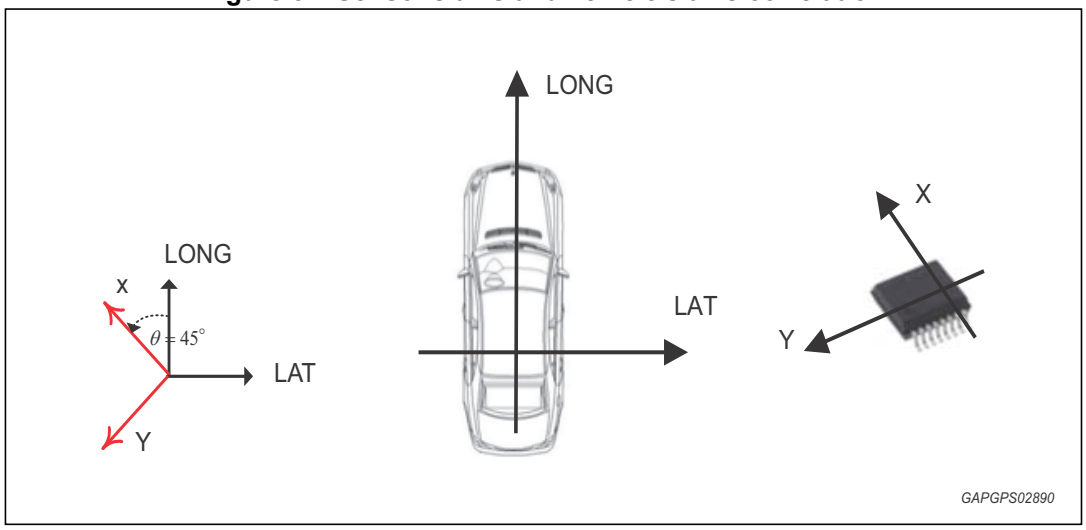
- Safing record pairs x, x+1 (1, 2 / 3, 4)
- Safing record (x) = data(x) + data (x+1)
- Safing record (x+1) = data(x) - data (x+1)

Since this elaboration is performed using two safing records, the elaboration itself is performed only after that the two safing records have been captured (CC_x=1).

In case of an on-board dual axis sensor which is offset 45 degrees from center, the IC allows vector addition/subtraction to refer the sensor data to the vehicle axis.

Here below it is shown how to refer to the data caught by the sensors, referred to their axis orientation, to the axis LONG and LAT of the vehicle, through simple sum or difference of sensor data, with the aim of managing the ARMING process.

Figure 32. Sensor's axis and vehicle's axis correlation



$$LONG = X * \cos(\theta) + Y * \cos(\theta + 90°)$$
$$LAT = X * \cos(\theta + 90) + Y\cos(\theta + 180)$$

if

$$\theta = 45°$$

$$LONG = X * \frac{\sqrt{2}}{2} - Y * \frac{\sqrt{2}}{2} = \frac{\sqrt{2}}{2}(X - Y)$$

$$LAT = -X * \frac{\sqrt{2}}{2} - Y\frac{\sqrt{2}}{2} = -\frac{\sqrt{2}}{2}(X + Y)$$

GAPGPS02889

Thresholds are referred to LONG and LAT reference system;

Data caught from sensor are referred to X and Y axis. These axis are rotated with respect to LONG and LAT axis, see *Figure 32*.

The thresholds (TH_LONG and TH_LAT thresholds) must consider the sensor direction with respect to the vehicle axis, so the trigonometric parameters (θ) are known and included in the thresholds:

$$|X - Y| > TH * \frac{2}{\sqrt{2}} = TH\_LONG$$

$$|X + Y| > TH * \frac{2}{\sqrt{2}} = TH\_LAT$$

GAPGPS02891

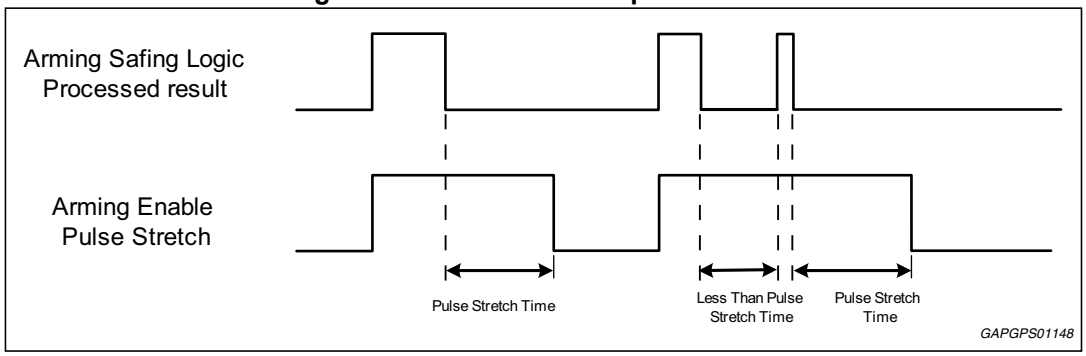
DWELL

Once an arming condition is detected, it remains valid for DWELL time (pulse stretch time) waiting for an eventual firing command.

DWELL time is blocked by SSM reset.

In case different values are configured for the four safing records, the value considered is the longest one.

Figure 33. ARMING enable pulse stretch



$EF SAF_CONTROL _1	Config. only in DIAG state
DWELLx (x=1), bit [9,8]	00 = 2048ms 01 = 256ms 10 = 32ms 11 = 0ms

Note: In SAFING configuration it is defined the time DWELL, that is the period of time in which the ARMING signal, once asserted, is valid waiting for a deployment command.

In DEPLOYMENT configuration it is defined the time DEPLOY EXPIRE TIME, that is the period of time in which the deployment command, once received, is valid waiting for the ARMING signal asserted.

ARM

There are two internal arming signals (ARM1INT, ARM2INT). The safing record can be assigned to one or both of them (ARM1INT ARM2INT).

This allows grouping the sensors that recognize a high acceleration based on their direction, representing then the crash event.

ARM1INT - ARM2INT are linked to the deployment loop following LOOP_MATRIX_ARMx set-up, $6E and $6F registers:

$6E LOOP_MATRIX_AMR1 **$6F LOOP_MATRIX_AMR2**	Config. only in DIAG state
ARMx_Li, i=3..0, b[3:0]	0 = ARMx not associated with loop i 1 = ARMx associated with loop i

In this way it is possible to establish which deployment loop will take place based on the direction of the crash event.

A safing record can be mapped at the same time both on ARM1INT and ARM2INT

x=1

$EF SAF_CONTROL _1	Config. only in DIAG state
ARM2x, bit 5	0 = safing record x not assigned to ARM2INT 1 = safing record x assigned to ARM2INT
ARM1x, bit 4	0 = safing record x not assigned to ARM1INT 1 = safing record x assigned to ARM1INT

CS

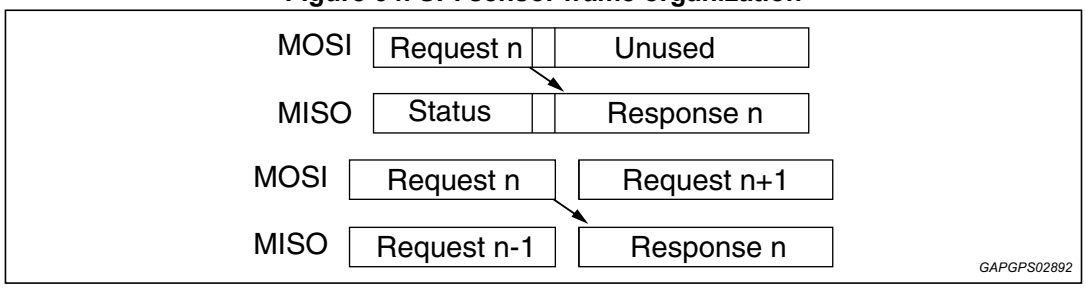
To associate a safing record with one of the 3 chip select, SPI_CS, SAF_CS0, SAF_CS1

$EF SAF_CONTROL _1	Config. only in DIAG state
CSx, bit [3÷1]	001 = SAF_CS0 associated to safing record x 010 = SAF_CS1 associated to safing record x 101 = SPI_CS associated to safing record x Others = no CS associated to safing record x

IN FRAME

As said, the answer (on MISO) can be in frame with the request (MOSI) or out of frame. The IC has to be aligned to the option chosen

Figure 34. SPI sensor frame organization



$EF SAF_CONTROL _1	Config. only in DIAG state
IF1, bit 0	0 = response out of frame (record x) 1 = response in frame (record x)

SPI is 32bit:

	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI_MOSI REQUEST	GID	RID[6:0]							WID[6:0]							WPAR
SPI_MISO REQUEST	GSW[10:0]											RPAR	READ[19:16]			

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI_MOSI REQUEST	WRITE[15:0]															
SPI_MISO REQUEST	READ[15:0]															

Compatible on board sensors are able to communicate over 16, 32 or even more bit frames.

Useful information for the microcontroller requests is supposed to be always located on the first 16MSBs (GID, RID, WID, WPAR), while the useful information for sensor response can be located both on first 16 or second 16bit block of response from sensor.

If the frame is shorter than 16 bit, the frame is discharged;

In case of frame length between 16 and 32 bit, 16 MSB are latched while the latter bits are ignored.

In case of 32 bit frame or longer, 32 MSB are latched, while the latter bits are ignored by the IC; in this case first or second 16 bit block is considered, depending on the configuration of the bit SPIFLDSEL1.

In order to extract useful information from the data sniffed over the SPI bus, IC provides definition for configurations of proper request masks and response masks, which can be programmed by the microcontroller, see *Section 7.2*.

Sensor data are processed as described below. The IC provides four 16 bit registers for safing records.

ENABLE

Each safing record has to be activated, through 7F SAF_ENABLE

These bit determine when a safing record is active or inactive.

$7F SAF_ENABLE	Config. in DIAG, SAFING, SCRAP, ARMING state
EN_SAF4, bit 3 EN_SAF3, bit 2 EN_SAF2, bit 1 EN_SAF1, bit 0	0 = disable 1 = enable

THRESHOLD

Each safing record has to be compared with a threshold, configured only in DIAGNOSTIC state, one threshold for each safing record:

$DF → SAF_THRESHOLD_x → x=1

$E0 → SAF_THRESHOLD_x → x=2

$E1 → SAF_THRESHOLD_x → x=3

$E2 → SAF_THRESHOLD_x → x=4

x=1,

$DF SAF_THRESHOLD_1	Config. only in DIAG state
	SAF_THRESHOLD_1, bit [15÷0]

The threshold is always programmed in absolute value, even if the safing record is a simple sensor data or data combination, sum or difference.

The safing record is compared with the SAF_TH.

Inside the IC there are two dedicated counters (POS_COUNT, NEG_COUNT) to count, with the weight ADD_VAL and SUB_VAL, how many times the safing record overcomes the threshold SAF_TH.

These counters (POS_COUNT, NEG_COUNT) have their limits, ARMN_TH, ARMP_TH. These parameters are defined in the SAF_ALGO_CONF register.

Counters' values are updated on each sensor sample received.

$66 SAF_ALGO_CONF	Config. only in DIAG state
ADD_VAL, bit [2÷0]	incremental step size
SUB_VAL, bit [5÷3]	decremental step size
ARMN_TH, bit [13÷10]	Threshold negative counter
ARMP_TH, bit [9÷6]	Threshold positive counter

If safing record > +|SAF_TH| → it is incremented the positive counter, POS_COUNT of ADD_VAL; otherwise the counter is decremented of SUB_VAL

If safing record < -|SAF_TH| → it is incremented a negative counter, NEG_COUNT of ADD_VAL; otherwise the counter is decremented of SUB_VAL.

Values of NEG_COUNT and POS_COUNT counters are compared with two thresholds, ARMN_TH and ARMP_TH respectively to define ARMN and ARMP signals

If POS_COUNT ≥ ARMP_TH → ARMP set

If NEG_COUNT ≥ ARMN_TH → ARMN set

Based on ARMSEL bit (register $EF-$F2) and on ARMP/ARMN result, the internal arming flags will be asserted or not.

Once the sample cycle time is elapsed, the sensor data received (CC_x=1) flags that the arming processing has been successfully run.

In case of no data received (CC_x=0) it is possible to configure how the IC behaves. A possibility is to consider the "no data" received event in a similar way than a data below threshold (arming counters decremented) or by taking a more severe alternative (arming counters reset).

$66 SAF_ALGO_CONF	Config. only in DIAG state
NO_DATA, bit 15	0 = event count reset if CC=0 when SPI has read the SAF_CCx bit (no acceleration data received) 1 = event count decremented by SUB_VAL when SPI has read the SAF_CCx bit (no acceleration data received)

COMPARE COMPLETE

Data sample cycle is managed by the microcontroller reading CC_x bit;

typical timing for sampling cycle is 500 µs (sensor sampling period 475 µs ÷ 525 µs).

Once the IC receives data from sensors, post processing for arming purpose starts and any other valid data received from sensors are ignored until the microcontroller reads the CC_x bit.

Should the microcontroller read the CC_x bit and no valid data have been received yet, IC reacts according to the $66 SAF_ALGO_CONF register, NO DATA bit

$FF SAF_CC	
CC_4, bit 3 CC_3, bit 2 CC_2, bit 1 CC_1, bit 0	0 = compare not completed 1 = compare completed

7.2 SPI sensor data decoding - MASK

Sensors regularly exchange data with the main microcontroller through multiple SPI messages. Since not all communications between sensors and microcontroller contain data, it is important for the decoder to properly sort the communications and extract only the targeted data.

The solution is specific masking definition, per safing record:

REQUEST MASK, TARGET REQUEST (MOSI line)

RESPONSE MASK, TARGET RESPONSE (MISO line)

DATA MASK

SAFING THRESHOLD

REQUEST MASK (on MOSI line)

The registers to be configured are the four SAF_REQ_MASK _x. The explanation is the same for all the four registers, so only the first is considered:

$80 SAF_REQ_MASK_x → x=1

$81 SAF_REQ_MASK_x → x=2

$82 SAF_REQ_MASK_x → x=3

$83 SAF_REQ_MASK_x → x=4

Case x=1

$80 SAF_REQ_MASK_1	Config. only in DIAG state
	SAF_REQ_MASK_1, bit [15÷0]

Here is defined the position, inside the frame, of the bit to be considered in the data received. This is done by setting bit in SAF_REQ_MASK_x located in the position to be considered.

Extraction of data is done by putting in AND (bit per bit) the SAF_REQ_MASK with the data received.

TARGET REQUEST (on MOSI line)

The registers to be configured are the four SAF_REQ_TARGET_x. The explanation is the same for all the four registers, so only the first is considered:

$93 SAF_REQ_TARGET_x → x=1

$94 SAF_REQ_TARGET_x → x=2

$95 SAF_REQ_TARGET_x → x=3

$96 SAF_REQ_TARGET_x → x=4

Case x=1

$93 SAF_REQ_TARGET_1	Config. only in DIAG state
	SAF_REQ_TARGET_1, bit [15÷0]

Considering the bit selected through REQUEST MASK, the value they have to assume is indicated in the REQUEST TARGET.

So in REQUEST MASK it is defined which bit positions to look at; in REQUEST TARGET it is defined the expected value the bit in the positions defined through REQUEST MASK have to assume.

This procedure is the definition of the filter of all the data exchanged: which bit consider and which value they have to assume.

RESPONSE MASK (on MISO line)

The registers to be configured are the four SAF_RESP_MASK_x. The explanation is the same for all the four registers, so only the first is considered:

$A6 SAF_RESP_MASK_x → x=1

$A7 SAF_RESP_MASK_x → x=2

$A8 SAF_RESP_MASK_x → x=3

$A9 SAF_RESP_MASK_x → x=4

Case x=1

$A6 SAF_RESP_MASK_1	Config. only in DIAG state
	SAF_RESP_MASK_1, bit [15÷0]

Here is defined the position, inside the frame received from the sensor (MISO), of those bit to be considered. This is done setting the bit, in SAF_RESP_MASK_x, located in the position of the bit to be considered.

RESPONSE TARGET (on MISO line)

The registers to be configured are the four SAF_RESP_TARGET_x. The explanation is the same for all the four registers, so only the first is considered:

$B9 SAF_RESP_TARGET_x → x=1

$BA SAF_RESP_TARGET_x → x=2

$BB SAF_RESP_TARGET_x → x=3

$BC SAF_RESP_TARGET_x → x=4

Case x=1

$B9 SAF_RESP_TARGET_1	Config. only in DIAG state
	SAF_RESP_TARGET_1, bit [15÷0]

Considering all the data that can pass on MISO line, the only one that has to be considered is the data that in the position defined in RESP MASK has the value defined in the RESP TARGET.

So in RESP MASK it is defined which bit look at; in REQUEST TARGET it is defined the expected value these bit, coming from the sensors (MISO line), should have.

This procedure is the definition of the filter of all the data exchanged: which bit consider and which value they should have, sensor side.

DATA MASK (on MISO line) definition, only in DIAGNOSTIC state

The registers to be configured are the four SAF_DATA_MASK_x. The explanation is the same for all the four registers, so only the first is considered:

$CC SAF_DATA_MASK_x → x=1

$CD SAF_DATA_MASK_x → x=2

$CE SAF_DATA_MASK_x → x=3

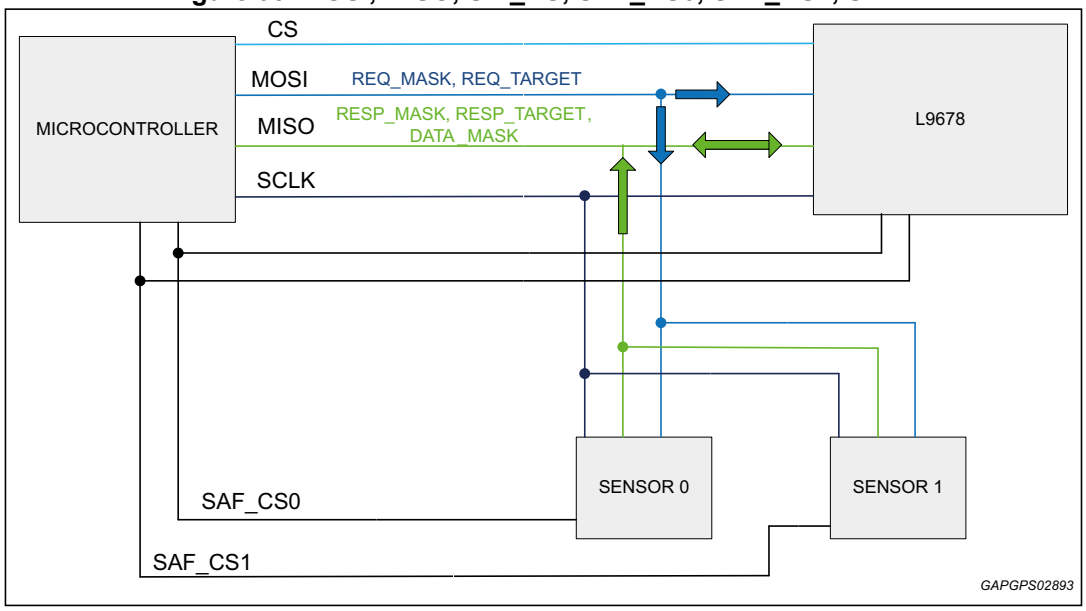
$CF SAF_DATA_MASK_x → x=4

Case x=1

$CC SAF_DATA_MASK_1	Config. only in DIAG state
	SAF_DATA_MASK _1, bit [15÷0]

Here is defined the position, inside the frame received from the sensor (MISO), of those bit to be considered. This is done by setting the bit, in SAF_DATA_MASK_x, located in the position of the bit to be considered.

Figure 35. MOSI, MISO, SPI_CS, SAF_CS0, SAF_CS1, SCLK



Example to explain safing MASK definition

1. Data sensors are so organized:

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SO	0	CH1	CH0	P	ST1	ST0	Sensor data									

bit [14,13]= channel selection, x or y

bit [9÷0]=data sensor

bit 12=patity bit

bit [11,10]=type of data in sensor data field (acceleration, self-test, error)

2. If the microcontroller request is (MOSI):

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SI	0	0	1	X	1	0	X						1	0	1	0

bit [14,13]= 01 this codification can be for example the sensor for x acceleration

bit 12=don't care

bit [11,10]=01 this codification can indicate for example that in the data field are reported acceleration data

bit [9÷4]= don't care. These are the data the sensor sends. Microcontroller reads them, but doesn't write them

bit [3÷0]=address of the sensor, for example 1010 that is sensor n.3

3. Based on this information, the MASK will be:

SAF_REQ_MASK - microcontroller requires something

SAF_REQ_MASKx[15:0]																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1	1	1	0	1	1	0						1	1	1	1

bit [15:13], b[11:10], b[3:0] set

bit 12, bit [9:4] correspond to the position of bit not to be considered; they are left at 0.

4. SAF_REQ_target - microcontroller says what it expects

SAF_REQ_TARGETx[15:0]																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	1	X	1	0	X						1	0	1	0

bit 15 is expected @0

bit [14:13] is expected to be 01 being x axis data

bit 12 is the parity bit, so a priori is not defined

bit [11:10] is expected to be 10 because it is supposed that the data received from sensor will be the acceleration

bit [9:4] are the data, so their value is not known a priori.

bit [3:0] is expected to be 1010 being selected the sensor number 3

SI[15:0] AND SAF_REQ_MASKx[15:0] = SAF_REQ_TARGETx[15:0]

5. The SAF_RESP_MASK allows choosing one message among those in the SPI bus.

In this case, the IC sniffs the frame on MISO line. It considers the position of bit at 1 in the frame.

SAF_RESP_MASKx[15:0]																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1	1	1	0	1	1	0									

bit 15 is expected at 0, so bit 15 is in a relevant position and bit 15 in the SAF_RESP_MASK is set.

bit [14:13] expected 01 so bit 14 and bit 13 are set

bit 12 don't care so bit 12 is left at 0

bit [11:10] expected 10 so bit 11, bit 10 are set

bit [9:0] are the data, not useful to identify which message has to be chosen, so they are left 0

6. The SAF_RESP_TARGET indicates which values are expected for the bit selected in the SAF_RESP_MASK:

SAF_RESP_TARGETx[15:0]																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	1	X	1	0	X									

bit 15=0

bit [14:13]=01

bit 12 don't care → x

bit [11:10]=10

bit [9:0] this field corresponds to the data that are not known a priori → x

The SAF_DATA_MASK indicates where the data bits are localized, in the frame received from the sensor and identified through SF_RESP_MASK, SAF_RESP_TARGET

In this example data are supposed to be localized in position 9:0, so the correspondent bit [9:0] =1

SO[15:0] AND SAF_RESP_MASKx[15:0] = SAF_RESP_TARGETx[15:0]

SAF_DATA_MASKx[15:0]																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	1									

7.3 SPI sensor data decoding - Example arming without on board sensor

Here below there is an example to show how to configure the MASK registers in case of an ECU where on board sensors are not present and the arming is driven by the data received through RSU interface. It is supposed a 10 bit PSI5 data transmission (L9678-S case).

In particular, the example shows how to assign safing record 1 to the data from RSU0; safing record 1 is assigned to both ARM1int and ARM2int signals.

1. Enable safing record #1

$7F SAF_ENABLE	Config. in DIAG, SAFING, SCRAP, ARMING state
EN_SAF1 (bit 0) =1	0 = safing record 1 disabled 1 = safing record 1 enabled ← chose

2. Configure SAF_REQ_MASK for safing record 1

$80 SAF_REQ_MASK_1	Config. only in DIAG state
SAF_REQ_MASK1 bit[15:0]	0111 1111 0000 0000 = $7F00

SAF_REQ_MASK is applied on messages recorded from MOSI pin; peripheral device identification is through GID, RID, WID that are located always in the first 16 bit:

- bit 31 = GID, not important in this exercise
- bit [30:24] = RID [6:0] is the address of the data to be read.
- bit [23:17] = WID [6:0] is the address of the data to be written, not relevant
- bit 16 = WPAR not relevant

SAF_REQ_MASK defines, in the first 16 bit, which of them have to be considered, here all the RIDs, moved to the range [15:0] instead of [31:0]. They are set.

3. Configure SAF_REQ_TARGET for safing record 1

$93 SAF_REQ_TARGET_1	Config. only in DIAG state
SAF_REQ_TARGET1 bit[15:0]	0101 0000 0000 0000 = $5000

Here is defined the expected value of the bit extracted through the request MASK.

When bit[14:8] = RID[6:0] = 101, $50, of MOSI frame sent to the IC, a reading access to the sensor register RSDR0.

All remaining bit, bit[7:0] in the register $93, are don't care in this exercise.

4. Configure SAF_RESP_MASK for safing record 1

$A6 SAF_RESP_MASK_1	Config. only in DIAG state
SAF_RESP_MASK1 bit[15:0]	1111 1100 0000 0000 = $FC00

SAF_RESP_MASK is applied on messages sniffed from MISO pin;

Useful information from sensor is in the second 16 bit block, READ[15:0]

Here is defined which bit of the sensor RSU0 ($50 RSDR0 register) have to be considered. Useful information is FLT bit, ON/OFF bit and LCID[3:0] bit. All remaining bits are don't care in this exercise.

5. Configure SAF_RESP_ TARGET for safing record 1

$B9 SAF_RESP_TARGET_1	Config. only in DIAG state
SAF_RESP_TARGET1 bit[15:0]	0100 0000 0000 0000 = $4000

Here is defined the expected value of the bit extracted through the response MASK in the data received: IC is addressed to read RSU0 ($50 RSDR0 register).

In this step the following bit have to be considered (see RESP_MASK):

- bit 15 = FLT, expected to be 0
- bit 14 = on/off expected to be 1, that means the channel is ON
- bit [13:10] = LCID[3:0] expected to be 0000, RSU0

All remaining bits are don't care in this exercise.

6. Configure SAF_DATA_MASK for safing record 1

$CC SAF_DATA_MASK_1	Config. only in DIAG state
SAF_DATA_MASK1 bit[15:0]	0000 0011 1111 1111 = $03FF

Here is defined which bit of the sensor RSU0 ($50 RSDR0 register) have to be read as data from sensor.

Note: In case of 8 bit data transmission the data from sensor would be available on DATA[7:0] so mask would be $00FF

All remaining bits are don't care in this exercise.

7. Configure SAF_CONTROL for safing record 1

$EF SAF_CONTROL_1	Config. only in DIAG state
SPIFLDSEL1	bit 13 = 1sensor data are transmitted on second 16 bit block of MISO word
ARM21	bit 5 = 1 safing record 1 is assigned to both ARM1int and ARM2int signals.
ARM11	bit 4 = 1 safing record 1 is assigned to both ARM1int and ARM2int signals
CS1[3:1]	bit[3:1] = 101 SPI_CS is selected for safing record 1
IF1	bit 0 = 1 SPI MISO answer is in frame

All remaining bits are left at their default value

8. Enter in SAFING STATE

$31 SAFING_STATE	Config. only in DIAG state
	bit[15:0] = 1010110010101100 = $ACAC

This is needed to bring IC in SAFING state from DIAG state: safing records are enabled only in SAFING STATE.

9. Check IC STATE

$04 SYS_STATE	
	OPER_CTL_STATE bit [10:8] expected to be 010 ie. IC in SAFING STATE

This SPI reading command allows to check in which state the IC is (SAFING expected).

10. Read PSI5 Sensor data

$50 RSDR0	
FLT, bit 15	this bit is expected to be 0
on/off, bit 14	this bit is expected to be 1 channel ON
LCID[3:0], bit [13:10]	expected to be 0000, PSI5 channel 0
DATA, bit [9:0]	

11. Check SAF CC bit

$FF SAF_CC	
CC_1 bit 0	0 = compare not completed for record 1 1 = compare completed for record 1

Safing record can only be evaluated on the first matching input packet. Any further data packet matches are ignored (i.e. once CC is set, record can't be processed until next sensor sampling period)

7.4 Additional communication line (ACL)

The ACL pin is the Additional Communication Line input with the purpose of safely activate the arming output for disposal of restraints devices at the end of vehicle life.

A valid ACL allows the IC to pass from Scrap state to Arming state.

To remain in Arming state the IC must receive the correct ACL signal; this must occur before the scrap timeout timer expires:

$T_{disEOL} = (2*T_{ALC(min)}) \div (2*T_{ACL(max)})$

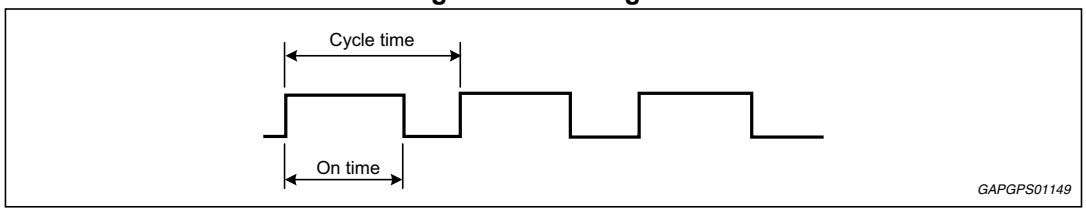
T_{ACL} = 187ms ÷ 213ms

A specific waveform needs to be present on this input in order to instruct the IC to arm all deployment loops and implement the scrapping feature. For those systems not having a proper ACL stimulus available, L9678 allows the ACL input pin to be successfully triggered also with one of the MCU output digital ports.

The disposal signal may come from either the vehicle's service connector, or the systems main microcontroller, depending on the customer's requirements.

The arming function monitors the disposal PWM input (ACL pin) for a command to arm all loops for vehicle end-of-life airbag disposal.

Figure 36. ACL signal



To remain in Arming state, at least three cycles of the ACL signal must be qualified. To qualify the ACL signal, the period and duty cycle are checked.

Two consecutive cycles of invalid disposal signal are enough to disqualify the ACL signal.

8 Deployment

Features:

- 4 independent loops composed by 4 independent high side and 4 independent low side
- all 4 squibs can deploy at the same time or according to a given sequence
- deployment granted in case of short to ground of the low side SRx.
- firing voltage capability across SSxy and SFi is maximum 25 V
- max resistance of high side and low side is 10 Ω.
- each loop can sustain 50 deployment max, waiting at least 10s between each of them
- 2 supply pins, SS01, SS23 directly connected to the High Side for each channel;
- 2 dedicated power ground SG01, SG23 each of them able to sustain the current of two channels simultaneously
- SGxy is connected to GNDSUB through a diode so that the device is able to fire in case of SGxy lost.

8.1 Deployment requirement

Deployment features are deploy current, deploy time and deployment expiration time. The deployment expiration time is the duration time in which the deploy command remains valid, once it is received, waiting for the arming signal.

These parameters are defined in DCR register, one per each channel.

Here are explained the commands to configure the IC deployment.

Deployment configuration is done through the four registers DCR_x, x=0-3, DIAG, SAFING, SCRAP, ARMING state.

$06 DCRx → x=0 channel 0

$07 DCRx → x=1, channel 1

$08 DCRx → x=2, channel 2

$09 DCRx → x=3, channel 3

All deployment configuration registers are reset by SSM reset.

Case x=0

Deploy current, Deploy time and Deploy expiration time definition

$06 DCR0	Config. in DIAG, SAFING, SCRAP, ARMING state
Dep_current, bit [5:4]	00 - 11= not used 01 = 1.75A min 10 = 1.2A min
Deploy_timer, bit [7:6]	00 = no deploy 01 = 0.5ms 10 = 0.7ms 11 = 2ms
Dep_expire_time, bit [3:2]	00 = 500ms 01 = 250ms 10 = 125ms 11 = 0ms

Note: DWELL time is defined in SAFING configuration. DWELL time is the period of time in which the ARMING signal once asserted, is valid waiting for a deployment command.

DEPLOY EXPIRE TIME is defined in DEPLOYMENT configuration. DEPLOY EXPIRE TIME is the period of time in which the deployment command, once received, is valid waiting for the ARMING signal asserted.

Note: The combination 1.75 A, 2 ms is not allowed.

If the above mentioned case should happen, the IC changes the set-up into 1.2 A / 500 µs and flags the bit CHxDD, in DSRx register (deployment status register), one per each channel:

$13 DSRx register, x=0, channel 0

$14 DSRx register, x=1, channel 1

$15 DSRx register, x=2, channel 2

$16 DSRx register, x=3, channel 3

$13 DSR0 register	
CH0DD, bit [13]	0 = correct current / time combination 1 = incorrect current / time combination.

Note: In order to perform deployment, all the deploy configurations registers ($06 DCR_0, $07 DCR_1, $08 DCR_2, $09 DCR_3) have to be accessed, also if the default values have just to be confirmed; otherwise IC inhibits deployment. DRCxERR bit in DSRx register reports this information.

$13 DSR0 register	
DRCxERR, bit [12]	0 = Deploy configuration change accepted and stored in memory 1 = Deploy configuration change rejected because deploy is in progress (or DEP_EXPIRE_TIME changed when in DEP_ENABLED state)

For each channel, the deploy requires that high side and low side are enabled first and then switched on.

The pictures below show the states of the IC and the signal paths which enable the high side and low side.

Figure 37. Device functionality

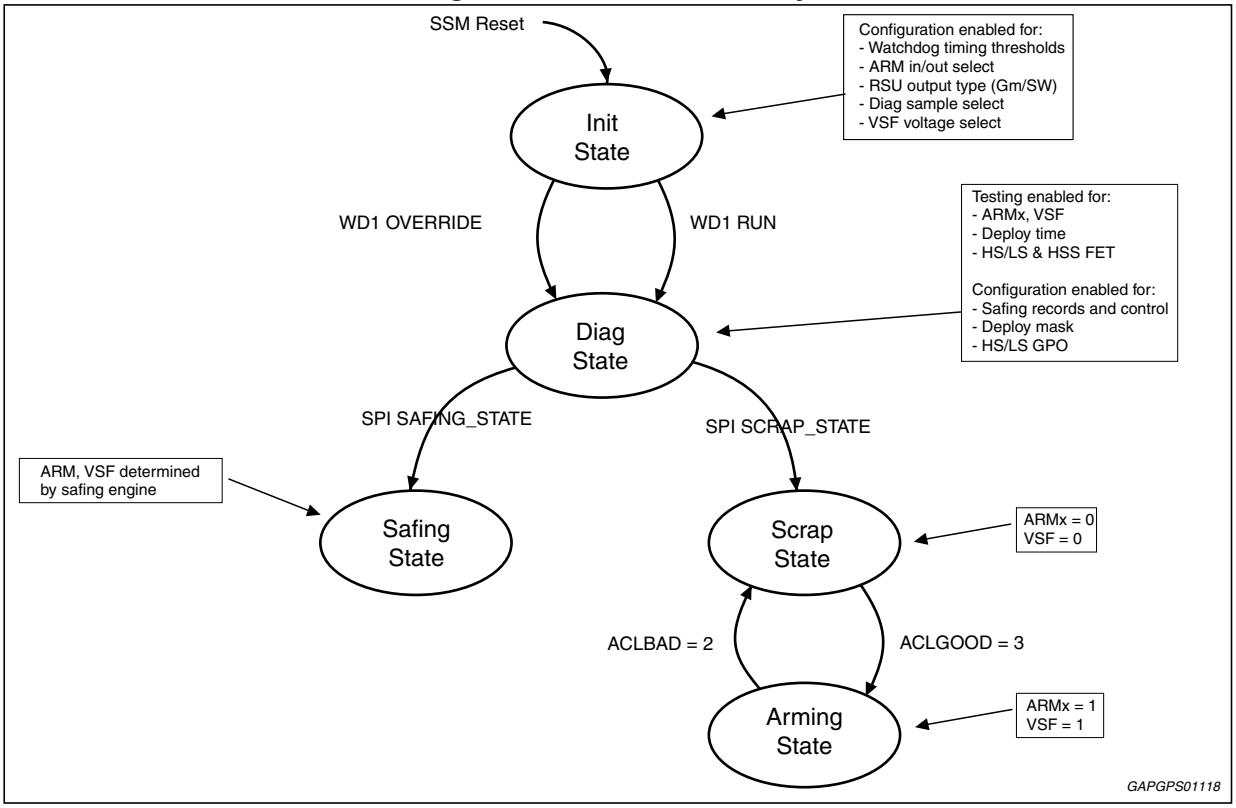
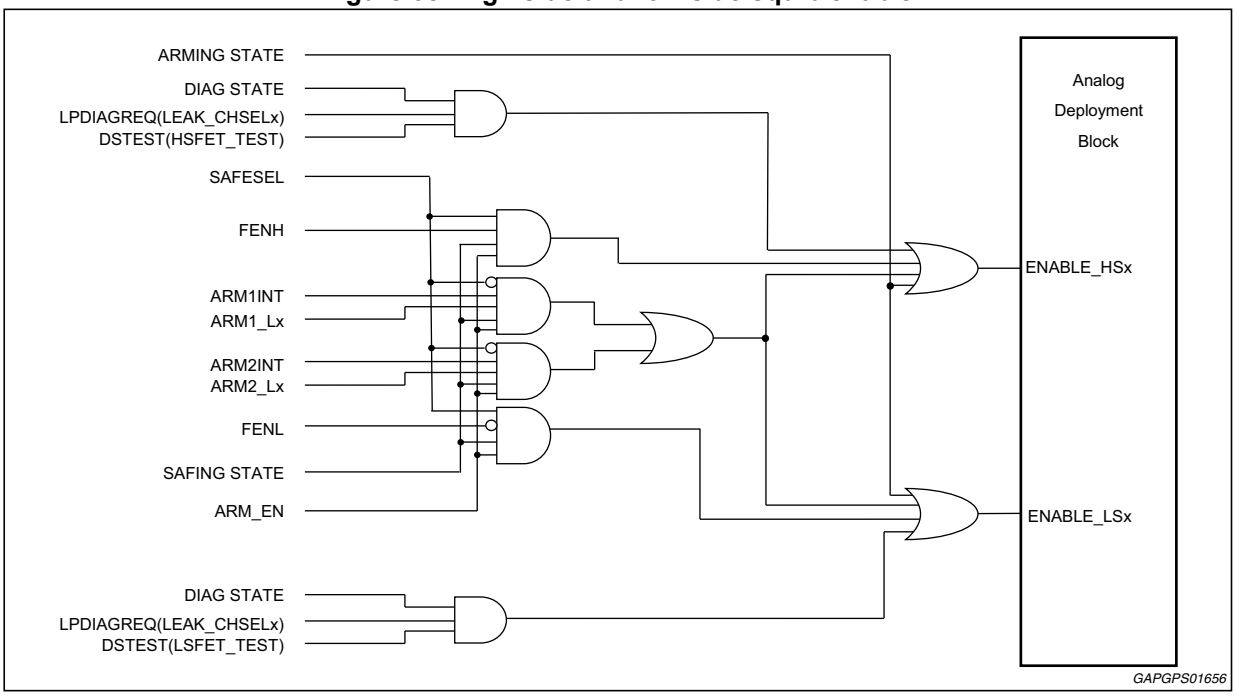


Figure 38. High side and low side squib enable

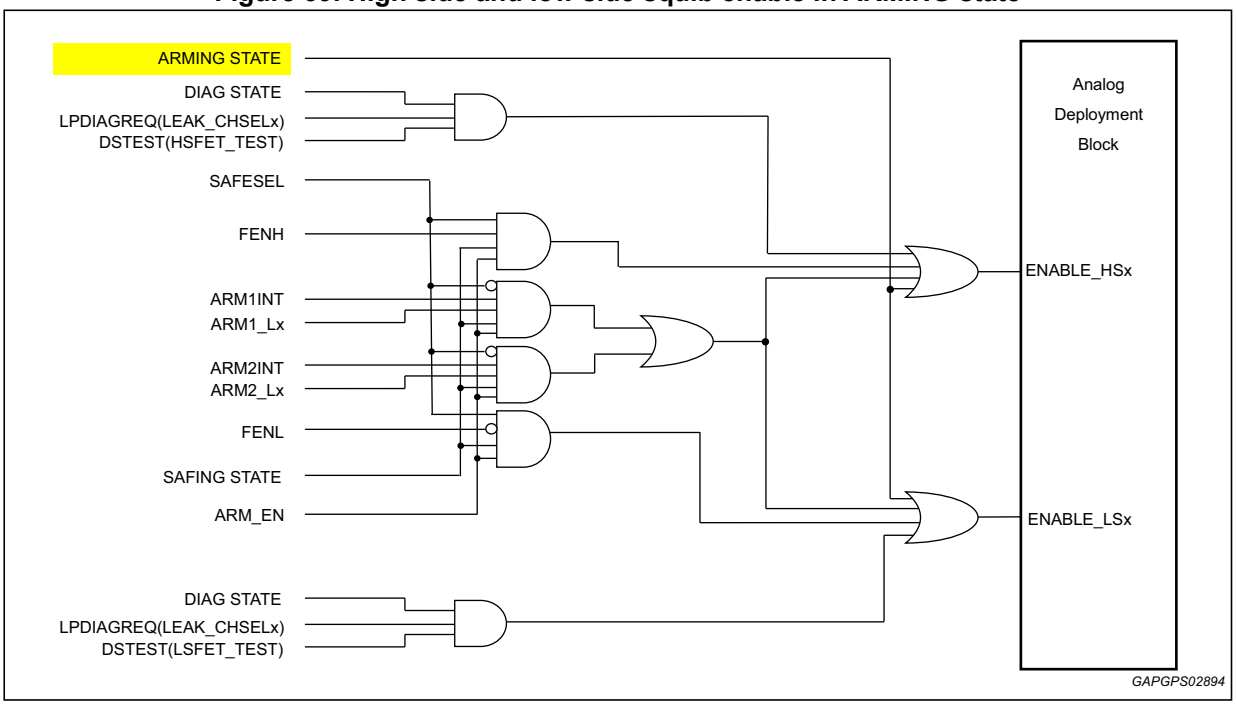


In the next pages the conditions required to deploy are considered.

