

# Programmable solenoid controller

The PT2000 is a SMARTMOS programmable gate driver IC for precision solenoid control applications, which makes the component very flexible and relieves the main microcontroller from the heavy task of the actuator control. The chip integrates six microcores used to control, seven external MOSFET high-side pre-drivers, eight external MOSFET low-side pre-drivers (two of them with higher switching frequency can be used for DC/DC converters), an integrated end of injection detection, six current measurements, and diagnostics for both the high-side and low-side.

PT2000 includes two internal regulators with overvoltage and undervoltage monitoring and protection,  $V_{CCP}$  fed from the battery line supplying the pre-driver section and  $V_{CC2P5}$  fed from an external 5.0 V generating supply for the digital block.

Interface with the MCU is achieved via serial peripheral interface (SPI) and 16 configurable I/O signals. I/O are supplied by an external 3.3 V or 5.0 V regulator connected to  $V_{CCIO}$ .

These features along with cost effective packaging, make the PT2000 ideal for power train engine control applications.

## Features

- Battery voltage range,  $5.0\text{ V} < V_{BATT} < 72\text{ V}$
- Battery and boost voltage monitoring
- Pre-drive operating voltage up to 72 V
- Seven high-side/ eight low-side pre-drive PWM capability up to 100 kHz-30 nC
- All pre-drivers have four selectable slew rates
- Eight selectable, pre-defined  $V_{DS}$  monitoring thresholds
- Measurement function for end of injection detection
- Encryption for microcode protection
- Integrated 1.0 MHz back-up clock



## Applications

- Automotive (12 V), truck and industrial (24 V) power train
- Diesel and gasoline direct injection (three banks)
- Transmission
- Valve control

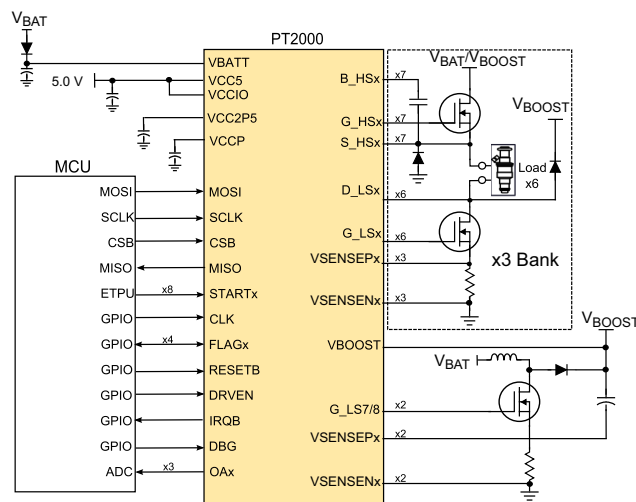


Figure 1. PT2000 simplified application diagram

# Table of Contents

1	Orderable parts	4
1.1	Cipher key	4
2	Internal block diagram	5
3	Pin connections	6
4	Functional description	10
4.1	Introduction	10
4.2	Features	10
5	Electrical characteristics	11
5.1	Maximum ratings	11
5.2	Power supply electrical characteristics	13
5.3	High-side pre-driver electrical characteristics	16
5.4	Low-side (LS1-LS6) pre-driver electrical characteristics	17
5.5	Low-side high-speed (LS7-LS8) pre-driver electrical characteristics	19
5.6	High-side VDS VSRC monitoring electrical characteristics	20
5.7	Low-side VDS VSRC monitoring electrical characteristics	22
5.8	Load bias electrical characteristics	23
5.9	Current measurement electrical characteristics	24
5.9.1	Current measurement for positive current	24
5.9.2	Current measurement for negative currents	26
5.10	Analog output (OAx) electrical characteristics	28
5.11	Clock / PLL electrical characteristics	29
5.12	Digital input/output electrical characteristics	30
5.13	Serial peripheral interface electrical characteristics	32
6	Functional block description	34
6.1	Power supplies	34
6.1.1	VCC5	34
6.1.2	VCCIO	34
6.1.3	VCC2P5 regulator	34
6.1.3	VCC2P5 regulator	34
6.1.4	VCCP regulator	35
6.1.5	Battery voltage monitor	36
6.1.6	Boost voltage monitor	36
6.1.7	Charge pump	38
6.2	Clock subsystem	38
6.3	High-side pre-driver	39
6.3.1	High-side pre-driver slew rate control	40
6.3.2	Safe state of high-side pre-driver	40
6.4	High-side VDS and VSRC monitor	42
6.4.1	HS1, 3, 5, 7 VDS monitoring	42
6.4.2	HS2, 4, 6 VDS monitoring	43
6.5	Low-side pre-driver (LS1-6)	44
6.5.1	Low-side pre-driver slew rate control	45
6.5.2	LS1 - LS6 VDS monitor	46
6.6	Low-side pre-driver for DC/DC converter (LS7 and LS8)	47
6.6.1	Low-side pre-driver slew rate control (LS7 and LS8)	48
6.6.2	Low-side VDS monitor D_Is7/D_Is8 for DC/DC	48

6.7	Current measurement	49
6.7.1	General purpose current measurement block	49
6.7.2	Current measurement for DC/DC	52
6.8	OA_x output pins, multiplexer and T & H	53
6.8.1	General features	53
6.8.2	OA_2 Pin digital I/O function	55
6.8.3	OAx output offset and offset error	56
7	Functional device operation	57
7.1	Power-up/down sequence	57
7.1.1	Power-up sequence of VCC5, VCC2P5, and reset	57
7.1.2	Power-up sequence VCCP and bootstrap capacitors	58
7.2	DC/DC converter control (LS7/8)	61
7.2.1	General description	61
7.3	Device clock manager and PLL init	63
7.4	SW initialization flow	63
7.4.1	Power supply, reset, and clock	63
7.4.2	SPI configuration	64
7.4.3	Clock monitor, flash enable, and DrvEn	64
7.5	BIST	64
7.5.1	MBIST	65
7.5.2	LBIST	65
7.6	Reset sources	65
7.7	Cipher unit	66
7.8	Ground connections	66
7.9	Shutoff path via the DrvEn pin	66
8	Digital core	68
8.1	Logic channels description	68
8.1.1	Microcores	69
8.1.2	Dual microcore arbiter	70
8.1.3	Signature unit	72
8.1.4	SPI backdoor	72
8.1.5	CRAM	73
8.1.6	DRAM	73
8.2	Serial peripheral interface	74
8.2.1	SPI read access	75
8.2.2	SPI write access	76
8.2.3	SPI protocol	77
8.3	SPI address map	79
8.3.1	Selection register (3FFh)	89
8.3.2	Configuration register	90
8.3.3	IO configuration registers	100
8.3.4	Main configuration registers	117
8.3.5	Diagnostics configuration registers	132
9	Typical applications	139
9.1	Application diagram: 3 bank, 6 cylinder with DC/DC	139
9.2	Application diagram: 3 bank, 3 cylinder (full overlap) with DC/DC	140
10	Packaging	141
10.1	Package mechanical dimensions	141
11	Reference section	145
12	Revision history	146

# 1 Orderable parts

Table 1. Orderable part variations

Part number	Temperature (T <sub>A</sub> )	Package
MC33PT2000AF <sup>(1)</sup>	-40 °C to 125 °C	80-pin LQFP with exposed pad

Notes

1. To order parts in tape and reel, add the R2 suffix to the part number.

## 1.1 Cipher key

Contact a NXP sales representative to obtain devices with a specific encryption key and the associated code encryptor.

## 2 Internal block diagram

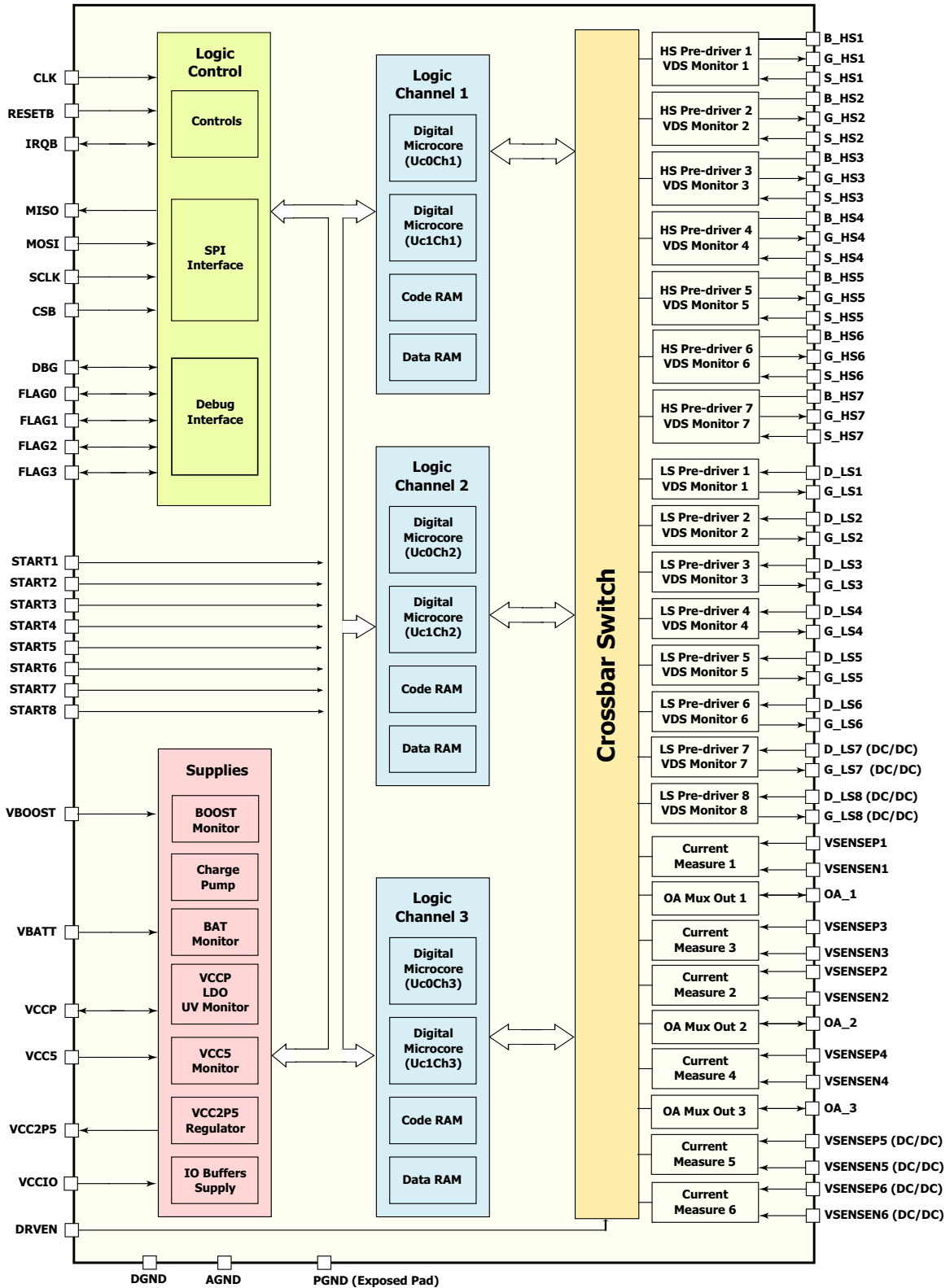


Figure 2. PT2000 simplified internal block diagram

### 3 Pin connections

Transparent top view

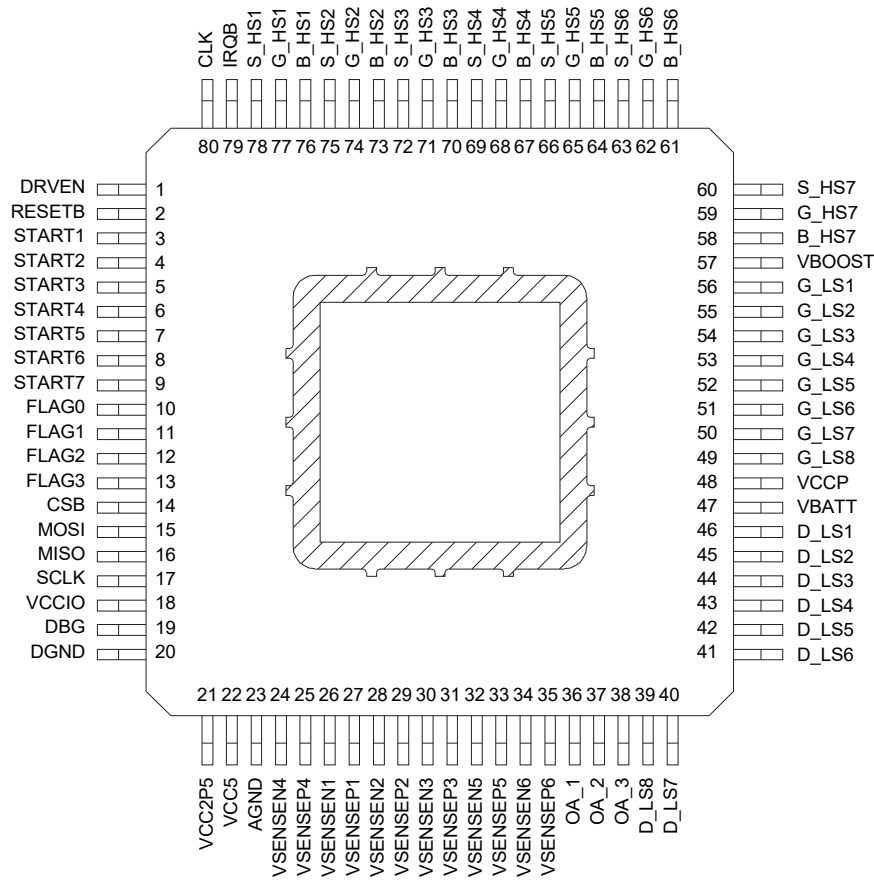


Figure 3. PT2000 pin connections

Functional descriptions of many of these pins can be found in the [Functional block description](#) section beginning on [page 34](#).

Table 2. PT2000 pin definitions (2), (3), (4)

Pin	Pin name	Pin function	Pull configuration	Definition
1	DRVEN	Input	Weak PD	Driver enable input
2	RESETB	Input	Weak PU	Reset pin
3	START1	Input/output	PU/PD Configurable	Trigger pin actuator 1 / Flag_bus(5)
4	START2	Input/output	PU/PD Configurable	Trigger pin actuator 2 / Flag_bus(6)
5	START3	Input/output	PU/PD Configurable	Trigger pin actuator 3 / Flag_bus(7)
6	START4	Input/output	PU/PD Configurable	Trigger pin actuator 4 / Flag_bus(8)
7	START5	Input/output	PU/PD Configurable	Trigger pin actuator 5 / Flag_bus(9)

**Table 2. PT2000 pin definitions (2), (3), (4)(continued)**

Pin	Pin name	Pin function	Pull configuration	Definition
8	START6	Input/output	PU/PD Configurable	Trigger pin actuator 6 / Flag_bus(10)
9	START7	Input/output	PU/PD Configurable	Trigger pin actuator 7 / Flag_bus(11)
10	FLAG 0	Input/output	Weak PD	Flag_bus(0) (general purpose I/O)/command for external free-wheeling MOSFET pre-driver 1
11	FLAG 1	Input/output	Weak PD	Flag_bus(1) (general purpose I/O)/command for external free-wheeling MOSFET pre-driver 2
12	FLAG 2	Input/output	Weak PD	Flag_bus(2) (general purpose I/O)/command for external free-wheeling MOSFET pre-driver 3
13	FLAG 3	Input/output	Weak PD	Flag_bus(3) (general purpose I/O)/command for external free-wheeling MOSFET pre-driver 4
14	CSB	Input	PU	SPI chip select
15	MOSI	Input	Weak PU	SPI slave input
16	MISO	Output	—	SPI slave output
17	SCLK	Input	Weak PU	SPI clock
18	VCCIO	Input	—	Digital I/O voltage supply 3.3 V or 5.0 V (supplied externally)
19	DBG	Input/output	Weak PU	Debug port / Flag_bus (15)
20	DGND	Ground	—	Digital ground
21	VCC2P5	Output	—	Internal 2.5 V voltage regulator decoupling
22	VCC5	Input	—	Power supply 5.0 V (supplied externally)
23	AGND	Ground	—	Analog ground
24	VSENSEN4	Input/output	PU/PD Configurable	Current sense input comparator - / Start8 / Flag(12)
25	VSENSEP4	Input/output	Weak PD	Current sense input comparator + /Flag(4) (general purpose I/O)
26	VSENSEN1	Input	—	Current sense input comparator 1 -
27	VSENSEP1	Input	—	Current sense input comparator 1 +
28	VSENSEN2	Input	—	Current sense input comparator 2 -
29	VSENSEP2	Input	—	Current sense input comparator 2 +
30	VSENSEN3	Input	—	Current sense input comparator 3 -
31	VSENSEP3	Input	—	Current sense input comparator 3 +
32	VSENSEN5	Input	—	DC-DC current sense input comparator -
33	VSENSEP5	Input	—	DC-DC current sense input comparator +
34	VSENSEN6	Input	—	DC-DC current sense input comparator -
35	VSENSEP6	Input	—	DC-DC current sense input comparator +
36	OA_1	Output	—	Analog output 1
37	OA_2	Input/output	Weak PD	Analog output 2/Flag_bus (14)
38	OA_3	Output	—	Analog output 3
39	D_LS8	Input	—	Drain pin low-side MOSFET for DC/DC converter
40	D_LS7	Input	—	Drain pin low-side MOSFET for DC/DC converter
41	D_LS6	Input	—	Drain pin low-side MOSFET actuator 6
42	D_LS5	Input	—	Drain pin low-side MOSFET actuator 5