8.1.1 ARMING state

This state corresponds to the disposal of the vehicle (see figures *37* and *39* for correspondent signals). In this state the high side and low side are enabled.

Figure 39. High side and low side squib enable in ARMING state



8.1.2 DIAGNOSTIC state

In DIAGNOSTIC state (see *Figure 37*) it is possible to perform the high side FET test and low side FET test.

These tests require a sequence of steps, as listed here below. Correspondent signals are summarized in *Figure 40*.

1. Chose the channel:

$38 LPDIAGREQ	Config. in DIAG, SAFING, SCRAP, ARMING state
LEAK_CHSEL, bit [3:0]	0000= CHANNEL 0 0001= CHANNEL 1 0010= CHANNEL 2 0011= CHANNEL 3

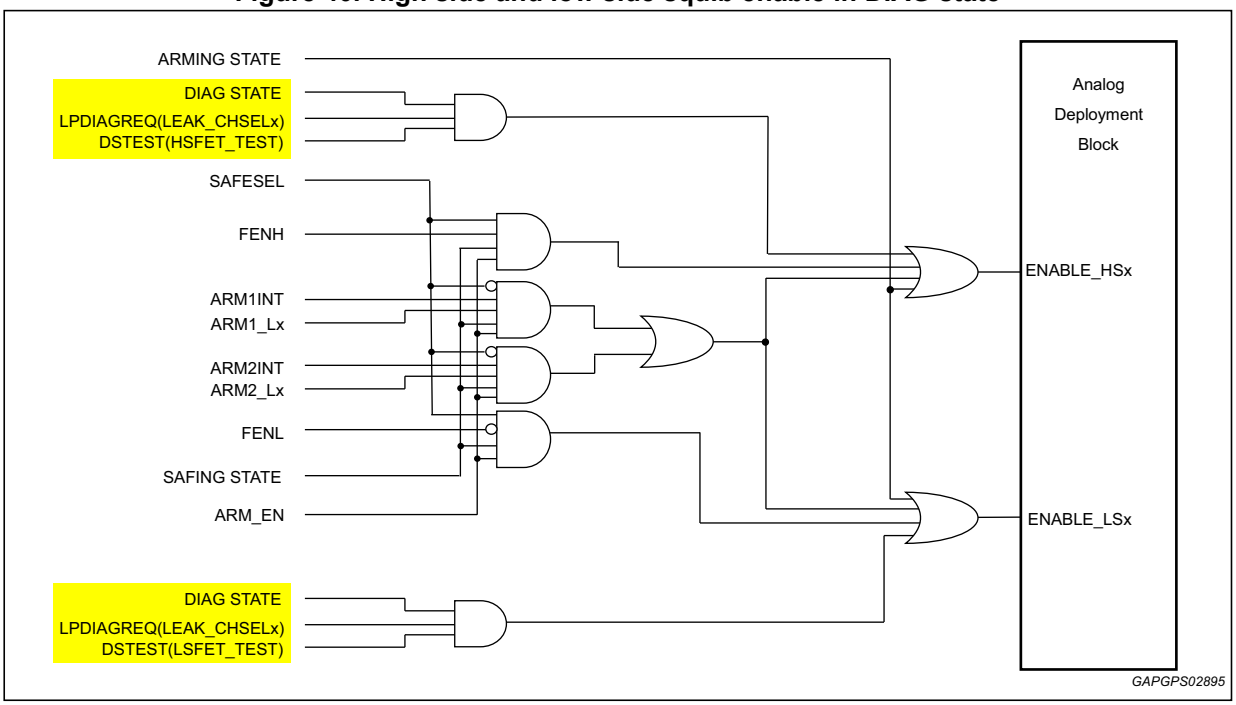
2a. Chose the test, high side FET:

$36 SYSDIAGREQ	Config. in DIAG state
DSTEST [3:0]0	0111 = high side FET test active

2b. Chose the test, low side FET:

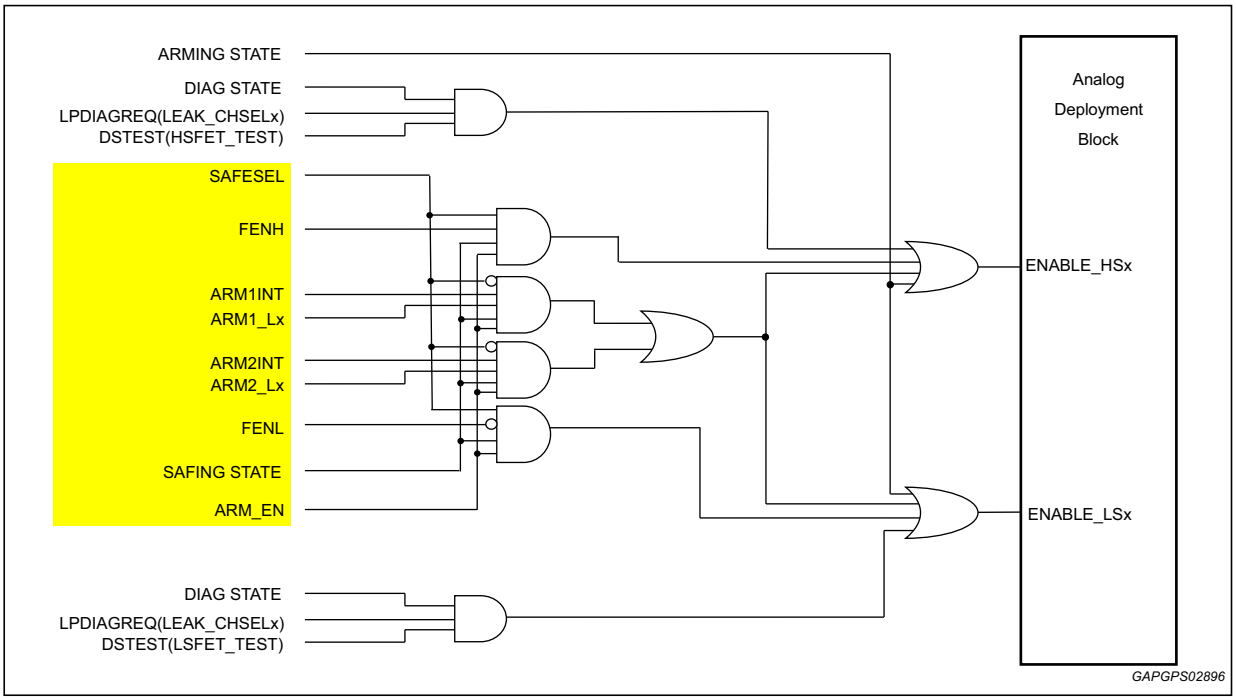
$36 SYSDIAGREQ	Config. in DIAG state
DSTEST [3:0]0	1000 = low side FET test active

Figure 40. High side and low side squib enable in DIAG state



8.1.3 SAFING state

Figure 41. High side and low side squib enable with ARMING signal



If WD_LOCKOUT is not set (that is ARM_EN=1 in the *Figure 41*), high side and low side enable depends on the safing engine machine, internal or external, in according to SAFESEL bit.

$01 SYS_CFG	Config. in INIT state
SAFSEL, bit 3	0 = internal safing engine 1 = external safing engine - default

In case of **external** safing engine (SAFESEL=1), the signals to be considered are FENH, active high, and FENL, active low.

Note: If the external safing engine is used, FENH and FENL drive directly the output. So, if their status changes during a deployment, passing from their active state to their inactive state, the deployment is immediately interrupted.

If the internal safing engine is chosen, FENH and FENL are ignored.

Note: The internal arming signals (ARM1INT, ARM2INT) drive, at the same time, the high side and the low side;

The external arming signals (FENH, FENL) drive the high side and the low side separately.

Note: If the internal safing engine is used, it is recommended keeping FENH and FENL in their inactive status, FENH=L, FENL=H to prevent that in case of safing internal engine fault, the arming signal is set.

In case of **internal** safing engine (SAFESEL=0), the signals to be considered are **ARM1INT**, **ARM2INT, ARM1_Lx, ARM2_LX**.

ARM1_Lx, ARM2_Lx signals are used to link the ARM signals, to the deployment loop.

Case x=1

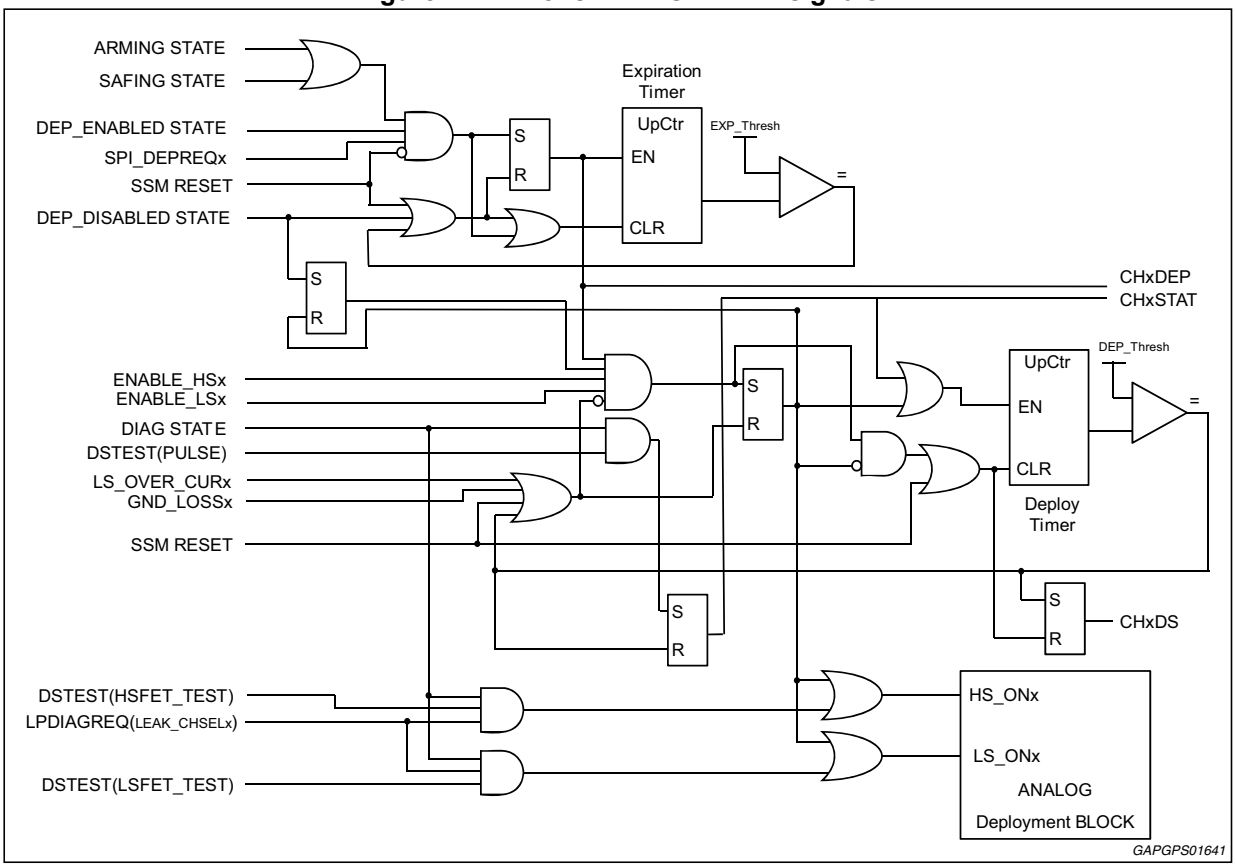
$6E LOOP_MATRIX_ARM1	Config. in DIAG state
ARM1_L3, bit 3	0 = ARM1 not associated to loop 3 1 = ARM1 associated to loop 3
ARM1_L2, bit 2	0 = ARM1 not associated to loop 2 1 = ARM1 associated to loop 2
ARM1_L1, bit 1	0 = ARM1 not associated to loop 1 1 = ARM1 associated to loop 1
ARM1_L0, bit 0	0 = ARM1 not associated to loop 0 1 = ARM1 associated to loop 0

The same for the other ARM2, mapped on register

$6F LOOP_MATRIX_ARMx, x=2

8.1.4 DEPLOYMENT driver

Figure 42. Driver's DEPLOYMENT signals



List of the requirements to deploy

1. high side and low side are enabled, see previous point (ENABLE_HS & ENABLE_LS =1)
2. no problem on SSM reset that means internal voltage references are at their correct value (POR=0), VDD5 & VDD3V3 voltage regulators at their correct value too (WSM_reset=0) and WD asserted or overridden (SSM_reset=0).
3. IC in ARMING or SAFING state
4. SPI DEPREQx, x=0, 1, 2, 3

$12 DEPCOM	Config. in SAFING and ARMING state
CHxDEPREQ, bit x	0 = no change to deploy. control ch x 1 = clear and start the expiration timer in ARMING, SAFING and DEP_ENABLED state

5. DEP_ENABLE_STATE SPI, unlock command:

$25 SPIDEPEN	Config. in SAFING and ARMING state
DEPEN_WR, bit [15:0]	$0FF0 = LOCK enter deploy disable state
	$F00F = UNLOCK enter deploy enable state

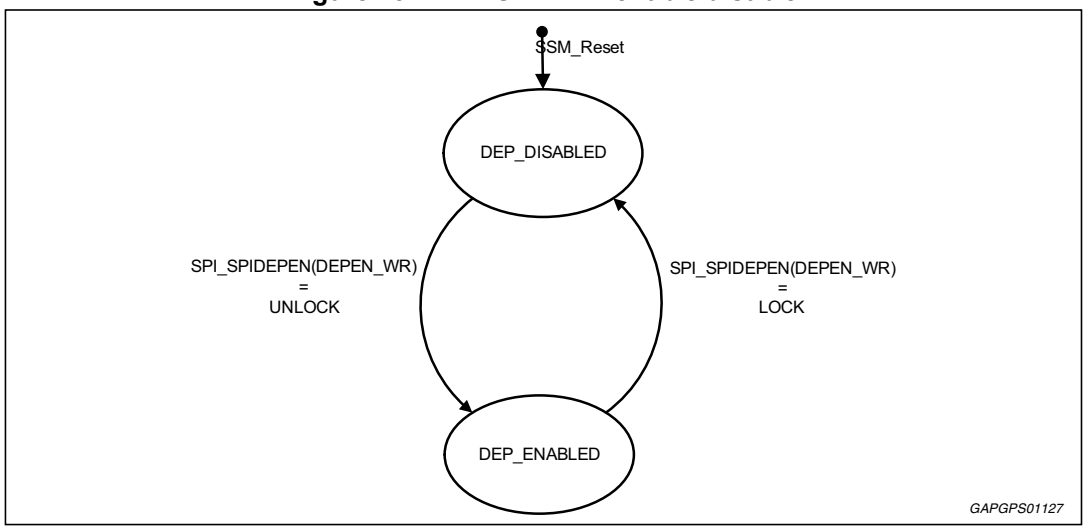
As the SSM_Reset is released, to perform the deployment the DEPEN_WR = UNLOCK (F00F) is required for the channel that has to deploy.

After a deploy, the next deployment requires a toggle DEPEN_WR = UNLOCK(F00F) - LOCK(0FF0)-UNLOCK (F00F) for the channel that has to deploy.

The same is necessary in case of a multiple deployment request, after each deployment event.

Figure 43 summarizes these requirements:

Figure 43. DEPLOYMENT enable/disable



Once the 5 above points are satisfied, the Expiration time counter starts.

This counter takes into account the feature of the IC to accept a deploy command even if the arming is not yet serviced. If the arm command occurs inside the expiration time, the deployment takes place otherwise the deployment command is discharged.

Dep_exp_time is defined in DRCx (x=0-3) registers, together with the Deploy_timer and Dep_current,

$06 DCRx → x=0 channel 0

$07 DCRx → x=2, channel 1

$08 DCRx → x=3, channel 2

$09 DCRx → x=4, channel 3

Case x=0:

$06 DCR0	Config. in DIAG, SAFING, SCRAP, ARMING state
Dep_current, bit [5:4]	00 - 11= not used 01 = 1.75A min 10 = 1.2A min
Deploy_timer, bit [7:6]	00 = no deploy 01 = 0.5ms 10 = 0.7ms 11 = 2ms
Dep_expire_time, bit [3:2]	00 = 500ms 01 = 250ms 10 = 125ms 11 = 0ms

Once the deployment is started, any DEP_EN = $0FF0 (that means deploy disable), is ignored. If the same command arrives before the deployment has been started, the deployment is really disabled and the deploy command ignored.

Once the deployment is started, it can be interrupted by

over-current in the low side

GND loss

SSM reset

End of deployment time

Status of the deployment is reported to the microcontroller through SPI read out in deployment status register:

$13 DSRx register, x=0, channel 0

$14 DSRx register, x=1, channel 1

$15 DSRx register, x=2, channel 2

$16 DSRx register, x=3, channel 3

Case x=0, the same for the others:

$13 DSR0	
CH0STAT, bit [14]	0 = deployment not in progress 1 = deployment in progress
CH0DS, bit [15]	0 = deployment not successful 1 = deployment successful

If the deployment command lasts for the programmed deploy time, the flag CHxDS (deploy success) is set.

The event is also reported in GSW, DEPOK bit that is the "OR" of the deployment success of all the four channels.

MISO BIT	31	30	29	28	27	26	25	24	23	22	21
MISO	SPIFLT	DEPOK	RSFLT	WDTDIS_S	ERSTATE	POWERFLT	FLT	CONVRDY2	CONVRDY1	ERR_WID	ERR_RID
GSW BIT	10	9	8	7	6	5	4	3	2	1	0

GSW	DEPOK, bit 30 (bit 9 of GSW)
	0 = all DSR_x / CHxDS bits are = 0 (no deployment success on all channel)
	1 = at least a deployment successful on the channels.

In case the deploy success=1, this doesn't mean that the current is really passed through the squib for the programmed time.

This bit means only that no inhibition of deployment has been received (in external safing engine FENH/FENL could have disabled the high side or the low side).

In order to know if the current is really passed through the squib the Deploy Current_Mon registers have to be read.

8.2 Deployment driver protection

In order to avoid to damage the IC due to eventual free wheeling, two protections are implemented.

1. after a deployment, once the High Side is switched off, the low side is kept on for $t_{DEL\_SD\_LS}$ (50 µs min.) in order to allow fly-back.
2. once low side is switched off, a protection against the overvoltage through a clamp structure is implemented.

On the Low Side there is a current limitation and overcurrent protection circuit that attends limiting the current at I_{LIM_SR} (2.2 A ÷ 4 A) I_{OC_SR} (2.2 A ÷ 4 A) avoiding, in case of pin **short to battery**, the channel's damage. If the malfunction lasts over $t_{FLT\_ILIM\_LS}$ (100 µs typ) the whole channel (High and Low Side) is switched off until a new deployment command, via SPI_DEPEN occurs.

The squib driver can stand the **short to ground** of the pins during the deployment, because the high side current is limited by the high side itself.

It can manage also the case of SR short to ground after an open circuit, because it is able to detect the open circuit condition and then limiting the current overshoot as the open circuit disappears.

In case of squib's intermittence during deployment phase, current limitation is ensured by the Low Side current limitation, I_{LIM_SR}. If the condition lasts longer than t_{LIMOS} (20 µs max) the High Side is switched off for $t_{OFF\_OS\_HS}$ (4 µs ÷ 12 µs) and then on again.

This allows distinguish Open Load and Low Side short to battery cases and then proper manage them.

8.3 Deployment driver example

Table 4. Deployment driver example

Register	(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description
$01 SYS_CFG	I	W	X	X	X	X	X	0	1	0	0	0	0	X	0	0	X	1	10: 01=short time 9: RSU switch 8, 7: 8 sample DC-squib-temp measure 6, 5: 4 sample 3: internal saf eng 2: VSF=20V 0:timeout disable
$04 SYS_STATE	-	R						0	0	0						0	1	0	10, 9, 8 000=INIT 2, 1, 0: 010=RUN
$2A WDTCR	I	W	X	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	14: WDTMODE = FAST 13÷7: WDTMIN = 400us 6÷0: WDT DELTA = 200us
2C WD_STATE	-	R						0	0	0									10÷8: WD1_STATE=INITIAL
2B WDIT	-	W																	Service watchdog A/B/A….
$04 SYS_STATE	-	R						0	0	1						0	1	0	10, 9, 8: 001=DIAG 2, 1, 0: 010=RUN
$7F SAF_ENABLE	(I)	W	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	1	3÷0: SAF1 enabled, other disabled
$80 SAF_REQ_MASK_1	D	W	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	14÷8: chose RID [6:0] bit
$93 SAF_REQ_TARGET_1	D	W	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	14÷8: RID [6:0] bit selection (50=RSU1)
$A6 SAF_RESP_MASK_1	D	W	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	15÷10: chose RSU status bit
$B9 SAF_RESP_TARGET_1	D	W	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15÷10: expected bit status of RSU
$CC SAF_DATA_MASK_1	D	W	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	9÷0: 10 data RSO data bit selection

Table 4. Deployment driver example (continued)

Register	(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description
$EF SAF_CONTROL_1	D	W	0	0	1	X	0	0	0	0	X	X	1	1	1	0	1	1	14, 15: 00=arming on ARMP or ARMN 13: 1=last 16 bit in RESP_MASK/DATA_MASK 11: DATA RANGE LIMIT 0=disable 10: COMB 0=disabled 9, 8: DWELL 00 = 2048ms 5: 1=SAF_RECORD_1 assigned to ARM2INT 4: 1=SAF_RECORD_1 assigned to ARM1INT 3÷1: 101=SPI_CS 0: 1=SPI MISO answer in frame
$DF SAF_THRESHOL_1	D	W																	Set threshold
$66 SAF_ALGO_CONF	D	W	0	X	0	0	1	1	0	0	1	1	0	1	1	0	0	1	15: counter reset 13÷10: 0011=NEGATIVE EVENT COUNT=3 9÷6: 0011=POSITIVE EVENT COUNT=3 5÷3: 011=-3 2÷0: 001=1
$6E LOOP_MATRIX_ARM1	D	W	X	X	X	X	X	X	X	X	X	X	X	X	1	0	1	0	3, 1: ARM1 assigned to L1 and L3
$6F LOOP_MATRIX_ARM2	D	W	X	X	X	X	X	X	X	X	X	X	X	X	0	1	0	1	2, 4: ARM2 assigned to L0 and L2
$02 SYS_CTL	-	W	X	X	X	X	X	X	0	X	1	1	1	0	X	X	X	X	9: ERBOOST=23V 7: ER_CHARGE ON 6: ER_BOOST ON 5: VSUP ON 4: SPI_OFF not required

Table 4. Deployment driver example (continued)

Register					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description
$05 POWER STATE					-	R	0	0	0	0	0	0	0	1	1	X	X	0	1	1	1	1	15: 0=VIN>VINGOOD expected 14: 0=CDD3V3>VDD3V3_UV expected 13: 0= CDD3V3<VDD3V3_OV expected 12: 0=ER_BOOST>ER_BOOST_OK expected 11: 0= VDD5>VDD5_UV expected 10: 0=VDD5<VDD5_OV expected 9: 0=VTHLSUP<VSUP<VTHHSUP expected 8: 1=ERBOOST ON expected 7: 1=ER_CHARGE ON expected 4: 0=ER_SWITCH OFF expected 3: 1=VDD5 ramp up or ON expected 2: 1= VSUP ramp up or ON expected 1: 1=VSUP ON expected 0: 1= VSF_EN=1 expected
(3)	19	18	17	16																			19: 1=WAKEUP>WU_on expected 18: 0=VBATMON>VBBAD expected 17: 0=VBATMON>VBGOOD expected 16: 0=VIN>VINBAD expected
	1	0	0	0																			
$4A RSCR1					D	W	X	X	X	X	X	0	X	X	X	X	X	X	0	0	1	0	10: blanktime=5ms 3÷0: 0010=PSI5async, 10bit, 125khz
$4E RSCTRL					(I)	W	X	X	X	X	X	X	X	X	X	X	X	X	0	X	1	X	3: 0=CH1EN OFF 1: 1= CH0EN ON
$06 DCR_0					(I)	W	X	X	X	X	X	X	X	X	0	1	0	1	0	0	X	X	7, 6: 01=0.5ms deploy time 5, 4: 01=1.75A deploy current 3, 2: 00=500ms deploy expiration time

Table 4. Deployment driver example (continued)

Register	(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description
$07 DCR_1	(I)	W	X	X	X	X	X	X	X	X	1	0	1	0	0	0	X	X	7, 6: 10=0.7ms deploy time 5, 4: 10=1.2A deploy current 3, 2: 00=500ms deploy expiration time
$08 DCR_2	(I)	W	X	X	X	X	X	X	X	X	0	1	0	1	0	0	X	X	7, 6: 01=0.5ms deploy time 5, 4: 01=1.75A deploy current 3, 2: 00=500ms deploy expiration time
$09 DCR_3	(I)	W	X	X	X	X	X	X	X	X	1	0	1	0	0	0	X	X	7, 6: 10=0.7ms deploy time 5, 4: 10=1.2A deploy current 3, 2: 00=500ms deploy expiration time
$13 DSR_0	-	R			0														13: 0=CH0DD correct time/current expected
$14 DSR_1	-	R			0														13: 0=CH1DD correct time/current expected
$15 DSR_2	-	R			0														13: 0=CH2DD correct time/current expected
$16 DSR_3	-	R			0														13: 0=CH3DD correct time/current expected
$31 SAFING_STATE	D	W	1	0	1	0	1	1	0	0	1	0	1	0	1	1	0	0	15÷0: ACAC from DIAG to SAFING state
$04 SYS_STATE (4)	-	R						0	1	0						0	1	0	10, 9, 8: 010=SAFING expected 2, 1, 0: 010=RUN expected
$6A ARM_STATE	-	R													1	1			3: 1=ARMINT_2 ok expected 2: 1=ARMINT_1 ok expected
$50 RSDR0 (5)	-	R	0	1															15: FLT=0 expected 14: ON/OFF=1 expected 13÷10: LCID 0000 expected 9÷0: DATA

Table 4. Deployment driver example (continued)

Register	(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Description
$FF SAF_CC (5)	-	R													1	1	1	1	3: 1=CC_4 expected 2: 1=CC_3 expected 1: 1=CC_2 expected 0: 1=CC_1 expected
$25 SPIDEPEN	S, A	W	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	$F00F=UNLOCK
$12 DEPCOM	S, A	W	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	3: 1=CH3DEPLOY 2: 1= CH2DEPLOY 1: 1= CH1DEPLOY 0: 1= CH0DEPLOY
$25 SPIDEPEN	-	R	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	$0FF0=LOCK
$13 DSR_0	-	R	1																15: 1=CH0DS deployment successful
$14 DSR_1	-	R	1																15: 1=CH1DS deployment successful
$15 DSR_2	-	R	1																15: 1=CH2DS deployment successful
$16 DSR_3	-	R	1																15: 1=CH3DS deployment successful

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES (I) = no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
   W = WRITE
3. Further bit over the 16 standard.
4. Once a deployment successful, restart from point *5 if no parameters have to be changed.
5. repeat the sequence read $50 read $FF up to the data received is greater than threshold or lower than the -threshold defined in $DF for a number of time equal to that defined in $66.

9 Diagnostic

For all channels the following diagnostics are implemented:

- High voltage leak test, for SFx, SRx oxide isolation
- Leakage to battery/ground for SFx SRx with/ without squib
- Loop to loop short diagnostic
- Squib resistance measurement -leakage cancellation
- High squib resistance, 500 ÷ 2000
- SSxy, SFx, VER voltage monitor
- High & Low FET diagnostics
- High side driver diagnostic
- Loss of ground
- High Side Safing FET diagnostic
- Deployment timer diagnostic

These diagnostics data are elaborated by a 10 bit ADC converter.

Diagnostic can be done in two ways: high level or low level.

In **high level diagnostic**, the set-up for each requested measurement is managed by the device itself.

In **low level diagnostic**, the set-up for each requested measurement is managed by an external logic, step by step.

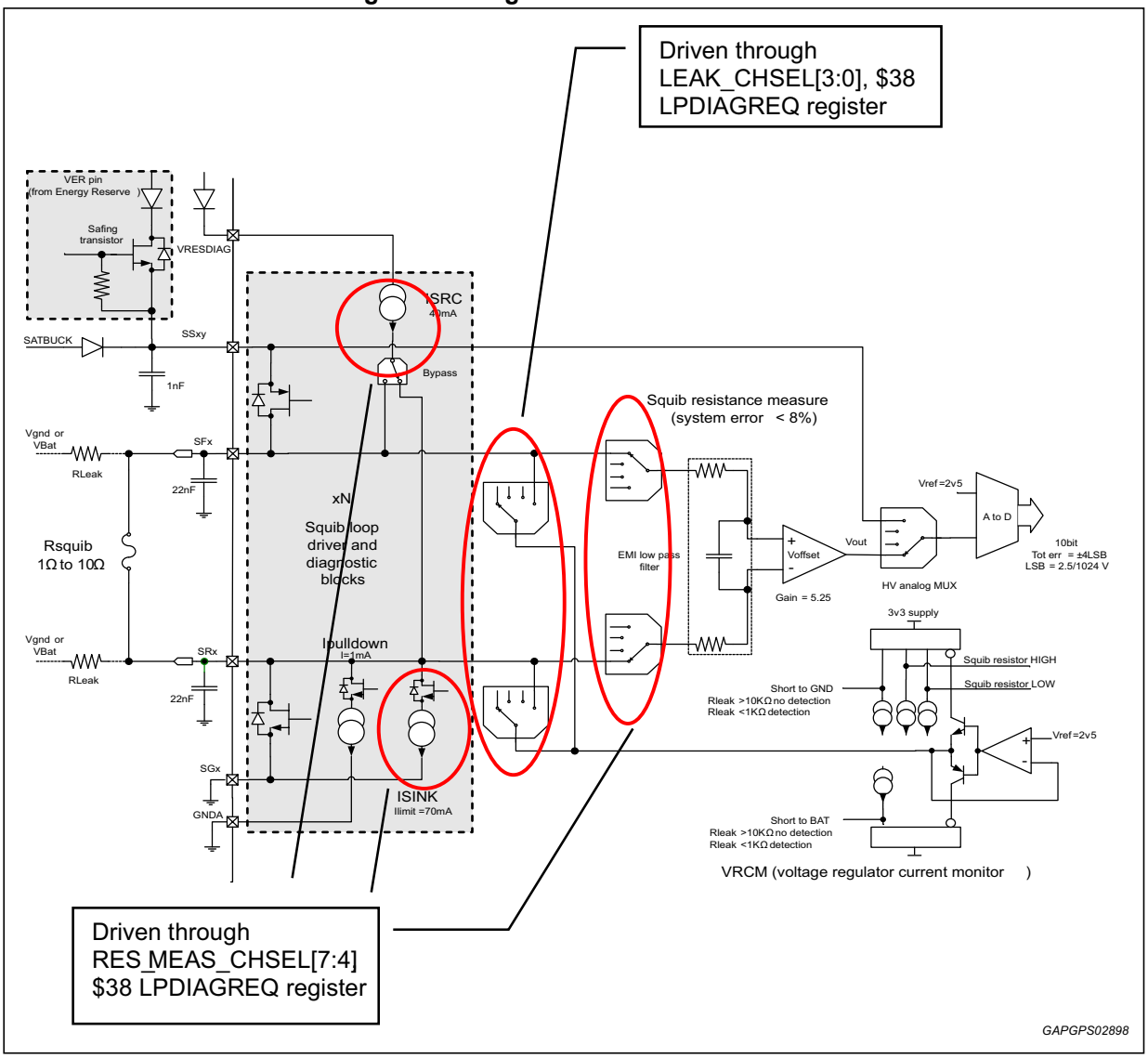
The choice of high level or low level diagnostic is done via SPI:

$38 LPDIAGREQ	Config. in DIAG, SAFING, SCRAP, ARMING state
DIAG_LEVEL, bit 16	0 = low level 1 = high level

In the next figure the relevant blocks used for the diagnostic are reported.

In particular there are a Voltage Regulator Current Monitor (VRCM) and three current generators that withstand diagnostic operations, ISRC (40 mA), ISNK (limit 70 mA), pull-down (1mA).

Figure 44. Diagnostic - blocks overview



9.1 Low level

1. ER charge has to be previously turned ON before running the diagnostic;
2. verify that the IC is in DIAG state, reading register $04;
3. decide, writing the appropriate bit in reg. $38, which diagnostic mode is used;

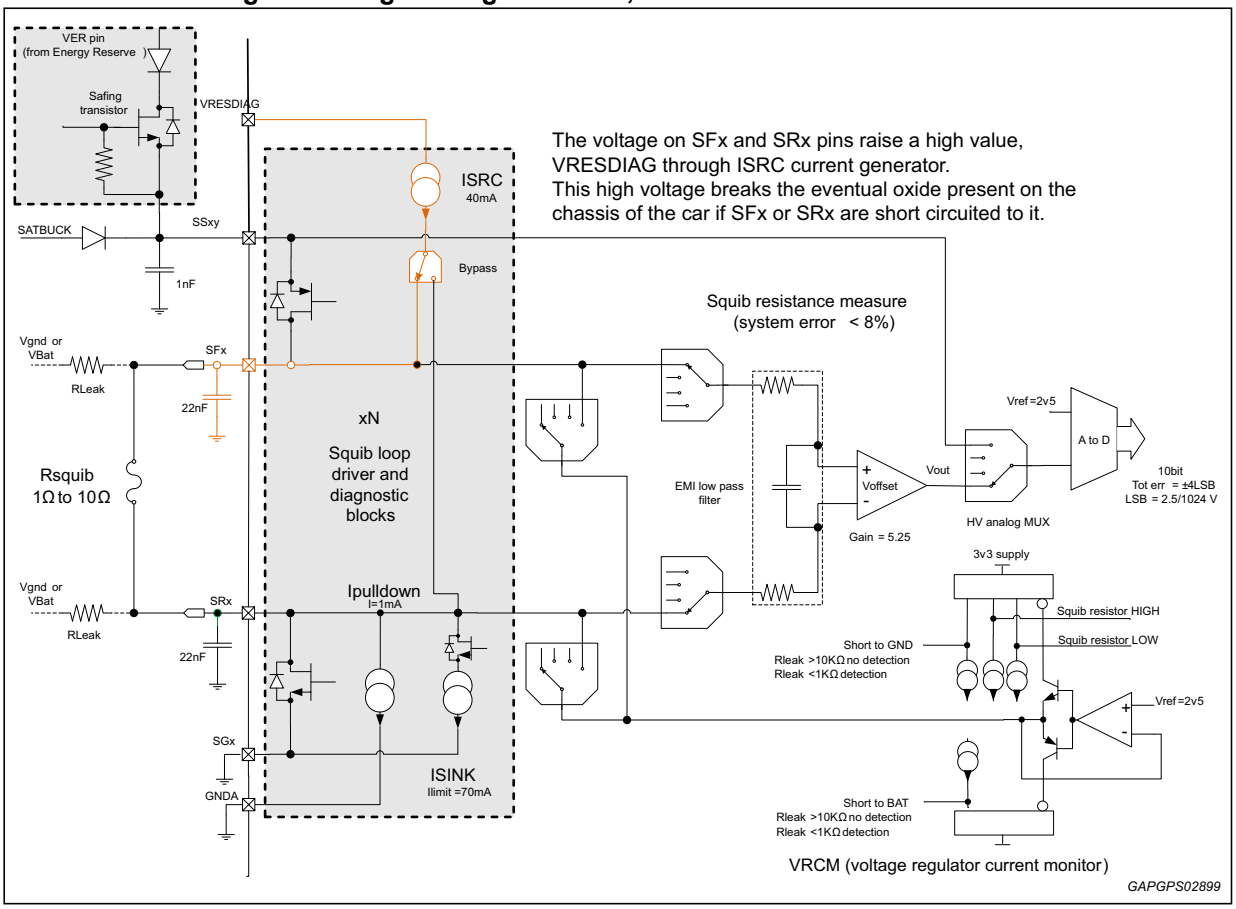
		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	**$02 SYS_CTL**	-	W	X	X	X				0	X	1	1	0	0	X	X	X	X	12:, VIN_TH_SEL depends on application; 11,10: VBATMON_TH_SEL depend on application; 9: ER_BST_V 0=23V, 1=33V 7: ER_CUR_EN 0=OFF, 1=ON 6: ER_BST_EN 0=OFF, 1=ON, 5: VSUP_EN 0=OFF, 1=ON 4: SPI_OFF 0=no effect, 1=POWER OFF required
2	**$04 SYS_STATE**		R						0	0	1									10, 9, 8 001=DIAG 2, 1, 0: 010=RUN
3	**$38 LPDIAGREQ**	(I)	W	0	14:0 define the test, see next chapters															15: 0=LOW LEVEL diag setup
4	**$37 LPDIAGSTAT**		R	0	14:0 define the test, see next chapters															15: 0=LOW LEVEL diag
	(3) 19 18 17 16																			
4	**$3xDIAGCTRL_x X=A, B, C, D**		W		X	X	X	X	X	X	X	X	6:0 ADC address							
	(3) 19 18 17 16			16:10 ADC address						9:0 ADC result										19:1=conversion finished
	1																			

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

In low level mode, the IC performs the measurement, following external requests. Each test set-up is driven, step by step, by the microcontroller and the timing for the measurement is fixed by the microcontroller too.

9.1.1 High voltage leak test, oxide isolation IC-car chassis

Figure 45. High voltage leak test, oxide isolation IC-car chassis



This test is mandatory and verifies that no leakages are present on SFx or SRx pins when high voltage is applied.

ISRC current generator is ON and addressed on SFx.

If there is no leakage, SFx raises up to VRESDIAG and, being the impedance between SFx and SRx very low (squib connected), SRx follows SFx.

Confirmation of this is done through an ADC measurement request of SFx voltage value.

Supply= VRESDIAG

SET-UP, see *Figure 45*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	0	1	0	0	0	RES_MEAS_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				LEAK_CHSEL 0100:1111				15: 0=DIAG LOW LEVEL 13:1= pull-down curr. OFF all ch 12,11: 01=ISRC=40mA RES_MEAS_CHSEL, OFF for the others 10: 0= ISINK all OFF 9,8:00 VRCM not connected 7:4 RES_MEAS_CHSEL 3:0 0100-1111 not selected

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I) = no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

SFx voltages and VRESGIAG are readable by the microcontroller through the ADC converter in the registers

$3X DIAGCTRL_X → X=A, B, C, D

Case X=A:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3A DIAGCTRL_A					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $46 = SF0 $47 = SF1 $48 = SF2 $49 = SF3 $42 = VRESDIAG							
(3)	19	18	17	16																			
	1	0	0	ADCREQ_A	-	R	ADCREQ_A $46 = SF0 $47 = SF1 $48 = SF2 $49 = SF3 $42 = VRESDIAG						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I) = no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio of the ADC. In case of SFx and VRESDIAG, it is 15:1.

Table 5. VRESDIAG and SFx measurement of the value ratio ADC

Measurements	Divider Ratio				
	15:1	10:1	7:1	4:1	1:1
VRESDIAG	√				
SFx	√				

As a conversion example, let's consider the case where the VRESDIAG conversion has been requested and the readout of the ADC register is done.

VRESDIAG = 22.6 V measured

ADC = $(100110100)_2 = (616)_{10}$

In order to obtain the result in Volt, being the ADC characteristic linear,

$$2.5 : 1024 = x : ADC \rightarrow x = \frac{616 \cdot 2.5V}{1024} = 1.5V$$

GAPGPS02900

Considering the divider ratio (DR) stated above, the result is x * DR = 1.5 * 15 = 22.6 V

Test result:

In case of leakage on High (SFx) or Low Side (SRx), SFx voltage is not able to reach VRESDIAG and the microcontroller can detect the leakage problem, both on the high side or on the low side, with no possibility, at this stage, to distinguish which of them is involved in the problem.

9.1.2 VRCM test validation

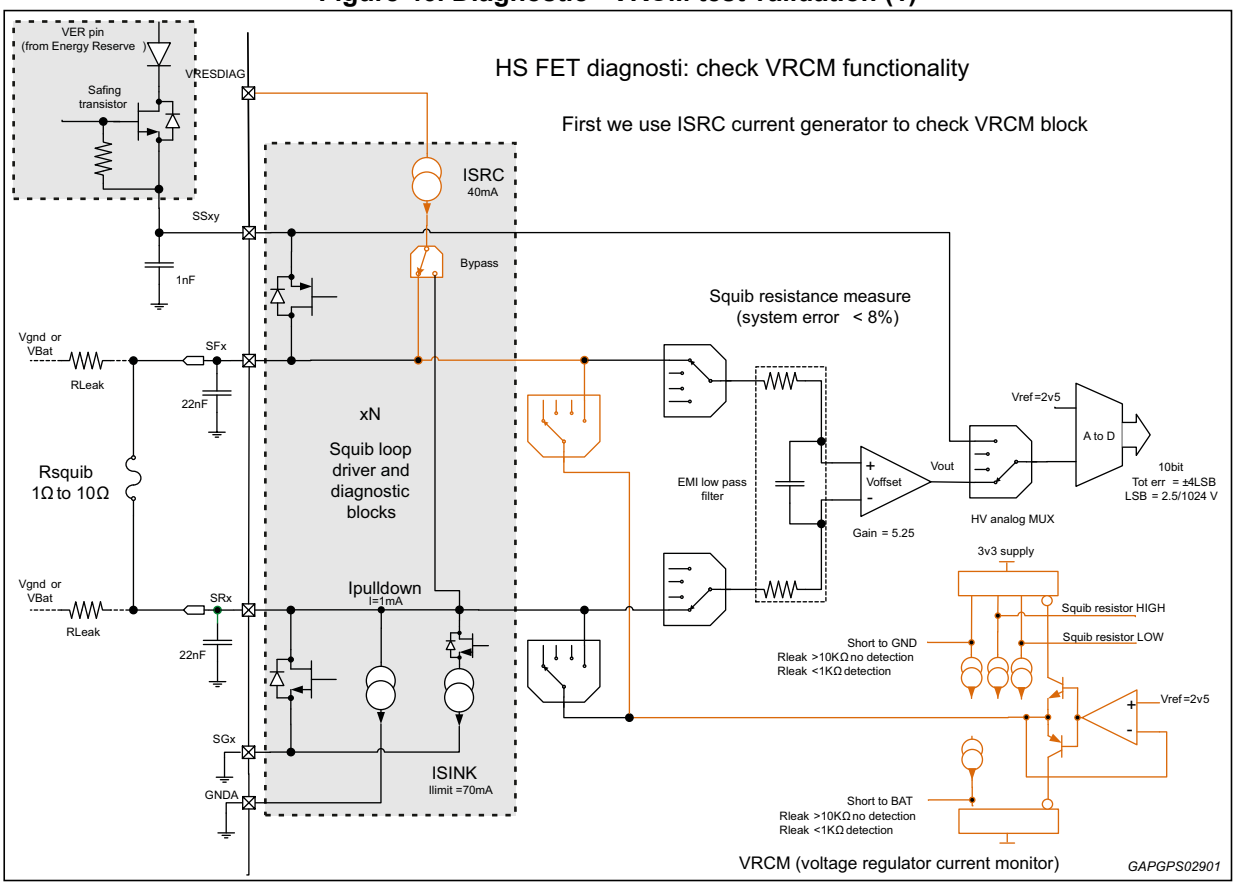
Before using VRCM block, that is used in many IC diagnostic, it is necessary a test for its validation. The test is done through short to battery and short to ground flag verification.

Measurement set-up is composed by 2 steps:

VRESDIAG supplied

1st step

Figure 46. Diagnostic - VRCM test validation (1)



For set up, refer to *Figure 46*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	0	1	0	VRCM 01		RES_MEAS_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 0=DIAG LOW LEVEL 13: 1=pulldown curr OFF all ch 12,11: 01=ISRC=40mA (RES_MEAS_CHSEL), OFF for the others 10: 0= ISINK all OFF 9,8:01 VRCM connected to SFx (LEAK_CHSEL ch) 7:4 RES_MEAS_CHSEL 3:0 LEAK_CHSEL

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R=READ / W = WRITE

RES_MEAS_CHSEL, bit[7:4] and LEAK_CHSEL, bit[3:0] must refer to the same channel.

Test result:

Being ISRC and VRCM connected to SFx, if VRCM works correctly, short to battery, readable in LPDIAGSTAT register, is asserted for the channel selected:

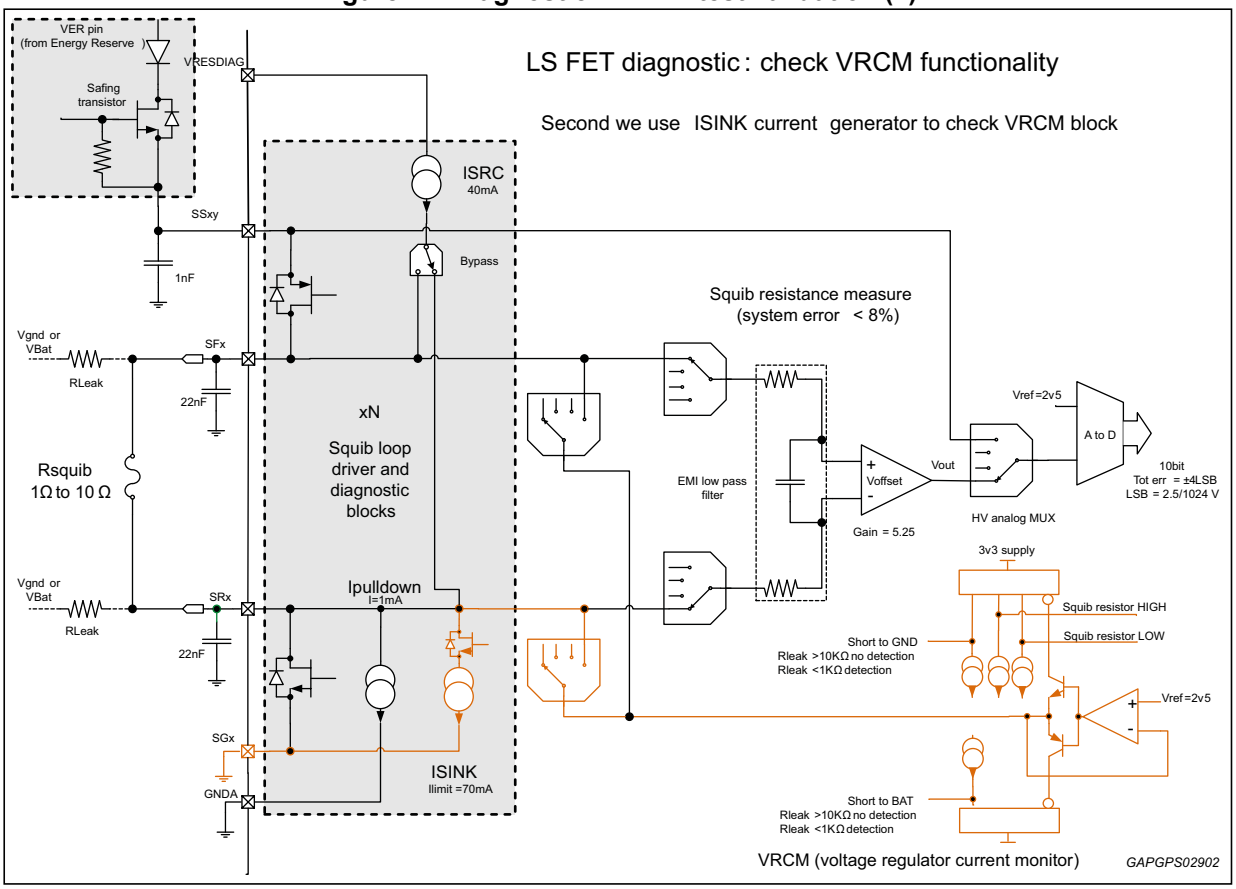
						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R					RES_MEAS_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				X	X	1	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 0= LOW LEVEL 5: 1=STB expected 4: 1=test on SFx
	(3)	19	18	17	16		R																	
		0	X	0																				

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R=READ / W = WRITE
3. further bit over the 16 standard

If the first step of VRCM test is passed, proceed with the second step.

2nd step

Figure 47. Diagnostic - VRCM test validation (2)



For set up, refer to *Figure 47*.

		(1)	(2)	15	14	13	12	11	10	9 8	7 6 5 4	3 2 1 0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	0	0	1	VRCM 10	RES_MEAS_ CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3	15: 0=DIAG LOW LEVEL 13: 1=pulldown curr OFF all ch 12,11: 00/11=ISRC OFF all channel 10: 1= ISINK ON (RES_MEAS_CHSEL) 9,8:10 VRCM connected to SRx (LEAK_CHSEL) 7:4 RES_MEAS_CHSEL 3:0 LEAK_CHSEL

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE

RES_MEAS_CHSEL, bit[7:4] and LEAK_CHSEL, bit[3:0] must refer to the same channel.

Test result:

Being ISNK and VRCM connected to SRx, if VRCM works correctly, short to ground, readable in LPDIAGSTAT register, is asserted for the channel selected:

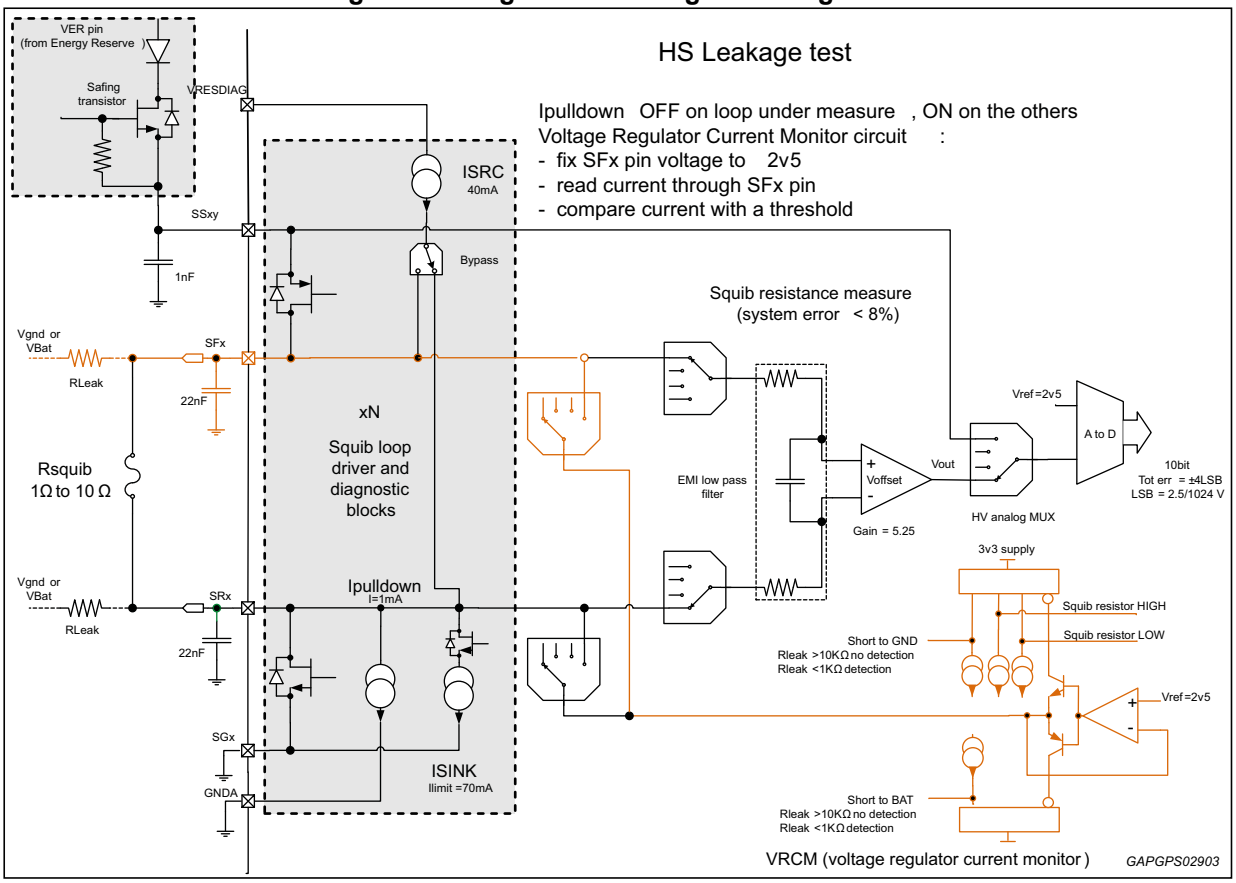
						(1)	(2)	15	14	13	12	11 10 9 8	7	6	5	4	3 2 1 0	
4	**$37 LPDIAGSTAT**						R					RES_MEAS_ CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3	X	1	X	0	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3	19: 0= LOW LEVEL 6: 1=STG expected 4: 0=test on SRx
	(3)	19	18	17	16		R											
		0	X	0														

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE
3. further bit over the 16 standard

If the second step of VRCM test is passed too, VRCM test is validated.

9.1.3 Leakage test - High side

Figure 48. Diagnostic - leakage test - high side



For set up, refer to *Figure 48*.

ISRC and ISINK are kept off and VRCM is connected to SFx, chosen through LEAK_CHSEL.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	0	0	0	0	0	1	RES_MEAS_ CHSEL 0100:1111				LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 0=DIAG LOW LEVEL 13: 0=pulldown curr OFF for VRCM ch; ON the others 12,11: 00/11=ISRCOFF on all channel 10: 0= ISINK all OFF 9,8:01 VRCM to SFx (LEAK_CHSEL) 7:4 0100-1111 no selection 3:0 LEAK_CHSEL

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE

Test result:

If there is no leakage on the high side, SFx voltage is equal to VREF=2.5V and no current is detected by VRCM itself.

SFx voltage is readable addressing the ADC read out on it. The registers involved in this operation are the four DIAGCTRL_x
$3X DIAGCTRL_X → X = A, B, C, D

Case X = A:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3A DIAGCTRL_A					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $46 = SF0 $47 = SF1 $48 = SF2 $49 = SF3							
(3)	19	18	17	16																			
	1	0	0	ADCREQ_A	-	R	ADCREQ_A $46 = SF0 $47 = SF1 $48 = SF2 $49 = SF3						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE
3. further bit over the 16 standard

Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio of the ADC. In case of SFx and VRESDIAG, it is 15:1 (see *Table 5*).

In case of a leakage (to ground or to battery), VRCM will sink or source a current to maintain SFx at VREF. As a consequence, STG or STB is set:

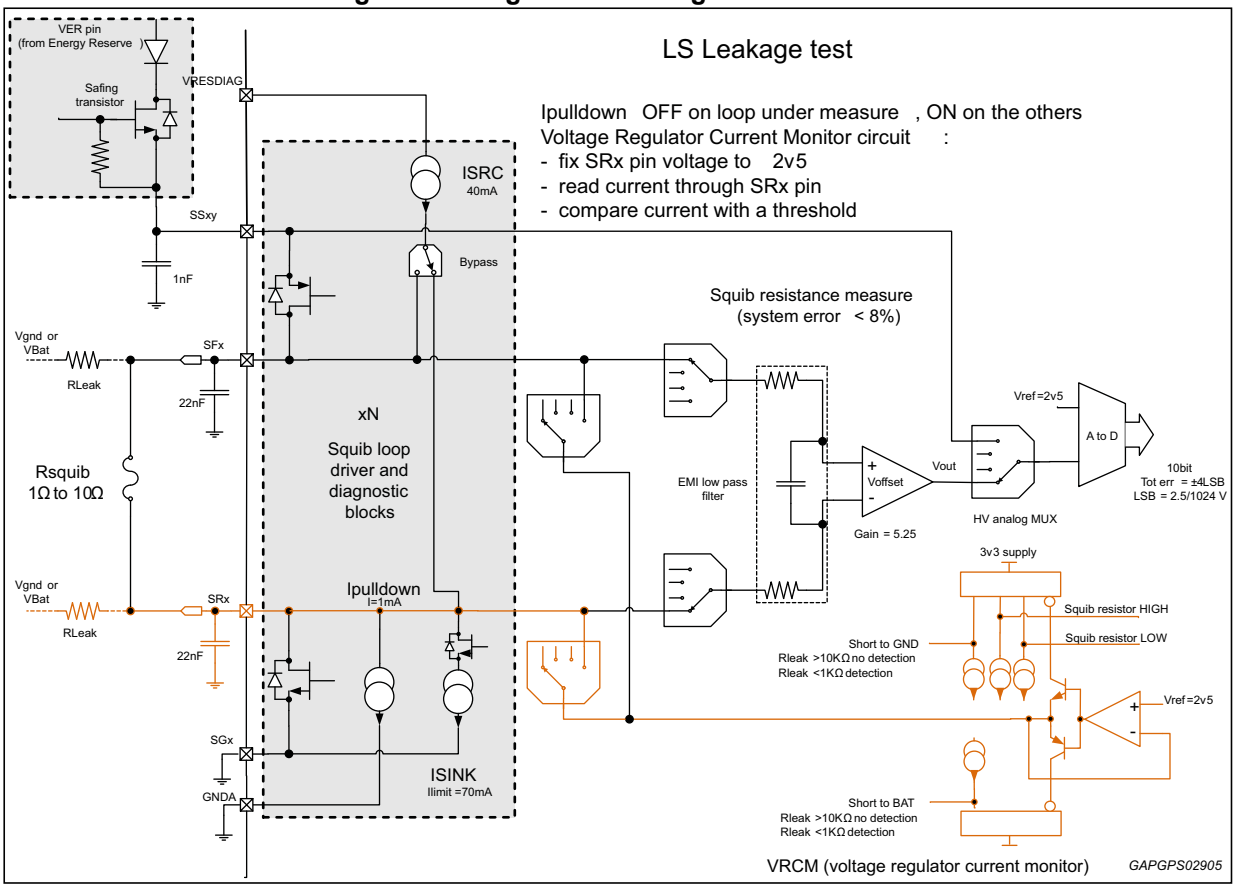
						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R					RES_MEAS_CHSEL 0100-1111				X	1	1	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 0 = LOW LEVEL 6: 1=STG if leak vs GND 5: 1= STB if leak vs BATT 4: 1 = test on SFx
	(3)	19	18	17	16		R																	
		0	X	0																				

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE
3. further bit over the 16 standard

Note: Pull-down current (1mA) is active on all the channels except the one under analysis. So, the STG requires further investigation to understand if it comes from a real short to ground of the channel itself or it comes from a short between the channel itself and another one.

9.1.4 Leakage test - low side

Figure 49. Diagnostic - leakage test - low side



For set up, refer to *Figure 49*.

ISRC and ISINK are kept off and VRCM is connected to SRx, chosen through LEAK_CHSEL.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	0	0	0	0	1	0	RES_MEAS_CHSEL 0100:1111				LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 0=DIAG LOW LEVEL 13: 0=pulldown curr OFF for VRCM ch; ON the others 12,11: 00/11=ISRCOFF on all channel 10: 0= ISINK all OFF 9,8:10 VRCM to SRx 7:4 0100-1111 no selection 3:0 channel selection

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE

Test result:

If there is no leakage on the high side, SRx voltage is equal to VREF=2.5V and no current is detected by VRCM itself.

Only if the squib is connected, SFx and SRx pin are at the same voltage, so SRx voltage is readable indirectly through SFx voltage, as done in case of high side leakage test.

SFx voltage is readable addressing the ADC read out on it. The registers involved in this operation are the four DIAGCTRL_x.

$3X DIAGCTRL_X → X=A, B, C, D

Case X = A:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3A DIAGCTRL_A					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $46 = SF0 $47 = SF1 $48 = SF2 $49 = SF3							
(3)	19	18	17	16																			
	1	0	0	ADCREQ_A	-	R	ADCREQ_A $46 = SF0 $47 = SF1 $48 = SF2 $49 = SF3						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE
3. further bit over the 16 standard.

Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio of the ADC. In case of SFx and VRESDIAG, it is 15:1 (see *Table 5*).

If the squib between SFx and SRx pin is not connected, SRx voltage read out is not possible, as it is not mapped into ADC request command.

In case of a leakage (to ground or to battery), VRCM will sink or source a current to maintain SFx at VREF. As a consequence, STG or STB is set:

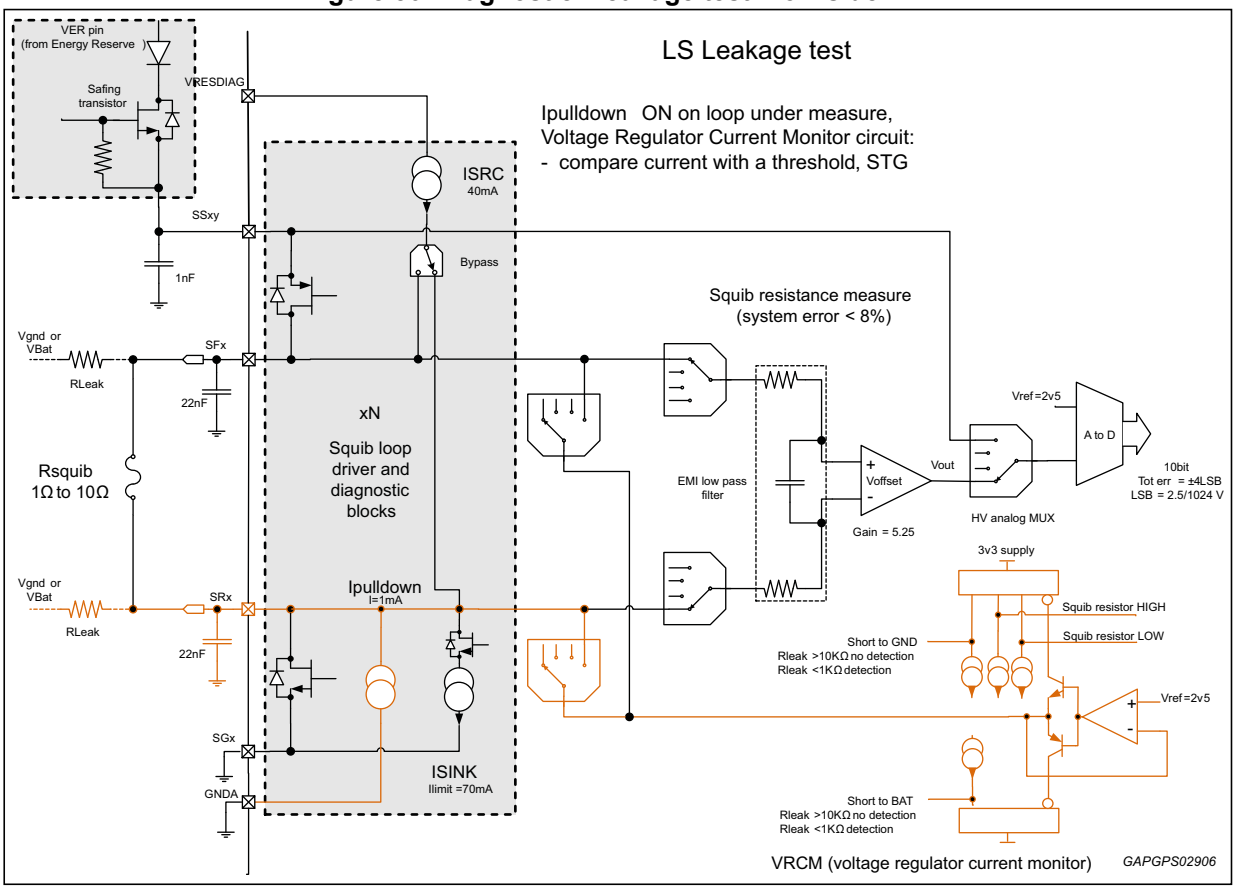
						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R					RES_MEAS_ CHSEL 0100-1111				X	1	1	0	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 0= LOW LEVEL 6: 1=STG if leak vs GND 5: 1= STB if leak vs BATT 4: 0=test on SRx
	(3)	19	18	17	16		R																	
		0	X	0																				

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE
3. Further bit over the 16 standard.

Note: Pull-down current (1mA) is active on all the channels except the one under analysis. So, the case of STG detection, further investigation is necessary to understand if it comes from a real short to ground of the channel or it comes from a short of the channel with another one.

9.1.5 Leakage test - low side IPD

Figure 50. Diagnostic - leakage test - low side IPD



For set up, refer to *Figure 50*.

After having verified that no HS/LS leakage is present, it is possible to verify if IPD is correctly working.

VRCM is connected to SRx, chosen through LEAK_CHSEL. IPD is switched on for that channel.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	0	0	0	0	1	1	RES_MEAS_ CHSEL 0100:1111				LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 0=DIAG LOW LEVEL 13: 0=pulldown curr OFF for VRCM ch; ON the others 12,11: 00/11=ISRC OFF on all channel 10: 0= ISINK all OFF 9,8:11 VRCM to SRx 7:4 0100-1111 no selection 3:0 channel selection

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE

Test result:

If IPD is working, SRx voltage is equal to VOUT_VRCM and VRCM shows STG.

If, in this condition, STG is not set, it means that there is something not correctly working in IPD.

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R					RES_MEAS_CHSEL 0100-1111				X	1	0	0	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 0= LOW LEVEL 6: 1=STG if OK 5: 0 STB 4: 0=test on SRx
	(3)	19	18	17	16		R																	
		0	X	0																				

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE
3. Further bit over the 16 standard.

9.1.6 Short between loops

Supposing the external squib is connected, a short to ground flag of SRx or SFx can be read as:

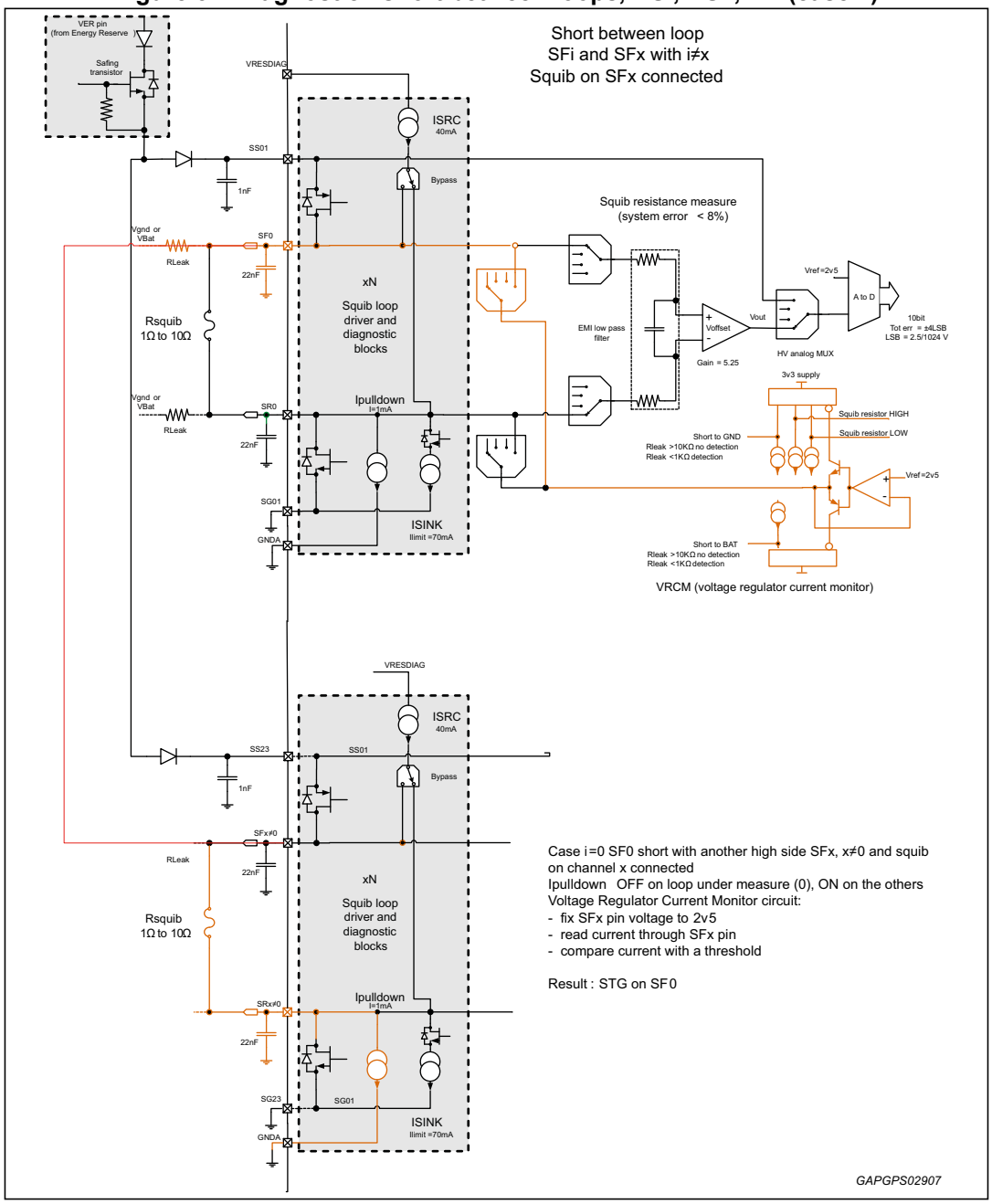
Short of the pin with SR or SF of another channel, both SR and SF

Real short of the pin SRx or SFx to GND

Below, four possible cases are presented.

Case 1

Figure 51. Diagnostic - short between loops, HSi, HSx, i≠x (case 1)

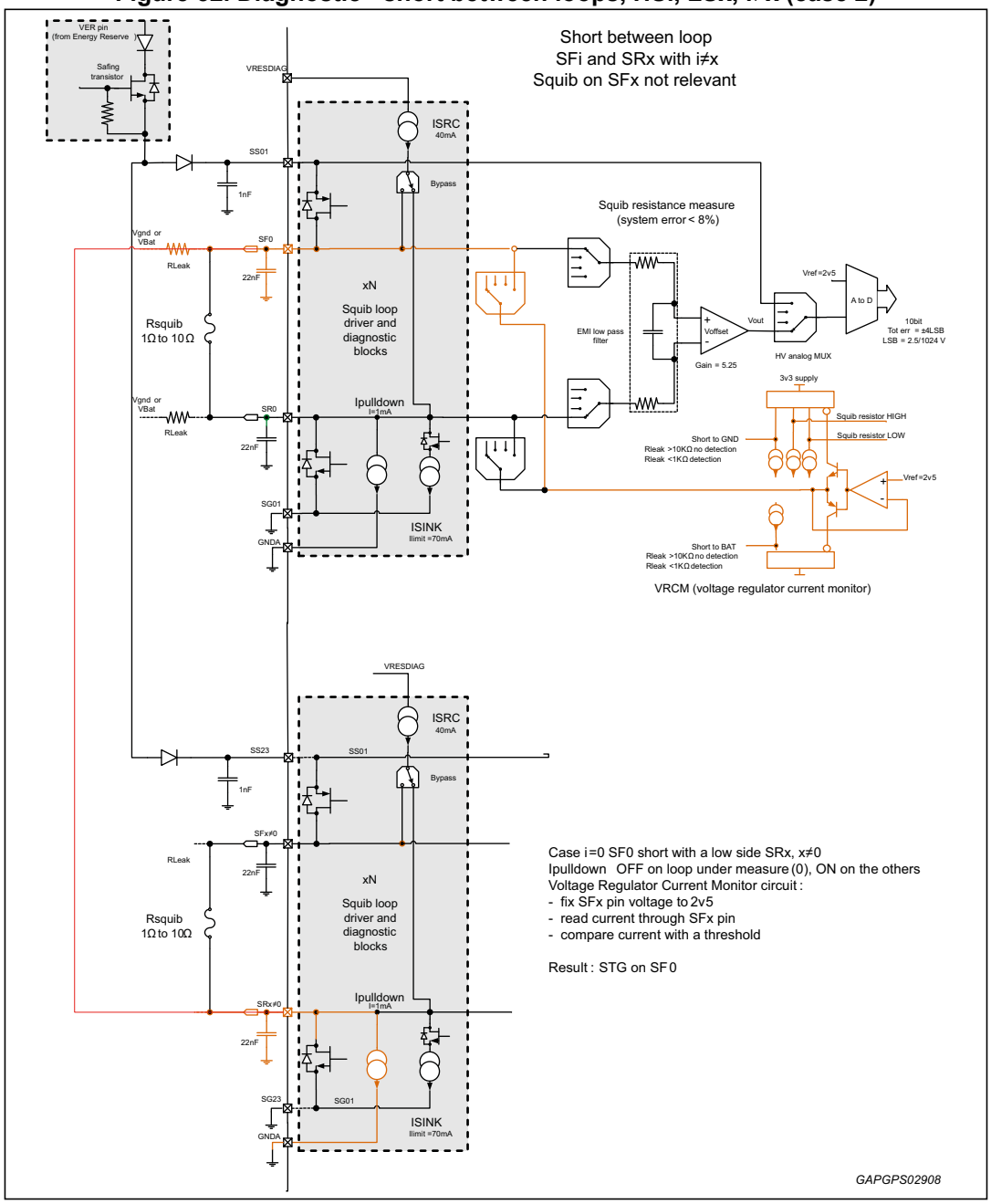


SFi and SFx, i≠x, squib on channel x connected.

If the squib on SFx is not connected, there is no path between SFi (0 in the example) and ground through SFx, squib, SRx and pull down current generator, so the short of a high side SFi with another high side SFx is not detectable.

Case 2

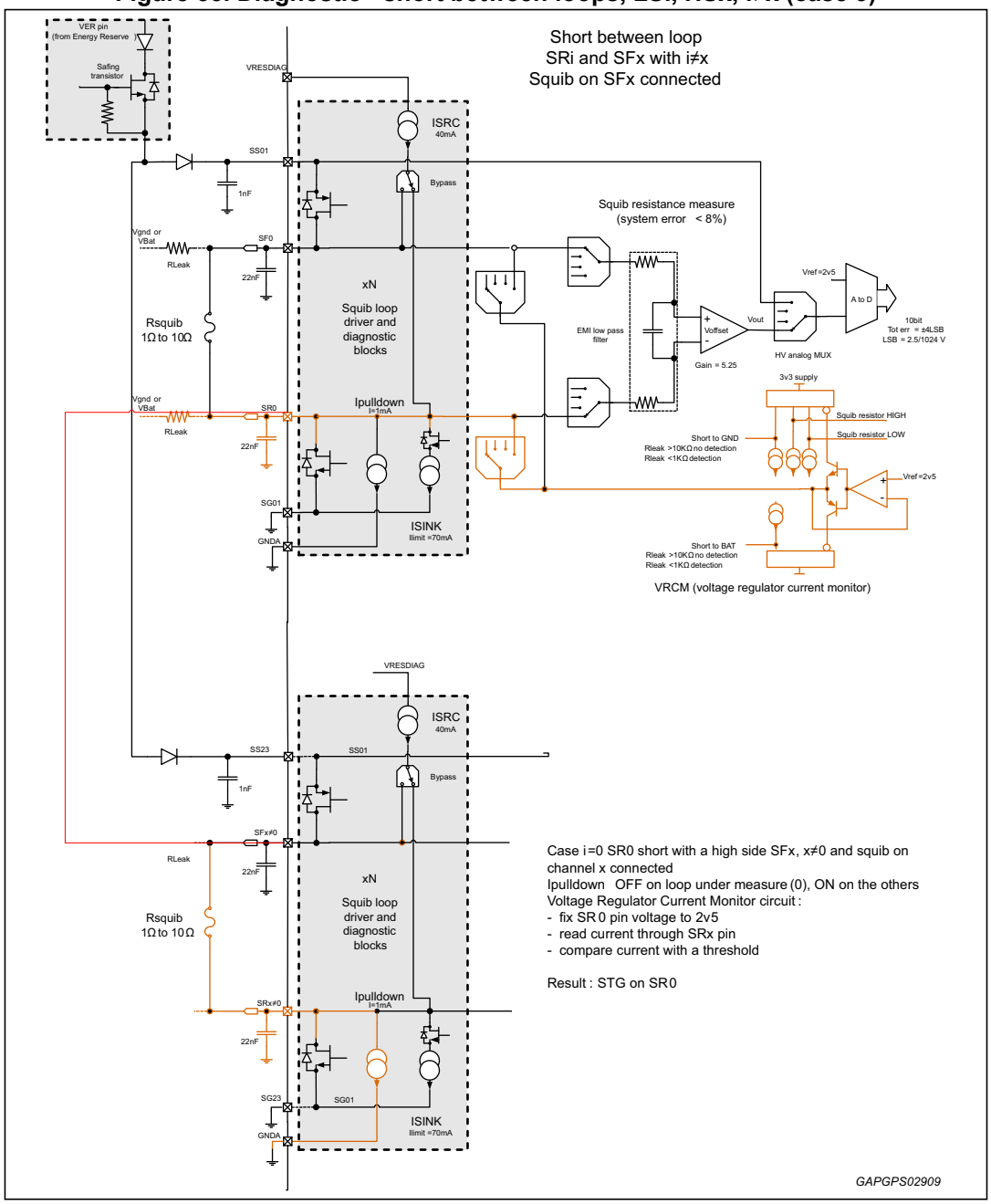
Figure 52. Diagnostic - short between loops, HSi, LSx, i≠x (case 2)



SFi and SRx, i≠x, squib on channel x not necessary connected.