**Table 2. PT2000 pin definitions (2), (3), (4)(continued)**

Pin	Pin name	Pin function	Pull configuration	Definition
43	D_LS4	Input	—	Drain pin low-side MOSFET actuator 4
44	D_LS3	Input	—	Drain pin low-side MOSFET actuator 3
45	D_LS2	Input	—	Drain pin low-side MOSFET actuator 2
46	D_LS1	Input	—	Drain pin low-side MOSFET actuator 1
47	VBATT	Input	—	Battery voltage input
48	VCCP	Input/output	—	Output: Internal 7.0 V voltage regulator Input: External 7.0 V voltage regulator (supplied externally)
49	G_LS8	Output	—	Gate pin low-side high speed MOSFET can be used for DC/DC converter
50	G_LS7	Output	—	Gate pin low-side high speed MOSFET can be used for DC/DC converter
51	G_LS6	Output	—	Gate pin low-side MOSFET actuator 6
52	G_LS5	Output	—	Gate pin low-side MOSFET actuator 5
53	G_LS4	Output	—	Gate pin low-side MOSFET actuator 4
54	G_LS3	Output	—	Gate pin low-side MOSFET actuator 3
55	G_LS2	Output	—	Gate pin low-side MOSFET actuator 2
56	G_LS1	Output	—	Gate pin low-side MOSFET actuator 1
57	VBOOST	Input	—	Boost voltage and drain pin for boost pre-drivers
58	B_HS7	-	—	Bootstrap pin high-side MOSFET 7
59	G_HS7	Output	—	Gate pin high-side MOSFET 7
60	S_HS7	Input	—	Source pin high side MOSFET 7
61	B_HS6	-	—	Bootstrap pin Boost MOSFET 6
62	G_HS6	Output	—	Gate pin Boost MOSFET 6
63	S_HS6	Input	—	Source pin Boost MOSFET 6
64	B_HS5	-	—	Bootstrap pin high-side MOSFET 5
65	G_HS5	Output	—	Gate pin high-side MOSFET 5
66	S_HS5	Input	—	Source pin high side MOSFET 5
67	B_HS4	-	—	Bootstrap pin boost MOSFET 4
68	G_HS4	Output	—	Gate pin boost MOSFET 4
69	S_HS4	Input	—	Source pin boost MOSFET 4
70	B_HS3	-	—	Bootstrap pin high-side MOSFET 3
71	G_HS3	Output	—	Gate pin high-side MOSFET 3
72	S_HS3	Input	—	Source pin high-side MOSFET 3
73	B_HS2	-	—	Bootstrap pin boost MOSFET 2
74	G_HS2	Output	—	Gate pin boost MOSFET 2
75	S_HS2	Input	—	Source pin boost MOSFET 2
76	B_HS1	-	—	Bootstrap pin high-side MOSFET 1
77	G_HS1	Output	—	Gate pin high-side MOSFET 1
78	S_HS1	Input	—	Source pin high-side MOSFET 1
79	IRQB	Input/output	Weak PD	Interrupt output/Flag_bus (13)
80	CLK	Input	Weak PU	Clock pin (low-frequency reference for internal PLL)



**Table 2. PT2000 pin definitions (2), (3), (4)(continued)**

Pin	Pin name	Pin function	Pull configuration	Definition
ePAD	PGND	Ground	—	Power ground (to be soldered on GND PCB)

## Notes

2. External 7.0 V is required in case the typical battery voltage is 24 V ([See External VCCP \(vccp\\_ext\\_en='1'\)](#) on page 35).
3. Except for supply and ground, it is guaranteed by design unused pins can be kept open without any impact on the device.
4. Unused VSENSEPx and VSENSEPx pins can both be connected to GND.

**Table 3. Resistor types**

Pin type	Description
PU	Pull-up to VCCIO (nominal value: 120 k $\Omega$ )
Weak PU	Weak pull-up to VCCIO (nominal value: 480 k $\Omega$ )
PD	Pull-down to AGND (nominal value: 120 k $\Omega$ )
Weak PD	Weak pull-down to AGND (nominal value: 480 k $\Omega$ )

# 4 Functional description

## 4.1 Introduction

The PT2000 is a mixed signal IC for engine injector and electrical valve control, which provides a cost effective, flexible, and smart, high-side and low-side MOSFET gate driver. The device includes both individual charge pump outputs for each high-side pre-driver and high-voltage DC/DC converter pre-driver. Gate drive, diagnostics, and protection against external faults, are managed through six independent and concurrent digital microcores. Each of the three logic channels including two microcores and their own code RAM and data RAM. The internal microcode is protected against theft via encryption and corruption via check sums. Those microcores are optimized to control power MOSFET with a small latency time. The PT2000 can control three banks of two injectors each or three banks with one injector per bank for full overlap,

## 4.2 Features

### High-side and low-side pre-drivers

- Seven high-side pre-drivers for logic level N-channel MOSFETs using four programmable slew rates
- Six low-side pre-drivers for logic level N-channel MOSFETs using four programmable slew rates
- Integrated bootstrap circuitry for each high-side pre-driver
- Integrated charge pump circuitry for each high-side pre-driver with 100% duty cycle capability
- Configurable automatic freewheeling capability between high-side and low-side

### DC/DC converter

- Two low-side pre-driver, for a logic level N-channel MOSFET, can be optionally dedicated to providing a boost DC-DC converter with four programmable slew rates
- Three different control modes to reduce power dissipation (manual, hysteretic, resonant)

### Current measurement

- Four independent current measurement blocks
- Two current measurements (channel 5 and 6) are optionally configurable to support DC/DC converters

### Diagnostics and monitoring

- $V_{DS}$  and  $V_{SRC}$  monitoring (programmable values) for fault protection and diagnostics
- $V_{BOOST}$  monitoring
- $V_{BAT}$  monitoring
- Temperature monitoring

### Integrated end of injection detection

- Accurate detection of end of injection for each high-side source and low-side drain without any external component needed.

### Power supplies

- Integrated 7.0 V linear regulator (VCCP) for the HS/LS gate power supply <sup>(2)</sup>
- Integrated 2.5 V linear regulator (VCC2P5) for the digital core supply based on the VCC5 input supply
- External 5.0 V supply (VCC5)
- Selectable VCCIO external supply (5.0 V or 3.3 V) for digital I/O

### Digital block

- Six digital microcores, each with their own ALU, and full access to the system crossbar switch
- Three memory banks: 1024 x 16-bit of code RAM with built-in error detection and 64 x 16-bit of data RAM
- Memory BIST and Logic BIST activated by the SPI, with pass/fail status

### Control interface

- 16-bit slave SPI up to 10 MHz - two protocols - programmable slew rate
- 16 general purpose digital IOs able to sustain up to 36 V
- Independent direct pre-driver inhibition input for safety purposes

# 5 Electrical characteristics

## 5.1 Maximum ratings

**Table 4. Maximum ratings**

All voltages are with respect to ground unless otherwise noted. Exceeding these ratings may cause a malfunction or permanent damage to the device.

Symbol	Ratings	Min.	Max.	Unit	Notes
<b>Electrical ratings</b>					
V <sub>BOOSTMAX</sub>	DC voltage at VBOOST	0	72	V	
V <sub>BATT</sub>	DC voltage at VBATT	-0.3	72	V	
V <sub>CC5</sub>	DC voltage at VCC5	-0.3	36	V	
V <sub>CCIO</sub>	DC voltage at VCCIO	-0.3	36	V	
V <sub>CC2P5</sub>	DC voltage at VCC2P5	-0.3	3.0	V	
V <sub>DIG</sub>	DC voltage at CLK, MISO, MOSI, SCLK, CSB, IRQB, RESETB	-0.3	36	V	
V <sub>DRV_EN</sub>	DC voltage at DRVEN	-0.3	36	V	
V <sub>STARTX</sub>	DC voltage at STARTx	-0.3	36	V	
V <sub>FLAGX</sub>	DC voltage at FLAGx	-0.3	36	V	
V <sub>START8_VSENSE</sub>	Start8 voltage of pins multiplexed with V <sub>SENSE</sub> N4	-1.0 -1.0	18 36	V	(5)
V <sub>FLAG4_VSENSE</sub>	Flag4 voltage of pins multiplexed with V <sub>SENSE</sub> P4	-2.5 -2.5	18 36	V	(5)
V <sub>DBG</sub>	DC voltage at DBG	-0.3	36	V	
V <sub>OA_OUTX</sub>	DC voltage at OA_1, OA_2, OA_3	-0.3	36	V	
V <sub>DGND</sub>	DC voltage at DGND	-0.3	0.3	V	
V <sub>AGND</sub>	DC voltage at AGND	-0.3	0.3	V	
V <sub>CCP</sub>	DC voltage at VCCP	-0.3	9.0	V	
V <sub>S_HSx</sub>	S_HSx • DC voltage • Transients t < 800 ns • Transients t < 400 ns	-3.0 -6.0 -8.0	V <sub>BOOSTMAX</sub> V <sub>BOOSTMAX</sub> V <sub>BOOSTMAX</sub>	V	— (6) (6)
V <sub>B_HSx</sub>	B_HSx • V <sub>BATT</sub> - V <sub>B_HSx</sub> must not exceed 40 V • Transients t < 800 ns • Transients t < 400 ns	-0.3 -2.0 -4.0	V <sub>S_HSx</sub> + V <sub>BS_HSx_CL</sub>	V	— (6) (6)
V <sub>G_HSx</sub>	DC voltage at G_HSx	V <sub>S_HSx</sub> - 0.3	V <sub>B_HSx</sub> + 0.3	V	
V <sub>G_LSx</sub>	G_LSx • DC voltage • Transients t < 5.0 μs; V <sub>CCP_MAX</sub> = 8.0 V; energy of pulses < 0 V or > V <sub>CCP</sub> is limited to 2.0 μJ due to capacitive coupling	-0.3 -1.5	V <sub>CCP</sub> + 0.3 V <sub>CCP</sub> + 1.5	V	— (6)
V <sub>D_LSx</sub>	D_LSx • DC voltage • Transients t < 400 ns	-3.0 -8.0	75 75	V	— (6)

**Table 4. Maximum ratings (continued)**

All voltages are with respect to ground unless otherwise noted. Exceeding these ratings may cause a malfunction or permanent damage to the device.

Symbol	Ratings	Min.	Max.	Unit	Notes
$V_{VSENSEN1/2/3}$	DC voltage at VSENSEN1/2/3 <ul style="list-style-type: none"> <li>• Static at VCC5 &lt; 10 V</li> <li>• Dynamic for max 5.0 <math>\mu</math>s, 1.0 kHz repetition rate at VCC5, 5.25 V</li> <li>• Dynamic for max 1.0 <math>\mu</math>s at VCC5 &lt; 5.25 V</li> </ul>	-1.0 -5.0 -15	1.0 5.0 15	V	— (6) (6)
$V_{VSENSEP1/2/3}$	DC voltage at VSENSEP1/2/3 <ul style="list-style-type: none"> <li>• DC voltage at VCC5 &lt; 10 V</li> <li>• Dynamic for max 5.0 <math>\mu</math>s, 1.0 kHz repetition rate at VCC5 &lt; 5.25 V</li> <li>• Dynamic for max 1.0 <math>\mu</math>s at VCC5 &lt; 5.25 V</li> </ul>	-2.5 -5.0 -15	2.5 5.0 15	V	— (6) (6)
$V_{VSENSEN5/6}$	DC voltage at VSENSEN5/6 <ul style="list-style-type: none"> <li>• DC voltage at VCC5 &lt; 10 V</li> <li>• Dynamic for max 5.0 <math>\mu</math>s, 1.0 kHz repetition rate at VCC5 &lt; 5.25 V</li> <li>• Dynamic for max 1.0 <math>\mu</math>s at VCC5 &lt; 5.25 V</li> </ul>	-3.0 -5.0 -15	1.0 5.0 15	V	— (6) (6)
$V_{VSENSEP5/6}$	DC voltage at VSENSEP5/6 <ul style="list-style-type: none"> <li>• DC voltage at VCC5 &lt; 10 V</li> <li>• Dynamic for max 5.0 <math>\mu</math>s, 1.0 kHz repetition rate at VCC5 &lt; 5.25 V</li> <li>• Dynamic for max 1.0 <math>\mu</math>s at VCC5 &lt; 5.25 V</li> </ul>	-4.2 -5.0 -15	2.5 5.0 15	V	— (6) (6)

**ESD voltage**

$V_{ESD-HBM1}$	ESD voltage <ul style="list-style-type: none"> <li>• Human body model (HBM) VBOOST, VBATT, S_HSx</li> </ul>	-4000	4000	V	(7), (8)
$V_{ESD-HBM2}$		-8000	8000		
$V_{ESD-HBM3}$		-2000	2000		
$V_{ESD-CDM1}$	Machine model <ul style="list-style-type: none"> <li>• Corner pins</li> </ul>	-750	750		
$V_{ESD-CDM2}$		All other pins	-500		

**Thermal ratings**

$T_A$	Operating temperature <ul style="list-style-type: none"> <li>• Ambient</li> <li>• Junction</li> </ul>	-40	125	°C	
$T_J$		-40	150		
$T_{THRESHOLD}$	Temperature monitoring threshold	167	187	°C	
$T_{STG}$	Storage ambient temperature	-55	150	°C	

**Thermal resistance**

$R_{\theta JA}$	Thermal resistance junction to ambient	—	25.3	°C/W	(9)
$R_{\theta JCTOP}$	Thermal resistance junction to case top	—	13.2	°C/W	(10)
$R_{\theta JCBOTTOM}$	Thermal resistance junction to case bottom	—	0.8	°C/W	(11)

**Notes**

- With series resistor of 3.3 k $\Omega$   $\pm$ 20% at the pin
- Guaranteed by design.
- Human body model (HBM) per JESD22-A114 - 100 pF, 1.5 k $\Omega$
- Charge device model (CDM) per JESD22-C101.
- Per JEDEC JESD51-6 with the board (JESD51-7) horizontal
- Thermal resistance between the die and the case top surface as measured by the cold plate method (MIL SPEC - 883 Method 1012.1).
- Thermal resistance between the die and the solder pad on the bottom of the package based on the simulation without internal resistance

## 5.2 Power supply electrical characteristics

**Table 5. PT2000 static electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>VBATT input supply</b>						
$V_{BATT}$	VBATT power supply input voltage, normal operation <ul style="list-style-type: none"> <li>Internal VCCP regulator</li> <li>External VCCP regulator</li> </ul>	5.0 5.0	13.5 —	16 72	V	(12)
$V_{BATT\_LOADDUMP}$	VBATT power supply input voltage during load dump duration < 500 ms <ul style="list-style-type: none"> <li>Internal VCCP regulator</li> </ul>	18	—	40	V	
$I_{VBATT\_LEAK}$	VBATT power supply current in reset state, $V_{CC5} = V_{CCIO} = 0.0\text{ V}$ <ul style="list-style-type: none"> <li><math>V_{BATT} = 13.5\text{ V}</math></li> <li><math>V_{BATT} = 40\text{ V}</math></li> </ul>	— —	150 600	180 800	$\mu\text{A}$	
$I_{VBATT\_OPER}$	VBATT power supply current in normal operation $V_{BATT} = 16\text{ V}$ <ul style="list-style-type: none"> <li>DRVEN low, internal VCCP reg. off</li> <li>DRVEN low, Internal VCCP reg. on</li> <li>DRVEN high, VCCP max. load 100 mA</li> </ul>	— — —	0.9 4.5 104.5	2.5 6.0 106	mA	
$I_{VBATT\_LEAK}$	VBATT power supply current in reset state, $V_{CC5} = V_{CCIO} = 0.0\text{ V}$ <ul style="list-style-type: none"> <li><math>V_{BATT} = 13.5\text{ V}</math></li> <li><math>V_{BATT} = 40\text{ V}</math></li> </ul>	— —	150 600	180 800	$\mu\text{A}$	
<b>VBOOST input supply</b>						
$I_{VBOOST\_LEAK}$	Leakage current from $V_{BOOST}$ , during reset state with $V_{CC5} = V_{CCIO} = 5.0\text{ V}$ <ul style="list-style-type: none"> <li><math>V_{BOOST} = V_{BAT} = 13.5\text{ V}</math></li> <li><math>V_{BOOST} = V_{BAT} = 40\text{ V}</math></li> <li><math>V_{BOOST} = V_{BAT} = 65\text{ V}</math></li> </ul> Contributors (13.5 V): $V_{BOOST}$ volt. div.: $65\text{ }\mu\text{A}$	65 240 400	— — —	90 370 600	$\mu\text{A}$	
$I_{VBOOST\_OPER}$	Operating current from $V_{BOOST} = 65\text{ V}$	—	3.9	5.75	mA	
<b>VCC5 input supply</b>						
$V_{CC5}$	VCC5 supply input voltage	4.75	5.0	5.25	V	
$V_{CC5\_DIGITAL}$	VCC5 supply input voltage for digital part functional only	4.0	5.0	5.25	V	(12)
$I_{VCC5}$	VCC5 supply current <ul style="list-style-type: none"> <li><math>f_{SYS} = 24\text{ MHz}</math>, 7 HS load biasing enabled, no microcore running</li> <li><math>f_{SYS} = 24\text{ MHz}</math>, 7 HS load biasing enabled, all microcores running</li> <li>LBIST running, bias disabled</li> </ul>	— — —	53 65 40	66.4 81.4 50	mA	(12)
$V_{OVVCC5}$	VCC5 overvoltage threshold	7.5	8.5	10	V	
$V_{OVVCC5\_VCCP}$	VCC5 overvoltage threshold for VCCP shutdown	6.2	6.9	7.5	V	
$V_{UVVCC5-}$	VCC5 undervoltage low-voltage threshold	4.3	4.45	4.7	V	
$V_{UVVCC5+}$	VCC5 undervoltage high-voltage threshold	4.35	4.5	4.75	V	
$V_{UVVCC5\_HYST}$	VCC5 undervoltage hysteresis	30	50	85	mV	