Case 3

Figure 53. Diagnostic - short between loops, LSi, HSx, i≠x (case 3)

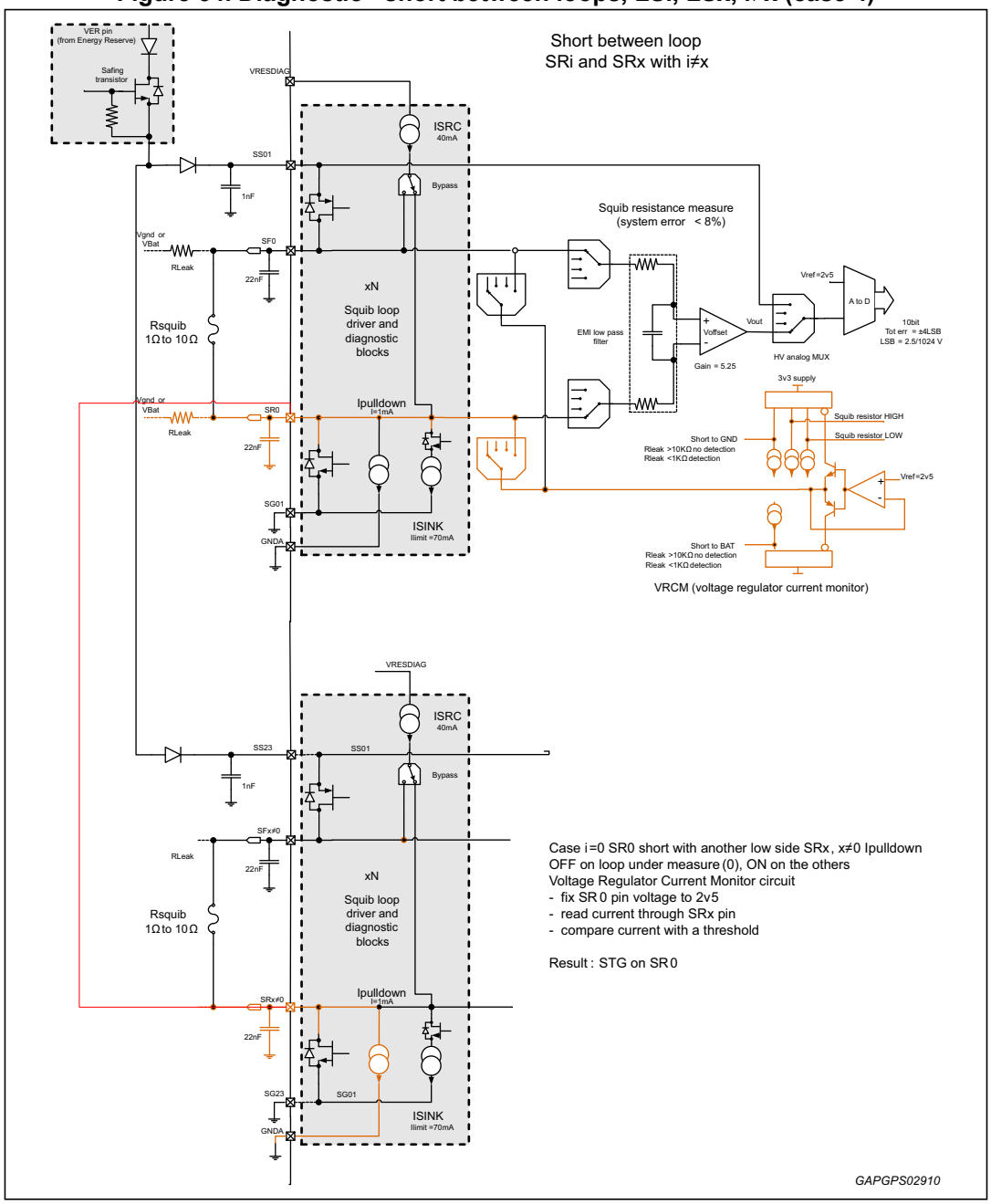


SRi and SFx, i≠x, squib on channel x connected

If the squib on SFx, is not connected, there is no path between SRi (0 in the example) and ground through SFx, squib, SRx and pull down current generator, so the short of a low side SRi with another high side SFx is not detectable.

Case 4

Figure 54. Diagnostic - short between loops, LSi, LSx, i≠x (case 4)

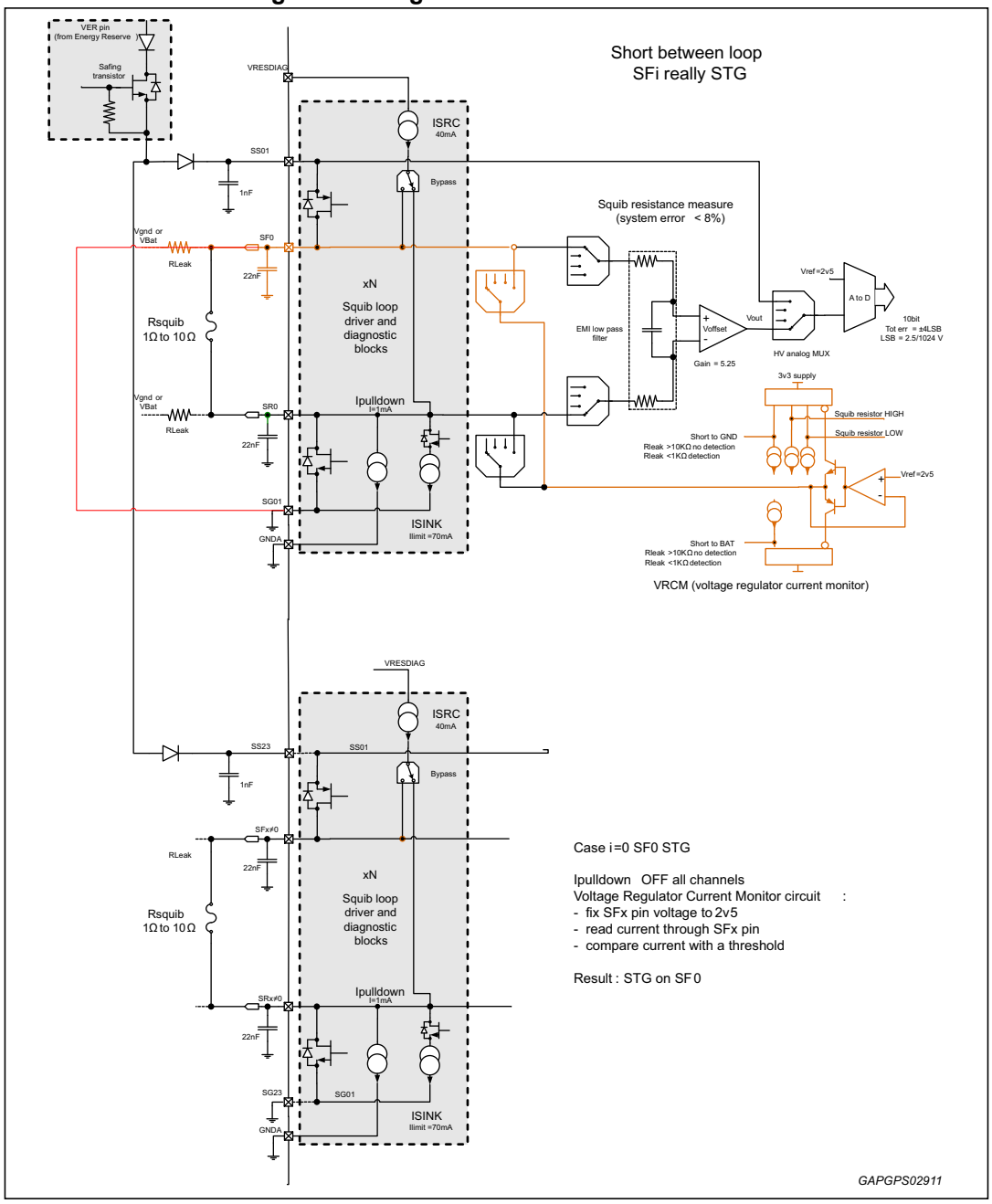


SRi and SRx, i≠x, squib on channel x not necessary.

If a STG has been detected, to identify its origin, it is necessary to understand if it is a real short to ground of the channel or a short between loops.

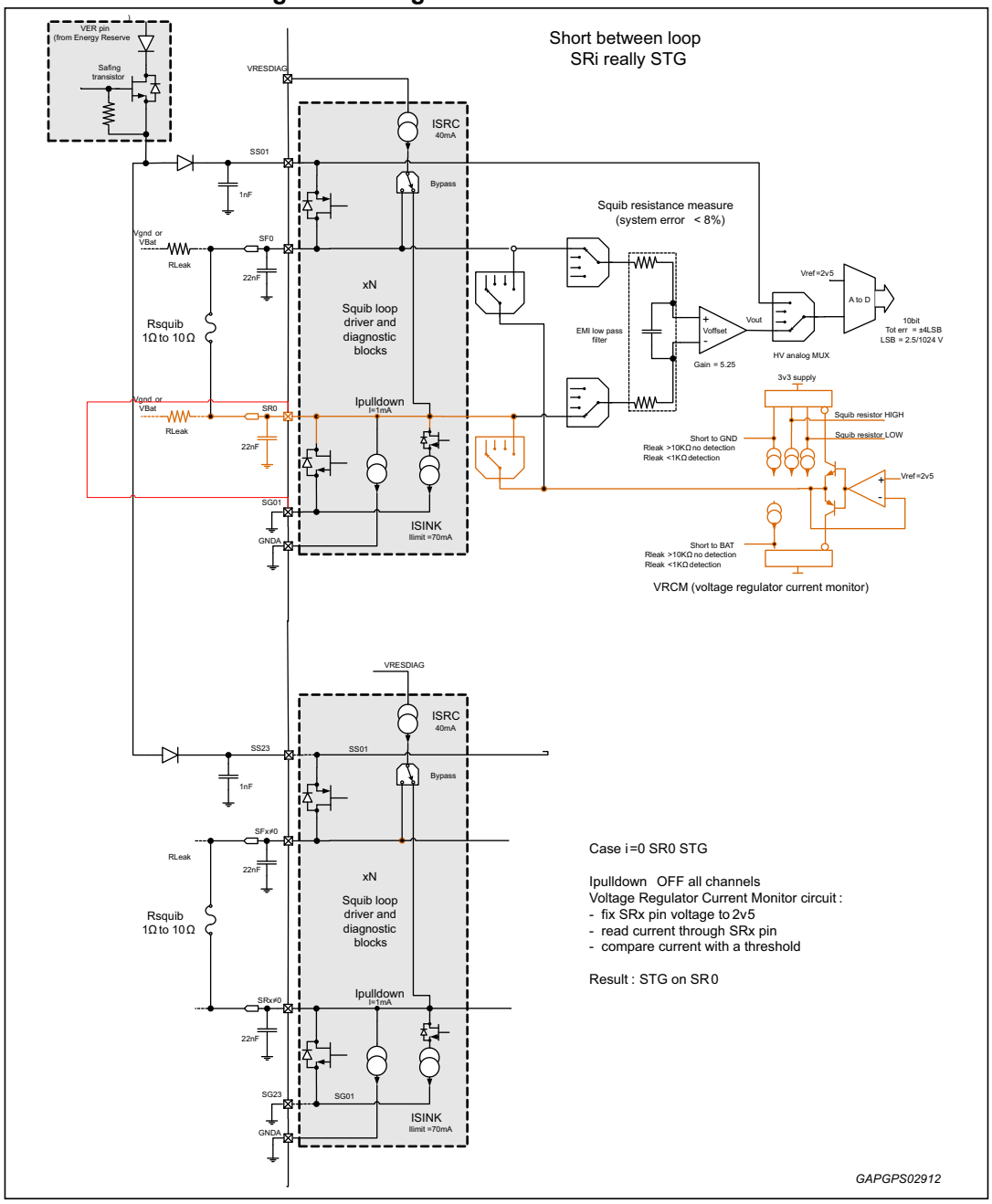
With respect to the case of STG investigation, in this test the pull down current generators are switched off for all channels. If the STG is still present, it means a real STG of the channel under test, otherwise the STG is a short between loops.

Figure 55. Diagnostic - HS short to Ground



Real SFi STG in case of STG flag still set and SR pull-down current switched off for all channels.

Figure 56. Diagnostic - LS short to Ground



Real SRi STG in case of STG flag still set and SR pull-down current switched off for all channels.

In order to understand which pairs loops are involved in the short, each pair has to be checked. Correspondent set up is done by setting $38 LPDIAG properly:

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	0	1	0	0	0	0/1	1/0	RES_MEAS_CHSEL 0100:1111				LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 0=DIAG LOW LEVEL 14: 0=ISRC=40mA 13: 1=pulldown curr OFF for all channels 12,11: 00/11=ISRCOFF on all channel 10: 0= ISINK all OFF 9,8:01= VRCM to SFx 10 VRCM to SRx 7:4 0100-1111 no selection 3:0 channel selection

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

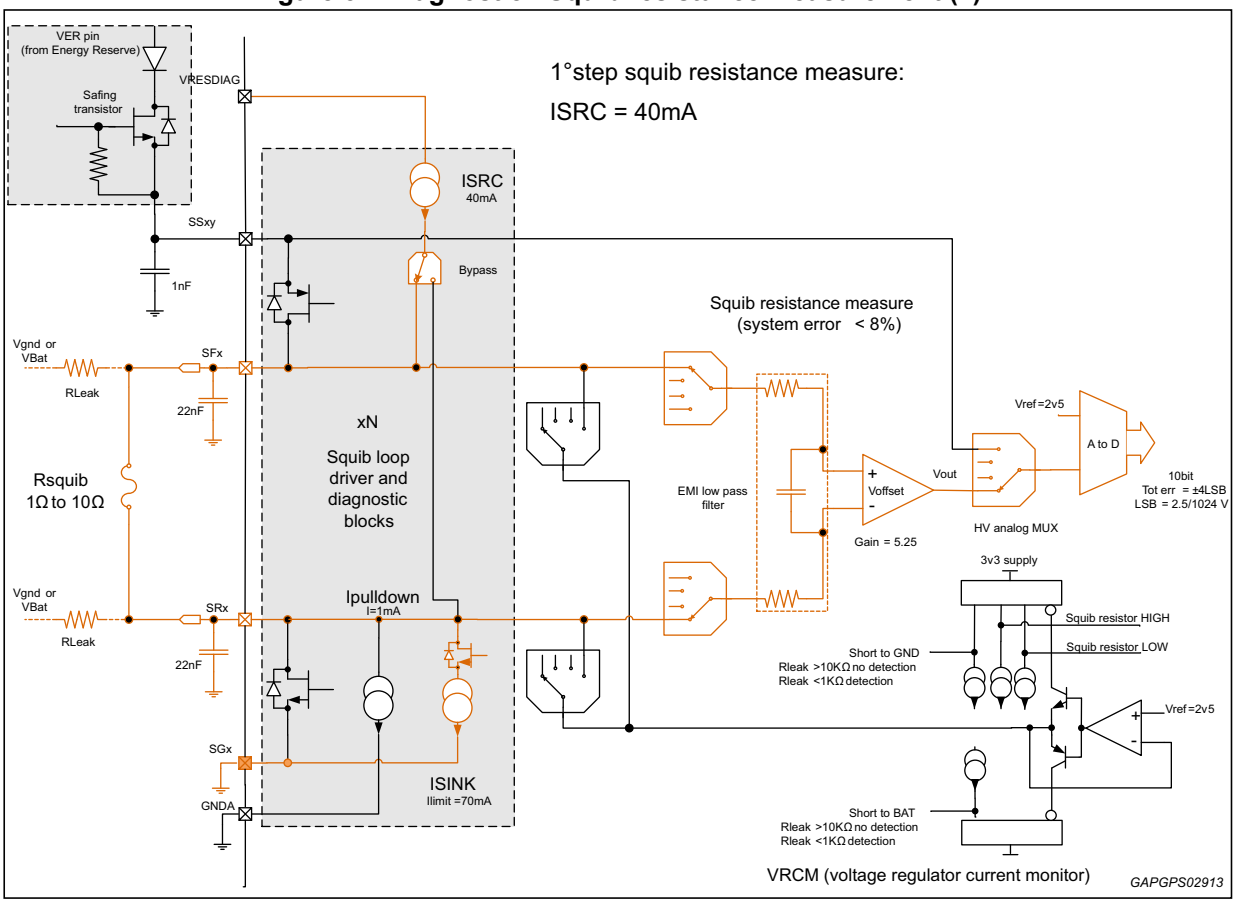
9.1.7 Squib resistance measurement

IC allows measuring the squib resistance value in the range of 1 Ω ÷10 Ω with overall 8% precision.

This is a two-step process:

1st step

Figure 57. Diagnostic - Squib resistance measurement (1)



For set up, refer to *Figure 57*.

Through this set-up, the ISRC is connected to the SFx.

The squib is correctly connected between SFx and SRx.

SRx is internally connected to ISINK that is able to sink the current.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	0	1	1	0	0	RES_MEAS_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				LEAK_CHSEL 0100:1111				15: 0=DIAG LOW LEVEL 13:1= pull down curr OFF all ch 12,11: 01=ISRC=40mA (RES_MEAS_CH), OFF in the others 10: 1= ISINK (RES_MEAS_CH) ON, OFF the others 9,8:00 VRCM not connected 7:4 RES_MEAS_CHSEL 3:0 0100-1111 not selected

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

The first step of the measurement is the read out of the voltage between SFx and SRx that is named resistance into ADC addressing.

This parameter is readable by the microcontroller, via 10bit ADC, through a dedicated request.

The registers to be read are still the four DIAGCTRL_x. Again the explanation of the first register (x=A) is true also for the other three registers:

$3X DIAGCTRL_X → X=A, B, C, D
Case X=A:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3A DIAGCTRL_A					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $06 = squib x resistance							
(3)	19	18	17	16																			
	1	0	0	ADCREQ_A	-	R	ADCREQ_A $06 = squib x resistance						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

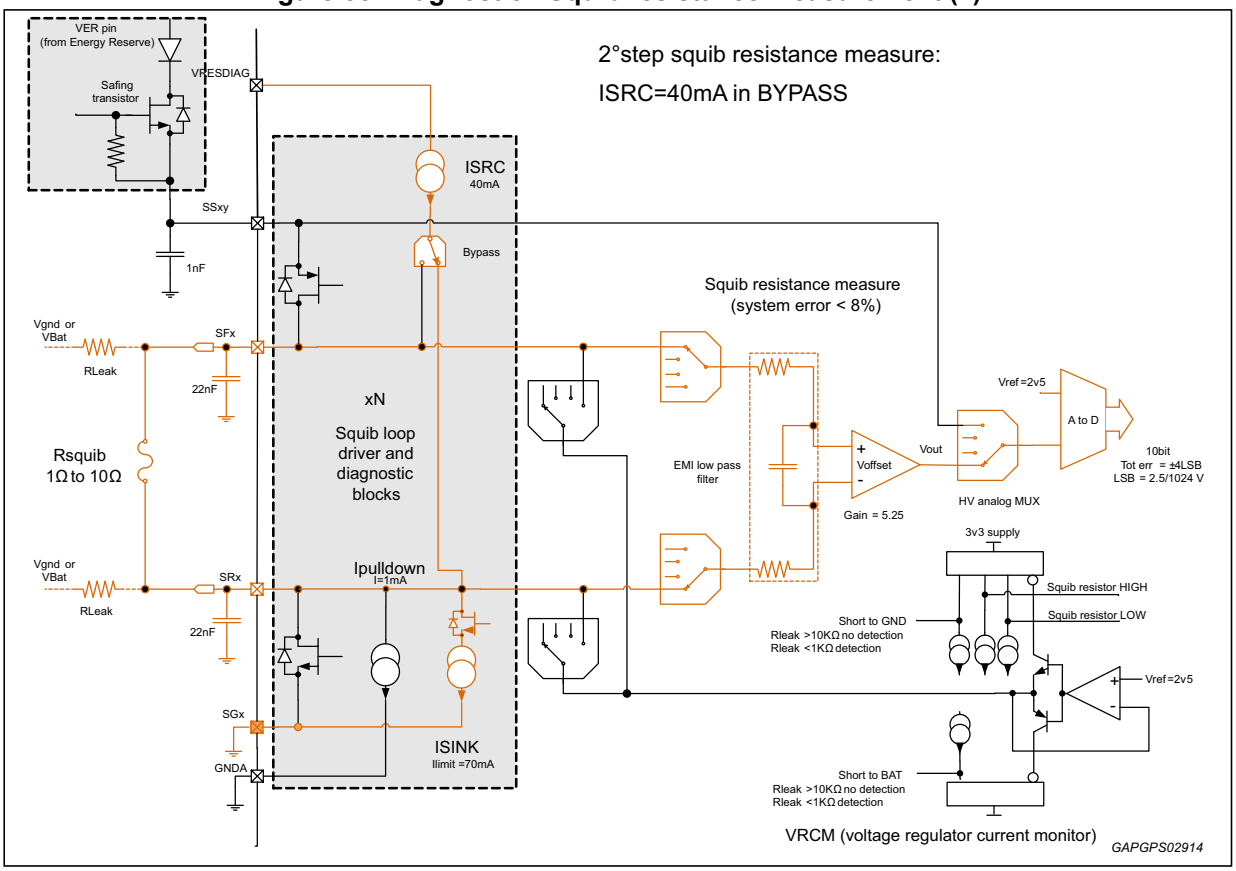
Once read the ADC measurement, to obtain the value it is necessary to consider the divider ratio of the ADC. In case of resistance x, it is 1:1.

Table 6. Squib x resistance measurement of the value ratio ADC

Measurements	Divider Ratio				
	15:1	**10:1**	**7:1**	**4:1**	**1:1**
Squib x resistance					√

2nd step

Figure 58. Diagnostic - Squib resistance measurement (2)



For set up, refer to *Figure 58*.

The set-up is the ISRC connected to the SRx. The squib is correctly connected between SFx and SRx. SRx is internally connected to ISINK that is able to sink the current.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	1	0	1	0	0	RES_MEAS_ CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				LEAK_CHSEL 0100:1111				15: 0=DIAG LOW LEVEL 13:1= pull down curr OFF all ch 12,11: 10=ISRC=40mA by pass (RES_MEAS_CH), OFF in the others 10: 1= ISINK (RES_MEAS_CH) ON, OFF the others 9,8:00 VRCM not connected 7:4 RES_MEAS_CHSEL 3:0 0100-1111 not selected	

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE.

The second step of the measurement is the read out of the voltage between SFx and SRx, named resistance into ADC addressing.

This measurement takes into account the leakage that may be present on SFx SRx pins.

As the previous measurement, also this is readable by the microcontroller, via 10bit ADC, through the same dedicated request.

The registers to be read are still the four DIAGCTRL_x. Again the explanation of the first register (x=A) is true also for the other three registers:

$3X DIAGCTRL_X → X=A, B, C, D

Case X=A:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3A DIAGCTRL_A					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $06 = squib x resistance							
	19	18	17	16																			
(3)	1	0	0	ADCREQ_A	-	R	ADCREQ_A $06 = squib x resistance						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Once read the ADC measurement, to obtain the value it is necessary to consider the divider ratio of the ADC. In case of resistance x, it is 1:1.

Measurements	**Divider Ratio**				
	15:1	**10:1**	**7:1**	**4:1**	**1:1**
Squib x resistance					√

In LPDIAGSTAT is possible to verify on which channel the resistance measurement has been performed:

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R					RES_MEAS_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				X	X	X	X	LEAK_CHSEL			19: 0= LOW LEVEL
	(3)	19	18	17	16		R																
		0																					

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Having the microcontroller these two measurements (that are two voltage drops across SF and SR), the squib resistance is so calculated:

$$\Delta Vout = (SFx - SRx)_1 - (SFx - SRx)_2$$

GAPGPS02915

$$Rsquib = \frac{\Delta Vout}{G * ISRC}$$

GAPGPS02916

G = 5.25 ±2% differential amplifier gain

ISRC = 40 mA ±5%

Let's consider an example where

$ADC_{1ST\ CONVERSION} = (0100111000)_2 = (312)_{10}$

$ADC_{2ND\ CONVERSION} = (0010000001)_2 = (129)_{10}$

$\Delta_{ADC} = 312\text{-}129 = 183$

In order to obtain the result in Volt, being the ADC characteristic linear,

$$2.5 : 1024 = x : \Delta_{ADC} \rightarrow x = \frac{183 \cdot 2.5V}{1024} = 0.44V$$

GAPGPS02917

In order to obtain resistance value, considering typical factors

$$Rsquib = \frac{x}{G * ISRC} = \frac{0.44V}{5.25 \cdot 40\,mA} = 2.12\Omega$$

GAPGPS02918

Note: Immediately after the ADC read-out, ISRC is automatically switched OFF to reduce the power consumption.

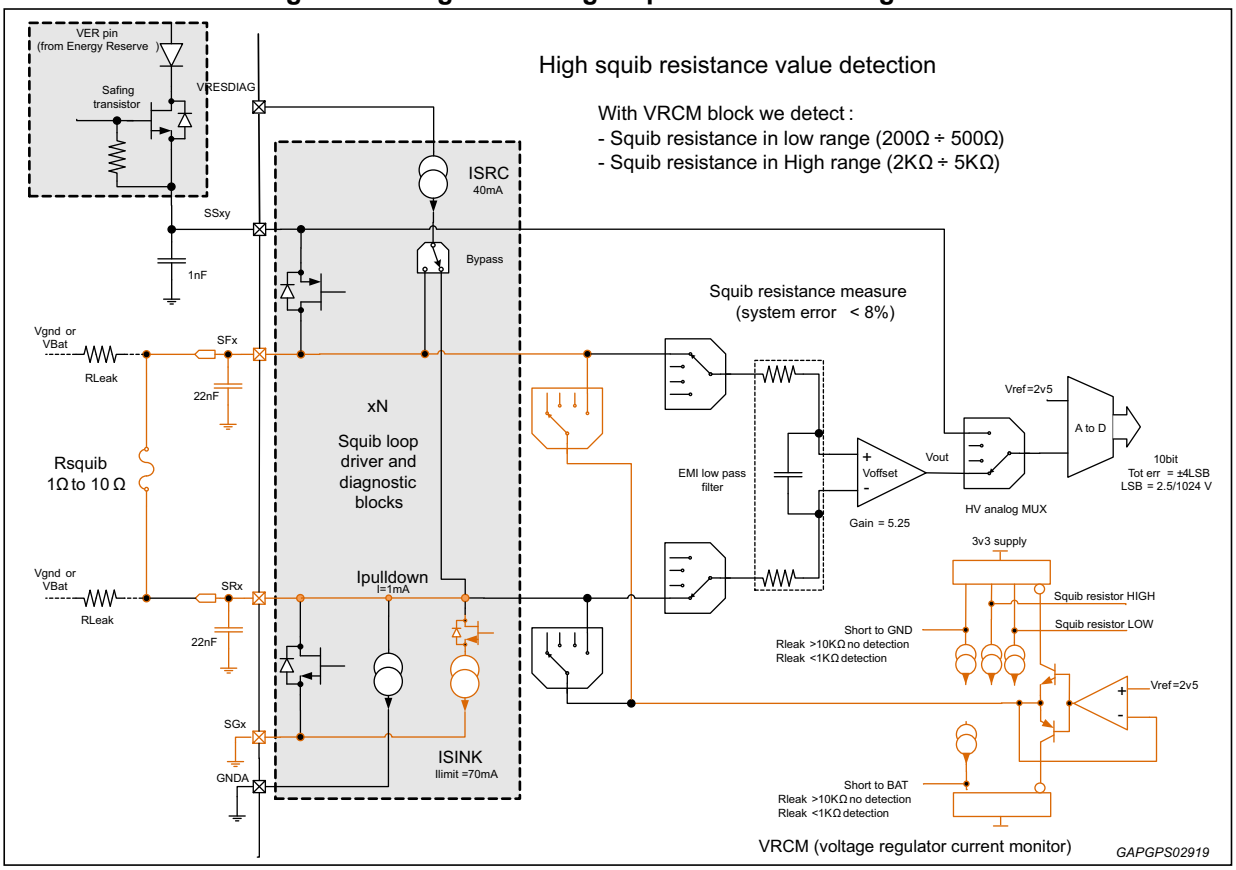
9.1.8 High squib resistance diagnostic

The aim of the test is to understand if the squib resistor is below 200 Ω, between 500 Ω and 2000 Ω or beyond 5000 Ω

In case of a very high squib resistance, there is the possibility to set a lower ISRC current, through ISRC_CURR_SEL bit, bit 14, $LPDIAGREQ. In this way, ADC maintains a good dynamic.

The following description, referred to ISRC = 40 mA, is true also in case of ISRC = 8 mA.

Figure 59. Diagnostic - High squib resistance diagnostic



For set up, refer to *Figure 59*.

The set-up is the ISINK connected to the SRx. The squib is correctly connected between SFx and SRx. SRx is internally connected to ISINK that is able to sink the current.

		(1)	(2)	15	14	13	12	11	10	9	8	7 6 5 4	3 2 1 0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	0/1	0/1	1	0	1	RES_MEAS_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3	15: 0=DIAG LOW LEVEL 13:1= pull down curr OFF all ch 12,11: 00 or 11 ISRC OFF all channels 10: 1= ISINK (RES_MEAS_CH) ON, OFF the others 9,8:01 VRCM SFx LEAK_CHSEL) 7:4 RES_MEAS_CHSEL 3:0 LEAK_CHSEL

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Note: ISINK and VRCM have to be addressed to the same channel, that means RES_MEAS_CHSEL, bit[7:4] and LEAK_CHSEL, bit[3:0]are equal.

If there is a wrong selection in the two fields RES_MEAS_CHSEL / LEAK_CHSEL, there is no notice of the mistake.

Test result:

Through this set-up, the VRCM is connected to SFx and ISINK to SRx. Current flowing through SFx is measured and compared with ISRlow, ISRhigh thresholds to identify in which range the resistor measured is.

HSR HIGH = R_{Sqhigh} = 2 kΩ ÷ 5 kΩ

HSR LOW = R_{Sqlow} = 200 Ω ÷ 500 Ω

In case of low resistance value, VRCM sees a path from SRx and GND so STG (very low impedance towards ground) could be detected.

Read out of these bit has to be done before the next diagnostic request, because these bits are not latched.

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	$37 LPDIAGSTAT					R			0/1	0/1	RES_MEAS_ CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				X	0/1	X	0/1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 0= LOW LEVEL 13: 0 = resis < HSR HIGH 1 = resis > HSR HIGH 12: 0 = resis > HSR LOW 1 = resis < HSR LOW 6: STG 1=yes/0=no 4:VRCM connected to: 0= SR 1=SF
	(3)	19	18	17	16		R																
		0																					

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

9.1.9 High side FET diagnostic

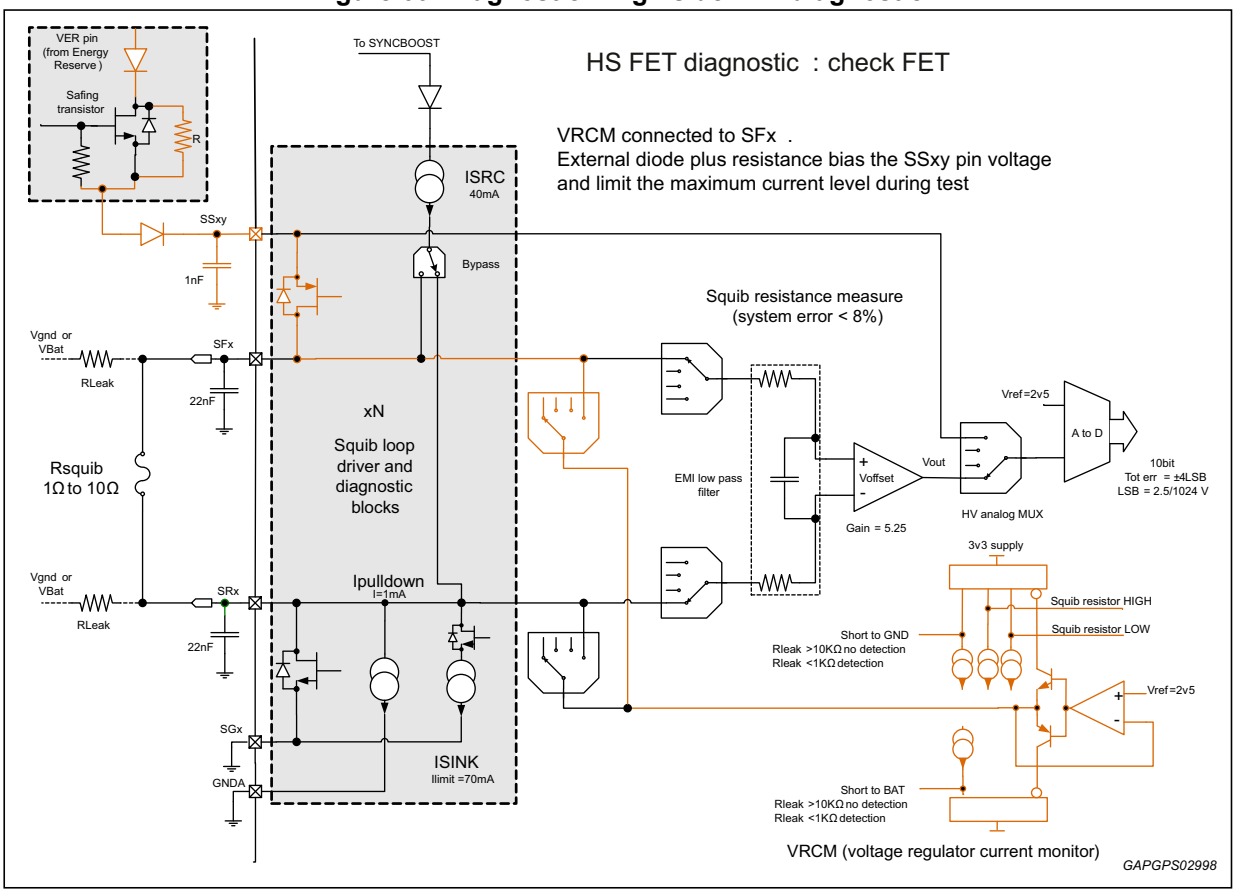
The test is possible only in the diagnostic phase (see *Section 4.2*).

These are two tests performed separately for the high side and the low side, with dedicated commands.

Note: Before running this test, VRCM has to be previously validated (see Section 9.1.2: VRCM test validation) and leakage tests have to be already performed with no fails found (see Section 9.1.3: Leakage test - High side and Section 9.1.4: Leakage test - low side).

Only if these mentioned tests have been successfully done, the HIGH SIDE (and LOW SIDE) FET tests can be performed.

Figure 60. Diagnostic - High side FET diagnostic



For set up, refer to *Figure 60*.

ISRC and ISINK are kept off and VRCM is connected to SFx, chosen through LEAK_CHSEL.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	0	0	0	0	1	RES_MEAS_ CHSEL 0100:1111				LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 0=DIAG LOW LEVEL 13: 1=pulldown curr OFF for all ch. 12,11: 00/11=ISRCOFF on all channel 10: 0= ISINK all OFF 9,8: 01 VRCM to SFx (LEAK_CHSEL) 7:4 0100-1111 no selection 3:0 LEAK_CHSEL
3	**$36 SYSDIAGREQ**	D	W	X	X	X	X	X	X	X	X					0	1	1	1	0111: DSTEST=HSFET active

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Test result:

High side FET test turns ON the HS power: if it turns ON correctly, SFx is connected to SSxy which is at VER voltage through the resistor R in parallel to the safing FET.

During the test, the device monitors the current flowing through VRCM.

If the high side FET works properly, this current exceeds the thresholds IHSFET (high side) and the channel is immediately turned off,

$$I_{HSFET} = 1.8mA \pm 10\%$$

In case the current doesn't exceed the limit mentioned, after a fixed time, $T_{FETTIMEOUT}$, the test is terminated and the output is turned off.

$$T_{FETTIMEOUT} = 200\ \mu s$$

During $T_{FETTIMEOUT}$ period, FET activation is flagged through a bit, FETON, readable via SPI:

In any condition, current in SFx doesn't exceed I_{SVRCM} and during the FET test the energy provided to the squib is limited at $E_{FETtest}$.

$$I_{SVRCM}: I_{LIM\_SRC} = -20mA \div -10mA;\ I_{LIM\_SNK} = 10mA \div 20mA$$

$$E_{FETtes}t < 170\mu J$$

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	$37 LPDIAGSTAT						R	0/1				RES_MEAS_CHSEL 0100-1111				0	0/1	0/1	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 0= LOW LEVEL 15: 0=FET OFF during diag 1= 0=FET ON during diag 6: 0=STG 5: 1=STB 4: 1=test on SFx
	(3)	19	18	17	16		R																	
		0	X	0																				

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Possible results for high side:

STB=1 & STG=0 **ok**

STB=0 or STG=1 missing SSxy connection during FET test or high side not switched ON or short to GND during FET test

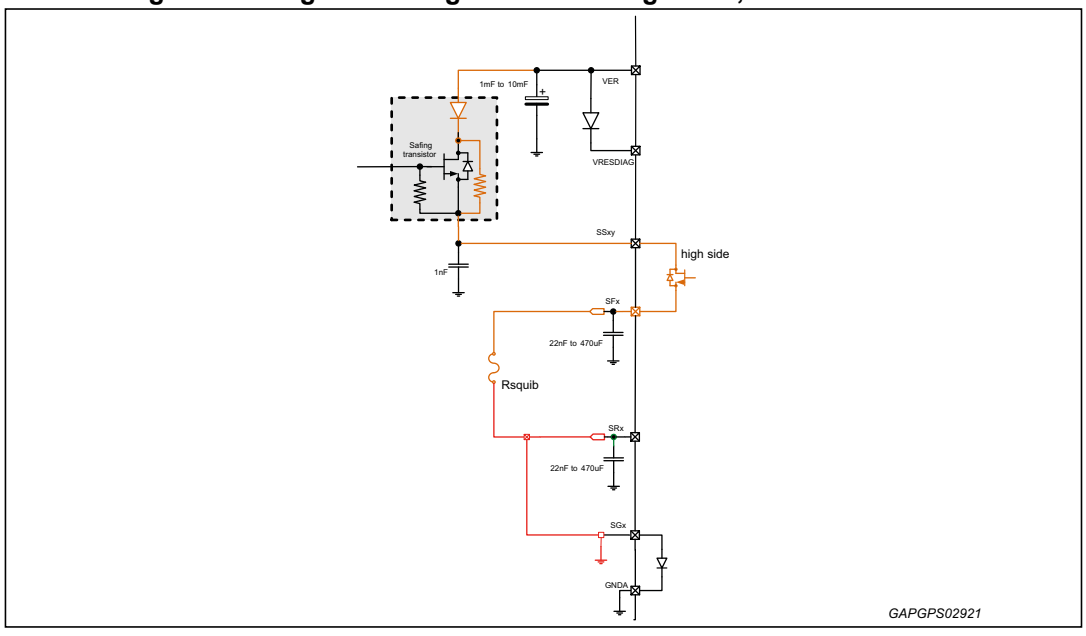
STG & STB, after FET test, are latched.

They are cleared through a new LPDIAGREQ or a new SYSDIAGREQ.

Note: If VRCM is not previously connected to the SFx and the test is run, a dangerous condition could happen.