Notes

12. Guaranteed by design.

**Table 5. PT2000 static electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>VCC5 input supply (continued)</b>						
$T_{\text{FILTER\_UVVCC5}}$	VCC5 UV anti-glitch filter delay time	0.8	1.3	2.0	$\mu\text{s}$	
<b>VCCIO INPUT SUPPLY</b>						
$V_{\text{CCIO}}$	VCCIO supply input voltage	3.0	—	5.25	V	
$I_{\text{VCCIO}}$	VCCIO supply current <ul style="list-style-type: none"> <li><math>f_{\text{SYS}} = 24\text{ MHz}</math>, no microcore running</li> <li><math>f_{\text{SYS}} = 24\text{ MHz}</math>, all microcores running</li> </ul>	— —	38 1.5	70 —	$\mu\text{A}$ mA	— (13)
<b>VCCP input supply</b>						
$V_{\text{CCP}}$	VCCP output voltage, $0.0\text{ mA} < I_{\text{VCCP}} < 100\text{ mA}$	6.5	7.0	7.5	V	
$V_{\text{CCP\_EXT}}$	VCCP input voltage range ( $V_{\text{CCP}}$ externally supplied)	5.0	—	9.0	V	
$C_{\text{VCCP}}$	VCCP external output capacitor	1.0	4.7	14	$\mu\text{F}$	(14)
$\Delta V_{\text{VCCP}}$	$V_{\text{BATT}}$ to $V_{\text{CCP}}$ voltage dropout <ul style="list-style-type: none"> <li><math>V_{\text{BATT}} = 5.0\text{ V}</math> and <math>I_{\text{VCCP}} = -50\text{ mA}</math></li> <li><math>V_{\text{BATT}} = 5.0\text{ V}</math> and <math>I_{\text{VCCP}} = -30\text{ mA}</math></li> <li><math>V_{\text{BATT}} = 5.0\text{ V}</math> and <math>I_{\text{VCCP}} = -10\text{ mA}</math></li> <li><math>V_{\text{BATT}} = 5.0\text{ V}</math> and <math>I_{\text{VCCP}} = -100\text{ mA}</math></li> </ul>	— — — —	— — — —	180 110 40 350	mV	
$V_{\text{UVVCCP-}}$	VCCP undervoltage low-voltage threshold	4.3	4.5	4.68	V	
$V_{\text{UVVCCP+}}$	VCCP undervoltage high-voltage threshold	4.4	4.55	4.73	V	
$V_{\text{UVVCCP\_HYST}}$	VCCP undervoltage hysteresis	30	50	70	mV	
$I_{\text{VCCP}}$	$V_{\text{CCP}}$ output current (average during PWM operation) $9.0\text{ V} < V_{\text{BATT}} < 18\text{ V}$	—	—	100	mA	
$I_{\text{VCCP\_MAX}}$	VCCP output current limitation	150	200	250	mA	
$t_{\text{FILTER\_UVVCCP}}$	VCCP UV anti-glitch filter delay time	0.8	1.3	2.0	$\mu\text{s}$	
<b>VCC2P5 internal regulator</b>						
$V_{\text{CC2P5}}$	VCC2P5 supply output voltage	2.375	2.5	2.625	V	
$V_{\text{cc5\_BGmin}}$	Voltage required on VCC5 to start VCC2P5	—	—	3.8	V	
$C_{\text{VCC2P5}}$	VCC2P5 external output capacitor	0.5	1.0	3.0	$\mu\text{F}$	(15)
$I_{\text{VCC2P5}}$	VCC2P5 supply output current <ul style="list-style-type: none"> <li><math>f_{\text{SYS}} = 24\text{ MHz}</math>, all microcores running</li> </ul>	—	-15	-50	mA	
$I_{\text{VCC2P5\_LIM}}$	VCC2P5 supply output current limit	-50	93	140	mA	
$V_{\text{PORESETB-}}$	VCC2P5 voltage threshold for asserting PORESETB	2.0	2.11	2.21	V	
$V_{\text{PORESETB+}}$	VCC2P5 voltage threshold for deasserting PORESETB	2.07	2.19	2.3	V	
$V_{\text{PORESETB\_HYST}}$	PORESETB voltage hysteresis	50	75	100	mV	
$t_{\text{D\_PORESETB}}$	PORESETB switching time	—	0.7	1.5	$\mu\text{s}$	

**Notes**

- Guaranteed by design.
- For VCCP: "For EMC purpose adding  $1.0\text{ }\mu\text{F} + 100\text{ nF}$  caps in parallel connected to PGND is recommended
- For VCC2P5: "For EMC purpose adding  $1.0\text{ }\mu\text{F} + 100\text{ nF}$  caps in parallel connected to DGND is recommended

**Table 5. PT2000 static electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Battery voltage monitor</b>						
$V_{\text{BATT\_MONITOR}}$	Input voltage range	5.0	—	36	V	(16)
$R_{\text{VBATT\_IN}}$	Input impedance	350	500	—	k $\Omega$	
$G_{\text{VBATT\_DIV}}$	$V_{\text{BATT}}$ voltage divider ratio	—	1/16	—		
$f_{\text{CVBATT\_DIV}}$	$V_{\text{BATT}}$ analog filter cutoff frequency	10	15.5	20	kHz	
$V_{\text{VBATT\_REF}}$	DAC reference voltage	2.475	2.5	2.525	V	
$V_{\text{VBATT\_DAC\_LSB}}$	DAC LSB	—	39.06	—	mV	
$V_{\text{VBATT\_DAC\_OUT\_MIN}}$	DAC minimum output voltage • DAC code = 0h	—	0.0	—	V	
$V_{\text{VBATT\_DAC\_OUT\_MAX}}$	DAC maximum output voltage • DAC code = 3Fh	—	2.461	—	V	
$\epsilon_{\text{VBATT}}$	$V_{\text{BATT}}$ measurement total error ( $V_{\text{BATT}} > 5.0\text{ V}$ )	-5.0	1.0	5.0	%	
$t_{\text{VBATT\_DAC}}$	$V_{\text{BATT}}$ DAC settling time	—	—	0.9	$\mu\text{s}$	
<b>Boost voltage monitor</b>						
$V_{\text{BOOSTMAX}}$	Input voltage range	0.0	—	72	V	
$R_{\text{VBOOST\_IN}}$	Input impedance	400	640	—	k $\Omega$	
$G_{\text{VBOOST\_DIV}}$	$V_{\text{BOOST}}$ voltage divider ratio (boost monitor mode)	1/32* 0.996	1/32	1/32* 1.004		
$G_{\text{UV\_VBOOST\_DIV}}$	$V_{\text{BOOST}}$ voltage divider ratio (UV $V_{\text{BOOST}}$ mode)	1/4* 0.996	1/4	1/4* 1.004		
$V_{\text{VBOOST\_DAC\_LSB}}$	DAC LSB	—	9.77	—	mV	
$\epsilon_{\text{VBOOST}}$	$V_{\text{BOOST}}$ measurement total error (4.85 V to 72 V)	-2.0	—	2.0	%	

**Notes**

16. This limitation is only for the  $V_{\text{BAT}}$  ADC, if  $V_{\text{BAT}}$  is  $> 36\text{ V}$ , then  $V_{\text{BAT}}$  monitoring results will saturate. It means in case  $V_{\text{BAT}} > 36\text{ V}$ , the  $V_{\text{BAT}}$  monitoring feature will not work but device will be 100 % functional until  $V_{\text{BAT}} = 72\text{ V}$ .

## 5.3 High-side pre-driver electrical characteristics

**Table 6. High-side pre-driver electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>High-side pre-driver</b>						
$V_{S\_HSX}$	S_HSx pin operating voltage Transients $t < 400\text{ ns}$ Transients $t < 800\text{ ns}$	-3.0 -6.0 -8.0	— — —	$V_{\text{BOOSTMAX}}$ — —	V	(17)
$V_{B\_HSX}$	B_HSx pin operating voltage	$V_{S\_HSX} + 4.0$	—	$V_{S\_HSX} + 8.0$	V	(17)
$V_{BS\_HSX\_CL}$	B_HSx-S_HSx clamp voltage	6.5	7.3	8.0	V	(18)
$V_{G\_HSX}$	G_HSx operating voltage	$V_{S\_HSX}$	—	$V_{B\_HSX}$	V	(17)
$I_{S\_HSX\_SINK\_OFF}$	S_HSx leakage current biasing switched Off: • $V_{S\_HSX} = V_{\text{BOOSTMAX}}$ • $V_{S\_HSX} = 13.5\text{ V}$ • $V_{S\_HSX} = 7.0\text{ V}$ • $V_{S\_HSX} = 4.0\text{ V}$	— — — —	— — — —	1000 250 120 100	$\mu\text{A}$	
$I_{S\_HSX\_SINK\_ON}$	S_HSx leakage current when pre-driver on (biasing switched Off) • $V_{S\_HSX} = 7.0\text{ V}$	—	—	220	$\mu\text{A}$	
$f_{G\_HSX\_PWM}$	PWM frequency • Internal $V_{\text{CCP}}$ and $V_{\text{BATT}} \geq 9.0\text{ V}$ • Internal $V_{\text{CCP}}$ and $5.0\text{ V} \leq V_{\text{BATT}} \leq 9.0\text{ V}$ • External $V_{\text{CCP}}$ and $9.0\text{ V} \leq V_{\text{BATT}}$	0.0 0.0 0.0	— — —	100 50 100	kHz	(17)
$DC_{G\_HSX}$	Duty cycle	0.0	—	100	%	
$t_{\text{ON\_HSX\_MIN}}$	High-side driver minimum PWM on time	—	—	1.0	$\mu\text{s}$	(17)
$Q_{G\_HSX}$	External high-side MOSFET effective gate charge • $f_{\text{PWM}} \leq f_{G\_HSX\_PWM}$ • $f_{\text{PWM}} \leq 67\text{ kHz}$	— —	40 55	50 75	nC	
$I_{G\_HSX\_PWM}$	G_HSx current (average during PWM operation) $Q_G = Q_{G\_HSX}$ , $f_{\text{PWM}} = 100\text{ kHz}$	—	4.0	5.0	mA	(17)
$I_{G\_HSX\_SRC}$	Peak source gate drive current	—	230	—	mA	(17)
$I_{G\_HSX\_SINK}$	Peak sink gate drive current	—	440	—	mA	(17)

### High-side pre-driver dynamic

$t_{R\_G\_HSX}$	Turn on rise time, 10%-90% of out voltage, $V_{\text{CCP}} = 7.0\text{ V}$ , at open pin	4.5	—	25	ns	(17)
$t_{F\_G\_HSX}$	Turn off fall time, 90%-10% of out voltage, $V_{\text{CCP}} = 7.0\text{ V}$ , at open pin	5.0	—	25	ns	(17)
$SR_{S\_HSX}$	Max permissible slew rate at the S_HSx pin	-125	—	600	V/ $\mu\text{s}$	(17)
$t_{\text{DON\_G\_HSX\_300}}$	Turn on propagation delay at 300 V/ $\mu\text{s}$ slew rate	40	—	100	ns	(17)(19)
$t_{\text{DOFF\_G\_HSX\_300}}$	Turn off propagation delay at 300 V/ $\mu\text{s}$ slew rate	40	—	100	ns	(17)(19)
$t_{\text{DON\_G\_HSX\_50}}$	Turn on propagation delay at 50 V/ $\mu\text{s}$ slew rate	65	—	125	ns	(17)(19)
$t_{\text{DOFF\_G\_HSX\_50}}$	Turn off propagation delay at 50 V/ $\mu\text{s}$ slew rate	50	—	100	ns	(17)(19)

### Notes

- Guaranteed by design.
- $V_{B\_HSX}$  has to be 2.0 V above PGND for full function (switch on) of the pre-driver
- 10% of output voltage change,  $C_{\text{LOAD}} = 4.7\text{ nF}$ ;  $R_G = 40.2\text{ }\Omega$ ,  $V_{\text{CCP}} = 7.0\text{ V}$



**Table 6. High-side pre-driver electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>High-side pre-driver dynamic (continued)</b>						
$t_{\text{DON\_G\_HSX\_25}}$	Turn on propagation delay at 25 V/ $\mu\text{s}$ slew rate	100	—	200	ns	(20)(21)
$t_{\text{DOFF\_G\_HSX\_25}}$	Turn off propagation delay at 25 V/ $\mu\text{s}$ slew rate	70	—	150	ns	(20)(21)
$t_{\text{DON\_G\_HSX\_12.5}}$	Turn on propagation delay at 12.5 V/ $\mu\text{s}$ slew rate,	160	—	310	ns	(20)(21)
$t_{\text{DOFF\_G\_HSX\_12.5}}$	Turn off propagation delay at 12.5 V/ $\mu\text{s}$ slew rate	90	—	170	ns	(20)(21)

**HS pre-driver safe off**

$R_{\text{PD\_HSX}}$	G_HSx to S_HSx pull-down resistor	500	—	2000	k $\Omega$	
----------------------	-----------------------------------	-----	---	------	------------	--

**Slew rate control**

$R_{\text{DS\_HSX\_P (00)}}$	G_HSx pMOS $R_{\text{DS(on)}}$ (00) 300 V/ $\mu\text{s}$	7.5	14.6	31.4	$\Omega$	
$R_{\text{DS\_HSX\_N (00)}}$	G_HSx nMOS $R_{\text{DS(on)}}$ (00) 300 V/ $\mu\text{s}$	2.5	5.9	16.5	$\Omega$	
$R_{\text{DS\_HSX\_P (01)}}$	G_HSx pMOS $R_{\text{DS(on)}}$ (01) 50 V/ $\mu\text{s}$	61	85	115	$\Omega$	
$R_{\text{DS\_HSX\_N (01)}}$	G_HSx nMOS $R_{\text{DS(on)}}$ (01) 50 V/ $\mu\text{s}$	23	35	50	$\Omega$	
$R_{\text{DS\_HSX\_P (10)}}$	G_HSx pMOS $R_{\text{DS(on)}}$ (10) 25 V/ $\mu\text{s}$	122	169	230	$\Omega$	
$R_{\text{DS\_HSX\_N (10)}}$	G_HSx nMOS $R_{\text{DS(on)}}$ (10) 25 V/ $\mu\text{s}$	47	69	100	$\Omega$	
$R_{\text{DS\_HSX\_P (11)}}$	G_HSx pMOS $R_{\text{DS(on)}}$ (11) 12.5 V/ $\mu\text{s}$	245	337	460	$\Omega$	
$R_{\text{DS\_HSX\_N (11)}}$	G_HSx nMOS $R_{\text{DS(on)}}$ (11) 12.5 V/ $\mu\text{s}$	94	138	199	$\Omega$	

## Notes

20. Guaranteed by design.

21. 10% of output voltage change,  $C_{\text{LOAD}} = 4.7\text{ nF}$ ;  $R_G = 40.2\text{ }\Omega$ ,  $V_{\text{CCP}} = 7.0\text{ V}$

## 5.4 Low-side (LS1-LS6) pre-driver electrical characteristics

**Table 7. Low-side pre-driver electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Low-side pre-driver LS1-6</b>						
$V_{\text{G\_LSx}}$	G_LSx operating voltage	0.0	—	$V_{\text{CCP}}$	V	(22)
$I_{\text{S\_LSx\_sink}}$	D_LSx leakage current (biasing switched off) • $V_{\text{D\_LSx}} = 13.5\text{ V}$ • $V_{\text{D\_LSx}} = 40\text{ V}$	10 10	— —	110 320	$\mu\text{A}$	
$f_{\text{G\_LSx\_PWM}}$	PWM frequency • Nominal • Short period of switching during 50 $\mu\text{s}$ every 1ms	0.0 0.0	— —	100 200	kHz	(22)
$\text{DC}_{\text{G\_LSx}}$	Duty cycle	0.0	—	100	%	(22)
$Q_{\text{G\_LSx}}$	External low-side MOSFET effective gate charge • $f_{\text{PWM}} \leq f_{\text{G\_LSx\_PWM}}$ • $f_{\text{PWM}} \leq 67\text{ kHz}$ • $f_{\text{PWM}} \leq 50\text{ kHz}$	— — —	30 55 —	50 75 100	nC	

## Notes

22. Guaranteed by design.

**Table 7. Low-side pre-driver electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
--------	----------------	------	------	------	------	-------

**Low-side pre-driver LS1-6 (continued)**

$I_{G\_LSx\_PWM}$	G_LSx current (average during PWM operation) • $Q_G = Q_{G\_LSx}$ ; $f_{PWM} = 100\text{ kHz}$	—	3.0	5.0	mA	(23)
$I_{G\_LSx\_SRC}$	Peak source gate drive current	—	230	—	mA	(23), (24)
$I_{G\_LSx\_SINK}$	Peak sink gate drive current	—	440	—	mA	(23), (24)

**Dynamic low-side pre-driver LS1-6**

$T_{R\_G\_LSx}$	Turn on rise time, 10% to 90% of output voltage; $V_{CCP} = 7.0\text{ V}$ ; at open pin	5.0	—	25	ns	(23)
$T_{F\_G\_LSx}$	Turn off fall time, 90% to 10% of output voltage; $V_{CCP} = 7.0\text{ V}$ ; at open pin	5.0	—	25	ns	(23)
$T_{DON\_G\_LSx\_300}$	Turn on propagation delay at 300 V/ $\mu\text{s}$ slew rate	10	—	70	ns	(23), (25)
$T_{DOFF\_G\_LSx\_300}$	Turn off propagation delay at 300 V/ $\mu\text{s}$ slew rate	10	—	70	ns	(23), (25)
$T_{DON\_G\_LSx\_50}$	Turn on propagation delay at 50 V/ $\mu\text{s}$ slew rate	10	—	80	ns	(23), (25)
$T_{DOFF\_G\_LSx\_50}$	Turn off propagation delay at 50 V/ $\mu\text{s}$ slew rate	10	—	80	ns	(23), (25)
$T_{DON\_G\_LSx\_25}$	Turn on propagation delay at 25 V/ $\mu\text{s}$ slew rate	15	—	120	ns	(23), (25)
$T_{DOFF\_G\_LSx\_25}$	Turn off propagation delay at 25 V/ $\mu\text{s}$ slew rate	15	—	120	ns	(23), (25)
$T_{DON\_G\_LSx\_12.5}$	Turn on propagation delay at 12.5 V/ $\mu\text{s}$ slew rate	15	—	150	ns	(23), (25)
$T_{DOFF\_G\_LSx\_12.5}$	Turn off propagation delay at 12.5 V/ $\mu\text{s}$ slew rate	15	—	150	ns	(23), (25)

**LS pre-driver safe off**

$R_{PD\_LSx}$	G_LSx to PGND pull-down resistor	25	50	90	k $\Omega$	
---------------	----------------------------------	----	----	----	------------	--

**Low-side pre-driver LS1-6**

$R_{DS\_LSx\_P(00)}$	G_LSx pMOS $R_{DS(on)}$ (00) 300 V/ $\mu\text{s}$	7.5	14.6	31.3	$\Omega$	
$R_{DS\_LSx\_N(00)}$	G_LSx nMOS $R_{DS(on)}$ (00) 300 V/ $\mu\text{s}$	2.5	5.9	16.5	$\Omega$	
$R_{DS\_LSx\_P(01)}$	G_LSx pMOS $R_{DS(on)}$ (01) 50 V/ $\mu\text{s}$	61	84	115	$\Omega$	
$R_{DS\_LSx\_N(01)}$	G_LSx nMOS $R_{DS(on)}$ (01) 50 V/ $\mu\text{s}$	23	35	50	$\Omega$	
$R_{DS\_LSx\_P(10)}$	G_LSx pMOS $R_{DS(on)}$ (10) 25 V/ $\mu\text{s}$	122	170	230	$\Omega$	
$R_{DS\_LSx\_N(10)}$	G_LSx nMOS $R_{DS(on)}$ (10) 25 V/ $\mu\text{s}$	47	69	100	$\Omega$	
$R_{DS\_LSx\_P(11)}$	G_LSx pMOS $R_{DS(on)}$ (11) 12.5 V/ $\mu\text{s}$	245	337	460	$\Omega$	
$R_{DS\_LSx\_N(11)}$	G_LSx nMOS $R_{DS(on)}$ (11) 12.5 V/ $\mu\text{s}$	94	138	199	$\Omega$	

**Notes**

23. Guaranteed by design.
24.  $V_{CCP} = V_{GS} = 7.0\text{ V}$  and fastest slew rate
25. 10% of output voltage change;  $C_{LOAD} = 4.7\text{ nF}$ ;  $R_G = 40.2\text{ }\Omega$ ;  $V_{CCP} = 7.0\text{ V}$

## 5.5 Low-side high-speed (LS7-LS8) pre-driver electrical characteristics

**Table 8. High-speed low-side pre-driver electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Low-side pre-driver 7 and 8</b>						
$V_{G\_LS7/8}$	G_LS7/8 operating voltage	0.0	—	$V_{CCP}$	V	(26)
$f_{G\_LS7/8\_PWM}$	PWM frequency	0.0	—	400	kHz	(27)
$DC_{G\_LS7/8}$	Duty cycle	0.0	—	100	%	(27)
$Q_{G\_LS7/8}$	External low-side MOSFET gate charge • $f_{PWM} = 400\text{ kHz}$ • $f_{PWM} = 300\text{ kHz}$ • $f_{PWM} = 240\text{ kHz}$	— — —	30 30 30	60 75 100	nC	
$I_{G\_LS7/8\_PWM}$	G_LS7/8 current (average during PWM operation) • $f_{PWM} = 400\text{ kHz}$ • $f_{PWM} = 300\text{ kHz}$ • $f_{PWM} = 100\text{ kHz}$ • $f_{PWM} = 50\text{ kHz}$	— — — —	12 9.0 3.0 1.5	24 22.5 7.5 3.75	mA	(27)
$I_{G\_LS7/8\_SRC}$	Peak source gate drive current	—	680	—	mA	(27), (27)
$I_{G\_LS7/8\_SINK}$	Peak sink gate drive current	—	2200	—	mA	(27), (27)

### Dynamic low-side pre-driver L7 and 8

$t_{R\_G\_LS7/8\_1500}$	Turn on rise time • at 1500 V/ $\mu\text{s}$ slew rate 10% to 90% of out voltage; $V_{CCP} = 7.0\text{ V}$ ; at the open pin	3.5	—	11	ns	(27)
$t_{F\_G\_LS7/8\_1500}$	Turn off fall time • at 1500 V/ $\mu\text{s}$ slew rate 90% to 10% of out voltage; $V_{CCP} = 7.0\text{ V}$ ; at the open pin	3.5	—	11	ns	(27)
$t_{R\_G\_LS7/8}$	Turn on rise time • at 300-25 V/ $\mu\text{s}$ slew rate 10% to 90% of out voltage; $V_{CCP} = 7.0\text{ V}$ ; at the open pin	5.0	—	25	ns	(27)
$t_{F\_G\_LS7/8}$	Turn off fall time • at 300-25 V/ $\mu\text{s}$ slew rate 90% to 10% of out voltage; $V_{CCP} = 7.0\text{ V}$ ; at the open pin	5.0	—	25	ns	(27)
$t_{DON\_G\_LS7\_1500}$	Turn on propagation delay • at 1500 V/ $\mu\text{s}$ slew rate 10% of out voltage change	10	—	50	ns	(27), (28)
$t_{DOFF\_G\_LS7\_1500}$	Turn off propagation delay • at 1500 V/ $\mu\text{s}$ slew rate 10% of out voltage change	10	—	50	ns	(27), (28)
$t_{DON\_G\_LS7\_300}$	Turn on propagation delay • at 300 V/ $\mu\text{s}$ slew rate 10% of out voltage change	10	—	70	ns	(27), (28)
$t_{DOFF\_G\_LS7\_300}$	Turn off propagation delay • at 300 V/ $\mu\text{s}$ slew rate 10% of out voltage change	10	—	70	ns	(27), (28)

#### Notes

26. Guaranteed by design.
27. At the fastest slew rate setting with minimum  $R_{G\_LS8}$  of 2.0  $\Omega$  and  $V_{CCP}/V_{GS} = 7.0\text{ V}$
28.  $C_{LOAD} = 4.7\text{ nF}$ ;  $R_G = 40.2\text{ }\Omega$ ,  $V_{CCP} = 7.0\text{ V}$

**Table 8. High-speed low-side pre-driver electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Dynamic low-side pre-driver L7 and 8</b>						
$t_{\text{DON\_G\_LS7\_50}}$	Turn on propagation delay • at 50 V/ $\mu\text{s}$ slew rate 10% of out voltage change	15	—	100	ns	(29), (30)
$t_{\text{DOFF\_G\_LS7\_50}}$	Turn off propagation delay • at 50 V/ $\mu\text{s}$ slew rate 10% of out voltage change	15	—	100	ns	(29), (30)
$t_{\text{DOFF\_G\_LS7\_25}}$	Turn off propagation delay • at 25 V/ $\mu\text{s}$ slew rate 10% of out voltage change	15	—	120	ns	(29), (30)
$t_{\text{DOFF\_G\_LS7\_25}}$	Turn off propagation delay • at 25 V/ $\mu\text{s}$ slew rate 10% of out voltage change	15	—	120	ns	(29), (30)
$R_{\text{PD\_LS7/8}}$	G_LS7/8 to PGND pull-down resistor	25	50	90	k $\Omega$	

**Slew rate control low-side 7 and 8**

$R_{\text{DS\_HSX\_P (00)}}$	G_LS7/8 pMOS $R_{\text{DS(on)}}$ (00) 1500 V/ $\mu\text{s}$	2.6	5.0	10.7	$\Omega$	
$R_{\text{DS\_HSX\_N (00)}}$	G_LS7/8 nMOS $R_{\text{DS(on)}}$ (00) 1500 V/ $\mu\text{s}$	0.5	1.1	2.9	$\Omega$	
$R_{\text{DS\_HSX\_P (01)}}$	G_LS7/8 pMOS $R_{\text{DS(on)}}$ (01) 300 V/ $\mu\text{s}$	7.5	14.6	31.3	$\Omega$	
$R_{\text{DS\_HSX\_N (01)}}$	G_LS7/8 nMOS $R_{\text{DS(on)}}$ (01) 300 V/ $\mu\text{s}$	2.5	5.9	16.5	$\Omega$	
$R_{\text{DS\_HSX\_P (10)}}$	G_LS7/8 pMOS $R_{\text{DS(on)}}$ (10) 50 V/ $\mu\text{s}$	61	85	115	$\Omega$	
$R_{\text{DS\_HSX\_N (10)}}$	G_LS7/8 nMOS $R_{\text{DS(on)}}$ (10) 50 V/ $\mu\text{s}$	23	35	50	$\Omega$	
$R_{\text{DS\_HSX\_P (11)}}$	G_LS7/8 pMOS $R_{\text{DS(on)}}$ (11) 25 V/ $\mu\text{s}$	122	170	230	$\Omega$	
$R_{\text{DS\_HSX\_N (11)}}$	G_LS7/8 nMOS $R_{\text{DS(on)}}$ (11) 25 V/ $\mu\text{s}$	47	69	100	$\Omega$	

## Notes

29. Guaranteed by design.

30.  $C_{\text{LOAD}} = 4.7\text{ nF}$ ;  $R_G = 40.2\text{ }\Omega$ ,  $V_{\text{CCP}} = 7.0\text{ V}$ 

## 5.6 High-side VDS VSRC monitoring electrical characteristics

**Table 9. High-side VDS/SRC monitor electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>High-side <math>V_{\text{DS/SRC}}</math> monitor</b>						
$V_{\text{S\_HS\_VDS}}$	High-side $V_{\text{DS/SRC}}$ monitoring functional range S_HSx • DC voltage • Transients $t < 400\text{ ns}$ • Transients $t < 800\text{ ns}$	-3.0 -6.0 -8.0	— — —	72 72 72	V	(31)
$V_{\text{VBATT\_VDS}}$	High-side $V_{\text{DS/SRC}}$ monitoring functional range S_HSx • Full functionality • Limited functionality ( $V_{\text{DS\_HS\_Th}}$ 3.5 V is at 3.0 V min)	5.5 5.0	— —	72 5.5	V	
$V_{\text{VBOOST\_VDS}}$	High-side $V_{\text{DS/SRC}}$ monitoring functional range VBOOST • Full functionality • Limited functionality ( $V_{\text{DS\_HS\_Th}}$ 3.5 V is at 3.0 V min)	5.5 5.0	— —	72 5.5	V	

## Notes

31. Guaranteed by design.

**Table 9. High-side V<sub>DS</sub>/S<sub>R</sub>C monitor electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>High-side V<sub>DS</sub>/S<sub>R</sub>C monitor (continued)</b>						
V <sub>DS_HS_TH</sub> (0000)	High-side V <sub>DS</sub> threshold (0000)	-0.03	0.0	0.03	V	
V <sub>DS_HS_TH</sub> (1001)	High-side V <sub>DS</sub> threshold (1001)	0.07	0.10	0.13	V	
V <sub>DS_HS_TH</sub> (1010)	High-side V <sub>DS</sub> threshold (1010)	0.155	0.2	0.245	V	
V <sub>DS_HS_TH</sub> (1011)	High-side V <sub>DS</sub> threshold (1011)	0.25	0.3	0.35	V	
V <sub>DS_HS_TH</sub> (1100)	High-side V <sub>DS</sub> threshold (1100)	0.345	0.4	0.455	V	
V <sub>DS_HS_TH</sub> (0001)	High-side V <sub>DS</sub> threshold (0001)	0.44	0.5	0.56	V	
V <sub>DS_HS_TH</sub> (0010)	High-side V <sub>DS</sub> threshold (0010)	0.9	1.0	1.1	V	
V <sub>DS_HS_TH</sub> (0011)	High-side V <sub>DS</sub> threshold (0011)	1.35	1.5	1.65	V	
V <sub>DS_HS_TH</sub> (0100)	High-side V <sub>DS</sub> threshold (0100)	1.8	2.0	2.2	V	
V <sub>DS_HS_TH</sub> (0101)	High-side V <sub>DS</sub> threshold (0101)	2.29	2.45	2.61	V	
V <sub>DS_HS_TH</sub> (0110)	High-side V <sub>DS</sub> threshold (0110)	2.76	2.95	3.14	V	
V <sub>DS_HS_TH</sub> (0111)	High-side V <sub>DS</sub> threshold (0111) <ul style="list-style-type: none"> <li>• V<sub>BATT_VDS</sub> = 5.5 V to 72 V, V<sub>BOOST_VDS</sub> = 5.5 V to 72 V</li> <li>• V<sub>BATT_VDS</sub> = 5.0 V to 5.5 V, V<sub>BOOST_VDS</sub> = 5.0 V to 5.5 V</li> </ul>	3.23 3.00	3.45 3.45	3.67 3.67	V	
t <sub>TH_HS_VDS</sub>	High-side V <sub>DS</sub> /S <sub>R</sub> C threshold settling time	—	0.4	1.0	μs	
V <sub>SRC_HS_TH</sub> (0000)	High-side V <sub>SRC</sub> threshold (0000)	-0.03	0.0	0.03	V	
V <sub>SRC_HS_TH</sub> (1001)	High-side V <sub>SRC</sub> threshold (1001)	0.07	0.10	0.13	V	
V <sub>SRC_HS_TH</sub> (1010)	High-side V <sub>SRC</sub> threshold (1010)	0.155	0.2	0.245	V	
V <sub>SRC_HS_TH</sub> (1011)	High-side V <sub>SRC</sub> threshold (1011)	0.25	0.3	0.35	V	
V <sub>SRC_HS_TH</sub> (1100)	High-side V <sub>SRC</sub> threshold (1100)	0.345	0.4	0.455	V	
V <sub>SRC_HS_TH</sub> (0001)	High-side V <sub>SRC</sub> threshold (0001)	0.44	0.5	0.56	V	
V <sub>SRC_HS_TH</sub> (0010)	High-side V <sub>SRC</sub> threshold (0010)	0.9	1.0	1.1	V	
V <sub>SRC_HS_TH</sub> (0011)	High-side V <sub>SRC</sub> threshold (0011)	1.35	1.5	1.65	V	
V <sub>SRC_HS_TH</sub> (0100)	High-side V <sub>SRC</sub> threshold (0100)	1.8	2.0	2.2	V	
V <sub>SRC_HS_TH</sub> (0101)	High-side V <sub>SRC</sub> threshold (0101)	2.38	2.55	2.72	V	
V <sub>SRC_HS_TH</sub> (0110)	High-side V <sub>SRC</sub> threshold (0110)	2.85	3.0	3.15	V	
V <sub>SRC_HS_TH</sub> (0111)	High-side V <sub>SRC</sub> threshold (0111)	3.33	3.5	3.68	V	

## 5.7 Low-side VDS VSRC monitoring electrical characteristics

**Table 10. Low-side  $V_{DS/SRC}$  monitor electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Low-side <math>V_{DS}</math> monitor</b>						
$V_{D\_LSX\_VDS}$	Low-side $V_{DS}$ monitoring functional range $D\_LSx$ <ul style="list-style-type: none"> <li>DC voltage</li> <li>Transients <math>t &lt; 400\text{ ns}</math></li> </ul>	-3.0 -8.0	— —	75 75	V	
$V_{DS\_LS\_TH(0000)}$	Low-side $V_{DS}$ threshold (0000)	-0.03	0.0	0.03	V	
$V_{DS\_LS\_TH(1001)}$	Low-side $V_{DS}$ threshold (1001)	0.07	0.10	0.13	V	
$V_{DS\_LS\_TH(1010)}$	Low-side $V_{DS}$ threshold (1010)	0.155	0.2	0.245	V	
$V_{DS\_LS\_TH(1011)}$	Low-side $V_{DS}$ threshold (1011)	0.25	0.3	0.35	V	
$V_{DS\_LS\_TH(1100)}$	Low-side $V_{DS}$ threshold (1100)	0.345	0.4	0.455	V	
$V_{DS\_LS\_TH(0001)}$	Low-side $V_{DS}$ threshold (0001)	0.44	0.5	0.56	V	
$V_{DS\_LS\_TH(0010)}$	Low-side $V_{DS}$ threshold (0010)	0.9	1.0	1.1	V	
$V_{DS\_LS\_TH(0011)}$	Low-side $V_{DS}$ threshold (0011)	1.35	1.5	1.65	V	
$V_{DS\_LS\_TH(0100)}$	Low-side $V_{DS}$ threshold (0100)	1.8	2.0	2.2	V	
$V_{DS\_LS\_TH(0101)}$	Low-side $V_{DS}$ threshold (0101)	2.38	2.5	2.63	V	
$V_{DS\_LS\_TH(0110)}$	Low-side $V_{DS}$ threshold (0110)	2.85	3.0	3.15	V	
$V_{DS\_LS\_TH(0111)}$	Low-side $V_{DS}$ threshold (0111)	3.33	3.5	3.68	V	
$t_{TH\_LSVDS}$	Low-side $V_{DS}$ threshold settling time	—	0.4	1.0	$\mu\text{s}$	(32)
<b>Low-side <math>V_{DS}</math> monitor <math>D\_Is7/D\_Is8</math> for DC/DC</b>						
$V_{D\_LSX\_VDS}$	Low-side $V_{DS}$ voltage range $D\_LSx$	-3.0	—	75	V	
$V_{DS\_LS\_TH\_DC(0100)}$	Low-side $V_{DS}$ threshold for DC/DC (0100)	1.8	2.06	2.2	V	
$V_{DS\_LS\_TH\_DC(0101)}$	Low-side $V_{DS}$ threshold for DC/DC (0101)	2.25	2.5	2.75	V	
$t_{VDS\_DCDC\_PD}$	Comparator propagation delay time	—	—	50	ns	(32)
$R_{VDS\_78\_IN}$	Input impedance $V_{DS\_78}$	200	350	—	$\text{k}\Omega$	

**Notes**

32. Guaranteed by design.

## 5.8 Load bias electrical characteristics

**Table 11. Load bias electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Load biasing structures</b>						
$I_{\text{BIAS\_HS}}$	Current source normal S_HSx (x = 1...7)	2.8	4.0	5.2	mA	(33)
$I_{\text{BIAS\_HS\_STRONG}}$	Current source strong S_HSx (x = 2, 4)	4.2	6.0	7.8	mA	(34)
$I_{\text{BIAS\_HS\_BOTH}}$	Current source strong + normal S_HSx (x = 2, 4)	7.0	10	13	mA	(35)
$I_{\text{BIAS\_HS\_MAX}}$	Total maximum current source S_HSx source current	36.4	—	—	mA	
$I_{\text{BIAS\_LS}}$	Current source D_LSx sink saturation current	0.98	1.09	1.2	mA	
$V_{\text{BIAS\_HS}}$	S_HSx bias voltage regulation <ul style="list-style-type: none"> <li>• <math>V_{\text{BATT}} &gt; 8.0\text{ V}</math>, <math>V_{\text{CC5}} &gt; 4.75\text{ V}</math></li> <li>• <math>V_{\text{BATT}} &lt; 8.0\text{ V}</math>, <math>V_{\text{CC5}} &gt; 4.75\text{ V}</math></li> </ul>	3.8 ( $V_{\text{BATT}}/2$ ) -200 mV	— ( $V_{\text{BATT}}/2$ )	$V_{\text{CC5}}$ ( $V_{\text{BATT}}/2$ ) 200 mV	V	
$R_{\text{BIAS\_LS}}$	Equivalent resistance of LS current source <ul style="list-style-type: none"> <li>• <math>V_{\text{D\_LSx}} &lt; 1.0\text{ V}</math></li> </ul>	0.5	—	1.5	k $\Omega$	

**Notes**

- 33. Current source can be connected to maximum two D\_LSx
- 34. Current source can be connected to maximum three D\_LSx
- 35. Current source can be connected to maximum five D\_LSx

## 5.9 Current measurement electrical characteristics

### 5.9.1 Current measurement for positive current

This section is applicable for all current measurement paths for positive currents:

- Current measurement channel 1- 4
  - Differential amplifier 1- 4
  - DAC 1- 4
  - Comparator 1- 4
- Current measurement channel 5 - 6
  - Differential amplifier 5 - 6
  - DAC 5 - 6H and DAC 5 - 6L
  - Comparator 5 - 6H and 5 - 6L

**Table 12. Current measurement for positive currents**

Symbol	Characteristic	Statistically evaluated	Unit	Notes
$\epsilon_{CS}$	Overall current sense error including gain errors and offsets at DAC range of 75% - 100%, after analog offset compensation <ul style="list-style-type: none"> <li>• at GDA_diff (00) = 5.8</li> <li>• at GDA_diff (01) = 8.7</li> <li>• at GDA_diff (10) = 12.6</li> <li>• at GDA_diff (11) = 19.3</li> </ul>	±3.5 ±3.5 ±3.5 ±3.5	%	(36), (37)
	At DAC range of 25% to 75%, after analog offset compensation <ul style="list-style-type: none"> <li>• at GDA_diff (00) = 5.8</li> <li>• at GDA_diff (01) = 8.7</li> <li>• at GDA_diff (10) = 12.6</li> <li>• at GDA_diff (11) = 19.3</li> </ul>	±5.3 ±5.3 ±5.3 ±5.3	%	(36), (37)

Notes

36. Guaranteed by design.

37. The tolerance of the 10 mΩ shunt resistor is assumed as ±2.0% (at 4.5  $\sigma$ ). All other input tolerances from the device specification are assumed at 6  $\sigma$ .