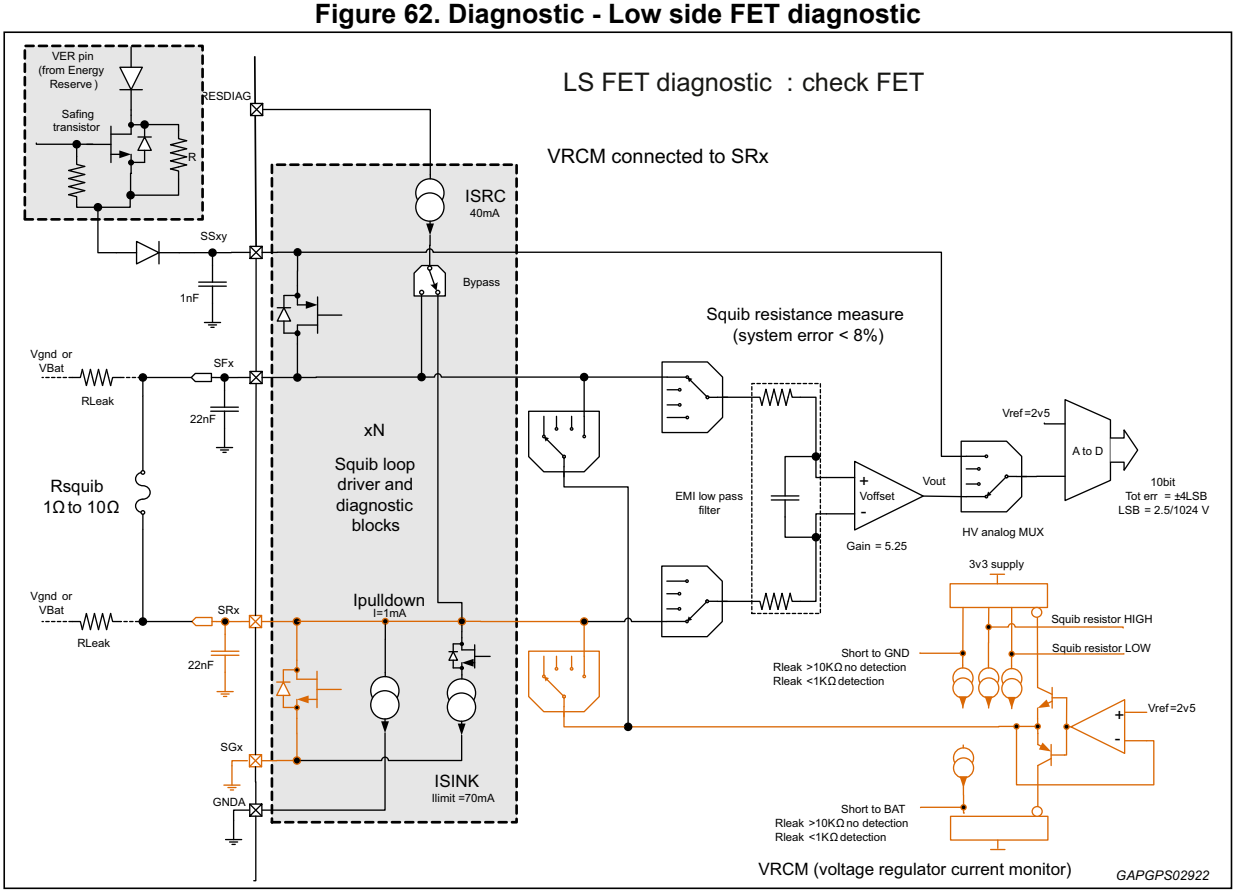
In case of SRx shorted to GND, when the HS is turned ON, even if the current flowing through the squib is greater than I_{HSFET}, the HS is not immediately turned off and the current flows through the squib until $T_{FETTIMEOUT}$ expires: this could determine an undesired deployment.

Figure 61. Diagnostic - High side FET diagnostic, SR short to GND



9.1.10 Low side FET diagnostic

Figure 62. Diagnostic - Low side FET diagnostic



For set up, refer to *Figure 62*.

ISRC and ISINK are kept off and VRCM is connected to SRx, chosen through LEAK_CHSEL.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	0	X	1	0	0	0	1	0	RES_MEAS_CHSEL 0100:1111				LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 0=DIAG LOW LEVEL 13: 1=pulldown curr OFF for all ch. 12,11: 00/11=ISRCOFF on all channel 10: 0= ISINK all OFF 9,8: 10 VRCM to SRx (LEAK_CHSEL) 7:4 0100-1111 no selection 3:0 LEAK_CHSEL
3	**$36 SYSDIAGREQ**	D	W	X	X	X	X	X	X	X	X					1	0	0	0	1000: DSTEST=LSFET active

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Test result:

Low side FET test turns ON the low side. If the low side turns ON correctly, SRx is connected to SGxy.

During the test, the device monitors the current flowing through VRCM.

If the FET works properly, this current exceeds the thresholds I_{LSFET} and the channel is immediately turned off,

$$I_{LSFET} = 450\mu A \pm 10\%$$

In case the current doesn't exceed the limit mentioned, after a fixed time, $T_{FETTIMEOUT}$, the test is terminated and the output is turned off.

$$T_{FETTIMEOUT} = 200\ \mu s$$

During $T_{FETTIMEOUT}$ period, FET activation is flagged through a bit, FETON, readable via SPI.

In any condition, current in SRx doesn't exceed I_{SVRCM} and during the FET test the energy provided to the squib is limited at $E_{FETtest}$.

$$I_{SVRCM}: I_{LIM\_SRC} = -20mA \div -10mA;\ I_{LIM\_SNK} = 10mA \div 20mA$$

$$E_{FETtest} < 170uJ$$

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**					R	0/1				RES_MEAS_CHSEL 0100-1111				0	0/1	0/1	0	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 0= LOW LEVEL 15: 0=FET OFF during diag 1= 0=FET ON during diag 6: 1=STG 5: 0=STB 4: 0=test on SRx
	(3)	19	18	17	16		R																
		0	X	0																			

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Possible results for low side:

STB=0 and STG=1 **ok**

STB=1 or STG=0 short to battery in LS or low side not switched ON.

Note: Ground loss (SGxy) is not detected through FET test, because there is a diode between SGxy and the substrate.

STG & STB, after FET test, are latched.

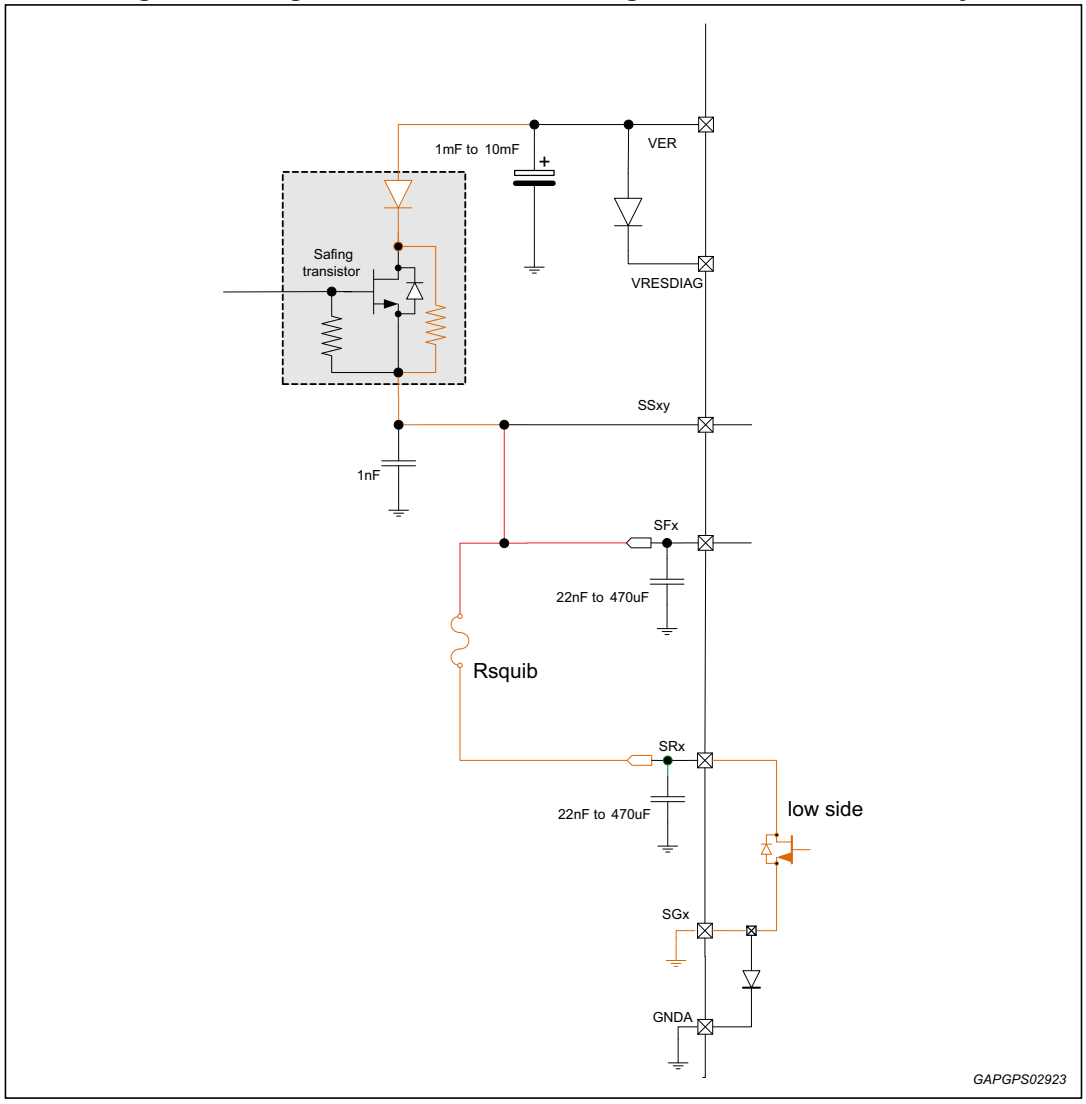
They are cleared through a new LPDIAGREQ or a new SYSDIAGREQ.

Note: If VRCM is not previously connected to the SRx and the test is run, a dangerous condition could happen.

In case of SFx shorted to SSxy, when the LS is turned ON, even if the current flowing through the squib is greater than I_{LSFET}, the LS is not immediately turned off and the current

flows through the squib until $T_{FETTIMEOUT}$ expires: this could determine an undesired deployment.

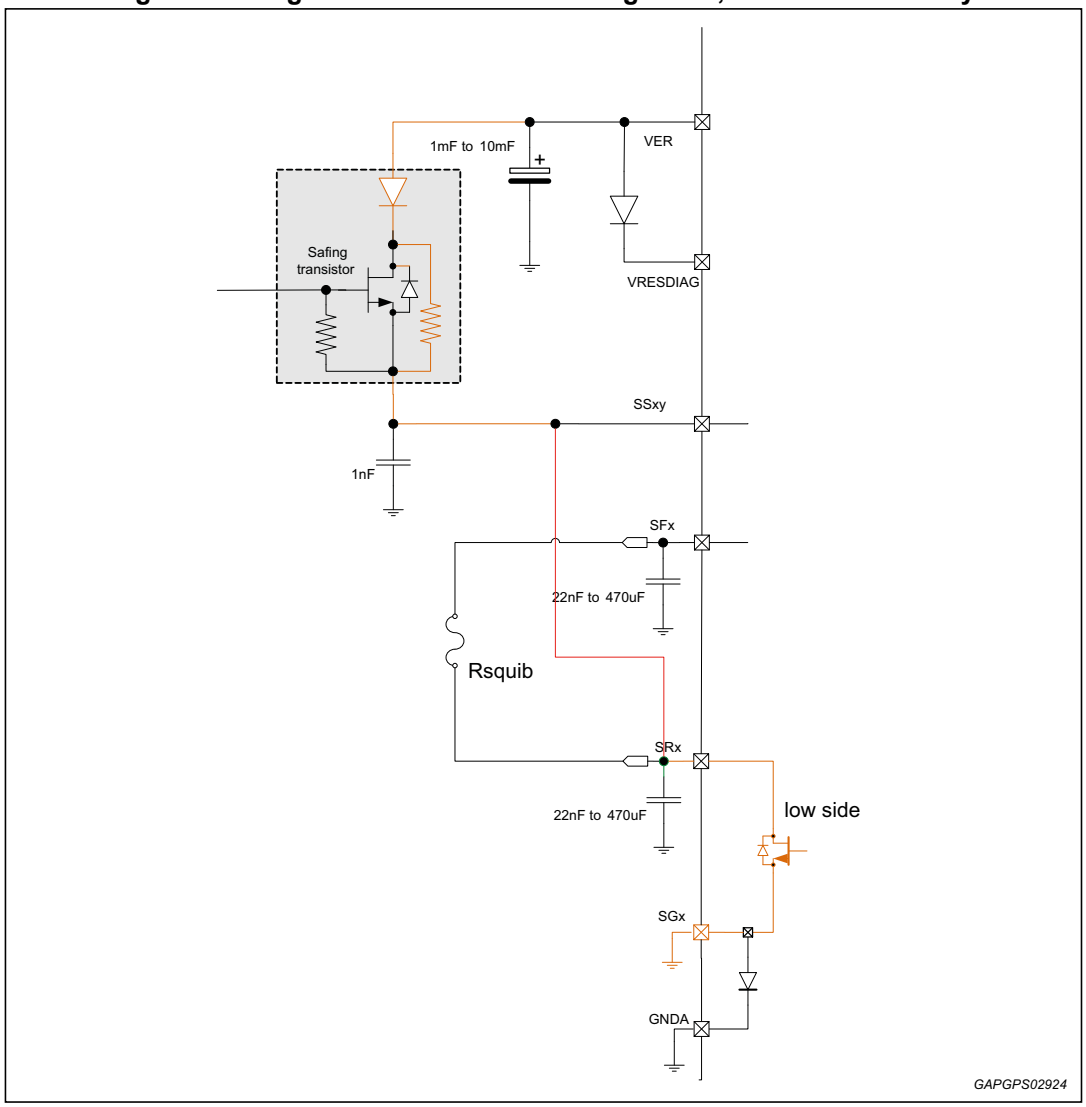
Figure 63. Diagnostic - Low side FET diagnostic, SF short to Battery



Note: If VRCM is not previously connected to the SRx and the test is run, a dangerous condition could happen.

In case of SRx shorted to SSxy, when the LS is turned ON, even if the current flowing through the squib is greater than I_{LSFET}, the LS is not immediately turned off and the current flows through the squib until $T_{FETTIMEOUT}$ expires: such a high current could damage the LS power.

Figure 64. Diagnostic - Low side FET diagnostic, SR short to Battery



9.1.11 LOSS of Ground

This test is based on the voltage of ground pin, SGxy, during the squib resistor measurement or the high side driver diagnostic, refers to those sketches.

Any voltage shift of SGxy pin over V_{SGopen} is considered loss of ground, readable in LP_GNDLOSS register.

$$V_{SGopen} = 400\text{-}600\text{-}800 \text{ mV}$$

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$26 LP_GNDLOSS**						R	0	0	0	0	0	0	0	0	0	0	0	0	CH3	CH2	CH1	CH0	0 = no loss of ground 1 = loss of ground i: chi i=3:0
	(3)	19	18	17	16		R																	
		0	X	0																				

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

GNDLOSSx is set considering t_{SGopen} filter time and it is cleared upon read

$$t_{SGopen} = 46\ \mu s - 50\ \mu s - 54\ \mu s$$

Note: Only two GND pins are available, SG01, SG23; IC is able to detect GND loss on CHx or CHy basing on the channel selected either for Section 9.1.7: Squib resistance measurement or Section 9.1.11: LOSS of Ground.

9.1.12 Safing FET diagnostic

The aim of the test is to verify the VSF and SSxy voltage level.

VSF is turned ON via SPI.

Set up

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$36 SYSDIAGREQ**	D	W	X	X	X	X	X	X	X	X					0	1	1	0	0110: DSTEST= VSF regulator active

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Test result:

VSF and SSxy voltage are readable by the microcontroller through the ADC converter in the registers:

$3X DIAGCTRL_X → X=A, B, C, D

Case X = A:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3A DIAGCTRL_A					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $36 = SS0 $37 = SS1 $38 = SS2 $399 = SS3 $2A = VSF							
(3)	19	18	17	16																			
	1	0	0	ADCREQ_A	-	R	ADCREQ_A $36 = SS0 $37 = SS1 $38 = SS2 $399 = SS3 $2A = VSF						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

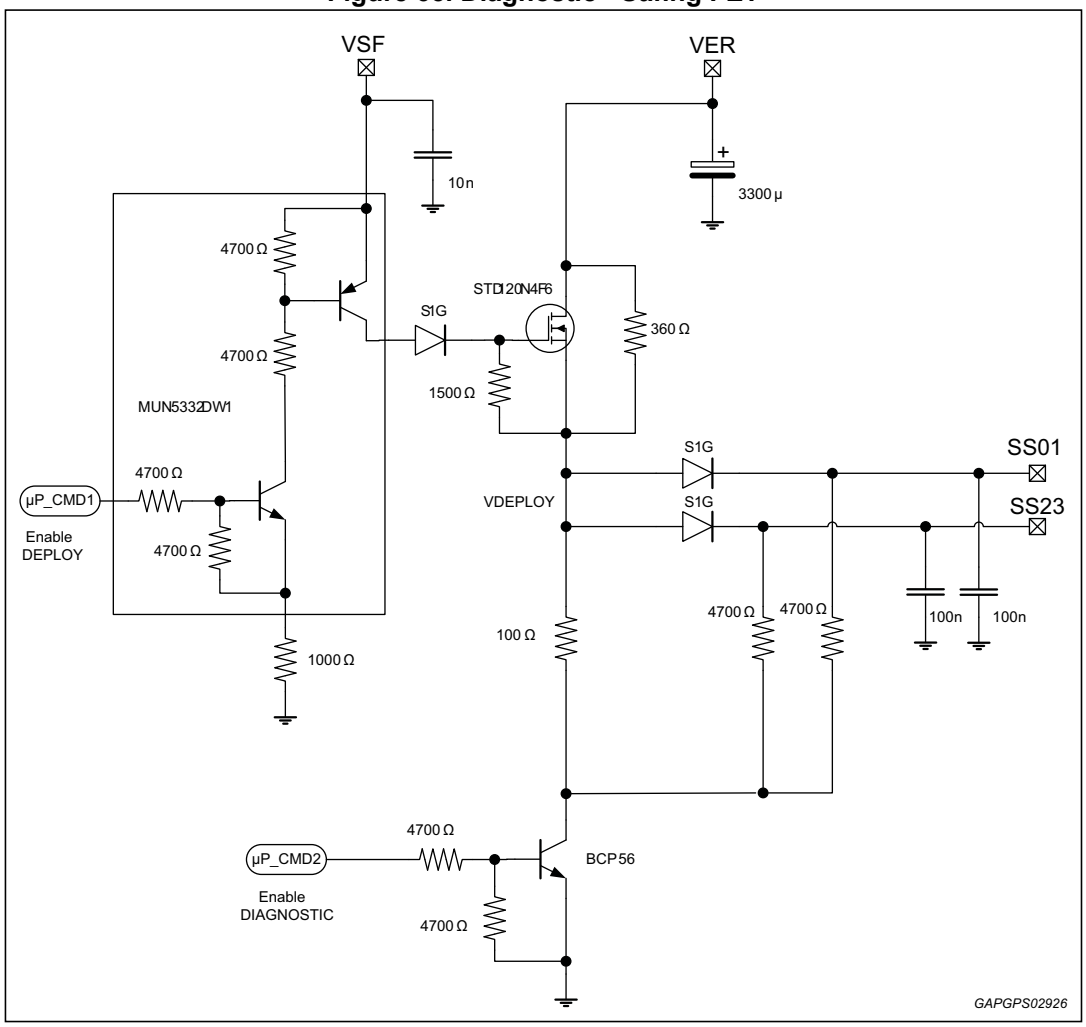
Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio of the ADC. In case of SS and VSF, it is 15:1.

Table 7. VSF and SS measurement of the value ratio ADC

Measurements	Divider Ratio				
	15:1	10:1	7:1	4:1	1:1
VSF	√				
SSx	√				

In the schematics here below a possible solution to perform the test is represented: such a solution allows performing SAFING FET test only if the external reserve capacitor CER has been charged.

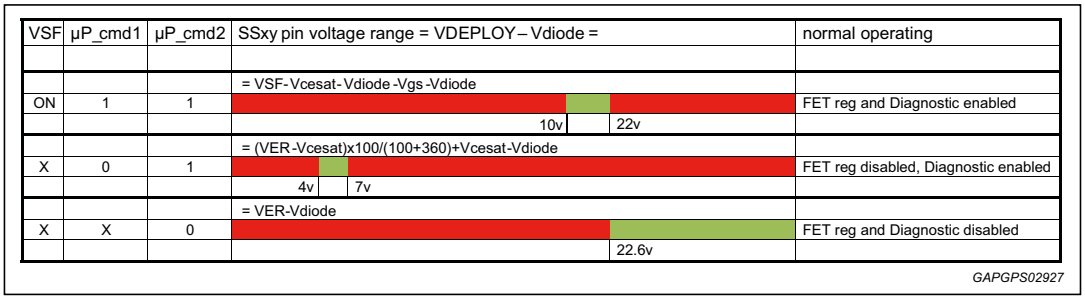
Figure 65. Diagnostic - Safing FET



The solution is based on the reading of the voltage on SSxy pins through ADC; it also requires an external component network and two commands from the microcontroller,

µP_cmd1 and µP_com2. Depending on the status of the VSF (ON or OFF) and on the commands from the microcontroller, the cases described below can occur.

Figure 66. Cases of status of the VSF (on or off) and on the commands from the microcontroller



In the first case of the table reported above, external FET is working in voltage regulator mode (VSF ON, µp_cmd1, µp_cmd2 set) and voltage on SSxy pin is:

$$V_{SS_{xy}} = VSF - V_{CE_{SAT}} - V_{DIODE} - V_{GS} - V_{DIODE}$$

GAPGPS02928

The expected value read on ADC, depending on all the parameter variation, is in the range of 10V÷22V.

In the second case the low side command of the diagnostic is enabled and voltage on SSxy pin is:

$$V_{SS_{xy}} = 100 \cdot \frac{VER - V_{CE_{SAT}}}{100 + 360} + V_{CE_{SAT}} - V_{DIODE}$$

GAPGPS02929

The expected value read on ADC, depending on all the parameter variation, is in the range of 4V÷7V.

In the last case everything is disabled so the voltage on SSxy is expected to be close to VER.:

$$V_{SS_{xy}} \approx VER - V_{DIODE}$$

GAPGPS02930

In case of an ADC reading out of the expected range, it has to be considered as a faulty condition.

Note: Once µp_cmd2 is active, capacitors on SSxy pins are discharged through 4.7k which requires about 1ms to reach steady state so a proper time should be elapsed before running ADC conversion.

Besides, in order to guarantee more safety, it is possible to read the voltage on V_{DEPLOY} net through a voltage divider which is sensed by ADC of the microcontroller.

In order to guarantee redundancy on safing FET enabling, two independent conditions must be verified. The assertion of the two conditions must come from two separate activation logics.

In the solution here presented, the first condition (VSF switch ON) comes from the IC in arming state while the second one (up_cmd1 asserted) comes from the microcontroller.

In case the ARMING algorithm is run by the microcontroller, the circuit which turns ON the safing FET can be removed (both MUN5332DW1 and S1G diode): VSF can be connected directly to the FET gate and *µp_cmd1* can be used to drive FENH and FENL.

9.1.13 Deployment time diagnostic

The aim of the test is to pass to the microcontroller the deploy time information that the IC has stored with the previous SPI commands (see *Section 8.1: Deployment requirement*).

Set up, only in DIAG state

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$36 SYSDIAGREQ**	D	W	X	X	X	X	X	X	X	X					1	0	0	1	1001: DSTEST= output timing on ARM pin

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Note: Once $36 SYSDIAGREQ register is set for output timing on ARM pin check, even if the test has been performed, it is not possible any modification in the deployment channel configuration, $06 DCR0, $07 DCR1, $08 DCR2, $09 DCR3.

This feature prevent any modification in the deployment time and deployment current after the test has been performed and, as a consequence, is no longer visible by the microcontroller.

To modify again the deployment channel configuration ($06 DCR0, $07 DCR1, $08 DCR2, $09 DCR3) it is first necessary to change the DSTEST request, and secondly to modify the deployment channel configuration itself as previously done.

Test result:

Once the test is ongoing, a signal 0V ->5/3.3V (depending on VDDQ) is output on ARM pin, which reports in sequence, from channel 0 to channel 3, the deployment time programmed, with a 8ms delay between each channel: starting from ch0, ARM signal is high for the deploy time of ch0; then remains low until the next pulse corresponding to the channel 1 occurs (8 ms delay between each pulse to start); the same happens with the next channels 2, 3.

The microcontroller can test the latest deployment time programmed in DRCx (see *Section 8.1: Deployment requirement*) measuring the duration of high ARM pulse.

If the test is performed on a channel with no deployment time previously configured, the high ARM pulse lasts 8 µs.

If the combination time/current deployment programmed for a channel is wrong, then, as explained in the DRCx (see *Section 8.1: Deployment requirement*) the combination time/current deployment turns back to the default value. In case the deployment time is then monitored through ARM signal, the default one is output.

Figure 67. Deployment timer diagnostic sequence

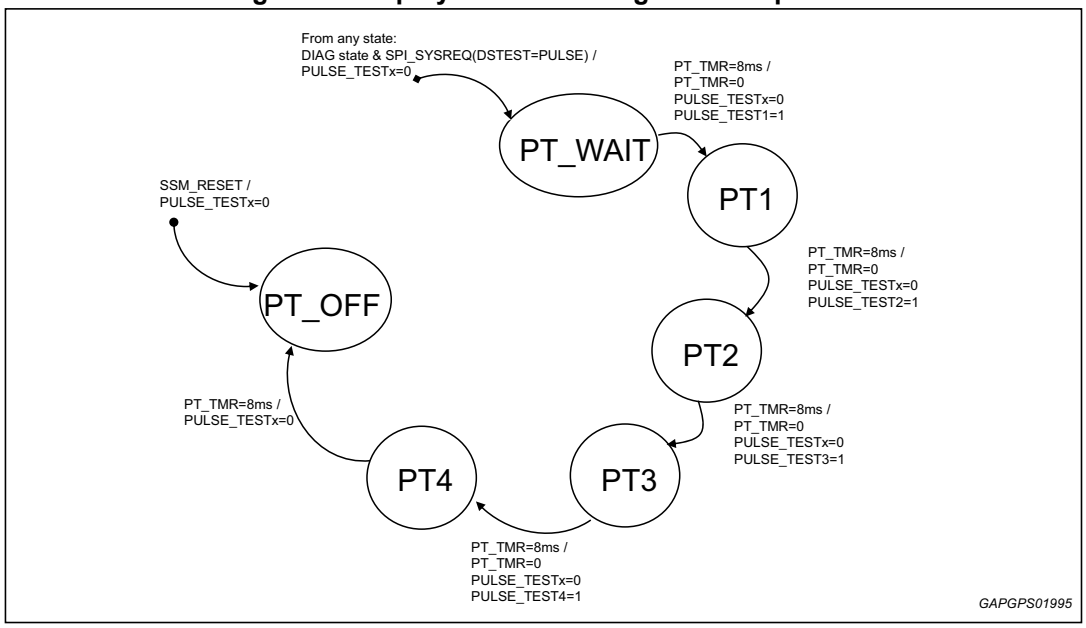
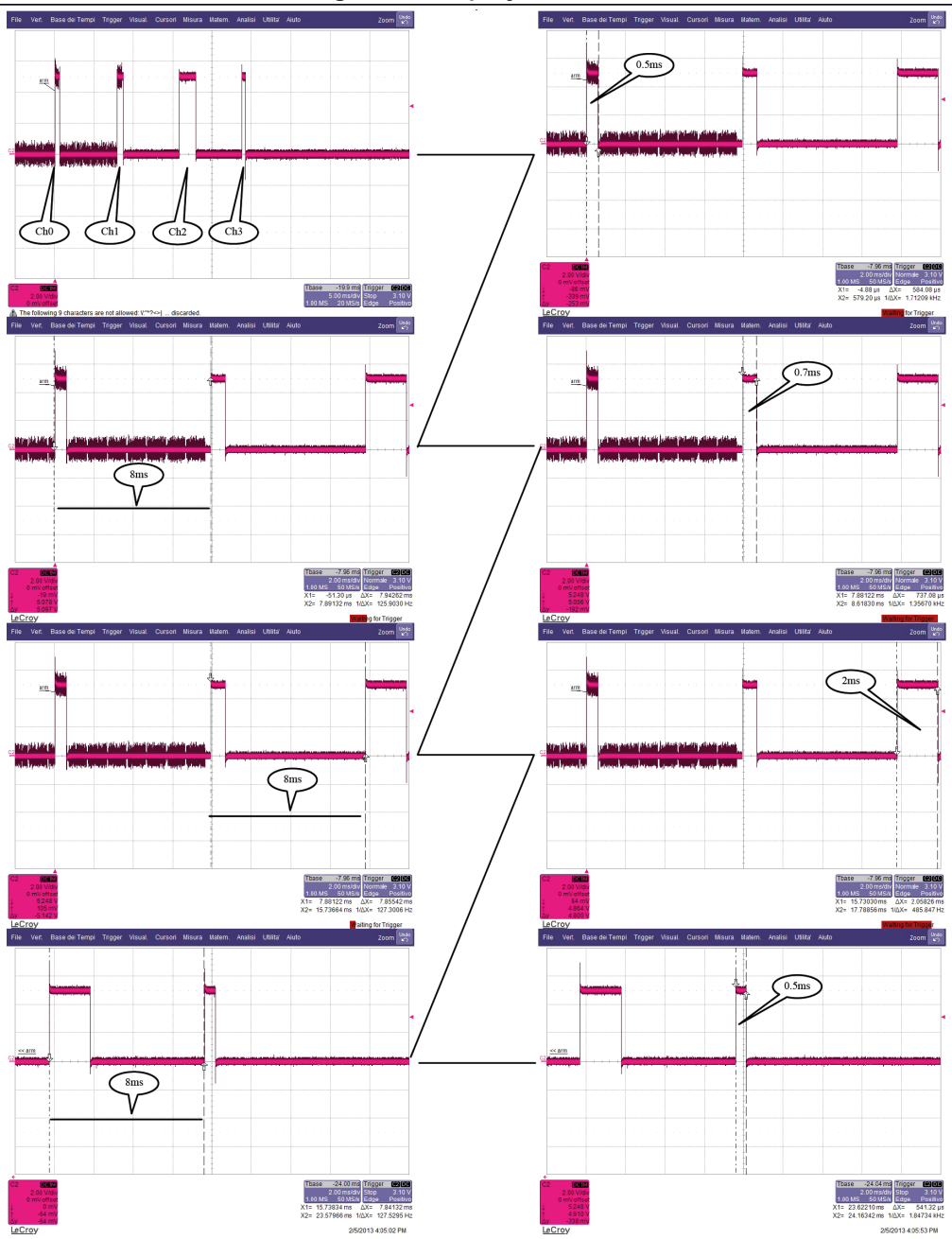


Figure 68. Deployment timer - no programmation

Figure 69. Deployment timer



Ch0: depl time=0.5ms, depl curr=1.2A,
Ch1: depl time=0.7ms, depl curr=1.2A,
Ch2: depl time=2ms, depl curr=1.2A,
Ch3: depl time=2ms, depl curr=1.75A → turns to depl time=0.5ms, depl curr=1.2A

9.2 High level

Figure 70. High level loop diagnostic flow 1

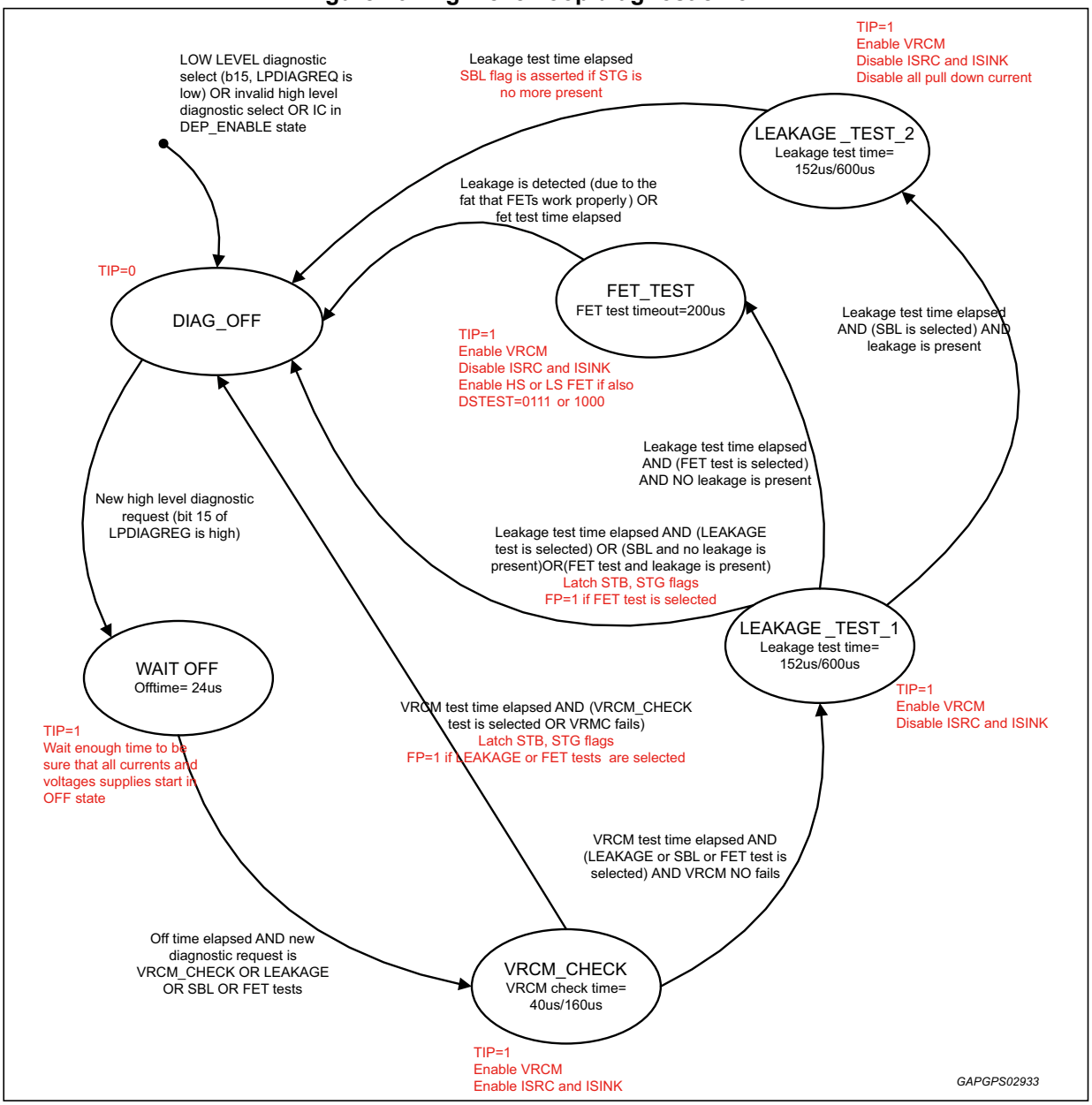
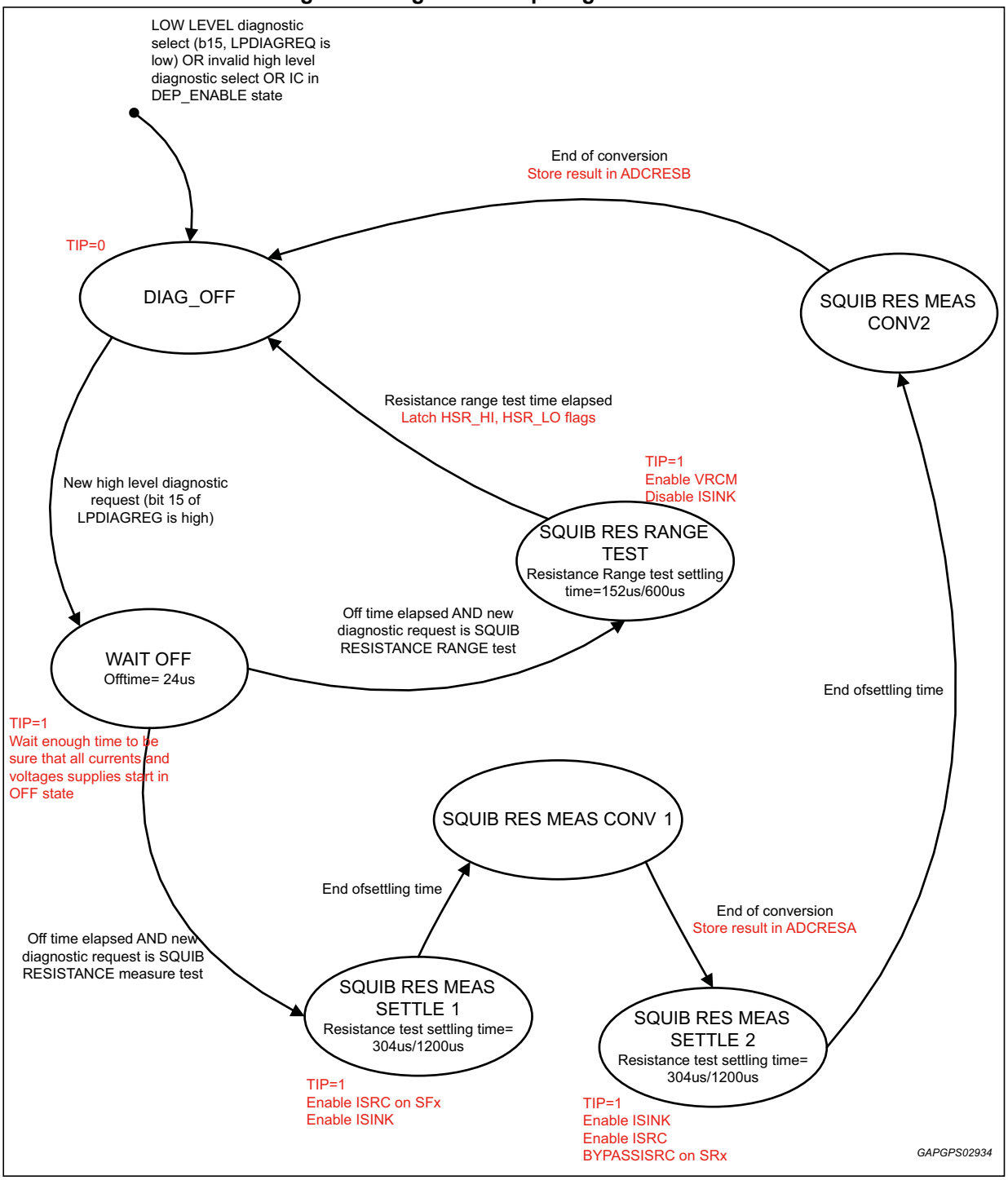


Figure 71. High level loop diagnostic flow 2



Device performs the measurement, as requested by the microcontroller, through LPDIAGREQ register.

Based on the requests from the microcontroller, diagnostics run according to the setups described for the low level mode but each test set up is driven step by step by the IC itself.