**Table 13. Differential amplifier 1, 2, 3, 4, 5, and 6**

Characteristics noted under conditions -40 °C < T<sub>A</sub> < +125 °C, referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at T<sub>A</sub> = 25 °C under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
V <sub>SENSEX_DA</sub>	Differential amplifier x functional range V <sub>SENSEX</sub> (x = 1...6)	-1.0	—	1.0	V	
V <sub>SENSEPX_DA</sub>	Differential amplifier x functional range V <sub>SENSEPX</sub> (x = 1...6)	-1.0	—	1.5	V	
V <sub>DA_DIFF_IN (00)</sub>	Differential input voltage range (00) • G <sub>DA_DIFF(00)</sub> = 5.8	-25.9	—	387	mV	
V <sub>DA_DIFF_IN (01)</sub>	Differential input voltage range (01) • G <sub>DA_DIFF(01)</sub> = 8.7	-17.3	—	258	mV	
V <sub>DA_DIFF_IN (10)</sub>	Differential input voltage range (10) • G <sub>DA_DIFF(10)</sub> = 12.6	-12	—	179	mV	
V <sub>DA_DIFF_IN (11)</sub>	Differential input voltage range (11) • G <sub>DA_DIFF(11)</sub> = 19.3	-7.8	—	116	mV	
G <sub>DA_DIFF (00)</sub>	Differential voltage gain (00)	5.71	5.79	5.87		
G <sub>DA_DIFF (01)</sub>	Differential voltage gain (01)	8.55	8.68	8.81		
G <sub>DA_DIFF (10)</sub>	Differential voltage gain (10)	12.32	12.53	12.74		



**Table 13. Differential amplifier 1, 2, 3, 4, 5, and 6 (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
$G_{DA\_DIFF(11)}$	Differential voltage gain (11)	18.92	19.25	19.58		
$t_{DA\_GAIN\_SW}$	Gain switching settling time		—	2.0	$\mu\text{s}$	(38)
$R_{VSENSEX\_IN}$	Input impedance $V_{SENSEX}$ ( $x = 1\dots4$ ) • 1.0 V common mode voltage	10	—	26	$\text{k}\Omega$	
$R_{VSENSEPX\_IN}$	Input impedance $V_{SENSEPX}$ ( $x = 1\dots4$ ) • 1.0 V common mode voltage	10	—	26	$\text{k}\Omega$	
$C_{VSENSE}$	Differential amplifier EMI filter C. It is recommended to place a filter C between $V_{SENSE}$ and P close to the IC for EMI.	—	330	—	$\text{pF}$	
$V_{DA\_BIAS}$	Output bias voltage	240	250	265	mV	
$V_{DA\_OUT\_OFF}$	Maximum output offset voltage error at maximum gain	-140	—	220	mV	
$V_{DA\_OUT}$	Differential amplifier x output voltage range	0.1	—	2.7	V	

**DAC 1, 2, 3, 4, 5L, 5H, 6H, and 6L (8-bit)**

$V_{DAC\_LSB}$	DAC LSB	—	9.77	—	mV	
$V_{DAC\_OUT\_MIN}$	DAC minimum output voltage • DAC code = 0h	—	0.0	—	V	
$V_{DAC\_OUT\_MAX}$	DAC maximum output voltage • DAC code = FFh	—	2.49	—	V	
$\epsilon_{DAC\_DNL}$	DAC differential linearity error	-0.5	—	0.5	LSB	
$\epsilon_{DAC\_INL}$	DAC integral linearity error	-1.0	—	1.0	LSB	
$V_{DAC\_OUT\_OFF}$	DAC maximum output offset	0.0	—	10	mV	
$t_{DAC}$	DAC settling time	—	—	0.9	$\mu\text{s}$	

**Voltage comparator 1, 2, 3, 4, 5H, 5L, 6H, and 6L**

$V_{COMP\_IN}$	Comparator input voltage	0.0	—	2.7	V	
$V_{COMP\_IN\_OFF}$	Comparator input offset voltage	-25	—	10	mV	

**Current measurement channel 1, 2, 3, 4, 5, and 6 detection delays**

$t_{D\_CS}$	Detection delay coming from differential amplifier and comparator at $G_{DA\_DIFF(00)} = 5.8$	20		500	ns	(38)
-------------	---	----	--	-----	----	------

## Notes

38. Guaranteed by design.

**Table 13. Differential amplifier 1, 2, 3, 4, 5, and 6 (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Differential amplifier 1, 2, 3, 4, 5L, 5H, 6L and 6H analog offset compensation</b>						
$V_{\text{OFFDAC\_OUT\_MAX\_POS}}$ $V_{\text{OFFDAC\_OUT\_MAX\_NEG}}$	Offset compensation voltage range referred to amplifier output offset at maximum gain • Offset DAC value = +31 • Offset DAC value = -31	150 -310	— —	310 -150	mV	(40)
$V_{\text{OFFDAC\_LSB}}$	Offset compensation step size referred to amplifier output offset at maximum gain	5.0	—	10	mV	
$V_{\text{CS\_OFF\_TEMP}}$	Differential amplifier output offset temperature drift	-5.0 -50	— —	5.0 50	LSB mV	(39)
$V_{\text{CS\_OFF\_GD}}$	Residual offset after offset compensation at diff amplifier output for path shunt $\geq$ comparator output	-0.61 -6.1	— —	0.39 3.9	LSB mV	(41)
$t_{\text{OFFCOMP\_STEP}}$	Offset compensation minimum step time	—	—	2.0	$\mu\text{s}$	(40)
$t_{\text{OFFCOMP}}$	Offset compensation runtime to finish compensation	—	—	$2.0 \times 31 = 62$	$\mu\text{s}$	(40)(42)

**Notes**

39. Guaranteed by design.

40. Gain set to  $G_{\text{DA\_DIFF}(11)} = 19.3$

41. The offset compensation algorithm is implemented so the compensation always stops when the comparator output signal is low, assuming a zero DAC gain error and INL.

42. Assuming the start from an offset compensation DAC value of 0 is worst case, it has to go to one extreme value (-31 or 31).

## 5.9.2 Current measurement for negative currents

This section is applicable for all current measurement paths for negative currents:

- Current measurement channel 5 and 6
  - Differential amplifier 5 and 6 negative
  - DAC 5 and 6 negative
  - Comparator 5 and 6 negative

**Table 14. PT2000 static electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Overall current sense performance for negative</b>						
$\epsilon_{\text{CSNEG}}$	Overall current sense error including gain errors and offsets • at DAC range of 75% to 100% at $G_{\text{DANEG\_DIFF}} = -2.0$ • at DAC range of 25% to 75% at $G_{\text{DANEG\_DIFF}} = -2.0$	— —	— —	$\pm 4.4$ $\pm 8.9$	%	(43), (44)
<b>Differential amplifier 5, 6 negative</b>						
$V_{\text{VSENSE5/6\_DANEG}}$	Differential amplifier 5 and 6 negative (negative currents) functional range $V_{\text{SENSE N5/6}}$	-3.0	—	1.0	V	(43)
$V_{\text{VSENSE5/6\_DANEG}}$	Differential amplifier 5 and 6 negative (negative currents) functional range $V_{\text{SENSE P5/6}}$	-4.2	—	1.0	V	(43)
$V_{\text{DANEG\_DIFF\_IN}}$	Differential input voltage range • $G_{\text{DANEG\_DIFF}} = -2.0$	-1.125	—	0.0	V	(43)
$G_{\text{DANEG\_DIFF}}$	Differential voltage gain	-1.966	-2.0	-2.034		
$R_{\text{VSENSE5/6\_IN}}$	Input impedance $V_{\text{SENSE N5/6}}$ • 1.0 V common mode voltage	6.0	—	14	k $\Omega$	

**Table 14. PT2000 static electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Differential amplifier 5, 6 negative (continued)</b>						
$R_{VSENSE5/6\_IN}$	Input impedance $V_{SENSE}$ P5/6 • 1.0 V common mode voltage	6.0	—	14	k $\Omega$	
$V_{DANEG\_IN\_OFF}$	Differential amplifier maximum input offset voltage	-20	—	20	mV	
$V_{DANEG\_BIAS}$	Output bias voltage	240	250	265	mV	
$V_{DANEG\_OUT\_OFF}$	Maximum output offset voltage error, including amplifier input offset and bias voltage offset.	-60	—	60	mV	
$V_{DANEG\_OUT}$	Differential amplifier x output voltage range	0.0	—	2.7	V	
<b>DAC 5 Neg and 6 Neg (4 Bit)</b>						
$V_{DACNEG\_LSB}$	DAC LSB	—	156.3	—	mV	
$V_{DACNEG\_OUT\_MIN}$	DAC minimum output voltage • DAC code = 0h	—	0.0	—	V	
$V_{DACNEG\_OUT\_MAX}$	DAC maximum output voltage • DAC code = Fh	—	2.344	—	V	
$\epsilon_{DACNEG\_GAIN}$	DAC maximum gain error • Error of bandgap reference voltage	-1.0	—	1.0	%	
$\epsilon_{DACNEG\_DNL}$	DAC differential linearity error	-0.063	—	0.063	LSB	
$\epsilon_{DACNEG\_INL}$	DAC integral linearity error	-0.063	—	0.063	LSB	
$V_{DACNEG\_OUT\_OFF}$	DAC maximum output offset	0.0	—	10	mV	
$t_{DACNEG}$	DAC settling time	—	—	0.9	$\mu\text{s}$	
<b>Voltage comparator 5 Neg and 6 Neg</b>						
$V_{COMP\_IN}$	Comparator input voltage	0.0	—	2.7	V	
$V_{COMP\_IN\_OFF}$	Comparator input offset voltage	-25	—	10	mV	
<b>Current measurement channel 5 Neg and 6 Neg detection delays</b>						
$t_{D\_CSNEG}$	Detection delay coming from differential amplifier and comparator • $GDANEG\_DIFF = -2.0$	20	—	500	ns	(45)

**Notes**

43. Guaranteed by design.
44. The tolerance of the 10 m $\Omega$  shunt resistor is assumed as  $\pm 2.0\%$  (at 4.5  $\sigma$ ). All other input tolerances from the device specification are assumed at 6  $\sigma$ .
45. Guaranteed by design.

## 5.10 Analog output (OAx) electrical characteristics

**Table 15. Analog output static electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>OAx output PiNS, multiplexer and T&amp;H</b>						
$V_{OAX}$	OAx output voltage range	0.0	—	$V_{CC5}$	V	
$BW_{OAX}$	OAx output bandwidth	100	—	—	kHz	(46)
$C_{OAX}$	OAx permissible capacitive load	—	—	50	pF	(46)
	• without series resistor, for digital function	1.0	—	5.0	nF	
	• $R_{MIN} = 200\ \Omega$	5.0	—	15	nF	
	• $R_{MIN} = 100\ \Omega$	15	—	50	nF	
	• $R_{MIN} = 75\ \Omega$	50	—	100	nF	
$PSRR_{OAX}$	OAx power supply rejection	—	—	103	dB	(46)
$G_{OAX(00)}$	OAx output gain (00)	1.303	1.33	1.357		
$G_{OAX(01)}$	OAx output gain (01)	1.940	2.0	2.060		
$G_{OAX(10)}$	OAx output gain (10)	2.91	3.0	3.090		
$G_{OAX(11)}$	OAx output gain (11)	5.17	5.33	5.49		
$G_{OAX(ADC)}$	OAx output gain (ADC)	0.98	1.0	1.02		
$t_{OAX\_GAIN}$	OAx output gain switching time	—	—	2.0	$\mu\text{s}$	(46)
$V_{OAX\_OFFSET}$	OAx output offset voltage from OAx amplifier	—	—	—	mV	
	• $G_{OAx} = 1.0$	-14	—	14		
	• $G_{OAx} = 1.33$	-18	—	18		
	• $G_{OAx} = 2.0$	-28	—	28		
	• $G_{OAx} = 3.0$	-30	—	30		
• $G_{OAx} = 5.33$	-53	—	53			
$R_{OA1/3\_EN0}$	OA1/3 input impedance when $OaENx = 0$ • 2.0 V, impedance to GND	—	—	8000	k $\Omega$	
$R_{OA2\_EN0}$	OA2 input impedance when $OaENx = 0$ • 2.0 V, impedance to GND	350	—	500	k $\Omega$	
$t_{OAX\_MUX}$	OAx multiplexer switching time	—	—	10	$\mu\text{s}$	(46)
$V_{OAX\_DRIFT\_ADC}$	OAx output voltage drift of T&H in ADC mode over time • at $V_{OAx} = 1.5\text{ V}$ and after 20 $\mu\text{s}$	-50	—	50	mV	

Notes

46. Guaranteed by design.

## 5.11 Clock / PLL electrical characteristics

**Table 16. Clock / PLL electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

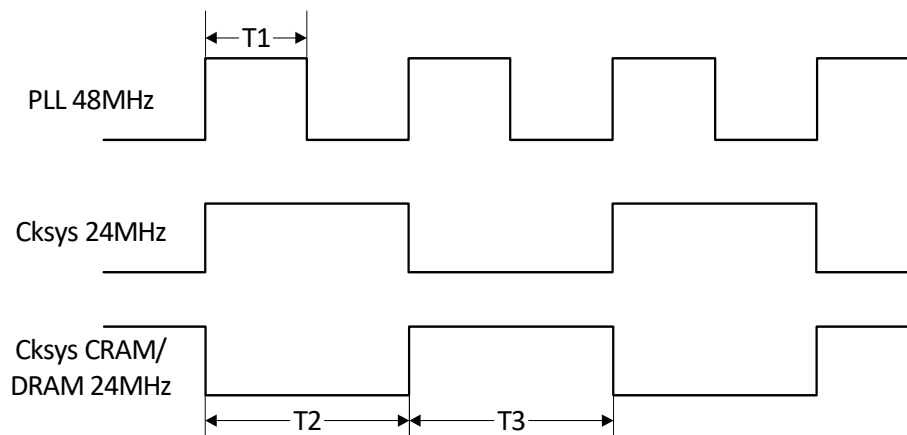
Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Backup clock</b>						
$f_{\text{CLK}}$	CLK pin input frequency	0.94	1.0	1.06	MHz	
$\text{DC}_{\text{CLK}}$	CLK pin input duty cycle	45	50	55	%	
$V_{\text{CLK}}$	CLK pin voltage	0.0	—	$V_{\text{CCIO}}$	V	
$V_{\text{IH\_CLK}}$	CLK pin high input voltage threshold	1.5	—	2.2	V	
$V_{\text{IL\_CLK}}$	CLK pin low input voltage threshold	1.0	—	1.65	V	
$V_{\text{HYST\_CLK}}$	CLK pin hysteresis	0.3	—	—	V	
$t_{\text{CLK\_JITTER}}$	CLK pin clock edge jitter	-25	—	25	ns	(47)
$f_{\text{CLK\_BACK}}$	Backup oscillator clock frequency	0.95	1.0	1.05	MHz	
$\text{DC}_{\text{CLK\_BACK}}$	Backup oscillator clock duty cycle	48	50	52	%	

### PLL

$f_{\text{CKSYS24}}$	Cksys output clock frequency 24 MHz	$f_{\text{CLK\_BACK}} \times 23.5$	$f_{\text{CLK\_BACK}} \times 24$	$f_{\text{CLK\_BACK}} \times 24.5$	MHz	
$f_{\text{CKSYS\_MOD}}$	Cksys modulation frequency	—	25	—	kHz	(48)
$t_{\text{PLL\_LOCK}}$	PLL lock time	—	25	40	$\mu\text{s}$	
$t_{\text{CKSYS\_T1}}$	Cksys rising edge to cksys_cram rising edge T1 CRAM address setup phase	8.75	—	12.32	ns	(49)
$t_{\text{CKSYS\_T2}}$	Cksys rising edge to cksys_c/dram rising edge T2 CRAM/DRAM address setup phase	19.44	—	—	ns	(49)
$t_{\text{CKSYS\_T3}}$	Cksys_c/dram rising edge to cksys rising edge T3 CRAM/DRAM access time	19.44	—	—	ns	(49)

### Notes

47. Guaranteed by design.
48. Divider value is changed every  $10\text{ }\mu\text{s}$
49. The following values take into account an input clock at 0.95 MHz to 1.05 MHz, a PLL multiplication factor of 47 to 49, and an output duty cycle of 45% to 55%.