IC timing schedule is selected through HI_LEV_DIAG_TIME bit in INIT:

	(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$01 SYS_CFG	I	W				X		0											10: HI_LEV_DIAG_TIME 0=short time 1=long time

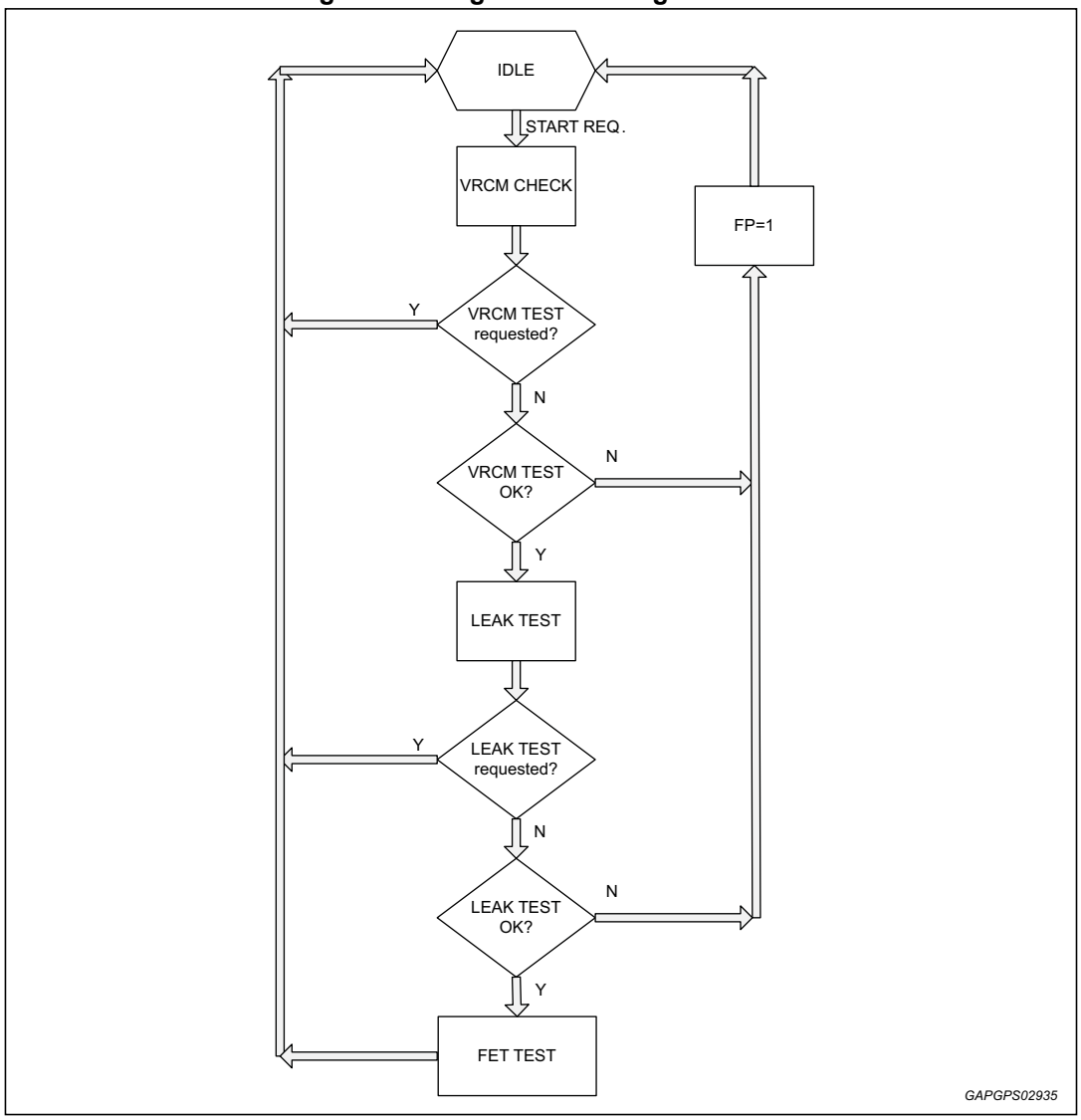
1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 000=No diag sel 001=VRCM Check 010=Leakage Check 011=Short Btw Loops Check 100 = Unused 101=Squib resist range Check 110=Squib resist measure 111=FET test			SQP	LOOP_DIAG_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4 SQP 0=SRx 1= SFx

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

In case of high level diagnostic selection, the IC automatically schedules the preparatory tasks to be eventually run in order to perform the required diagnostic.The following flow chart shows the time sequence implemented:

Figure 72. Diagnostic - Safing FET flow

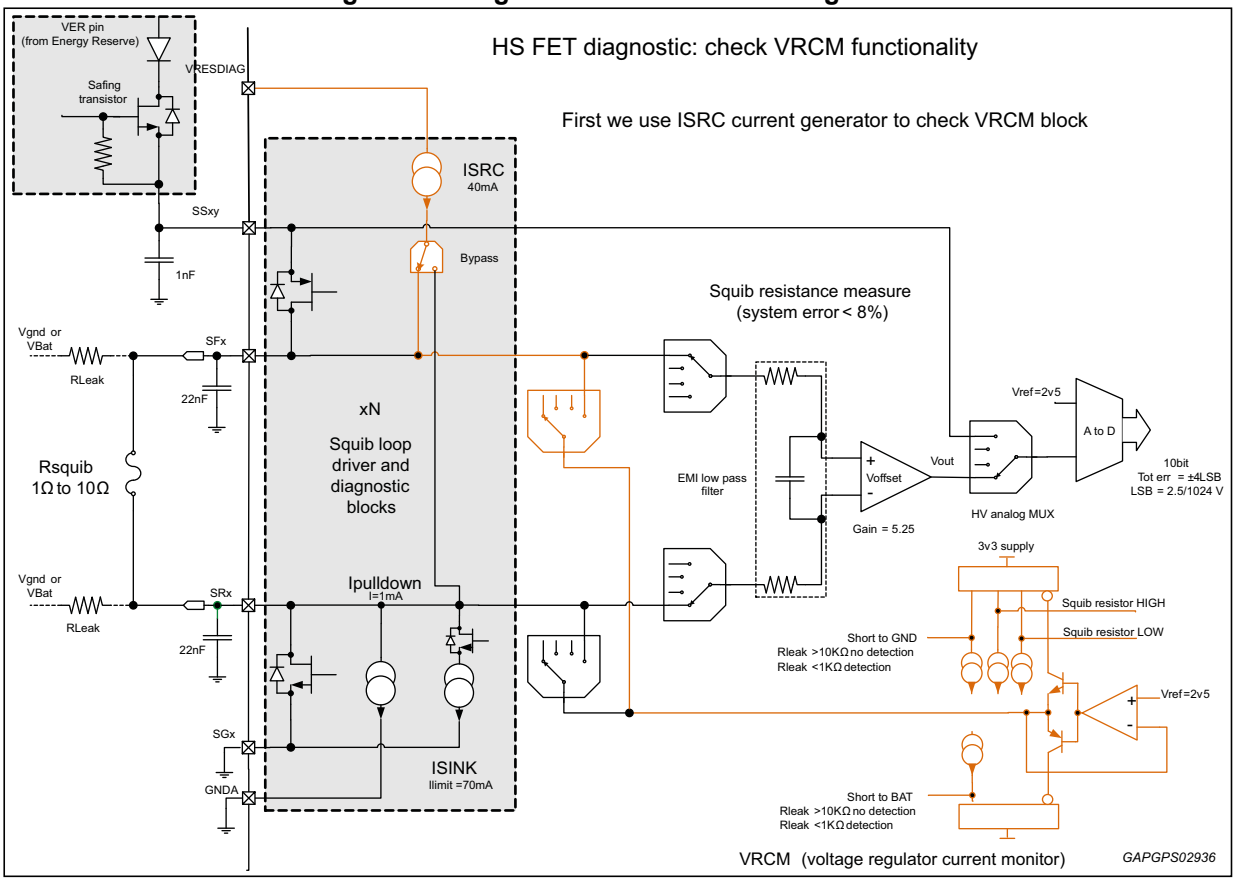


FP bit in LPDIAGSTAT register is available only in case of high level diagnostic selected; it is stuck at 0 otherwise.

Once a test which requires preliminary measurement phases is selected (ie leakage test, FET test), this bit is set if the diagnostic procedure has been stopped because of a fault recorded in such a preliminary steps.

9.2.1 VRCM check - High side

Figure 73. Diagnostic - VRCM check - High side

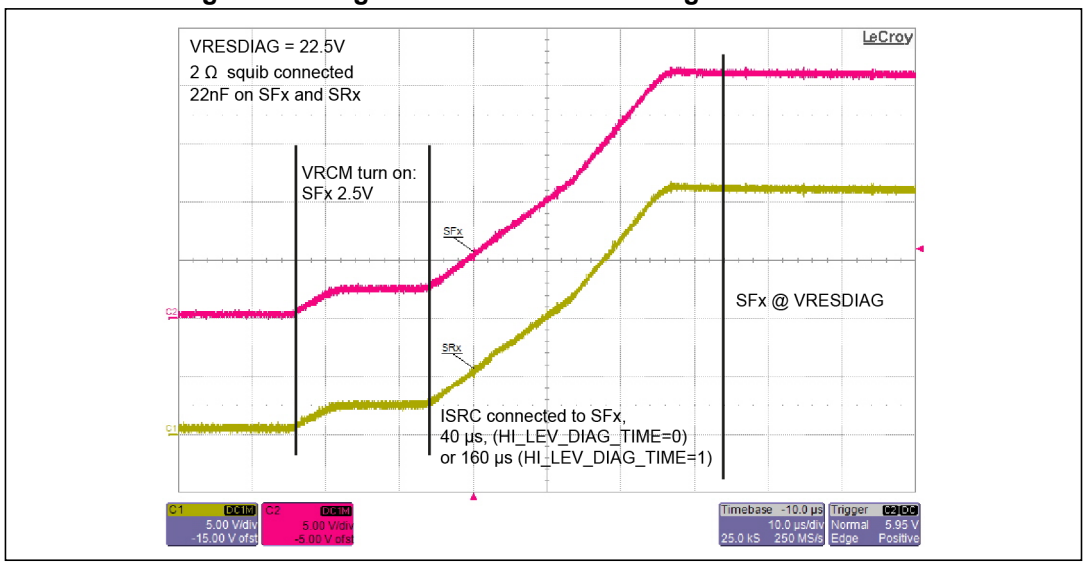


For set up, refer to *Figure 73*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 001=VRCM Check			1	LOOP_DIAG_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4: SQP 0=SRx 1= SFx

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
   (I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
   W = WRITE.

Figure 74. Diagnostic - VRCM check - High side waveform



Test result:

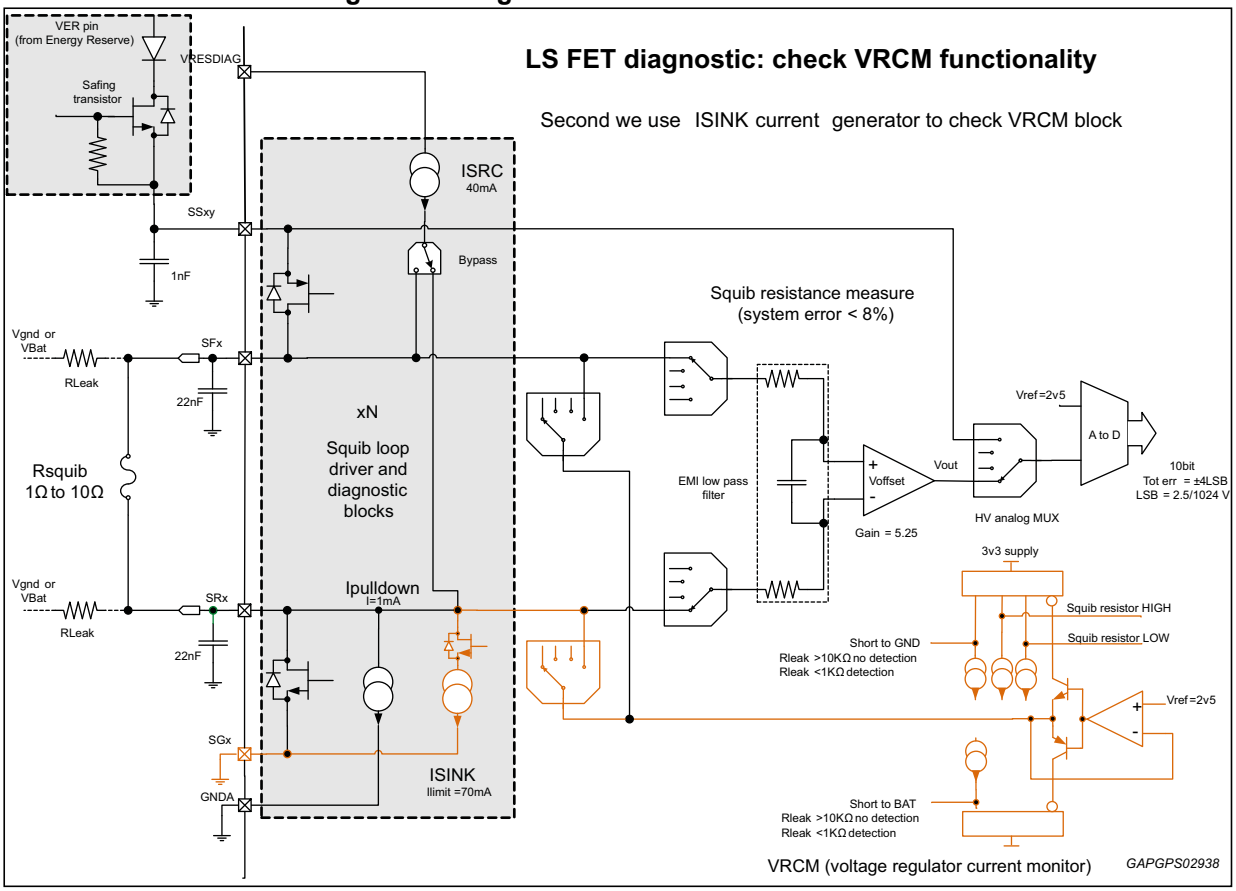
						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R	0				HIGH_LEVEL_DIAG_SEL 001=VRCM Check				0	0	1	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 0=FET off during diagn 7: SBL 0= no short loops 6: STG 0= no short GND 5: STB 1=short to battery 4:SQP=1 SFx 3:0 LEAK_CHSEL
	(3)	19	18	17	16		R																	
		1	0	0	0																			

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Note: VRCM check, once required, is not run one shot on both HS and LS, but microcontroller selects through SQP bit the high side or the low side.

9.2.2 VRCM check - Low side

Figure 75. Diagnostic - VRCM check - Low side



For set up, refer to *Figure 75*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 001=VRCM Check			0	LOOP_DIAG_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4: SQP 0=SRx 1= SFx

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Test result:

Being ISRC and VRCM connected to SFx, if VRCM works correctly, short to battery, readable in LPDIAGSTAT register, is asserted for the channel selected:

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R	0				HIGH_LEVEL_DIAG_SEL 001=VRCM Check				0	1	0	0	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 0=FET off during diagn 7: SBL 0= no short loops 6: STG 1= short GND 5: STB 0=no short to battery 4:SQP=0 SRx 3:0 LEAK_CHSEL
	(3)	19	18	17	16		R																	
		1	0	0	0																			

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Note: VRCM check, once required, is not run one shot on both HS and LS, but microcontroller selects through SQP bit the high side or the low side.

9.2.3 Leakage check - High side

Figure 76. Diagnostic - Leakage check - High side

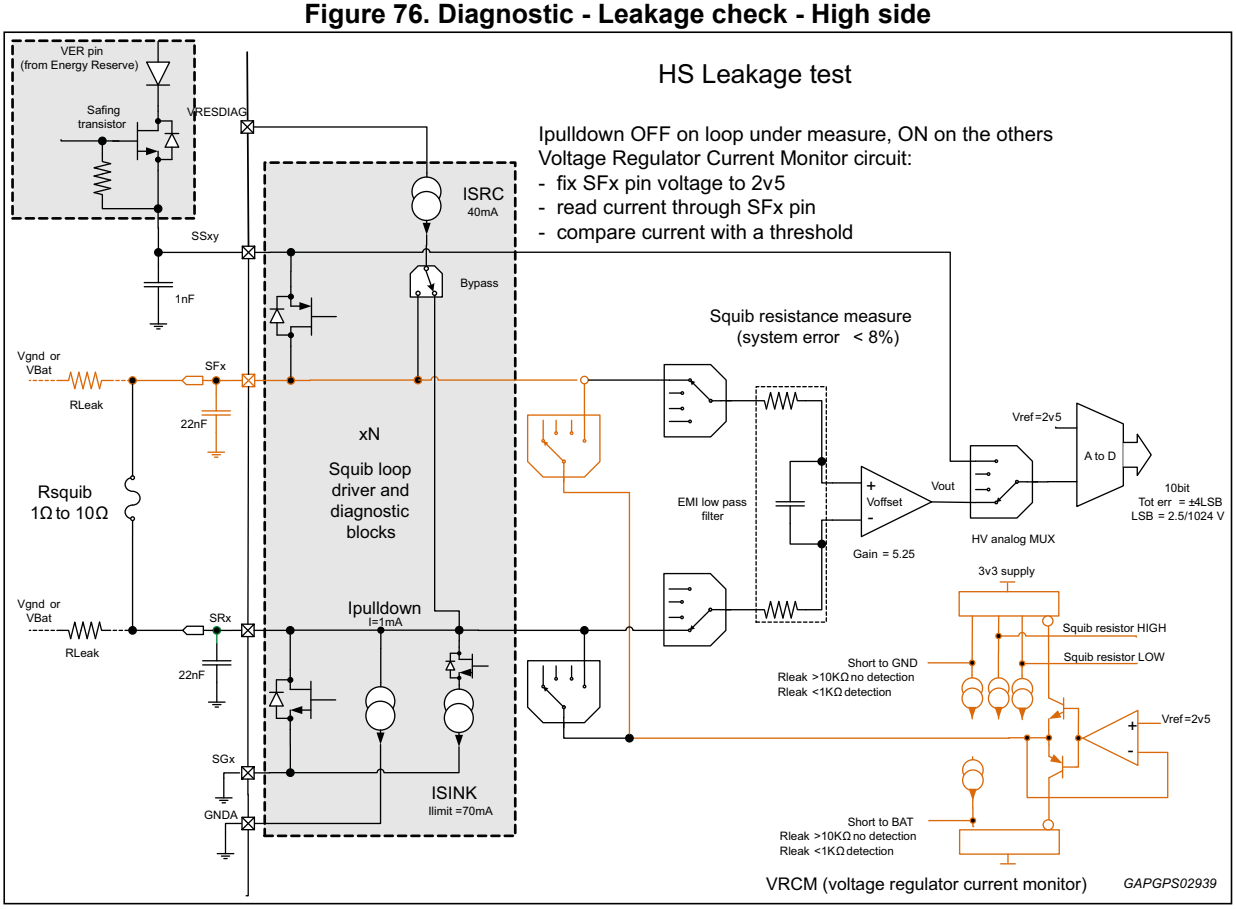
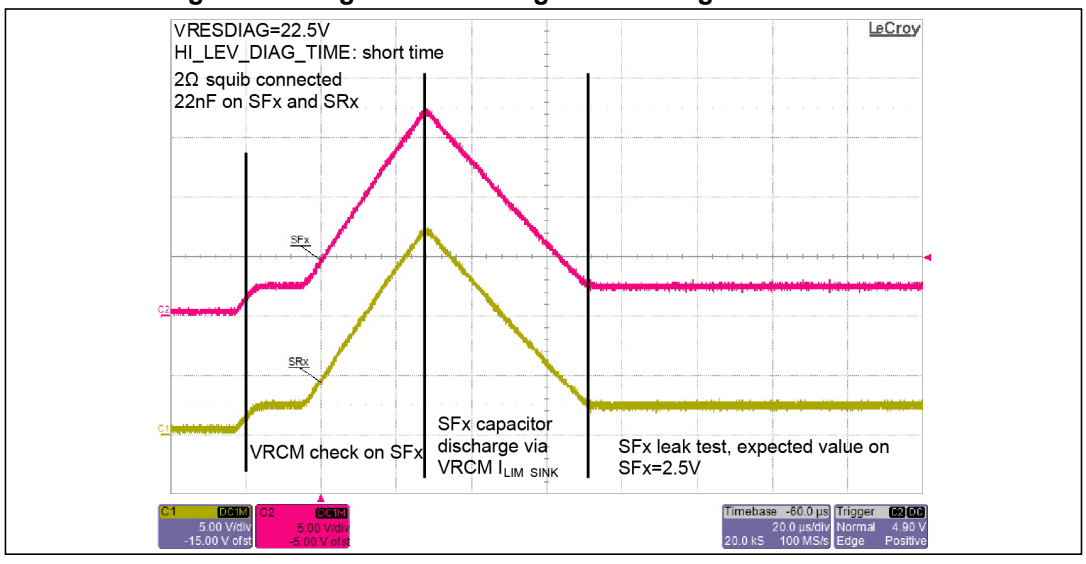


Figure 77. Diagnostic - Leakage check - High side waveform



For set up, refer to *Figure 76*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 010=leakage test			1	LOOP_DIAG_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4: SQP 0=SRx 1= SFx

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Test result:

								(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**								R	0				HIGH_LEVEL_DIAG_SEL 010=LEAKAGE Check				0	0	0	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 0=FET off during diagn 7: SBL 0= no short loops 6: STG 0= no short GND 5: STB 0= no short to batt. 4:SQP=1 SFx 3:0 LEAK_CHSEL
	(3)	19	18	17	16				R																	
		1	0	0	0																					

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Depending on the value of the capacitors mounted on the ECU, the same high level diagnostic can be performed setting HI_LEV_DIAG_TIME bit in order to increase the time of the internal diagnostic finite state machine operation.

This bit can be written only in INIT state.

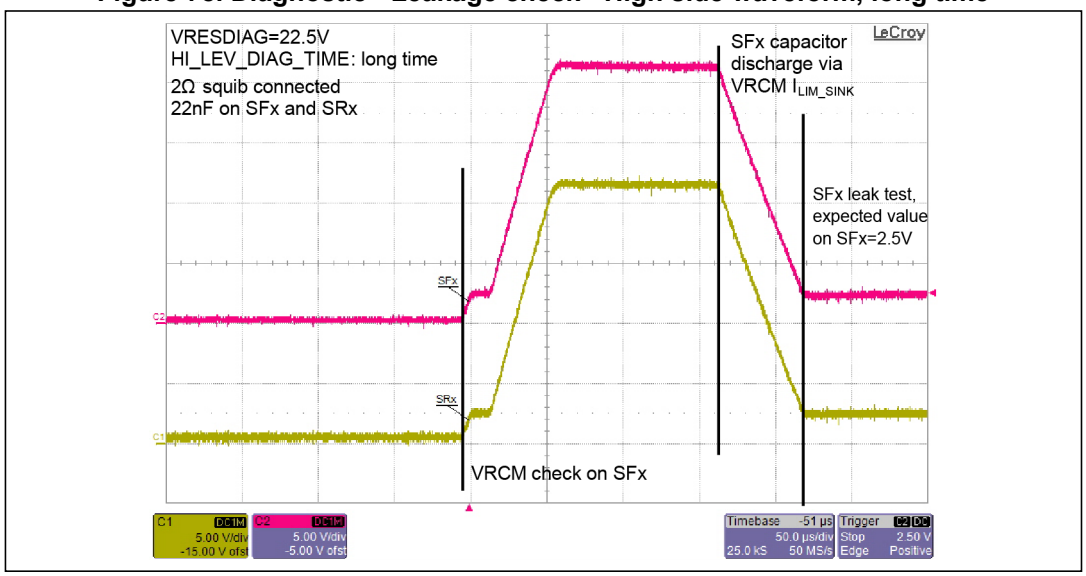
Note: In case HI_LEV_DIAG_TIME has to be written, microcontroller should do it before the RST activation after the initial 500ms are expired.

It could be necessary to disable the watchdog time out function through bit WD1_TOVR as described in *Section 5.1*.

	(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$01 SYS_CFG	I	W				X		1										1	10: HI_LEV_DIAG_TIME 0=short time 1=long time 0: WD1_TOVR 1=timeout disabled

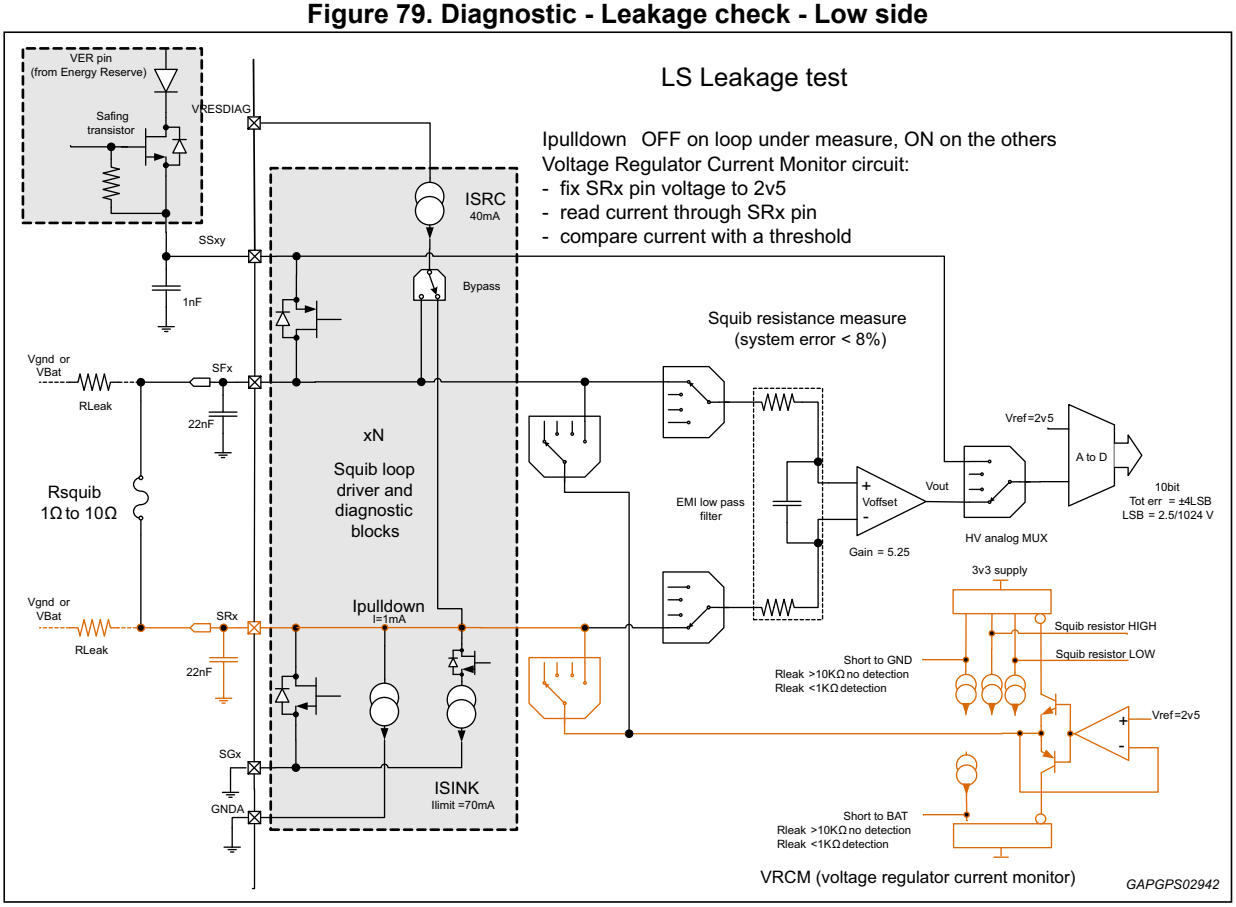
1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Figure 78. Diagnostic - Leakage check - High side waveform, long time



9.2.4 Leakage check - Low side

Figure 79. Diagnostic - Leakage check - Low side



For set up, refer to *Figure 79*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 010=leakage test			0	LOOP_DIAG_CH SEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4: SQP 0=SRx 1= SFx

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.

2. R = READ
W = WRITE.

Test result:

				(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**				R	0				HIGH_LEVEL_ DIAG_SEL 010=LEAKAGE Check				0	0	0	0	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 0=FET off during diagn 7: SBL 0= no short loops 6: STG 0=no short GND 5: STB 0=no short to batt. 4:SQP=0 SRx
	(3)	19	18	17	16		R															
		1	0	0	0																	

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

9.2.5 Short between loops

To be selected if the test is on SFx or SRx via SQP bit:

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL _DIAG_SEL 011=short between loop			0/1	LOOP_DIAG_C HSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4: SQP 0=SRx 1= SFx

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

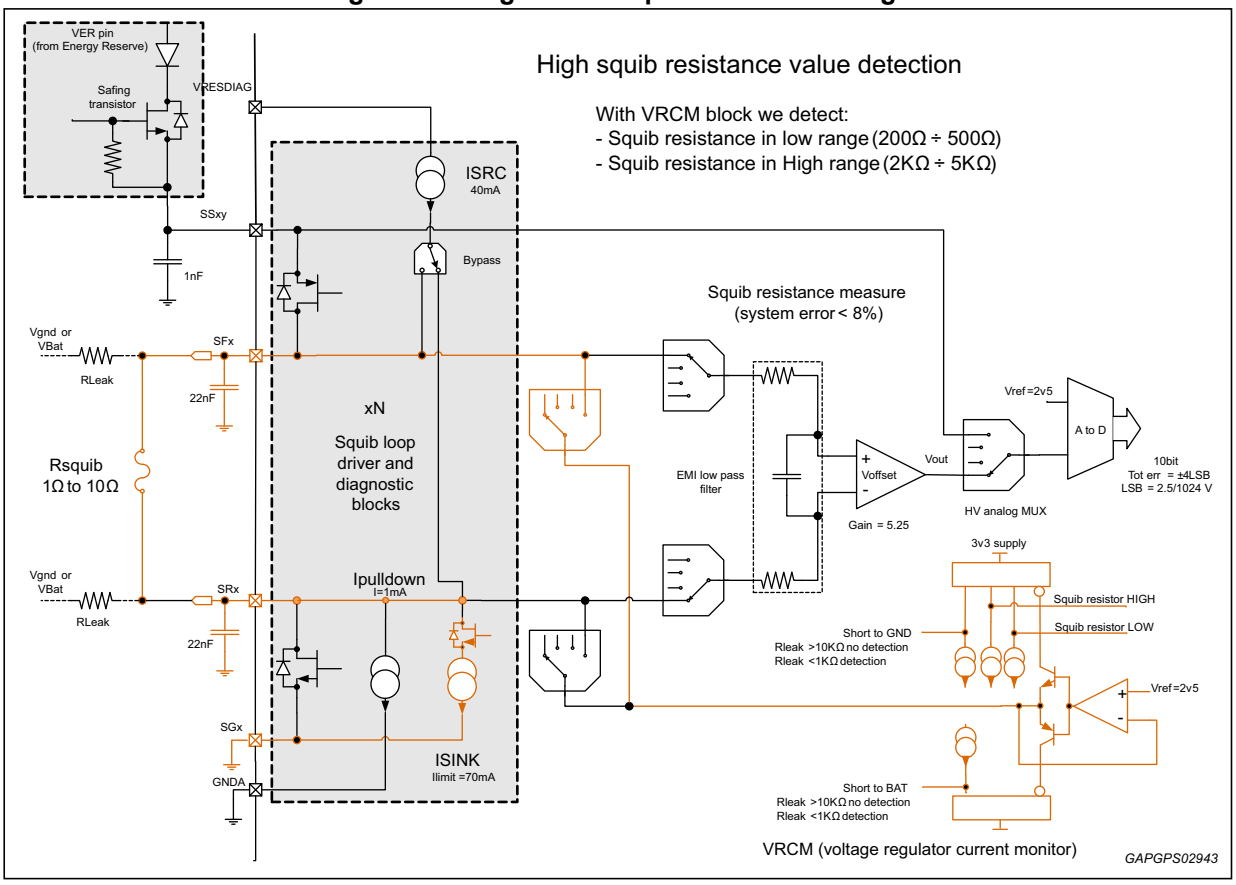
Test result:

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R	0				HIGH_LEVEL_DIAG_SEL 011=short between loop				0	0	0	0/1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 0=FET off during diagn 7: SBL 0= no short loops 6: STG 0=no short GND 5: STB 0=no short to batt. 4:SQP=0 SRx/1=SFz 3:0 LEAK_CHSEL
	(3)	19	18	17	16		R																	
		1	0	0	0																			

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

9.2.6 Squib resistance range

Figure 80. Diagnostic - Squib resistance range



For set up, refer to *Figure 80*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 101=squib res range			1	LOOP_DIAG_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4: SQP 0=SRx 1= SFx

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Test result (2 Ω squib):

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R	0		0	1	HIGH_LEVEL_DIAG_SEL 101=squib res range				0	1	0	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 0=FET off during diagn 7: SBL 0= no short loops 6: STG 1= short GND 5: STB 0=no short to batt. 4:SQP=1 SFx
	(3)	19	18	17	16		R																	
		1	0	0	0																			

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

STG=1 in case the squib has a very low resistive value

SQP=1 means that VRCM is connected to the high side

9.2.7 Squib resistance measurement

IC allows measuring the squib resistance value in the range of [1-10]Ω with overall 8% precision.

Two steps of the measurement described below are managed by the IC, which makes also ADC conversion results available.

Figure 81. Diagnostic - Squib resistance measurement (1)

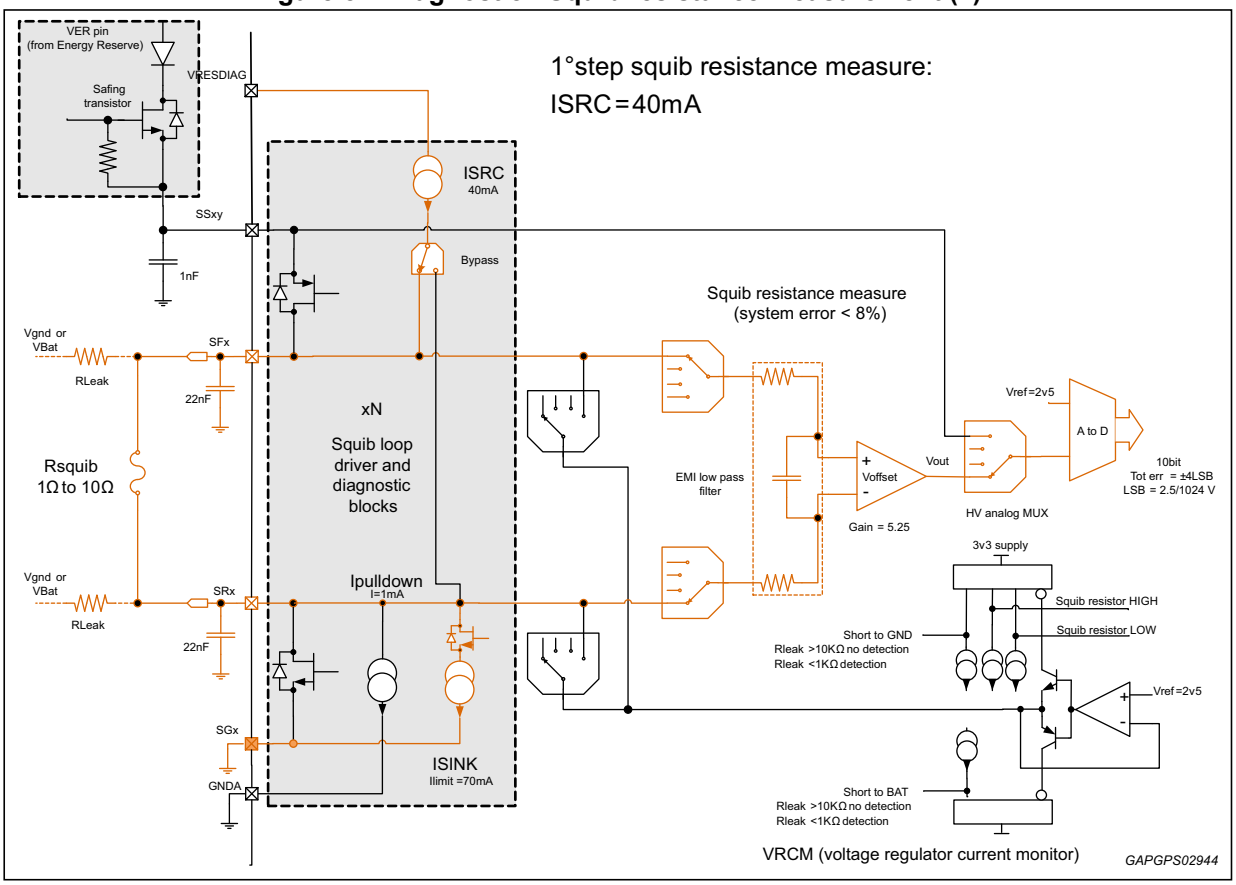
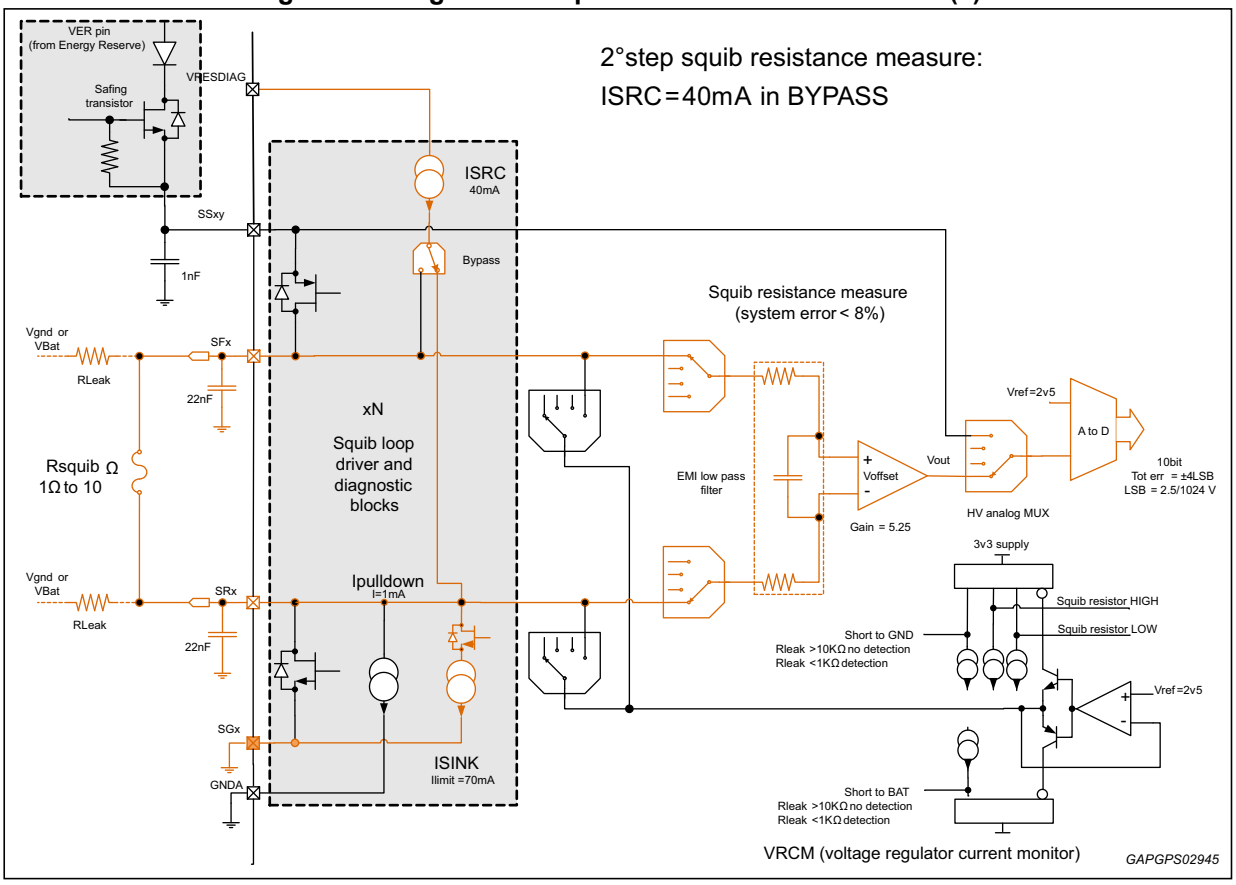


Figure 82. Diagnostic - Squib resistance measurement (2)



IC triggers at the end of each step above an ADC conversion; once the high level diagnostic has been performed, results of ADC conversions have to be read in the registers $3C, $3D DIAGCTRL_x by selection of SQUIB resistance measurement (bit [6:0]=$06).

For set up, refer to *Figure 82*.

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 110=squib res meas			X	LOOP_DIAG_CH SEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Test result:

The registers to be read are still the four DIAGCTRL_x. Again the explanation of the first register (x=A) is true also for the other three registers:

$3X DIAGCTRL_X → X= C, D

Case X = C:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3C DIAGCTRL_C					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $06 = squib x resistance							
(3)	19	18	17	16																			
	1	0	0	ADCREQ_A	-	R	ADCREQ_A $06 = squib x resistance						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Results read in the bit[9:0] will be named in the following as $ADC_{1ST\ CONVERSION}$.

Case X = D:

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
$3C DIAGCTRL_D					-	W	X	X	X	X	X	X	X	X	X	ADCREQ_A $06 = squib x resistance							
(3)	19	18	17	16																			
	1	0	0	ADCREQ_A	-	R	ADCREQ_A $06 = squib x resistance						ADCREQ_A 10bit ADC result										19:1=conversion finished

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Results read in the bit[9:0] will be named in the following as $ADC_{2ND\ CONVERSION}$.

Once read the ADC measurement, to obtain the value it is necessary to consider the divider ratio of the ADC. In case of resistance x, it is 1:1.