**Figure 4. PLL, DRAM/CRAM system clock**

## 5.12 Digital input/output electrical characteristics

**Table 17. PT2000 static electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Digital I/Os</b>						
$V_{IOVCCIO}$	Digital pins voltage (IRQB, MISO, MOSI, SCLK, CSB, Startx, Flagx, DBG, OAx)	0.0	—	$V_{CCIO}$	V	
$V_{IOVCC5}$	Digital pins voltage (RESETB, DRVEN)	0.0	—	$V_{CC5}$	V	
$t_{DIGIOREADY}$	Digital output ready time after PORResetB deactivation	—	—	100	$\mu\text{s}$	
$t_{FILT\_RESETB}$	RESETB filter time	0.2	—	2.0	$\mu\text{s}$	
$V_{IH\_IO}$	Digital pins high input voltage threshold (RESETB, IRQB, MOSI, SCLK, CSB, DRVEN, Startx, Flagx, DBG, OAx)	1.5	—	2.2	V	
$V_{IL\_IO}$	Digital pins low input voltage threshold (RESETB, IRQB, MOSI, SCLK, CSB, DRVEN, STARTx, FLAGx, DBG, OAx)	1.0	—	1.65	V	
$V_{HYST\_IO}$	Digital pins hysteresis (RESETB, IRQB, MOSI, SCLK, CSB, DRVEN, STARTx, FLAGx, DBG, OAx)	0.3	—	—	V	
$V_{OH\_IO}$	Digital pins high output voltage (IRQB, MISO, START1-7, FLAG0-3, DBG) • $I_{OUT} > -1.0\text{ mA}$ , no higher current at other I/Os	$V_{CCIO} - 0.3$	—	—	V	
$V_{OL\_IO}$	Digital pins low output voltage (IRQB, MISO, START1-7, FLAG0-3, DBG) • $I_{OUT} < 1.0\text{ mA}$ , no higher current at other I/Os	—	—	0.3	V	
$V_{OH\_START8/FLAG4}$	Digital pins high output voltage (Start8, Flag4) • $I_{OUT} > -200\text{ }\mu\text{A}$	$V_{CC0} - 0.6$	—	—	V	
$V_{OL\_START8/FLAG4}$	Digital pins low output voltage (Start8, Flag4) • $I_{OUT} > -200\text{ }\mu\text{A}$	—	—	0.6	V	
$V_{OH\_OA2}$	Digital pins high output voltage (OA2) • $G_{OAx} = 1.33$ , $I_{OUT} > -1.0\text{ mA}$ • $G_{OAx} = 2.0$ , $I_{OUT} > -1.0\text{ mA}$	2.8 $V_{CC5} - 0.6$	— —	— —	V	
$V_{OL\_OA2}$	Digital pins low output voltage (OA2), $I_{OUT} < 0.5\text{ mA}$	—	—	0.3	V	
$t_{R\_XXX}$	Digital pins output rise time (IRQB, START1-7, FLAG0-3, DBG) • $C_{LOAD} = 30\text{ pF}$ , 10%-90% of out voltage	3.0	—	12	ns	
$t_{F\_XXX}$	Digital pins output fall time (IRQB, START1-7, FLAG0-3, DBG) • $C_{LOAD} = 30\text{ pF}$ , 90%-10% of out voltage	3.0	—	12	ns	
$t_{D\_XXX}$	Digital pins output delay (IRQB, START1-7, FLAG0-3, DBG) • $C_{LOAD} = 30\text{ pF}$ , 10% of out voltage	2.0	—	10	ns	

**Table 17. PT2000 static electrical characteristics (continued)**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>Digital I/Os (continued)</b>						
$t_{R\_OA2}$	Digital pins output rise time (OA2), 10%-90% of out voltage • $C_{LOAD} = 30\text{ pF}$ with $V_{CCIO} = 3.3\text{ V}$ • $C_{LOAD} = 30\text{ pF}$ with $V_{CCIO} = 5.0\text{ V}$	— —	— —	1.4 2.0	$\mu\text{s}$	(50)
$t_{F\_OA2}$	Digital pins output fall time (OA2), 90%-10% of out voltage • $C_{LOAD} = 30\text{ pF}$ with $V_{CCIO} = 3.3\text{ V}$ • $C_{LOAD} = 30\text{ pF}$ with $V_{CCIO} = 5.0\text{ V}$	— —	— —	1.4 3.2	$\mu\text{s}$	(50)
$t_{D\_OA2}$	Digital pins output delay (OA2), 10% of out voltage • $C_{LOAD} = 30\text{ pF}$ with $V_{CCIO} = 3.3\text{ V}$ • $C_{LOAD} = 30\text{ pF}$ with $V_{CCIO} = 5.0\text{ V}$	— —	— —	2.7 3.0	$\mu\text{s}$	(50)
$t_{R\_START8/FLAG4}$	Digital pins output rise time (START8, FLAG4) • $C_{LOAD} = 30\text{ pF}$ , 10%-90% of out voltage	60	—	200	ns	(50)
$t_{F\_START8/FLAG4}$	Digital pins output fall time (START8, FLAG4) • $C_{LOAD} = 30\text{ pF}$ , 90%-10% of out voltage	60	—	200	ns	(50)
$t_{D\_START8/FLAG4}$	Digital pins output delay (START8, FLAG4) • $C_{LOAD} = 30\text{ pF}$ , 10% of out voltage	5.0	—	20	ns	(50)
$C_{PIN\_XXX}$	Digital pins equivalent pin capacitance (IRQB, STARTx, FLAGx, DBG)	—	—	10	pF	(50)
$C_{PIN\_MISO}$	Digital pin equivalent pin capacitance (MISO)	—	—	10	pF	(50)
$C_{PIN\_MOSI}$	Digital pin equivalent pin capacitance (MOSI)	—	—	10	pF	(50)

**Pull-up/down resistors**

$R_{W\_PU/PD}$	Weak pull-up/down resistor	200	480	800	$\text{k}\Omega$	
$R_{PU/PD}$	Pull-up/down resistor	50	120	200	$\text{k}\Omega$	

## Notes

50. Guaranteed by design.

## 5.13 Serial peripheral interface electrical characteristics

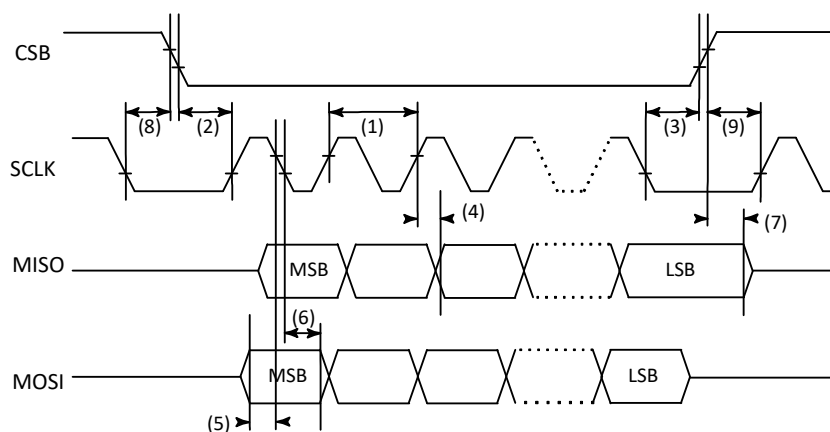


Figure 5. SPI timing

Table 18. SPI electrical characteristics

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>SPI</b>						
$f_{\text{SCLK}}$	SCLK pin input frequency - (1)	—	—	12.5	MHz	(51)
$t_{\text{CSBF\_SCLKR}}$	CSB fall to first SCLK rise - (2)	$1/f_{\text{SCLK}}$	—	—	ns	(51)
$t_{\text{SCLKF\_CSBR}}$	Last SCLK fall to CSB rise - (3)	$1/f_{\text{SCLK}}$	—	—	ns	(51)
$t_{\text{MISO\_VAL}}$	MISO valid time - (4)	—	—	$10 + t_{\text{R\_MISO}}$	ns	(51)
$t_{\text{MOSI\_SET}}$	MOSI setup time - (5)	10	—	—	ns	(51)
$t_{\text{MOSI\_HOLD}}$	MOSI hold time - (6)	12.5	—	—	ns	(51)
$t_{\text{CSBR\_MISOT}}$	CSB rise to MISO tri-state - (7)	—	—	15	ns	(51)
$t_{\text{SCLKF\_CSBF}}$	SCLK fall (other device) to CSB fall - (8)	13	—	—	ns	(51)
$t_{\text{CSBR\_CLKR}}$	CSB rise to SCLK rise (other device) - (9)	15	—	—	ns	(51)
$t_{\text{SPI\_RESETB\_t0}}$	SPI setup time after first RESETB rising edge	100	—	—	$\mu\text{s}$	(51)
$t_{\text{SPI\_RESETB}}$	SPI setup time after each following RESETB rising edge	30	—	—	$\mu\text{s}$	(51)

Notes

51. See Figure 5



**Table 18. SPI electrical characteristics**

Characteristics noted under conditions  $-40\text{ }^{\circ}\text{C} < T_A < +125\text{ }^{\circ}\text{C}$ , referenced to ground, unless otherwise noted. Typical values noted reflect the approximate parameter means at  $T_A = 25\text{ }^{\circ}\text{C}$  under nominal conditions, unless otherwise noted.

Symbol	Characteristic	Min.	Typ.	Max.	Unit	Notes
<b>SPI MISO driver with VCCIO = 3.3 V</b>						
$t_{R\_MISO\_S3.3}$	MISO rise time slow setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	10 20 40	— — —	30 50 90	ns	
$t_{F\_MISO\_S3.3}$	MISO fall time slow setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	10 20 40	— — —	30 50 90	ns	
$t_{R\_MISO\_F3.3}$	MISO rise time fast setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	1.5 2.7 4.4	— — —	13.4 17.1 23.9	ns	
$t_{F\_MISO\_F3.3}$	MISO fall time fast setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	1.5 2.7 4.4	— — —	13.4 17.1 23.9	ns	
<b>SPI MISO driver with VCCIO = 5.0 V</b>						
$t_{R\_MISO\_S5.0}$	MISO rise time slow setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	9.0 20 35	— — —	25 47 77	ns	
$t_{F\_MISO\_S5.0}$	MISO fall time slow setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	9.0 20 35	— — —	25 47 77	ns	
$t_{R\_MISO\_F5.0}$	MISO rise time fast setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	1.1 2.1 3.6	— — —	9.6 12.5 17.8	ns	
$t_{F\_MISO\_F5.0}$	MISO fall time fast setting: <ul style="list-style-type: none"> <li>• CL = 30 pF</li> <li>• CL = 75 pF</li> <li>• CL = 150 pF</li> </ul>	1.1 2.1 3.6	— — —	9.6 12.5 17.8	ns	

# 6 Functional block description

## 6.1 Power supplies

### 6.1.1 VCC5

The PT2000 works with an external supply voltage of 5.0 V connected to the VCC5 pin. This voltage is used for the VCC2P5 regulator, the load biasing, and to supply the analog blocks. The VCC5 overvoltage monitor is used to disconnect the device from the VCC5 supply in any overvoltage condition at this pin. This is done to guarantee 36 V robustness of the VCC5 pin.

The VCC5 undervoltage monitor is used to disable all the pre-drivers, as long as the supply voltage at the VCC5 pin is not high enough to guarantee full functionality of the analog modules of the device. An interrupt request (in case it is enabled) is issued to the microcontroller as soon as `uv_vcc5` is asserted.

### 6.1.2 VCCIO

The interfaces toward the ECU microcontroller uses the VCCIO voltage for the output drivers and receives the same levels on its inputs. The voltage level of the I/O pins interfacing the device with the microcontroller (STARTx, FLAGx, RESETB, DBG, SPI pins, CLK, IRQB, DRVEN, and OA\_x, when used as digital I/O) can be selected between 3.3 V and 5.0 V by supplying the device with the desired voltage at the VCCIO pin. This does not change the possible output voltage range of the analog outputs OA\_x when used as analog outputs because they are supplied by VCC5

Note that the DRVEN and RESETB input pins can operate with an input level of 5.0 V, even when VCCIO is 3.3 V.

### 6.1.3 VCC2P5 regulator

An integrated voltage regulator, fed by the VCC5 supply pin, provides 2.5 V to supply the logic core of the device. An external buffer capacitor (1.0  $\mu$ F recommended) needs to be connected at the VCC2P5 pin. The regulator, as well as the buffer capacitor, is referenced to digital ground (DGND pin).

If the VCC2P5 voltage is below the undervoltage threshold ( $V_{PORESETB-}$  for a minimum duration of  $t_{PORESETB}$ ), the power on reset signal (VCC2P5) is asserted to the logic core after a delay of  $t_{D\_PORESETB}$  and resets the logic core and all device internal modules.

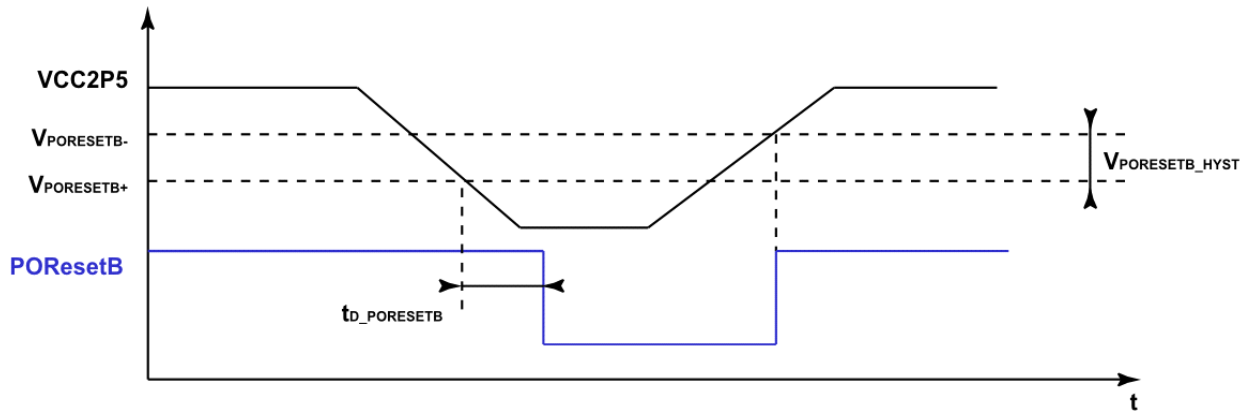


Figure 6. PORESETB

## 6.1.4 VCCP regulator

### 6.1.4.1 Internal V<sub>CCP</sub> (vccp\_ext\_en='0')

The voltage source at the VBATT input pin provides power for the V<sub>CCP</sub> regulator. This integrated linear regulator provides typically 7.0 V at the VCCP pin, to supply the pre-driver section of the device. The regulator uses low drop out features to extend the system's operating range when V<sub>BATT</sub> temporarily falls below its normal operating range, for example during engine crank conditions. This avoids problems caused by insufficient gate voltage, such as slow MOSFET switching and increased on-state losses. A capacitor (4.7 μF recommended) is required at the VCCP pin to provide the high peak currents required when charging a MOSFET gate.

At low V<sub>CC5</sub>, the regulator may be active, but with an increased dropout voltage. The low dropout mode of the regulator is active only when the voltage at VCC5 is above the V<sub>CC5</sub> undervoltage threshold V<sub>UVVCC5+</sub>.

If V<sub>CC5</sub> is not present or low, POResetB is active and disables the V<sub>CCP</sub> regulator.

V<sub>CCP</sub> during bootstrap initialization:

1. If the DBG pin is not used as a digital I/O, the DBG pin logic level is '1' during reset due to the device internal weak pull-up resistor. As result, the internal VCCP regulator is switched on during the HS pre-driver bootstrap initialization phase. (refer to [See Power-up sequence VCCP and bootstrap capacitors on page 58](#)).
2. If the DBG pin is used as a digital I/O. This means the DBG pin logic level is undefined during reset and may be '0'. As result, the internal VCCP regulator's On status during HS pre-driver bootstrap init phase can only be guaranteed if vccp\_ext\_en is set to '0'. The bit has to be configured as soon as possible during device init, to start the pre-charging of the bootstrap capacitors soon enough (refer to [Table 148, Driver\\_config\\_Part 2 \(1A6h\)](#)).

Note that this regulator also needs a charge pump voltage coming from the VBOOST pin. Due to this, the V<sub>CCP</sub> regulator is not functional when the voltage on the VBOOST pin is below 4.7 V.

### 6.1.4.2 External V<sub>CCP</sub> (vccp\_ext\_en='1')

The VCCP can also be powered by an external voltage source connected to the VCCP pin. The internal V<sub>CCP</sub> regulator is sized for 12 V system operation, including the ISO voltage transients specified for those systems. But for 24 V system operation, the internal V<sub>CCP</sub> linear regulator dissipates too much power. In this case, the internal V<sub>CCP</sub> regulator should be switched off by setting the vccp\_ext\_en bit of the driver\_config\_part2 register (1A6h) to '1' and an external regulator needs to be used.

VCCP during initialization VCCP:

1. If is not possible to turn on the internal V<sub>CCP</sub> regulator for a limited time in parallel to the external voltage source, the DBG pin has to be tied to logic level '0' and the SPI configuration bit has to stay at '1', to strictly avoid the internal regulator to be switched on at any time. The logic level of the DBG pin should be forced by a pull-down resistor towards GND.
2. On the other hand, if it is possible to switch on the internal regulator for some limited time in parallel with the external voltage supply without destroying it, no special measures have to be taken during startup. After bootstrap initialization, the vccp\_ext\_en bit has to be set to '1' (refer to [Table 148, Driver\\_config\\_Part 2 \(1A6h\)](#)).

### 6.1.4.3 V<sub>CCP</sub> undervoltage

V<sub>CCP</sub> voltage is internally monitored by a voltage comparator to detect if it is in the operating range. In case of an undervoltage when falling below the lower threshold, the gate driver outputs are switched off by the digital core.