Table 8. Resistance measurement of the value ratio ADC

Measurements	Divider Ratio				
	15:1	**10:1**	**7:1**	**4:1**	**1:1**
RESISTANCE					√

Being two measurements, the squib resistance is so calculated:

$$\Delta Vout = (SFx - SRx)_1 - (SFx - SRx)_2$$

$$Rsquib = \frac{\Delta Vout}{G * ISRC}$$

GAPGPS02946

G= 5.25 ±2% differential amplifier gain

ISRC = 40 mA ±5%

Let's consider an example where

ADC1ST CONVERSION = (0100111000)2 = (312)10

ADC2ND CONVERSION = (0010000001)2 = (129)10

Δ_{ADC} = 312 - 129 = 183

In order to obtain the result in Volt, being the A2D characteristic linear,

$$2.5 : 1024 = x : \Delta_{ADC} \rightarrow x = \frac{183 \cdot 2.5V}{1024} = 0.44V$$

GAPGPS02947

In order to obtain resistance value, considering typical factors

$$Rsquib = \frac{x}{G * ISRC} = \frac{0.44V}{5.25 \cdot 40mA} = 2.12\Omega$$

GAPGPS02948

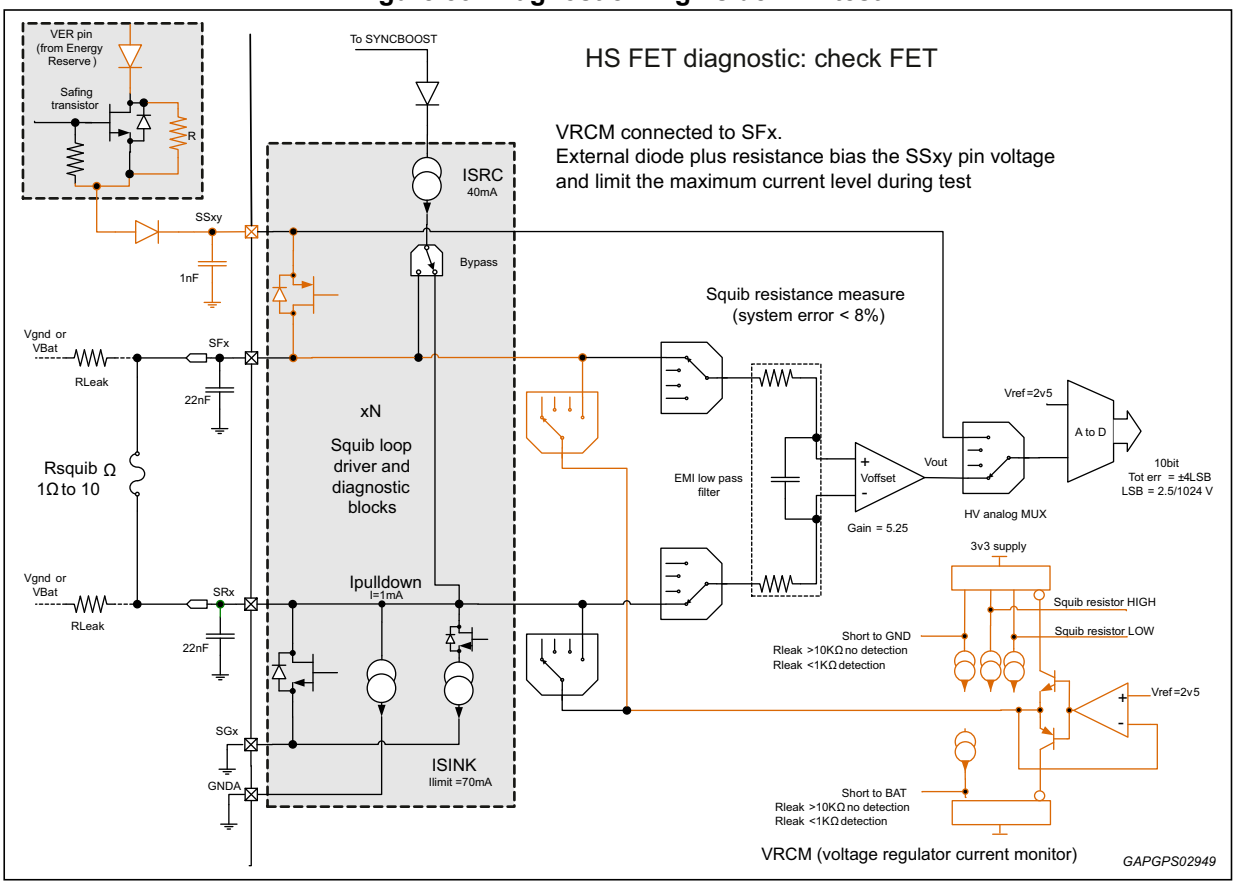
9.2.8 High side FET diagnostic

The test is possible only in the diagnostic phase (see *Section 4.2: Diagnostic*).

These are two tests performed separately for the high side and the low side, with dedicated commands.

Before running this test, IC validates VRCM, then performs leakage test and in case of no failures, high side FET tests is performed.

Figure 83. Diagnostic - High side FET test



For set up, refer to *Figure 83*.

Before selecting FET test, a writing access to the register $36 must be done, as it is shown here below:

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 111=FET test			SQP	LOOP_DIAG_CH SEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4 SQP 0=SRx 1= SFx
3	**$36 SYSDIAGREQ**	D	W	X	X	X	X	X	X	X	X					0	1	1	1	0111: DSTEST=HSFET active

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

High side FET test turns ON the HS power: if it turns ON correctly, SFx is connected to SSxy which is at VER voltage through the resistor R in parallel to the safing FET.

During the test, the device monitors the current flowing through VRCM.

If the high side FET works properly, this current exceeds the thresholds I_{HSFET} (high side) and the channel is immediately turned off,

$$I_{HSFET} = 1.8\ mA \pm 10\%$$

In case the current doesn't exceed the limit mentioned, after a fixed time, $T_{FETTIMEOUT}$, the test is terminated and the output is turned off.

$$T_{FETTIMEOUT} = 200\ \mu s$$

Test result:

Possible results for high side, readable in register $37:

STB=1 & STG=0 **ok**

STB=0 or STG=1 missing SSxy connection during FET test or high side not switched ON or short to GND during FET test

						(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**						R	1		0	1	HIGH_LEVEL_ DIAG_SEL 111 = FET test				0	1	0	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 1=FET off during diagn 7: SBL 0= no short loops 6: STG 0=no short GND 5: STB 1=short to batt. 4:SQP=1 SFx 3:0 LEAK_CHSEL
	(3)	19	18	17	16		R																	
		1	0	0	0																			

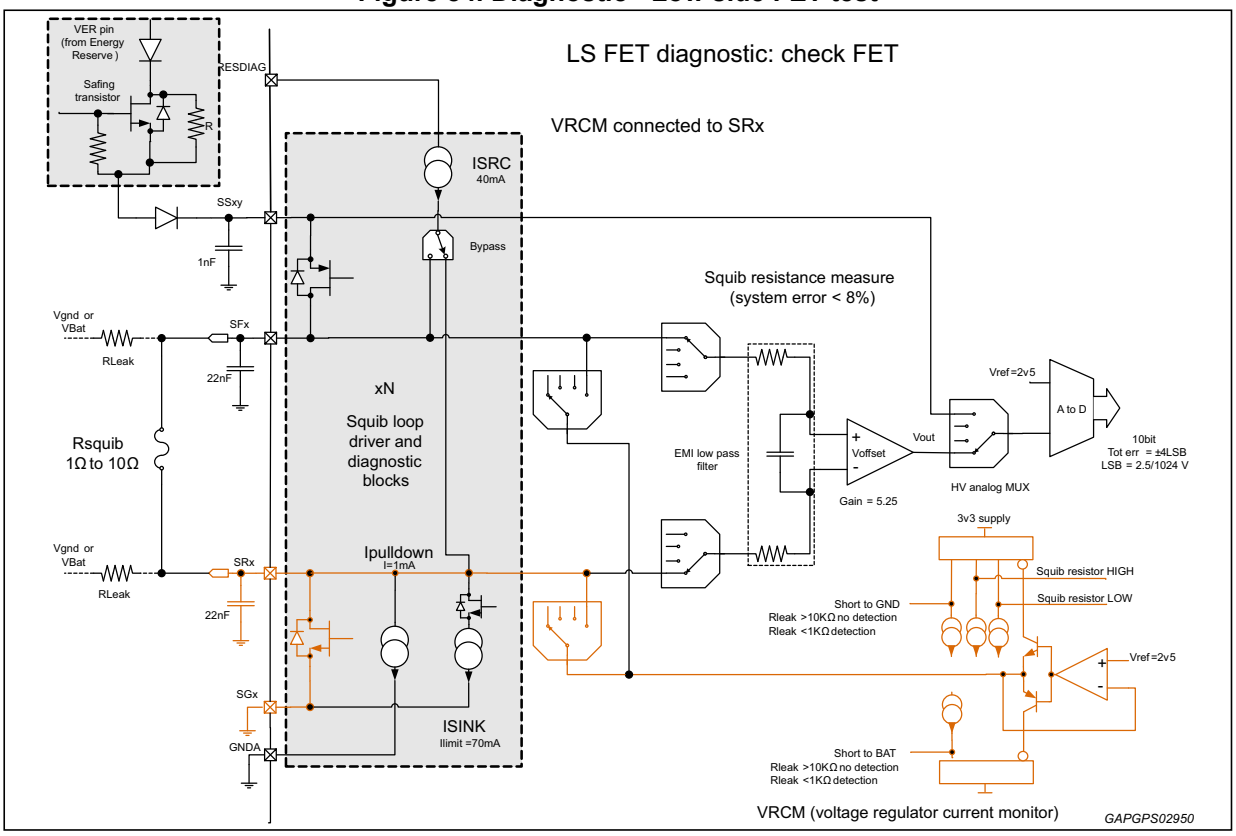
1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

STG & STB, after FET test, are latched. They are cleared through a new LPDIAGREQ or a new SYSDIAGREQ.

9.2.9 Low side FET diagnostic

Before running this test, IC validates VRCM, then performs leakage test and in case of no failures, low side FET test is performed.

Figure 84. Diagnostic - Low side FET test



For set up, refer to *Figure 84*.

Before selecting FET test, a writing access to the register $36 must be done, as it is shown here below:

		(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
3	**$38 LPDIAGREQ**	(I)	W	1	X	X	X	X	X	X	X	HIGH_LEVEL_DIAG_SEL 111FET test			SQP	LOOP_DIAG_CH SEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				15: 1=DIAG HIGH LEVEL 4 SQP 0=SRx 1= SFx
3	**$36SYSDIAGREQ**	D	W	X	X	X	X	X	X	X	X					0	1	1	1	1000: DSTEST=LSFET active

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.

Low side FET test turns ON the low side. If the low side turns ON correctly, SRx is connected to SGxy.

During the test, the device monitors the current flowing through VRCM.

If the FETs work properly, this current exceeds the thresholds I_{LSFET} and the channel is immediately turned off,

$$I_{LSFET} = 450\ \mu A \pm 10\%$$

In case the current doesn't exceed the limit mentioned, after a fixed time, $T_{FETTIMEOUT}$, the test is terminated and the output is turned off.

$$T_{FETTIMEOUT} = 200\ \mu s$$

Test result:

Possible results for low side:

STB = 0 and STG = 1 ok

STB = 1 or STG = 0 short to battery in LS or low side not switched ON.

					(1)	(2)	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
4	**$37 LPDIAGSTAT**					R	1		0	1	HIGH_LEVEL_DIAG_SEL 111 = FET test				0	1	0	1	LEAK_CHSEL 0000 = ch0 0001 = ch1 0010 = ch2 0011 = ch3				19: 1= HIGH LEVEL 18: TIP 0= high level diagn not running 16: FP no fault before test 15: 1=FET on during diagn 7: SBL 0= no short loops 6: STG 1=short GND 5: STB 0=no short to batt 4:SQP=1 SFx 3:0 LEAK_CHSEL
	(3)	19	18	17	16		R																
		1	0	0	0																		

1. I=INIT / D=DIAG / S=SAFING / C=SCRAP / A=ARMING / - = ALL STATES
(I)=no in INIT / (D)= no in DIAG / (S)= no in SAFING / (C)= no in SCRAP / (A)= no in ARMING.
2. R = READ
W = WRITE.
3. Further bit over the 16 standard.

Note: Ground loss (SGxy) is not detected through FET test, because there is a diode between SGxy and the substrate.

STG & STB, after FET test, are latched. They are cleared through a new LPDIAGREQ or a new SYSDIAGREQ.

10 Remote sensor interface - L9678-S only

This feature is available only for L9678-S version.

The 2 remote sensor interfaces (RSU0, RSU1) support protocol asynchronous PSI-5 1.3 version, 8 or 10 bit.

The two channels are independent from each other and a fault on a channel does not influence the other.

Each channel supplies an independently current limited DC voltage to its remote sensor derived from VSUP, and monitors the current to extract encoded data.

PSI5 (Peripheral Sensor Interface) is a two-wire protocol, where the active sensor modulates the load current in order to encode data to be transmitted It is a two current protocol, where the sensor applies a high level step superimposed to its normal current consumption.

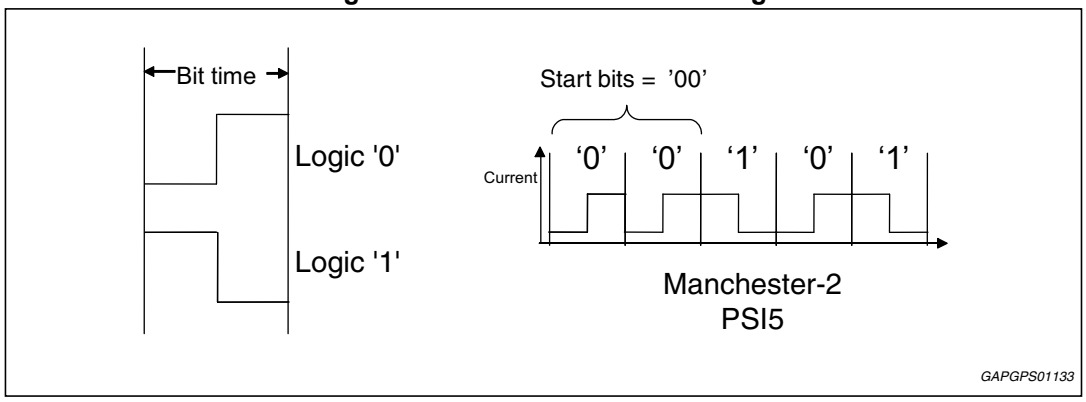
Current level detection threshold is automatically and independently defined for each channel by the IC.

The information, Manchester 2 encoded, is brought through current transitions in the middle of a bit time:

current rising slope = 0

current falling slope = 1

Figure 85. Manchester bit encoding



Features:

- Transmission speed: 125 k baud or 189 k baud
- Word data length: 8 & 10 bit
- 1 bit even parity

Registers:

- RSU configuration is done in DIAG state only:

$4A = RSCR1 Register	Config only in DIAG state
STARTBIT MEAS_DIS, bit 13	0 = used START bit period 1 = not used START bit period

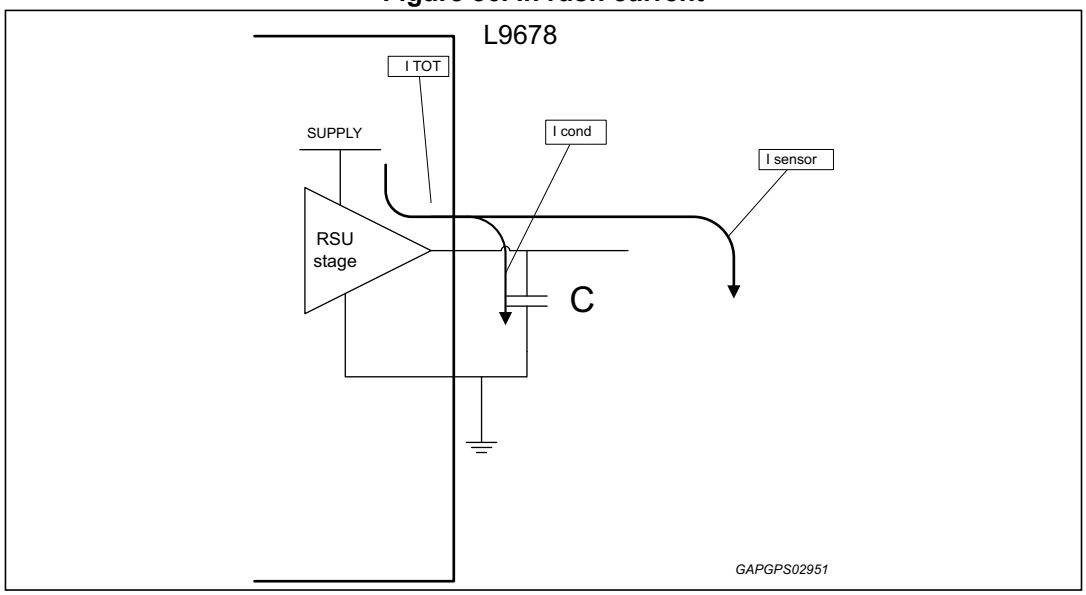
This bit allows to decide the bit period based on the first two start bit (STARTBIT MEAS_DIS, bit 13 =0) or based on the sensor parameter reported in the bit STSx[3:0] that means 1/125k or 1/189k.

$4A = RSCR1 Register	Config only in DIAG state
BLKTxSEL, x=1, bit 10	0 = 5ms blanking time 1 = 10ms blanking time

The same is for $4B register related to RSCR2

Blanking time is active at each channel power-up and masks the inrush current eventually needed to charge capacitors mounted on the line in order to avoid communication error wrong detection.

Figure 86. In rush current



STSx(3:0) bit to select the PSI5 characteristics: data length, transmission speed and parity bit.

$4A = RSCR1 Register	Config only in DIAG state
STSx, x=1, bit [3:0]	0000 = Async. PSI-5, parity 8bit, 125k
	0001 = Async. PSI-5, parity 8bit, 189k
	0010 = Async. PSI-5, parity 10bit, 125k
	0011-1111 = Async. PSI-5, parity 10bit, 189k

Control of RSU is in register $4E:

$4E RSCTRL Register	Config in DIAG, SAFING, SCRAP, ARMING state
CH1EN, bit 3	0 = OFF 1 = ON
CH0EN, bit 1	0 = OFF 1 = ON

Received data are stored into two stages FIFO readable by microcontroller via SPI (RSDRx).

Remote sensor data and fault are reported into two registers, one for each channel ($50 and $51). Content of these registers depends on the status of bit 15.

$50 RSDR0 Register	
FLT, bit 15	0 = no fault 1 = fault present

FLT = 0

$50 RSDR0 Register	
FLT, bit 15	**0 = no fault**
On/off, bit 14	0 = off 1 = on
LCID, bit[13:10]	0000 = RSU 0 0100 = RSU 1
DATA, bit[9:0]	Data
CRC, bit[19:17]	Calculated on the entire data received bit[16:0]. Polynomial calculation: $g(x) = 1+x+x^3$, initialized at 111

FLT = 1

When a fault condition occurs, RSFLT bit in GSW (bit 8 of GSW that is bit 29 of MISO frame) is set.

$50 RSDR0 Register	
FLT, bit 15	**1 = fault present**
On/off, bit 14	0 = off 1 = on
LCID, bit[13:10]	0000 = RSU 0 0100 = RSU 1
STG, bit9	0= no fault 1 = short to ground (in current limit condition)
STB, bit8	0= no fault 1 = short to battery
CURRENT_HI, bit7	0= no fault 1 = channel current exceeds I_{LKGG} for a determined time
OPENDET, bit6	0= no fault 1 = open sensor detected
RSTEMP, bit5	0= no fault 1 = over temperature detected
INVALID, bit4 (Cleared by STG, STB, CURRENT_HI, OPENDET, RSTEMP)	0= no fault 1 = frame with data length error or parity or bit time error.
NODATA, bit3 (Cleared by STG, STB, CURRENT_HI, OPENDET, RSTEMP)	0= no fault 1 = FIFO data empty

Whatever fault occurs, data stored into the 2 stages FIFO are lost and the fault occurred is immediately flagged via SPI.

If no fault has occurred, but no more data are received by the sensor, reading access to the register $50 determines a two stages FIFO pop event: once the FIFO is empty, a NODATA fault is flagged via SPI.

When a fault condition occurs, (except a NODATA), RSFLT bit in GSW (bit 8 of GSW that is bit 29 of MISO frame) is also set.

For the leakage to ground flag described above, the threshold is the following:

$$I_{LKGG} = 36mA \div 45mA$$

The above description referred to $50 RSDR0 is valid for $51 RSDR2 too.

Data stored into RSDRx register are latched until read via SPI.

10.1 Fault protection, short to GND, current limit

Each channel is separately protected against high current.

If the output current exceeds I_{LIMTH} for at least T_{ILIMTH} STG (bit9) register $50 RSDR1 and/or $51RSDR2 is set; the channel involved in the problem is then switched OFF.

$$I_{LIMTH} = 65mA \div 100mA$$

For what concerns the parameter T_{ILIMTH}, it depends on the operative state of the IC. At channel power-up, the inrush current due to the capacitive load must be masked so a short to ground condition is masked for a time period depending on the selection programmed via SPI (register $4A, 4B bit 10, BLKTxSEL)

$$T_{ILIMTH} = 5ms \text{ min if BLKTxSEL=0}$$

$$T_{ILIMTH} = 10ms \text{ min if BLKTxSEL=1}$$

Once initial blanking time is elapsed, the IC validates a short to ground condition after a time period

$$T_{ILIMTH} = 500\ \mu s \div 600\ \mu s$$

Once the channel has been switched off due to an overcurrent condition, in order to reactivate the channel, OFF/ON sequence is necessary.

Timer to count T_{ILIMTH} has a resolution of 25µs.

STG of RSU doesn't interfere with the normal operation of the IC.

10.2 Fault protection, short to battery

Each channel is separately protected against short to battery condition, disconnecting the channel from its supply rail.

Short to battery condition is detected when RSUx voltage pin rises over VSUP for at least T_{STBTH} time. STB (bit8) register $50 RSDR1 and/or $51RSDR2 is set.

$$T_{STBTH} = 12\ \mu s \div 16\ \mu s$$

The counter is cleared if the STB condition is not present for at least 1.5 µs

STB condition doesn't shut-down the channel involved in the STB condition.

STB bit is cleared upon read through SPI.

10.3 Cross link

This allows the verification of short between RSUx channels.

To perform the verification, one channel is turned ON, while the other is kept OFF and the voltage on the OFF channel is read.

$4E RSCTRL Register	Config in DIAG, SAFING, SCRAP, ARMING state
CH1EN, bit 3	0 = OFF 1 = ON
CH2EN, bit 1	0 = OFF 1 = ON

To read the pin voltage on the other channel:

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished - expected
ADCREQ_A, bit [6:0] request (MOSI)	$32 = RSU0 $33 = RSU1
ADCREQ_A, bit [16:10] readout (through MISO),	$32 = RSU0 $33 = RSU1
ADCRES_A, bit [9:0]	10bit ADC result correspondent to the ADCREQ_A, bit [9:0]

The same for the other registers, the choice depends on the user:

$3B DIAGCTRL, x = B

$3C DIAGCTRL, x = C

$3D DIAGCTRL, x = D

Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio of the ADC. In case of RSUxx, it is 4:1.

Table 9. RSUx measurement to obtain the voltage value

Measurements	**Divider Ratio**				
	15:1	**10:1**	**7:1**	**4:1**	**1:1**
RSUx				√	

An external logic has to manage the RSUx voltage read out and to distinguish the value expected with respect to the value read and then declare or not the cross link between loops

10.4 Leakage to battery / open condition

Open condition or leakage to battery is detected in the same way: current through RSUx falls down lower than I_{LKGB} for at least T_{RSUOP_FILT}, being

$$I_{LKGB} = 2\ mA \div 3\ mA$$

$$T_{RSUOP\_FILT}: 10\ \mu s \div 15\ \mu s$$

When detected this fault OPENDET (bit 6), register $50 RSDR1 and/or $51RSDR2 is set. The channel involved in the problem is not switched OFF.

$50 RSDR0 Register	
FLT, bit 15	**1 = fault present**
OPENDET, bit6	0= no fault 1 = open sensor detected

10.5 Leakage to ground

The leakage to ground is detected through high current, greater than I_{LKGG}, for at least T_{RSUCH_FILT}, being

$$I_{LKGG} \text{ for } = 36\ mA \div 45\ mA$$

$$T_{RSUCH\_FILT}: 10\ \mu s \div 15\ \mu s$$

When detected this fault, CURRENT_HI (bit7), register $50 RSDR1 and/or $51RSDR2 is set.

The channel involved in the problem is not switched OFF.

$50 RSDR0 Register	
FLT, bit 15	**1 = fault present**
CURRENT_HI, bit7	0= no fault 1 = channel current exceeds I_{LKGG} for a determined time

10.6 Thermal shut-down

Each channel is equipped with its dedicated over-temperature detection, each one independent from the other.

If the over-temperature on a channel is detected, the channel itself is switched off, without influencing the other.

In correspondence RSTEMP bit of register $50 RSDR1 and/or $51 RSDR2 is set.

$50 RSDR0 Register	
FLT, bit 15	**1 = fault present**
RSTEMP, bit5	0= no fault 1 = over temperature detected

To reactivate the channel after an over-temperature, the sequence OFF-ON of the channel ($4E RSCTRL register, bit 3 and/or1) is required.

10.7 Manchester decoding

PSI5 protocol encodes messages by modulating current sunk by the sensor through Manchester 2 codification: IC is able to sense current and decode the messages, in order to provide data via SPI register.

Once two valid start bits are detected, IC provides error detection mechanism by monitoring the period and the number of the bit sent in the frame.

Remote sensor interface errors are reported in the GSW bit8, RSFLT, that corresponds to bit 29 in MISO

MISO BIT	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**
MISO	SPIFLT	DEPOK	RSFLT	WDTDIS_S	ERSTATE	POWERFLT	FLT	CONVRDY2	CONVRDY1	ERR_WID	ERR_RID
GSW BIT	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**

10.8 Trip current auto adjust

Depending on the number of sensors connected to the line, the quiescent current sunk from each RSU interface may change.

IC provides the means to detect such a variation and is able to put an adaptive threshold in order to recognize the current transitions due to the signal modulation.

Once the quiescent current value has been detected, the IC recognizes a transition low → high if the current sunk by the sensor is higher than

$$I_{RSUxTH} = I_{BO} + I_{TH}$$

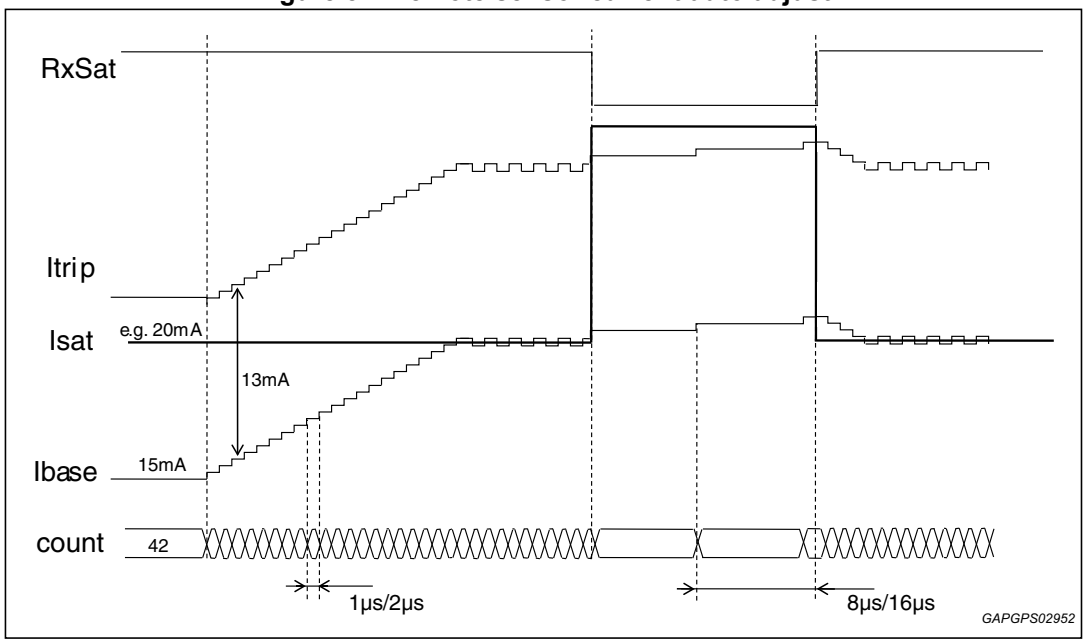
Where I_{BO} the base current which can span from 2.5mA up to 41mA covering PSI-5 specification range and I_{TH} is in the range of

[(12-9%),(12+9%)]mA

Referring to the current sensing auto adjust, there is the possibility to choose the counter frequency, see *Figure 87*, setting SLOWTRACK bit in register RSCR1 (the same for RSCR2 register)

$4A = RSCR1 Register	Config only in DIAG state
SLOWTRACK, bit 14	0: 8µs/1µs 1: 16µs/2µs

Figure 87. Remote sensor current auto adjust



11 DC sensor interface

These four interfaces are dedicated to positioning sensors (as, for example presence of the person on the seat, seat belt fasten and so on). They can be Hall Effect sensors, resistive sensors or simply switches.

Each channel has an internal pull down (100uA), default active, that is deactivated by switching ON the channel, or by requiring voltage measurement or by setting DCS_PD_CURR bit channel per channel or all the channels together.

In DIAG, SAFING; SCRAP or ARMING state, each channel, selected through CHID bit, is controlled and configured via SPI:

$39 SWCTRL register	Config in DIAG, SAFING; SCRAP or ARMING state
CDS_PD_CURR, bit7	0 = pull down OFF for the channel under voltage or current measurement, ON for all the others 1 = pull down OFF for all the channels
SWOEN, bit6	0 = OFF 1 = ON (40mA)
CHID, bit [3:0]	0000 = channel 0 0001 = channel 1 0010 = channel 2 0011 = channel 3 0100-1111 = not used

It is possible, through ADC, for each channel, to require the following measurements:

- Current measurement
- Voltage measurement, with the channel ON or OFF
- Resistor measurement

Current and resistor measurement require the channel ON (SWOEN=1).

As the measure has been performed, the channel is automatically switched off (to prevent power dissipation) or not, following the system configuration:

$01 SYS_CFG register	Config in INIT state
EN_AUTO_SWITCH_OFF, bit15	0 = auto switch off disable 1= auto switch off disable

Current measurement

Before running the current measurement, it is previously necessary to switch on the channel by accessing to register $39 as showed above.

The result of the measurement is addressed on $3x DIAGCTRLx, x = A...D.

Case X = A:

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished
ADCREQ_A, bit [6:0] request (MOSI)	$04 DCSi current
ADCREQ_A, bit [16:10] readout (through MISO),	$04 DCSi current
ADCRES_A, bit [9:0] (MISO)	10bit ADC result

Once the current measurement on the channel is completed, the channel is automatically switched off or not based on EN_AUTO_SWITCH_OFF bit configuration.

Current measurement is possible both in the range of 2mA÷22mA with a total error ±12% (including all errors which affect ADC measurement) or in the range of 1mA÷2mA: in this case the accuracy of the measurement becomes ±30%.

$$I_{DCSx} = \frac{1}{R_{REF1\_IDCSx}} * \frac{ADC_{REF\_hi}}{2^{ADC_{RES}}} * DIAGCTRLn\ (ADCRESn)$$

GAPGPS02953

where $R_{REF1\_IDCSx} = 83.333\Omega$

Example 1

Let's consider an example where 3kΩ resistor is mounted on DCSx:

when the channel is ON a voltage of typ 6.25 V is regulated so current flowing is ~2.1 mA

$3A DIAGCTRL_A, ADCRES_A, bit [9:0]:

ADC = $(0001001000)_2 = (72)_{10}$

In order to obtain the result in A, being the ADC characteristic linear,

$$2.5 : 1024 = x : ADC \rightarrow x = \frac{72 \cdot 2.5V}{1024} = 0.176V$$

GAPGPS02954

In order to obtain current value, considering typical factors,

$$I_{DCSx} = \frac{x}{83.3} = \frac{0.176V}{83.3\Omega} = 2.1mA$$

GAPGPS02955

Note: In case of a low value resistor mounted on DCSx channel, the current may be higher than $I_{DCS\_LIMIT\_L}$ = 24 mA ÷30 mA
In this case the IC is no more able to regulate 6.125 V on the channel but the current measurement still works fine.

Example 2

Let's consider an example where 65 Ω resistor is mounted on DCSx:

when the channel is ON a voltage of ~1.8 V is present with a current flowing ~27 mA (out of regulation)

$3A DIAGCTRL_A, ADCRES_A, bit [9:0]:

ADC = $(1110101001)_2 = (937)_{10}$

In order to obtain the result in A, being the ADC characteristic linear,

$$2.5 : 1024 = x : ADC \rightarrow x = \frac{937 \cdot 2.5V}{1024} = 2.287V$$

GAPGPS02956

In order to obtain current value, considering typical factors

$$I_{DCSx} = \frac{x}{83.3} = \frac{2.287V}{83.3\Omega} = 27mA$$

GAPGPS02957

Voltage measurement

IC is able to perform voltage measurement in the range of 1.5V÷10V with a total error ±8% (including all errors which affect ADC measurement).

For the voltage measurement, it is possible to run such a diagnostic both with channel on and off.: a double step measurement (ch ON and OFF) is recommended in order to reach a good precision by means of offset compensation.

In case of channel **ON** measurement, the channel has to be switched on, then the result of the measurement is addressed on $3x DIAGCTRLx, x=A...D.

Case X = A:

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished
ADCREQ_A, bit [6:0] request (MOSI)	$03 DCSi voltage
ADCREQ_A, bit [16:10] readout (through MISO),	$03 DCSi voltage
ADCRES_A, bit [9:0] (MISO)	10bit ADC result

Once this measurement is completed, the channel is not automatically switched off, but if necessary, it can be switched off by the microcontroller.

Voltage measurement with the channel **OFF** is done through DCSx bit, registers $3x DIAGCTRLx, x=A...D:

Case x = A

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished
ADCREQ_A, bit [6:0] request (MOSI)	$0B DCS0 voltage pin, channel off $0C DCS1 voltage pin, channel off $0D DCS2 voltage pin, channel off $0E DCS3 voltage pin, channel off
ADCREQ_A, bit [16:10] readout (through MISO),	$0B DCS0 voltage pin, channel off $0C DCS1 voltage pin, channel off $0D DCS2 voltage pin, channel off $0E DCS3 voltage pin, channel off
ADCRES_A, bit [9:0] (MISO)	10bit ADC result

In order to calculate the real voltage value from the result coming from an ADC conversion, 7.125 typ scaling factor must be taken into account.

Example 3

Let's consider an example where DCSx is at its voltage regulation value 6.25V typ

$ADCCH_{ON} = (0111101011)_2 = (491)_{10}$

$ADCCH_{OFF} = (0010001011)_2 = (139)_{10}$

$$\Delta_{ADC} = ADC_{CHON} - ADC_{CHOFF} = (352)_{10}$$

In order to obtain the result in V, being the ADC characteristic linear,

$$2.5 : 1024 = x : \Delta_{ADC} \rightarrow x = \frac{352 \cdot 2.5V}{1024} = 0.86V$$

GAPGPS02958

Considering typical scaling factor

$$V_{DCSx} = x \cdot 7.125 = 6.12\ V$$

Note: The voltage measurement is still available in case the current load is higher than $I_{DCS\_LIMIT\_L}$ = 24 mA ÷ 30 mA and channel is no more able to regulate 6.25 V typ.

Resistor measurement

IC is able to perform resistor measurement in the range of 65 Ω ÷ 3000 Ω with a total error defined by the combination of the accuracy obtained on the voltage and current measurements.

IC gives possibility to run one shot the resistor measurement by single access to the ADC.

In order to perform such measurement, the channel under test has to be previously switched on in the register $39.

Note: The one shot resistor measurement can be run only through $3A DIAGCTRL_A access.

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished
ADCREQ_A, bit [6:0] request (MOSI)	$05 DCSx resistance
ADCREQ_A, bit [16:10] readout (through MISO),	$05 DCSx resistance
ADCRES_A, bit [9:0] (MISO)	10bit ADC result

Once the one shot measurement is completed, the results are available in the register $3A DIACTRL_A (for the current flowing through the channel) and $3B DIACTRL_B (for the channel voltage).

Note: The result which is read on the register $3B corresponds to the voltage with channel ON. It is recommended to subtract from this voltage the offset which can be read with channel OFF.

Once current and voltage are available, the resistor can be computed as per the following:

$$I_{DCSx} = \frac{1}{R_{REF1\_IDCSx}} * \frac{ADC_{REF\_hi}}{2^{ADC_{RES}}} * DIAGCTRLn(ADC_{\$3A})$$

$$V_{DCSx} = RATIO_{VDCSx} * \left(\frac{ADC_{REF\_hi}}{2^{ADC_{RES}}} * DIAGCTRLn(ADC_{\$3BCHON}) - V_{OFF\_DCSx} \right)$$

$$R_{DCSx} = \frac{RATIO_{VDCSx} * R_{REF1\_IDCSx} * (DIAGCTRLn(ADC_{\$3BCHON}) - DIAGCTRLn(ADC_{CHOFF}))}{DIAGCTRLn(ADC_{\$3A})}$$

GAPGPS02959

Let's consider an example where a 3 kΩ resistor is mounted on DCSx

$DIAGCTRLn(ADC_{CHOFF}) = (001000101)_2 = (139)_{10}$

$DIAGCTRLn(ADC_{\$3BCHON}) = (0111101010)_2 = (490)_{10}$

$DIAGCTRLn(ADC_{\$3A}) = (0001000111)_2 = (71)_{10}$

$$R_{DCSx} = \frac{(490 - 139) * 7.125 * 83.3}{71} = 2934\,\Omega$$

GAPGPS02960

Once the channel resistor measurement is completed, the channel is automatically switched off or not based on EN_AUTO_SWITCH_OFF bit configuration

Note: The resistor measurement is still available in case the current load is higher than $I_{DCS\_LIMIT\_L}$ = 24 mA ÷ -30 mA and channel is no more able to regulate 6.25 V typ.

DCSx tolerate external ground shift up to ±1 V.

Short between channels:

By means of channel pull-down disabling, the IC in combination with the microcontroller can detect a short between channels, enabling one channel

$39 SWCTRL register	Config in DIAG, SAFING; SCRAP or ARMING state
DCS_PUCURR [bit 7]	0 = pull down OFF for the channel under voltage or current measurement, ON for all the others
SWOEN, bit6	1 = ON
CHID, bit [3:0]	0000 = channel 0 0001 = channel 1 0010 = channel 2 0011 = channel 3

and measuring the voltage of the others

$3X DIAGCTRL_X → X = A, B, C, D

Case X = A:

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished
ADCREQ_A, bit [6:0] request (MOSI)	$0B DCS0 voltage pin, channel off $0C DCS1 voltage pin, channel off $0D DCS2 voltage pin, channel off $0E DCS3 voltage pin, channel off
ADCREQ_A, bit [16:10] readout (through MISO),	$0B DCS0 voltage pin, channel off $0C DCS1 voltage pin, channel off $0D DCS2 voltage pin, channel off $0E DCS3 voltage pin, channel off
ADCRES_A, bit [9:0] (MISO)	10bit ADC result

Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio that is 7.125 in this measurement.

Protections:

Each channel is protected against:

- Overload
- Ground shift (±1V)
- Loss ECU battery
- Loss ground
- Short to ground.

12 GPO drivers

There are two GPO drivers configurable as high side or low side drivers.

Their configuration is done via SPI:

$42 GPOCR register	Config. in DIAG state only
GPO1HS, bit1	0 = high side driver configured for ch1 1 = low side driver configured for ch1
GPO0HS, bit0	0 = high side driver configured for ch0 1 = low side driver configured for ch0

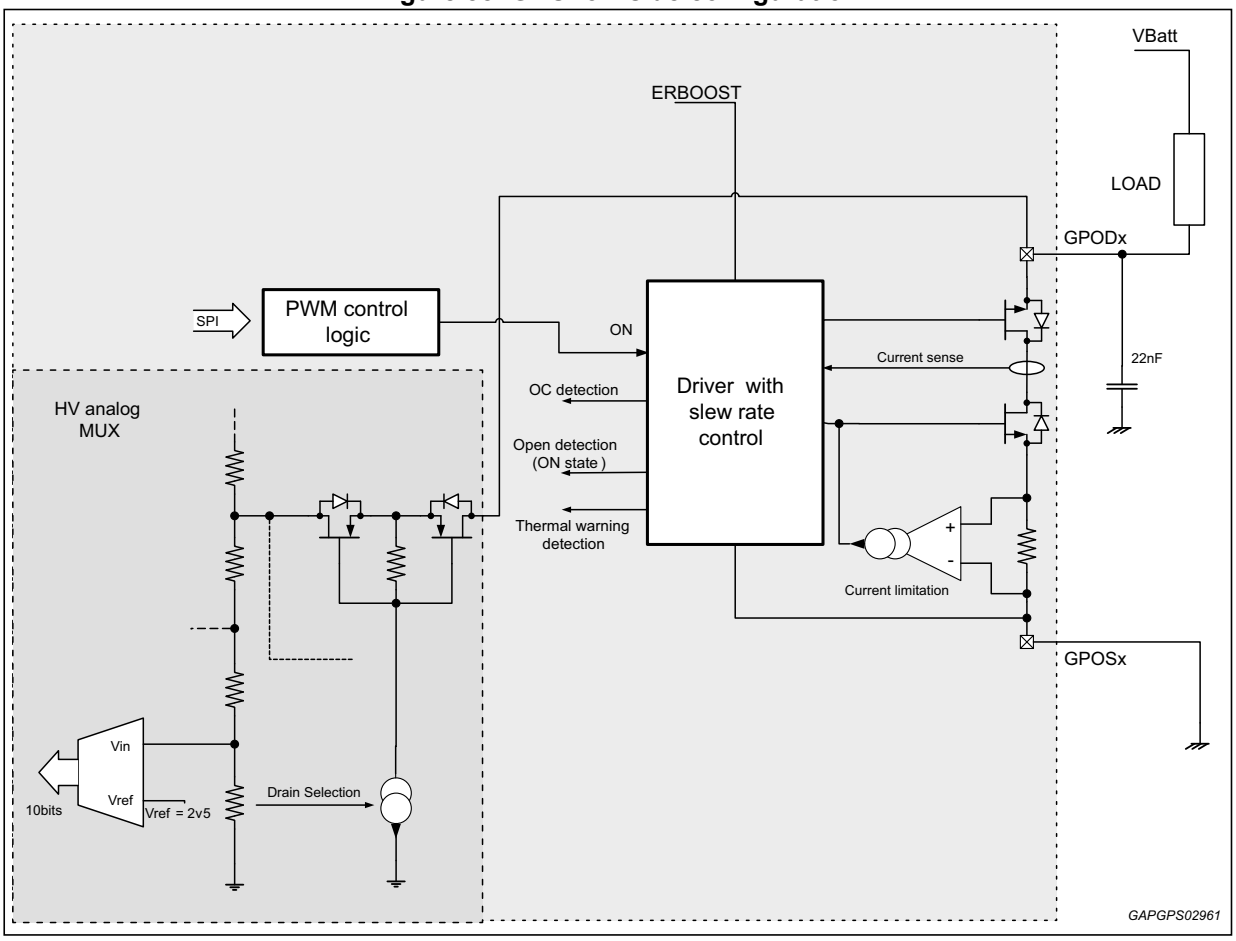
Low side

GPODx to the external load (that is connected to battery)

GPOSx to ground

x = 0, 1

Figure 88. GPO low side configuration



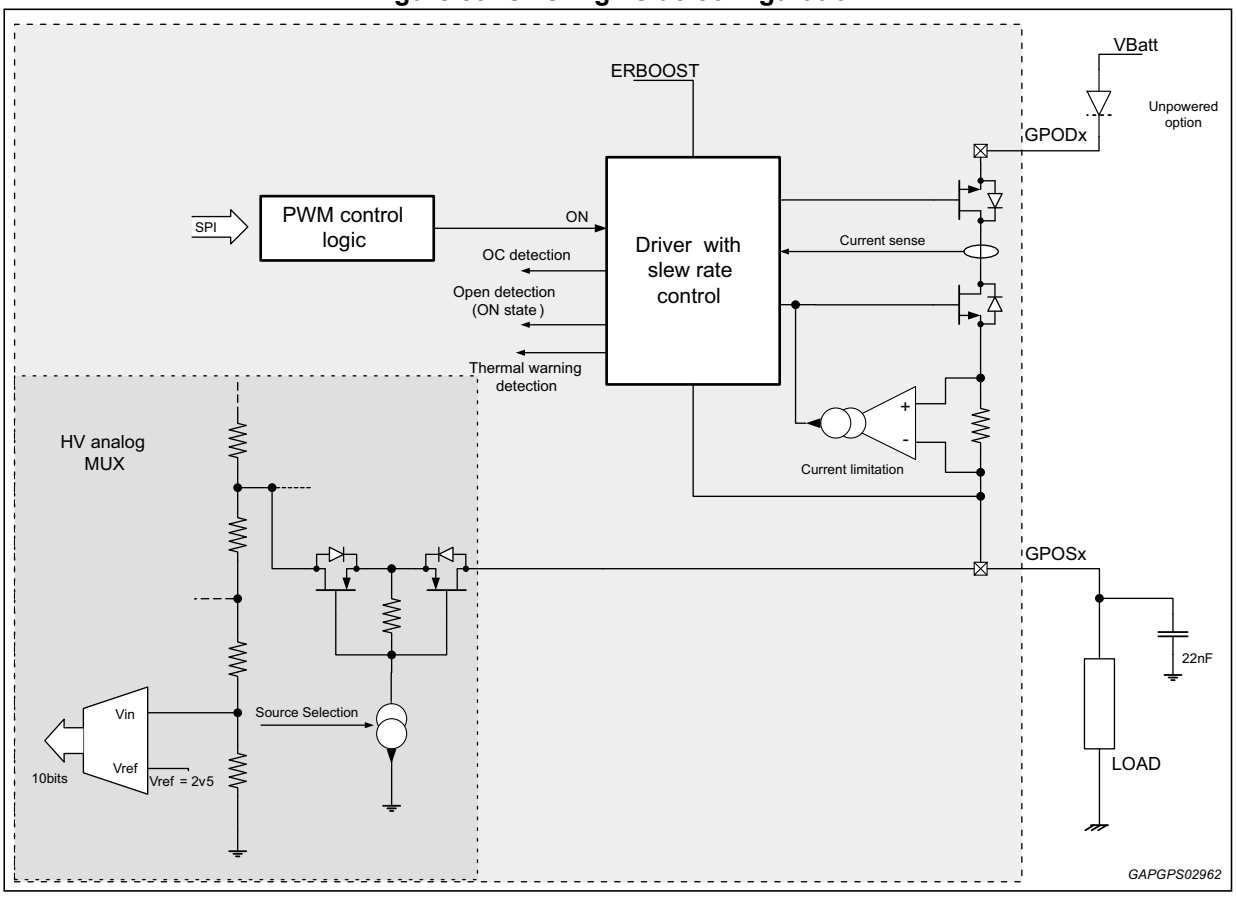
High side

GPODx to battery

GPOSx to the external load (that is connected to ground)

x = 0, 1

Figure 89. GPO high side configuration



GPOx are OFF by default.

Once they have been configured as high side or low side, they can be independently switched ON through GPOCTRLx register, GPOxPWM bit:

$43 GPOCTRL0 register	Config. in DIAG, SAFING, SCRAP, ARMING state
GPOxPWM bit[5:0]	PWM duty cycle, 1.6% per count

$44 GPOCTRL1 register	Config. in DIAG, SAFING, SCRAP, ARMING state
GPO1PWM bit[5:0]	PWM duty cycle, 1.6% per count

In case the channels are switched on without their configuration as high or low side, a fault bit (LAMPS_NOT_CONF in GPOFLTSR register) is set:

$46 GPOFLT register	
LAMPS_NOT_CONF bit 15	0= channels configured, activation permitted 1= channels not configured (default), activation not permitted

Output can work in a PWM configuration (125 kHz, 0-100% duty-cycle, depending on the GPOxPWM value).

Driver is kept OFF in case of 0% duty cycle programmed, full ON in case of 100% duty cycle programmed.

When both channels (it doesn't care HS or LS configuration) are used in PWM with the same duty-cycle, they are synchronized to provide parallel configuration capability.

Protections:

- Each channel can withstand -1V on the pin and +1V as reverse voltage across source and drain.
- Each channel is protected against short circuit
- Each channel is protected against thermal overload condition

All the diagnostics related to GPOx are reported in GPOFLTSR register

$46 GPOFLTSR	
GPO1TEMP, bit9 GPO0TEMP, bit4	0 = no fault 1 = fault
GPO1LIM, bit8 GPO0LIM, bit3	0 = no fault 1 = fault
GPO1OPN, bit7 GPO0OPN, bit2	0 = no fault 1 = fault

All faults (except thermal overload) are latched in the register until it is read.

Protection against short circuit is implemented by means of current limitation I_{LIM}, until thermal fault condition is detected.

$$I_{LIM} = 80\ mA \div 140\ mA$$

The limitation flag, GPOxLIM, is also asserted.

In case of thermal fault, the correspondent flag, GPO0TEMP, is asserted.

Thermal overload fault lasts until the over-temperature condition disappears: a hysteresis is applied for the cool down

$$T_{JSD} = 150°C \div 190°C$$

$$_{THYS\_JSD} = 5°C \div 15°C$$

The IC is able to detect open load in ON condition, comparing the current flowing into the output pin with a threshold

$$I_{Openload} = 3mA$$

If the current through the pin is lower than the threshold, the open load flag, GPOxOPN, is asserted.

For further diagnostics, IC provides the means to read via ADC the voltage on both drain and source pin for each GPOx driver. The registers involved in this operation are the four $3x DIAGCTRL_x → x=A, B, C, D

Case x = A

$3A DIAGCTRL_A	Config. in DIAG, SAFING, SCRAP, ARMING state
NEWDATA[19] new data available from the conversion	0 = cleared on read 1 = conversion finished
ADCREQ_A, bit [6:0] request (MOSI)	$2C = GPO D0 $2D = GPO S0 $2E = GPO D1 $2F = GPO S1
ADCREQ_A, bit [16:10] readout (through MISO),	$2C = GPO D0 $2D = GPO S0 $2E = GPO D1 $2F = GPO S1
ADCRES_A, bit [9:0]	10bit ADC result correspondent to the ADCREQ_A, bit [9:0]

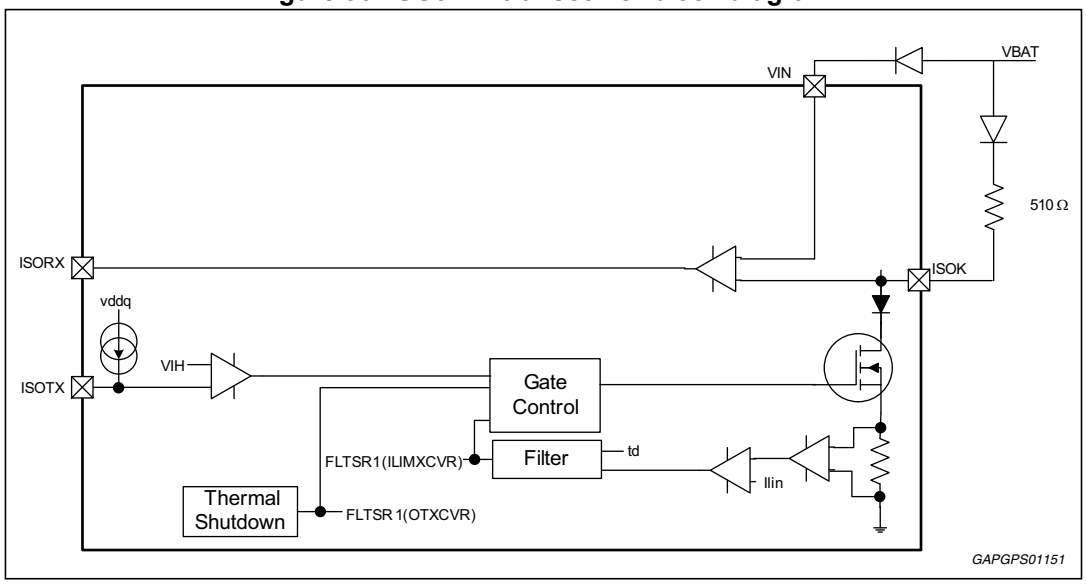
Once read the ADC measurement, to obtain the voltage value it is necessary to consider the divider ratio of the ADC.

Table 10. GPODx and GPOSx measurement of the value ratio ADC

Measurements	**Divider Ratio**				
	15:1	**10:1**	**7:1**	**4:1**	**1:1**
GPODx		√			
GPOSx		√			

13 ISO9141 transceiver

Figure 90. ISO9141 transceiver block diagram



ISOTX is pulled up to VDDQ to guarantee the output is disabled in case of an open load.

ISOK is an output pin, connected to BATTERY through an external resistor (510 Ω)

ISORX is an output pin referred to VDDQ. Value of this pin depends on the value of ISOK, and then on the value of ISOTX.

If ISOK is above ISOK input receiver threshold, V_{TH_REC}, ISORX is high, otherwise it is low.

$$V_{TH\_REC} = VIN*0.5 \div VIN*0.6 \text{ (typ = } VIN*0.55)$$

Protections:

- against short circuit
- current limitation detection, ILIM 50mA ÷ 100mA
- thermal shutdown

In $47 ISOFLTSR register the diagnostic results are reported:

$47 ISOFLTSR register	
ISOTEMP, bit1	0 = no fault 1 = fault
ISOLIM, bit0	0 = no fault 1 = fault

In current limit condition, the output stage runs until the thermal shutdown is reached.

Thermal shutdown switches OFF the output until the temperature falls down below limit threshold temperature, considering the hysteresis.

$$T_{JSD} = 150°C \div 190°C$$

$$T_{HYS\_JSD} = 5°C \div 15°C$$

14 System voltage diagnostic

IC is equipped with a 10 bit ADC running at 16 MHz.

Different measurements can be performed and then read via four dedicated commands, $3x_DIAGCTRL_x, x = A ÷ D.

All measurements can be addressed in one of these four registers, with the exception of:

DCS resistance measurement available in $3A_DIAGCTRL_A

Squib resistance measurement available in $3C_DIAGCTRL_C

In case of DCS resistance measurement, in $3A_DIAGCTRL_A is reported the current flowing through the pin and in $3B_DIAGCTRL_B is reported the voltage of the pin, as explained in *Section 11: DC sensor interface*.

In case of squib resistance measurement, result of ADC conversion is available in registers $3C_DIAGCTRL_C (current) and $3D_DIAGCTRL_D (voltage of the pin), see *Section 9.1.7: Squib resistance measurement*, and *Section 9.2.7*.

Structure of DIAGCTRL_A same for the other three registers.

$3A, DIAGCTRL_A

	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOSI					X	X	X	X	X	X	X	X	X	ADCREQ_A [6:0]						
MISO	NEWDATA_A	0	0	ADCREQ_A [6:0]							ADC_RES_A [9:0]									

To get four measurements, the four $3x, DIAGCTRL_x, x = A ÷ D commands have to be sent, regardless their order.

If the voltage to be measured requires a certain time to be stable, the requirement has to be done in advance. This is the case of:

Squib resistance measurement and diagnostic

DCS measurement

NEW_DATA_x =1 indicates that the conversion required is finished and the new data is available.

NEW_DATA_x is reported in GSW, elaborated in this way:

CONVRDY_0 = NEW_DATA_A or NEW_DATA_B

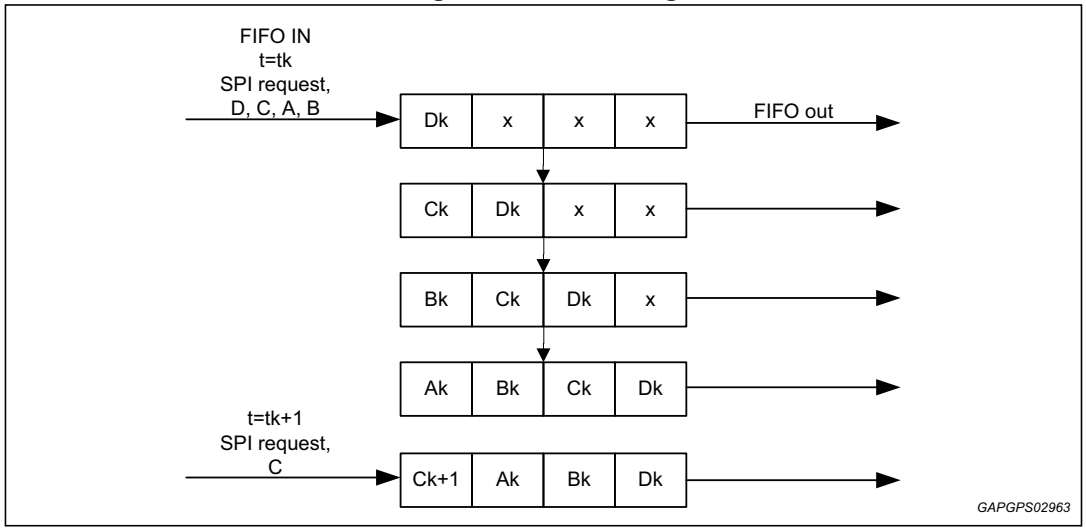
CONVRDY_1 = NEW_DATA_C or NEW_DATA_D

MISO BIT	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	**23**	**22**	**21**
MISO	SPIFLT	DEPOK	RSFLT	WDTDIS_S	ERSTATE	POWERFLT	FLT	CONVRDY2	CONVRDY1	ERR_WID	ERR_RID
GSW BIT	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**

Once the conversion has been read via SPI, bit NEW_DATA_x is cleared and results of conversion done, ADC_RES_A [9:0], is kept until a new conversion is available.

In case a new conversion command is received while a conversion is still running, a 4 stages FIFO queue is available in order to service all conversions commands with the integrated ADC.

Figure 91. FIFO filling



FIFO management, in a case of 4 conversion commands requested sequentially, is reported in the example above.

In case a new conversion is required (ie on DIAGCTRL_C) while a conversion is still running, the new request (C_{k+1}) overwrites the previous one (C_k) and a new conversion request is located at the end of the queue.

Voltage measurements require a proper scaling, as summarized here below:

Table 11. Voltage measurements

Measurements	Divider Ratio					
	15:1	10:1	7.125:1	7:1	4:1	1:1
VER	√					
ERBOOST	√					
VSF	√					
SSxy	√					
SFx	√					
GPODx		√				
GPOSx		√				
VIN		√				
VBATMON		√				
WAKEUP		√				
VSUP				√		

Table 11. Voltage measurements (continued)

Measurements	Divider Ratio					
	15:1	10:1	7.125:1	7:1	4:1	1:1
WDTDIS				√		
RSUx					√	
DCSx			√			
VDD5					√	
VDD3V3					√	
VINT3V3					√	
Band-gap (BGR/BGM)						√
TEMP						√

14.1 ADC algorithm

10 bit data are internally filtered.

The number of samples that is filtered depends on set-up of VMEAS bit (default 4 samples) in SYS_CFG register,

$01 SYS_CFG register	Config in INIT state only
VMEAS, bit[6:5]	00 = 4 samples 01 = 16 samples 10 = 8 samples 11 = 2 sample

Reference voltage for ADC is 2.5V.

Conversion times takes into account several factors:

- Number of measurements loaded into the queue
- Numbers of sample taken for each measurement
- Settling time

15 Temperature sensor

The aim of this sensor is to have a reference for the average junction temperature on the silicon on the surface.

The sensor is localized far from the power stages.

Temperature is available through ADC conversion, ADRREQ_x = $0A.

Once the temperature is available, the formula for the conversion is:

$$T(°C) = 180 - \left[\left(\frac{220}{1.652}\right) * \left(\frac{ADC_{REF\_hi}}{2^{ADC_{RES}}} * DIAGCTRLn(ADCRESn)\right) - 0.739\right]$$

$$@\ DIAGCTRLn(ADCREQn) = 0A_{hex}$$

GAPGPS02964

16 Footprint

L9678 and the other two devices of the same family have a concentric footprint, see next figures:

Figure 92. Footprint L9678-L9680

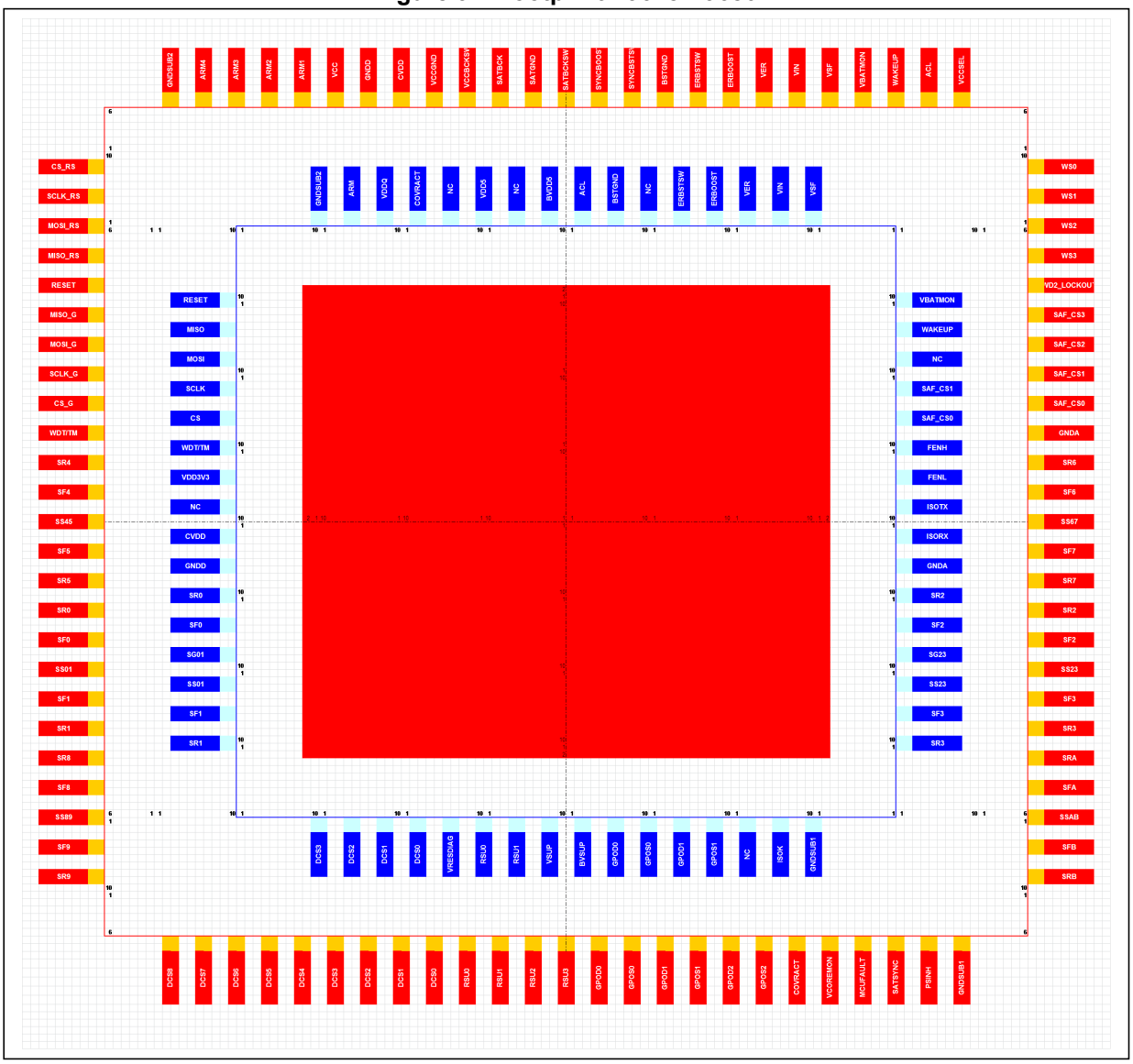
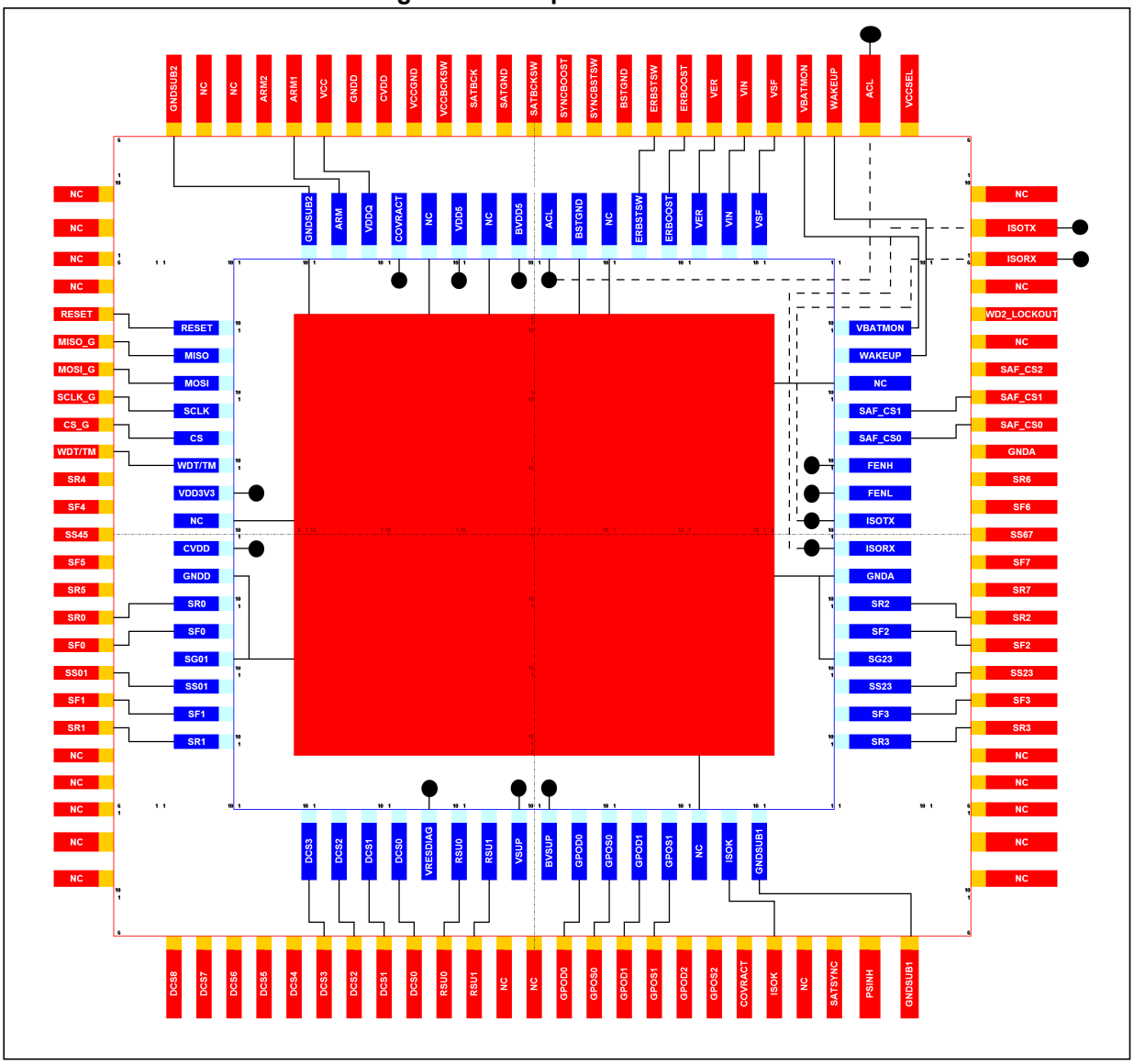


Figure 93. Footprint L9678-L9679



Appendix A Energy reserve capacitor

Energy reserve capacitor stores the necessary energy to operate the Airbag ECU during loss of battery.

System operating requirements influence device selection and calculations. The following example makes general operating assumptions and changing the assumptions may effect calculations and results.

During loss of battery operation, energy reserve operation can be mechanized as shown. To continue the analysis, system functional operating assumptions must be determined. These assumptions are shown in the drawing below. Operation assumes three states, sensing, deploy and shutdown. During sensing state, all functions operate normally. In deploy state, all functions remain operational and all squib deployment channels are fired. The final state, shutdown, reduces operation to only the microcontroller.

Figure 94. Blocks active in Autarchy mode

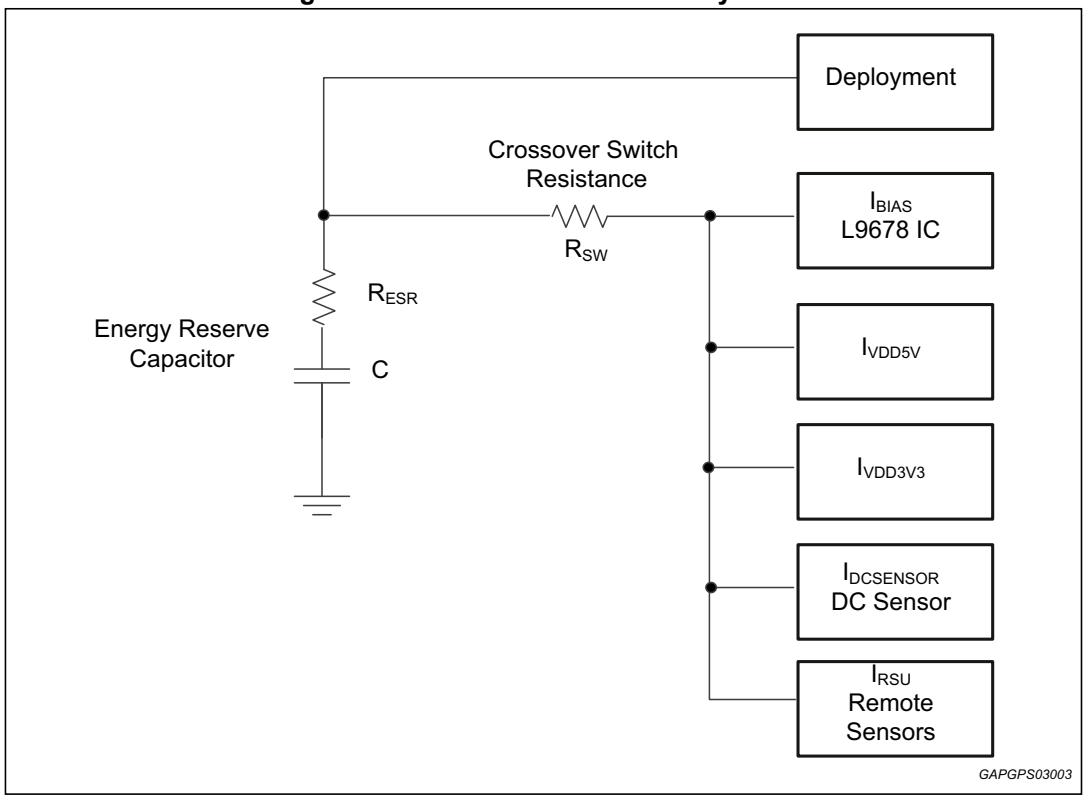
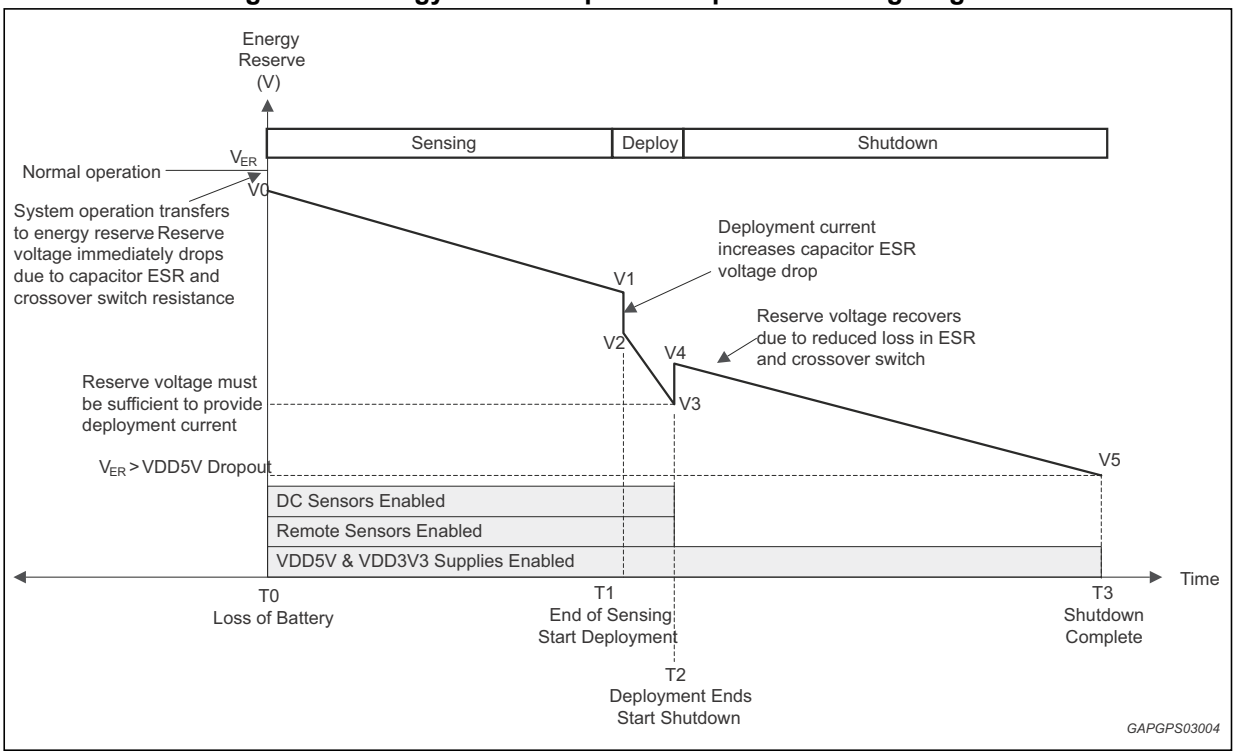


Figure 95. Energy reserve capacitor depletion - timing diagram



At t = T0 = 0, $V_0 = V_{ER} - I_{SYS} \bullet (R_{ESR} + R_{CO})$

Where,

V_{ER} = Energy reserve voltage just prior to loss of battery detection and crossover operation

I_{SYS} = System current consumption, L96xx bias, Voltage regulators, RSUs, DC Sensors

R_{ESR} = Energy capacitor's ESR

R_{CO} = Crossover Switch Resistance

$$I_{SYS} = C * \left(\frac{V_0 - V_1}{T_1 - T_0} \right)$$

GAPGPS03005

$$V_1 = V_0 - \frac{I_{SYS} * (T_1 - T_0)}{C}$$

GAPGPS03006

Equation 1

$$V1 = V_{ER} - I_{SYS} * (R_{ESR} + R_{CO}) - \frac{I_{SYS} * T_1}{C}$$

GAPGPS03007

Deployment begins at T1, thus increasing energy reserve current and effects due to ESR

Equation 2 $\mathbf{V2 = V1 - I_{DEPLOY} \bullet R_{ESR}}$

Where IDEPLOY = total deployment current controlled by L96xx

Substituting (1) into (2):

Equation 3

$$V2 = V_{ER} - I_{SYS} * (R_{ESR} + R_{CO}) - \frac{I_{SYS} * T_1}{C} - I_{DEPLOY} * R_{ESR}$$

GAPGPS03008

During deployment phase, reserve voltage behavior is characterized as:

$$I_{SYS} + I_{DEPLOY} = C * \frac{V2 - V3}{T2 - T1}$$

GAPGPS03009

Equation 4

$$V3 = V2 - \frac{T_{DEPLOY} * (I_{SYS} + I_{DEPLOY})}{C}$$

GAPGPS03010

Substituting (3) into (4):

$$V3 = (V_{ER} - I_{SYS} * (R_{ESR} + R_{CO}) - \frac{I_{SYS} * T_1}{C} - I_{DEPLOY} * R_{ESR}) - \frac{T_{DEPLOY} * (I_{SYS} + I_{DEPLOY})}{C}$$

GAPGPS03011

Once deployment is complete, the airbag module enters its final state, shutdown. Reserve current is reduced causing reserve voltage to increase due to less loss in capacitor CER and Cross over switch. The change in reserve voltage is calculated as:

Equation 5

$$V4 = V3 + I_{DEPLOY} * R_{ESR} + (I_{DCSENSOR} + I_{RSU}) * (R_{ESR} + R_{CO})$$

GAPGPS03012

Equation 6

$$V4 = (V_{ER} - I_{SYS} * (R_{ESR} + R_{CO}) - \frac{(I_{SYS} * T_1 + I_{SYS} * T_{DEPLOY} + I_{DEPLOY} * T_{DEPLOY})}{C} + - I_{DEPLOY} * R_{ESR} + I_{DEPLOY} * R_{ESR} + (I_{DCSENSOR} + I_{RSU}) * (R_{ESR} + R_{CO})$$

GAPGPS03013

In equation (5) above, the system disables current to all deployment drivers, DC sensor and Remote Sensor Interfaces.

During shutdown phase, only Voltage regulator and device (L96xx) bias current is needed from reserve.

To complete energy reserve capacitor estimate, the analysis must assume a final reserve voltage requirement. In the study, energy reserve must be higher than VDDx dropout voltage where VDDx is the supply of the microcontroller.

By assuming this requirement, the system is designed to operate for the desired reserve time.

Equation 7

$$V5 > VDDx_{DROPOUT}$$

Reserve voltage behavior follows as:

Equation 8

$$I_{SHUTDOWN} = C * \frac{V4 - V5}{T3 - T2}$$

GAPGPS03014

Where $I_{SHUTDOWN} = I_{VDD} + I_{BIAS}$

Re-arranging (8)

$I_{SHUTDOWN} \bullet (T_3 - T_2)\ C \bullet V4 - V5)$

Substituting (6) and (7):

$$I_{SHUTDOWN} * (T_3 - T_2) = C * ((V_{ER} - I_{SYS} * (R_{ESR} + R_{CO}) - \frac{(I_{SYS} * T_1 + I_{SYS} * T_{DEPLOY} + I_{DEPLOY} * T_{DEPLOY})}{C} +$$

$$- I_{DEPLOY} * R_{ESR} + I_{DEPLOY} * R_{ESR} + (I_{DCSENSOR} + I_{RSU}) * (R_{ESR} + R_{CO}) - VDDx_{DROPOUT})$$

GAPGPS03016

Simplify and arrange

$$C = \frac{I_{SHUTDOWN} * (T_3 - T_2) + (I_{SYS} * T_1 + I_{SYS} * T_{DEPLOY} + I_{DEPLOY} * T_{DEPLOY})}{V_{ER} - I_{SYS} * (R_{ESR} + R_{CO}) + (I_{DCSENSOR} + I_{RSU}) * (R_{ESR} + R_{CO}) - VDDx_{DROPOUT}}$$

GAPGPS03017

Revision history

Table 12. Document revision history

Date	Revision	Changes
15-Apr-2014	1	Initial release.
06-Mar-2015	2	Modified in *Section 9.1: Low level on page 100* for "**$04 SYS_STATE**" 010=DIAG in 001=DIAG.
12-Sep-2016	3	In *Section 5.7: VSF linear regulator on page 47* the paragraph 'VSF stability requires an external small capacitor 2.9 µF ÷ 14 µF' is changed in 'VSF .....2.9 nF ÷ 14 nF'. The section '**8.1.5 Deployment current monitor**' has been removed. Modified '**$38 LPDIAGREQ**' table on page 104. Modified '**$37 LPDIAGSTAT**' table on pages 111, 112, 127 and 132. The section '**9.1.11 High side driver diagnostic**' has been removed.