This prevents possible malfunctions and/or failures: in case of an undervoltage, operations are stopped before any malfunction, due to insufficient gate driver supply voltage. Moreover, in case of a battery voltage disconnection, all MOSFETs are switched off (and therefore inductive loads are disconnected) before the electrolytic capacitors on the V<sub>BATT</sub> line are completely discharged. This prevents any negative voltage on the V<sub>BATT</sub> line, which may cause failures on the VBATT, VCCP, D\_HSx and B\_HSx pins due to exceeding its maximum ratings.

## 6.1.5 Battery voltage monitor

The device includes a battery voltage measurement block which measures the voltage at the VBATT pin with a DAC and comparator running in ADC mode. Figure 7 shows the structure of the analog part of the block. The battery voltage is divided with a voltage divider by 16. The battery voltage can be measured in the range of 5.0 V to 36 V with a resolution of 6 bits. The digital core permanently performs the battery voltage measurements. The result is available both via a SPI register and the internal microcore memory map.

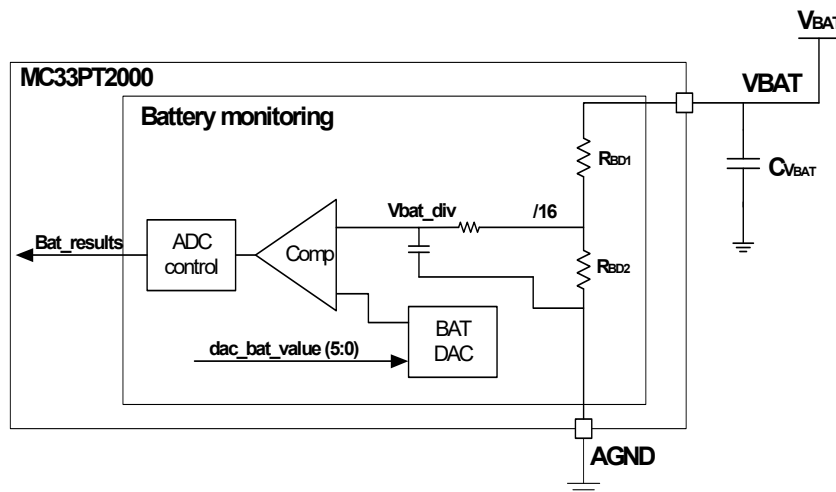


Figure 7. Battery voltage monitoring

The battery voltage threshold can be calculated using the following formula. Table 19 shows some example values.

$$V_{BAT} = (\text{DAC\_VALUE} * 39.06 \text{ mV}) * 16$$

Table 19.  $V_{BAT}$  voltage DAC values

DAC value		DAC output voltage/mV	$V_{BAT}$ upper threshold		
HEX	DEC		Min./V	Typ./V	Max./V
08	8	313	...	5.0	...
...	...	...	...	...	...
16	22	859.4	...	13.75	...
...	...	...	...	...	...
39	57	2226.6	...	35.63	...

## 6.1.6 Boost voltage monitor

The boost voltage monitor implements the voltage regulation of the DC/DC converter without external components. The boost voltage monitor contains:

- A high accuracy internal voltage divider dividing the voltage at the VBOOST pin to a smaller level  $V_{BOOST\_DIV}$
- A programmable DAC (8 bits) either by the SPI (refer to Table 113. Boost\_dac (17Fh)) or by microcode creating a reference voltage
- A comparator comparing the reference voltage with the  $V_{BOOST\_DIV}$

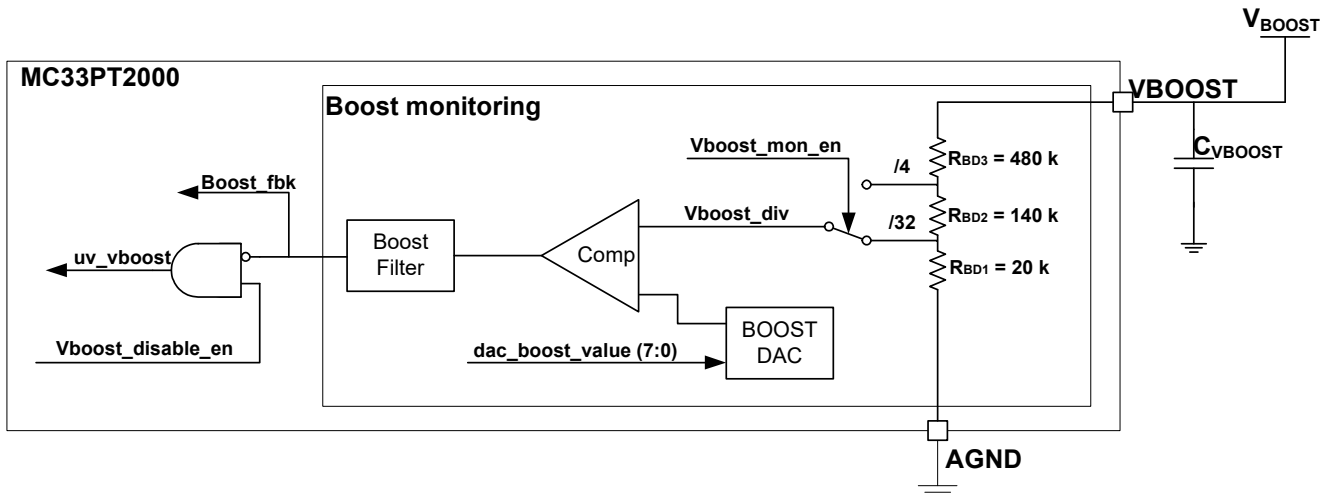


Figure 8. Boost voltage monitor block diagram

When boost voltage is required to drive injectors, the boost voltage monitor use a voltage comparator with a very accurate DAC threshold, whether the  $V_{\text{BOOST}}$  boost voltage exceeds the desired target value. All high-side pre-drivers are disabled when the voltage at the VBOOST pin is less than its undervoltage lockout threshold, which is around 4.7 V.

In applications without boost voltage, the battery voltage is connected to the VBOOST pin to supply the internal charge pump. In such applications, the boost voltage monitor can be used to detect an undervoltage at the VBOOST pin ( $v_{\text{boost\_mon\_en}} = 0$ ). This means the VBOOST pin is used to detect battery undervoltage.

### 6.1.6.1 Application with $V_{\text{BOOST}}$ voltage

The boost voltage threshold can be calculated using the following formula. Table 20 shows some example values.

- $V_{\text{BOOST}} = (\text{DAC\_VALUE} * 312.5 \text{ mV})$

Due to the compensation, concept values below 08h should not be used. Values higher than E1h must not be used, because this results in a boost voltage higher than 72 V which destroys the device. The real maximum value for the boost set point threshold in the application is even lower due to dynamic effects like voltage drop in the boost capacitor. It is not recommendable to use a DAC value above D0h (65 V).

Table 20. Boost voltage DAC values

DAC value		DAC output voltage/mV	$V_{\text{BOOST}}$ upper threshold		
HEX	BIN		Min./V	Typ./V	Max./V
08	8	78	2.45	2.50	2.55
...	...	...	...	...	...
9A	154	1504	47.16	48.13	49.09
...	...	...	...	...	...
B0	176	1719	53.90	55.00	56.10
...	...	...	...	...	...
D0	208	2031	63.70	65.00	66.30
...	...	...	...	...	...
E1	225	2197	68.91	70.31	71.72

### 6.1.6.2 Application without boost voltage

For this purpose, it is possible to change the internal voltage divider ratio from 1/32 to 1/4 by setting the signal `boost_mon_en` high. To detect undervoltage and use the signal `uv_vboost` to disable the pre-drivers, the `uv_vboost` bit needs to be set to "1" (refer to [Table 162, driver\\_status \(1B2h\)](#)). The `uv_vboost` signal goes high as soon as the voltage at the VBOOST pin is below the threshold, if the VBOOST UV monitor is enabled (`Vboost_disable_en = 1`).

The same digital filter used for the VBOOST voltage measurement is also used for the VBOOST UV monitoring mode. The DAC set point value in this mode has to be chosen to fulfill two requirements:

- The pre-drivers must not be disabled at a battery voltage above 5.0 V
- The device internal charge pump only works properly down to a battery voltage of 4.7 V

The VBOOST UV threshold can be calculated using the following formula.

- $V_{\text{BOOST}} = (\text{DAC\_VALUE} * 39.1 \text{ mV})$

### 6.1.7 Charge pump

The PT2000 provides one charge pump with independent outputs for each of the seven high-side drivers. The independent outputs allow complete flexibility of the topology used, meaning all high-sides can drive MOSFETs with the drain connected to VBOOST or VBAT. But there is a limitation on the diagnostics only HS2, 4, and 6 can use the VBOOST for monitoring (for example, VBAT or VBOOST).

In most operating topologies and conditions, the bootstrap is the primary source of charge for the bootstrap capacitor, and the charge pump sustains the voltage at each bootstrap capacitor when it is not being charged by low-side switching.

This charge pump allows 100% duty cycle operation of the high-side MOSFETs while the bootstrap circuitry is not operating (`VS_HSx` voltage never goes significantly below the `VCCP` voltage). In this condition, the charge pump provides current maintaining each bootstrap capacitor charged via independent current sources, to guarantee a minimum `VGS` voltage.

The charge pump, supplied by VBOOST, creates gate drive voltages of about 8.0 V greater than the voltage at VBOOST. However, their current capacity is sufficient only for low frequency switching. The charge pump is not running as long as the `POResetB` reset signal is active.

The internal CP can be used to charge the bootstrap capacitors during init with a guaranteed current of 20  $\mu\text{A}$  per HS pre-driver. This current is only available if there is no leakage current from `B_HSx` pin. Any possible leakage current has to be subtracted from this available charge current (See [Using the charge pump to charge bootstrap capacitors on page 59](#)).

## 6.2 Clock subsystem

The digital logic is supplied by a clock (`cksys`) generated by the PLL from the 1.0 MHz clock forced externally on CLK pin. Two internal clocks are derived from the PLL:

- the main logic clock `cksys`
- the code RAM clock `cksys_cram` inverted in respect to `cksys`
- the Data RAM clock `cksys_dram` inverted in respect to `cksys`

If an unsuitable signal is applied on the CLK pin, the device automatically switches to the internal backup clock. The PLL output frequency can be modulated for EMC purposes. Modulation activation is enabled by default, but can be disabled by the SPI. Refer to [Table 151 PLL\\_Config \(1A7h\)](#).

## 6.3 High-side pre-driver

The PT2000 provides seven independent high-side pre-drivers designed to drive the gate of external high-side configuration N-channel logic level MOSFETs. These pre-drivers are dedicated to load driving like injectors or solenoids, and integrate diagnostics features.

Internal to the device is a gate to source pull-down resistor holding the external MOSFETs in the off state while the device is in a power on reset state (RSTB low). The external FET can be connected to either VBATT or a higher voltage VBOOST, limitation in the diagnostics is described in the next chapter, only HS2, 4, and 6 can use VBOOST voltage for diagnostics.

The high-side pre-drivers are supplied by an external bootstrap capacitor connected between the S\_HSx and B\_HSx pins. The driver slew rate can be selected individually for each of the seven drivers, among a set of four value pairs by the SPI registers (refer to [Table 99](#), [Hs\\_slewrte \(171h\)](#)).

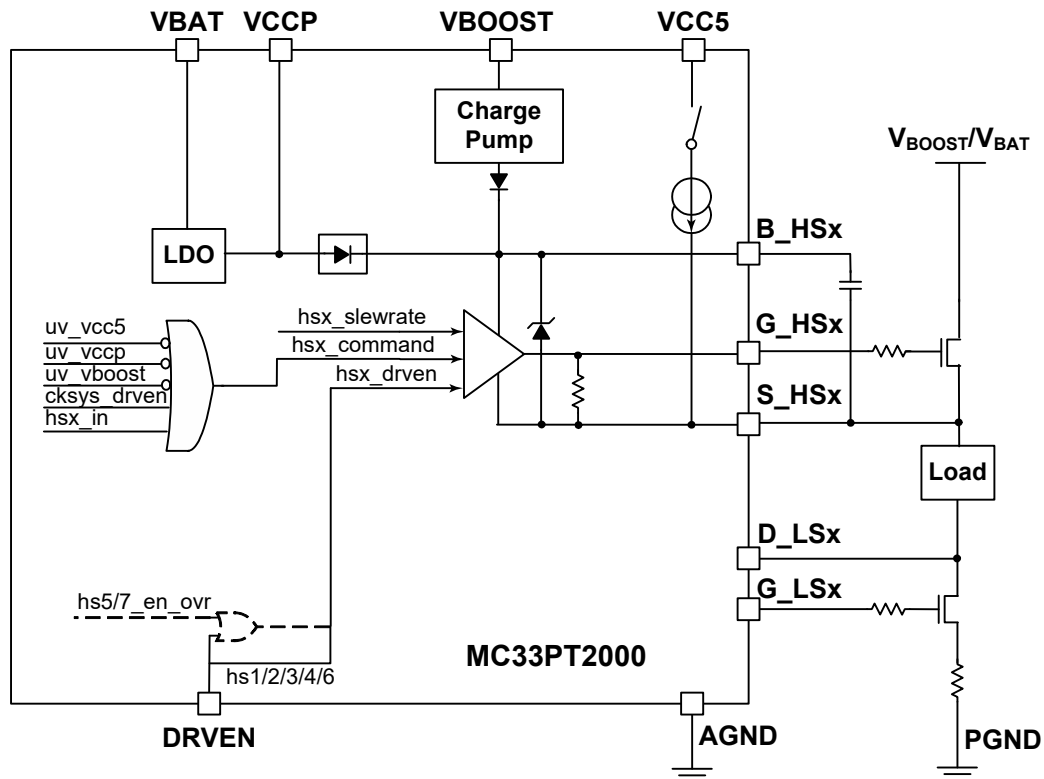


Figure 9. High-side pre-driver block diagram

The high-side pre-driver is intended to drive the gate of an external logic level MOSFET in a high-side configuration. The logic command, hsx\_command, to switch the external MOSFET, is provided by the microcores. This command is generated for taking into account the following signals:

- Logic command coming from channel logic (hsx\_in)
- VCCP undervoltage signals (uv\_vccp) from VCCP UV monitor: in case of an undervoltage, the external MOSFET is switched off
- VCC5 undervoltage signals (uv\_vcc5) from VCC5 UV monitor: in case of an undervoltage, the external MOSFET is switched off
- VBOOST undervoltage signals (uv\_vboost) from boost voltage monitor: in case of an undervoltage, the external MOSFET is switched off if this feature is enabled (refer to [Table 162](#), [driver\\_status \(1B2h\)](#))
- Signal cksys\_drven coming from the clock monitoring: in case of a missing clock (PLL not locked), the external MOSFET is switched off. This function is disabled by default and can be enabled by setting the cksys\_missing\_disable\_driver bit high (refer to [Table 152](#), [backup\\_clock\\_status \(1A8h\)](#))
- At the high-side pre-driver block signal, DrvEn is added to the control signal for the driver. As long as the DrvEn signal is negated (low), the high-side pre-driver is switched off. The high-side pre-driver 5 and 7 include a feature to override the switch off path via the DrvEn signal. As long as the signal hsx\_en\_ovr is high (only for HS5 and HS7), the pre-driver is not influenced by DrvEn. (refer to [Table 178](#), [HSx\\_output\\_config \(1DA, 1DDh, 1E0h, 1E3h, 1E6h, 1E9h\)](#))

The truth table describing the status of hsx\_command signal is given in [Figure 21](#).

**Table 21. High-side pre-driver truth table**

DRV_EN	hs5/7_en_ovr	uv_vccp	uv_vcc5	uv_vboost	cksys_drven	hsx_in	hsx_command	Driver status
0	-	-	-	-	-	-	0	off
-	-	1	-	-	-	-	0	off
-	-	-	1	-	-	-	0	off
-	-	-	-	1	-	-	0	off
-	-	-	-	-	0	-	0	off
-	-	-	-	-	-	0	0	off
0	1	0	0	0	1	1	1	on (hs5/7) / off (other hsx)
1	-	0	0	0	1	1	1	on

The pre-driver G\_HSx output is set according to hsx\_command:

- When the hsx\_command is high, the G\_HSx pin is driven high (pull-up to B\_HSx voltage);
- When the hsx\_command is low, the G\_HSx pin is driven low (pull-down to S\_HSx voltage).

### 6.3.1 High-side pre-driver slew rate control

The driver strength can be selected individually for each of the drivers among a set of values by the SPI registers. There are four selectable driver strengths. The strength for the rising and falling edge can be chosen individually for each driver. Changing the rising edge affects the falling edge such as to retain the same absolute slew rate. The given value of voltage slew rate is only an indication and is dependent on the used MOSFET and the additional gate circuit (refer to [Slew rate high-side and low-side selection register](#)).

**Table 22. Slew rate settings for HS pre-drivers**

hsx_slewrte(1:0)	Slew-rate/ V/ $\mu$ S	R <sub>DS(ON)_PMOS</sub> (switching on)/ $\Omega$	R <sub>DS(ON)_NMOS</sub> (switching off)/ $\Omega$
00	300	14.6	5.9
01	50	85	35
10	25	169	69
11	12.5	337	138

### 6.3.2 Safe state of high-side pre-driver

When reset (RSTB) is asserted or DRV\_EN is negated, the G\_HSx output is immediately forced to a low level, thus switching off the external MOSFET, to guarantee a safe condition while the device is not operating. In addition to this, an integrated pull-down resistor R<sub>PD\_HSx</sub> between G\_HSx and S\_HSx of about 1.0 M $\Omega$  keeps the external MOSFET in the off state even when the bootstrap voltage is low.



### 6.3.3 High-side pre-drivers in low-side configuration

All high-side pre-drivers can be used as low-side pre-drivers. In this configuration, an external bootstrap capacitor is still required because B\_HSx can't be connected to VCCP directly. The drain contact of this high-side pre-driver is still connected to V\_BOOST or V\_BATT internally, so the V\_DS monitoring for this low-side MOSFET is not functional.

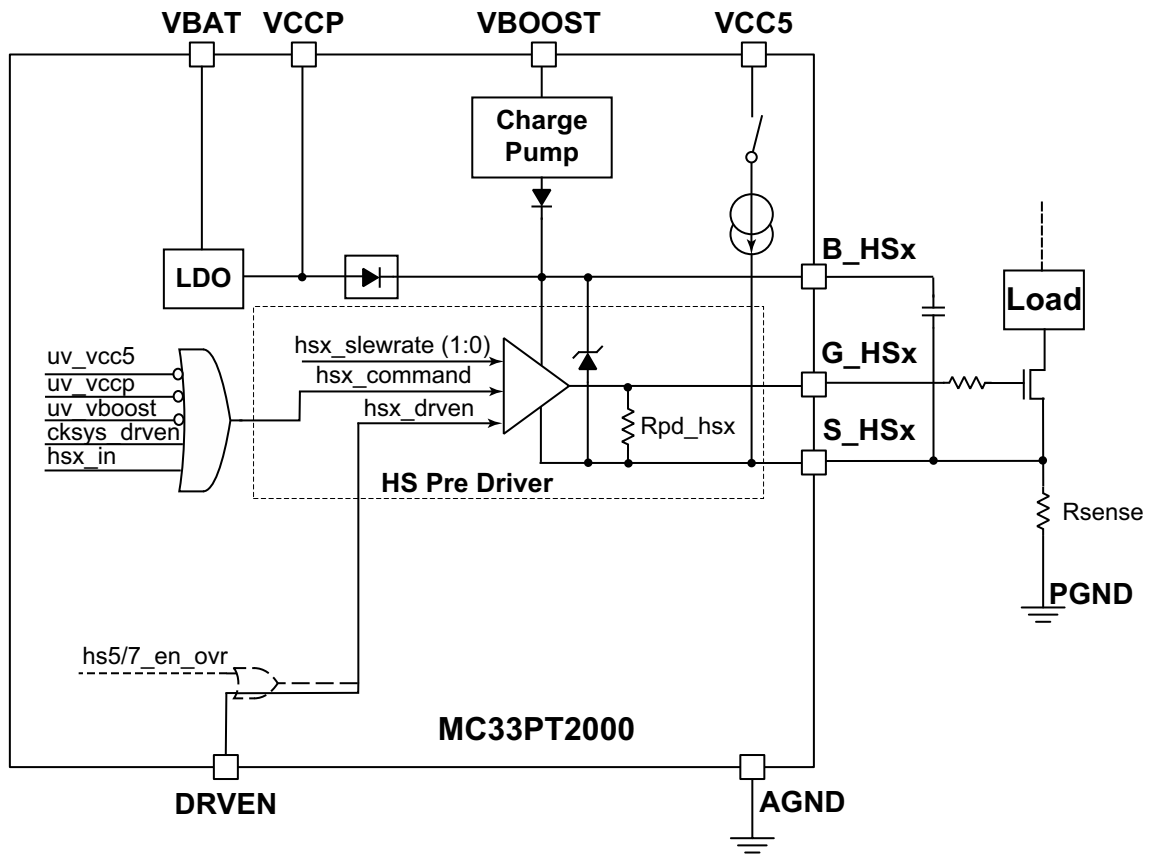


Figure 10. High-side pre-driver in low-side configuration

## 6.4 High-side VDS and VSRC monitor

The PT2000 monitors VDS and VSRC for diagnostic purposes across each high-side to detect any fault occurring on the external MOSFET. The drain and source voltages of the connected MOSFETs are needed for  $V_{DS}$  monitoring. To save pins, the high-side  $V_{DS}$  monitors have no dedicated drain pins, and the drain voltage of the MOSFETs is available via pins VBATT, for the MOSFETs connected to battery voltage (HS1, HS3, HS5, HS7) and  $V_{BOOST}$  or  $V_{BATT}$  for the MOSFETs connected to boost or battery voltage (HS2, HS4, and HS6).

### 6.4.1 HS1, 3, 5, 7 $V_{DS}$ monitoring

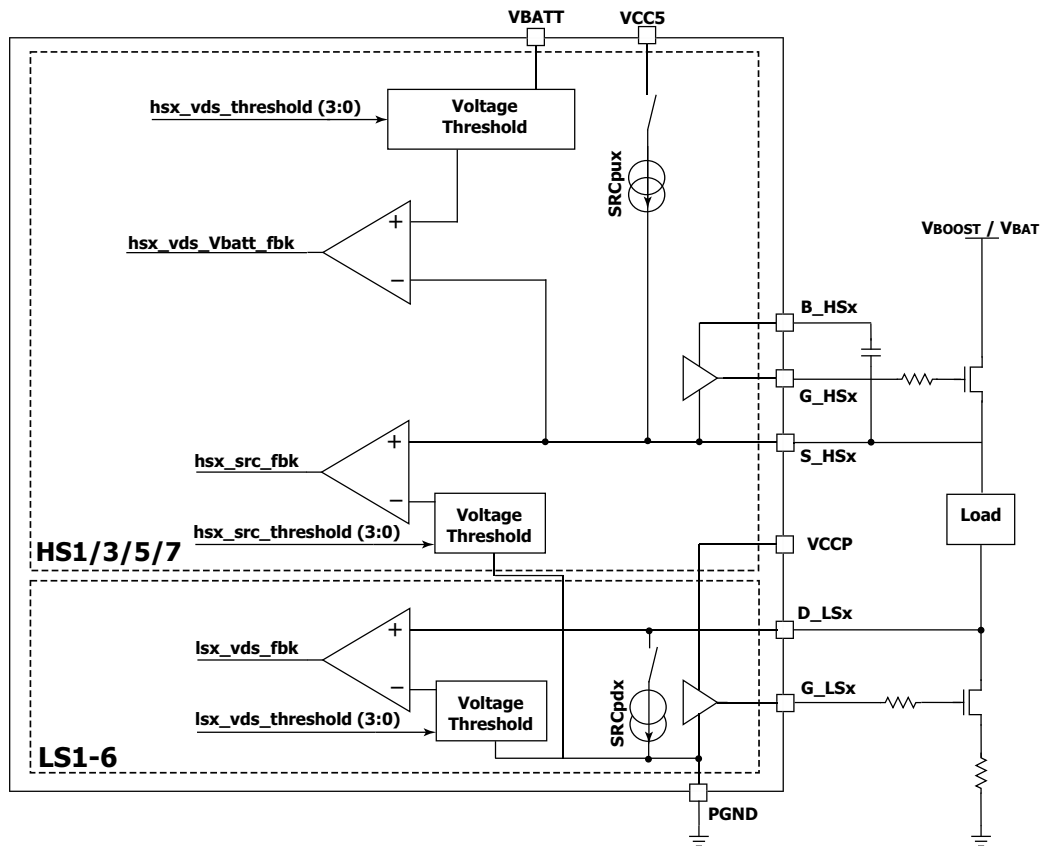


Figure 11. High-side 1, 3, 5, and 7  $V_{DS}$  VSRC and LS1-6  $V_{DS}$  monitoring

Four high-side  $V_{DS}$  monitors for high-side pre-drivers 1, 3, 5, and 7 are composed of two comparators with programmable thresholds, the first one sensing the voltage between  $V_{BATT}$  and the source pin  $S_{HSx}$  and the second one sensing the voltage between the source pin  $S_{HSx}$  and PGND (voltage across the free-wheeling element, either a diode or a MOSFET). A simplified schematic of the high-side  $V_{DS}$  monitors of HS pre-driver 1, 3, 5, and 7 is shown in Figure 11.

$V_{DS}$  and  $V_{SRC}$  threshold are selectable by the SPI using registers `vds_threshold_hs` and `vsrc_thresholds_hs` (refer to [VDS and VSRC threshold selection](#)). Selectable values are shown in [Table 23](#).

## 6.4.2 HS2, 4, 6 $V_{DS}$ monitoring

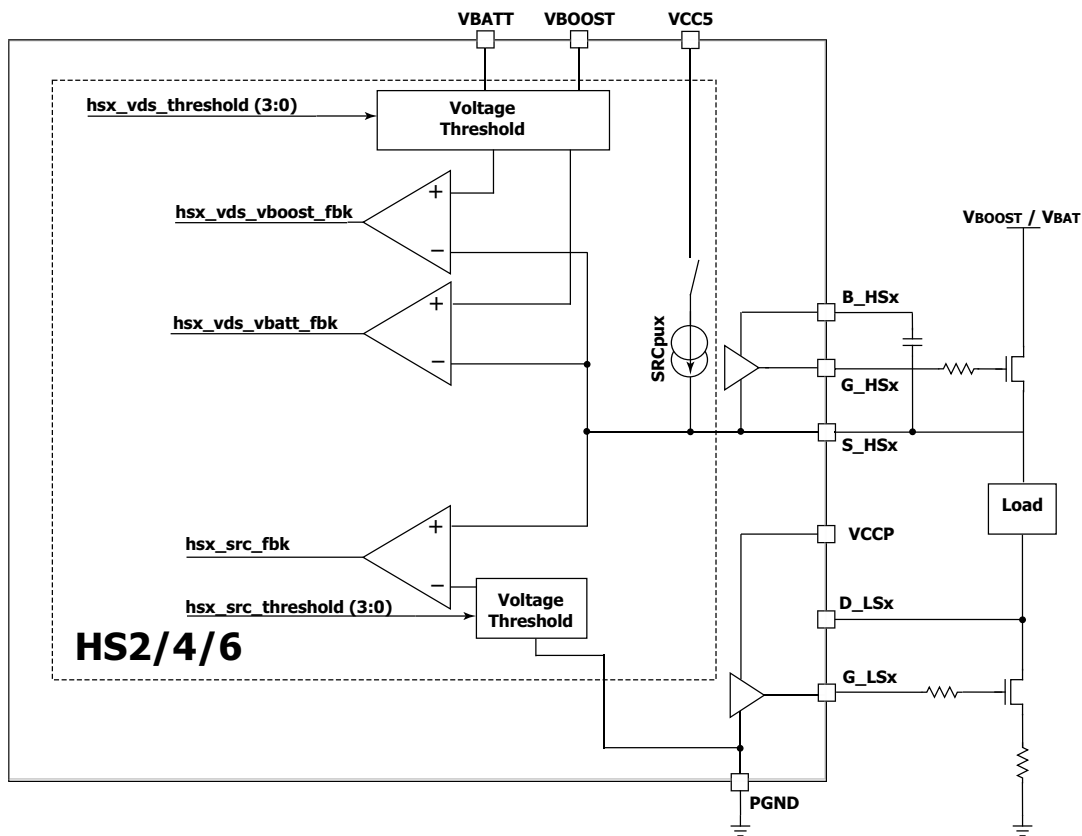


Figure 12.  $V_{DS}$  monitors and load biasing HS2, 4, and 6

The three high-side  $V_{DS}$  monitors of pre-drivers 2, 4, and 6 are composed of three comparators with programmable thresholds, the first one sensing the voltage between  $V_{BOOST}$  and the source pin  $S_{HSx}$  ( $V_{DS}$  of the high-side MOSFET used as the boost MOSFET) and the second one sensing the voltage between  $V_{BATT}$  and the source pin  $S_{HSx}$  ( $V_{DS}$  of the high-side MOSFET used for battery MOSFET and voltage information for voltage based diagnostics when the MOSFET is in boost configuration) and the third one sensing the voltage between the source pin  $S_{HSx}$  and  $PGND$  (voltage across the free-wheeling element, either a diode or a MOSFET). A simplified schematic of the high-side  $V_{DS}$  monitors of HS pre-driver 2, 4, and 6 is shown in Figure 12.

The selection between “ $V_{BATT}$ ” or  $V_{BOOST}$  is done with the microcode command `slfbk` (reference [Programming Guide and Instruction Set](#)).  $V_{DS}$  and  $V_{SRC}$  threshold are selectable by the SPI using registers `vds_threshold_hs` and `vsrsrc_thresholds_hs` (refer to [VDS and VSRC threshold selection](#)). Selectable values are shown in Table 23.

Table 23.  $V_{DS}$  monitor threshold selection

hsx_vds/src_threshold(3:0)	Threshold voltage HS VDS / HS VSRC
0000	0.00
1001	0.10
1010	0.20
1011	0.30
1100	0.40
0001	0.50
0010	1.0
0011	1.5

**Table 23.  $V_{DS}$  monitor threshold selection (continued)**

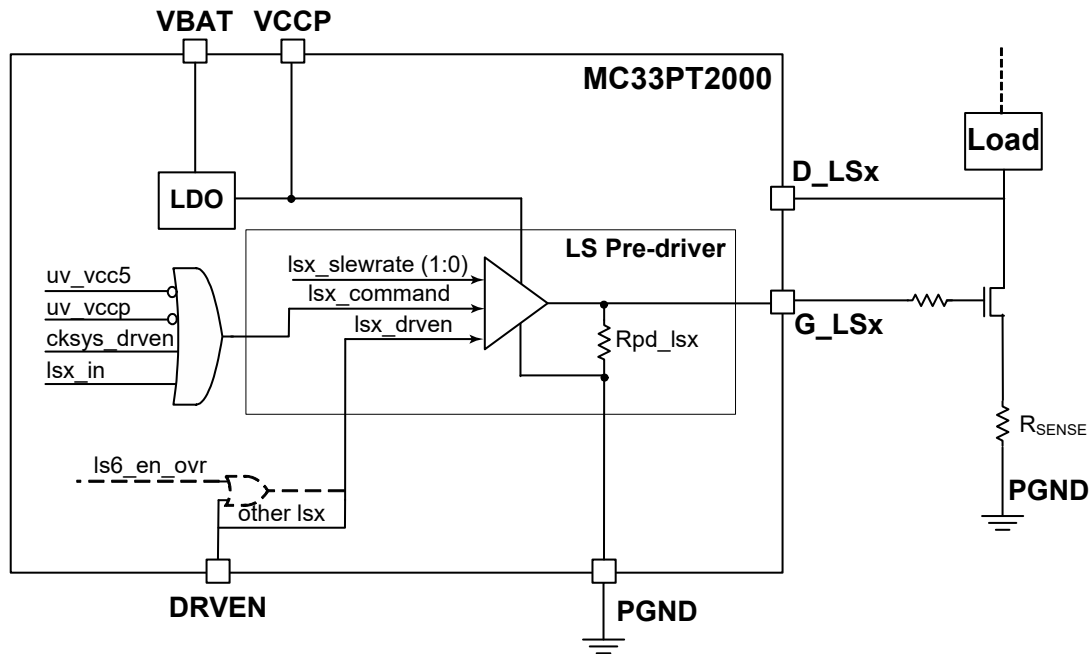
hsx_vds/src_threshold(3:0)	Threshold voltage HS VDS / HS VSRC
0100	2.0
0101	2.5
0110	3.0
0111	3.5

The high-side  $V_{DS}$  comparator compares  $V_{DS}$  of the high-side MOSFET, acquired through VBOOST, VBATT, and the S\_HSx pins, with the selected threshold. In actual implementation, the selected threshold voltage is subtracted to the VBOOST or VBATT voltage and compared to the S\_HSx voltage.

The high-side  $V_{SRC}$  comparator compares the voltage across the freewheeling element, acquired through the S\_HSx pins and PGND, with the selected threshold. In actual implementation, the selected threshold voltage is added to PGND voltage and compared to the S\_HSx voltage.

## 6.5 Low-side pre-driver (LS1-6)

There are six general purpose low-side pre-drivers and all drive external N-Channel logic level type power MOSFETs used in low-side configurations.



**Figure 13. Low-side pre-driver block diagram**

Internal to the device, a gate to source pull-down resistor RPD\_LSx holds the external MOSFETs in the off state, while the device is in a power on reset state (RSTB low).

The low-side pre-drivers are supplied by  $V_{CCP}$  voltage. The low-side pre-driver is intended to drive the gate of an external logic level MOSFET in low-side configuration. The logic command  $Isx\_command$ , to switch the external MOSFET, is provided by the digital block. This command is generated, taking into account the following signals:

- Logic command coming from channel logic ( $Isx\_in$ ).
- $V_{CCP}$  undervoltage signals ( $uv\_vccp$ ) from the VCCP UV monitor: In case of an undervoltage, the external MOSFET is switched off.
- $V_{CC5}$  undervoltage signals ( $uv\_vcc5$ ) from VCC5 UV monitor: In case of an undervoltage the external MOSFET is switched off.

- Signal cksys\_drven coming from the clock monitoring: In case of a missing clock (PLL not locked), the external MOSFET is switched off. This function is disabled by default and can be enabled by setting the cksys\_missing\_disable\_driver bit high (refer to [Table 152, backup\\_clock\\_status \(1A8h\)](#)).
- For safety purpose DRV\_EN is added to the control signal for the driver. As long as DRV\_EN signal is negated (low) the low-side pre-driver is switched off. The low-side pre-driver 6 includes a feature to override the switch off path via the DRV\_EN signal (refer to [Table 174, LSx\\_output\\_config \(1C2h, 1C5h, 1C8h, 1CBh, 1CEh, 1D1h\)](#)).

The truth table describing the status of lsx\_command signal is given in [Table 24](#).

**Table 24. Low-side pre-driver truth table**

DRV_EN	ls6_en_ovr	uv_vccp	uv_vcc5	cksys_drven	lsx_in	lsx_command	Driver Status
0	-	-	-	-	-	0	off
-	-	1	-	-	-	0	off
-	-	-	1	-	-	0	off
-	-	-	-	0	-	0	off
-	-	-	-	-	0	0	off
-	1	0	0	1	1	1	on
1	-	0	0	1	1	1	on

The pre-driver G\_LSx output is set according to lsx\_command:

- When lsx\_cmd is high, the G\_LSx pin is driven high (pull-up to V<sub>CCP</sub> voltage)
- When lsx\_cmd is low, the G\_LSx pin is driven low (pull-down to PGND voltage)

## 6.5.1 Low-side pre-driver slew rate control

All low-side pre-drivers feature a programmable slew-rate control. The settings can be changed independently for each pre-driver by the signals lsx\_slewrates(1:0) coming from the digital core. The given value of voltage slew-rate is only an indication and is dependant on the used MOSFET and the additional gate circuit.(refer to [Slew rate high-side and low-side selection register](#)).

**Table 25. Slew rate settings for LS pre-drivers 1-6**

lsx_slewrates(1:0)	Slew-rate/ V/μS	R <sub>DS(ON)_PMOS</sub> (switching on)/Ω	R <sub>DS(ON)_NMOS</sub> (switching off)/Ω
00	300	14.6	5.9
01	50	84	35
10	25	170	69
11	12.5	337	138

## 6.5.2 LS1 - LS6 $V_{DS}$ monitor

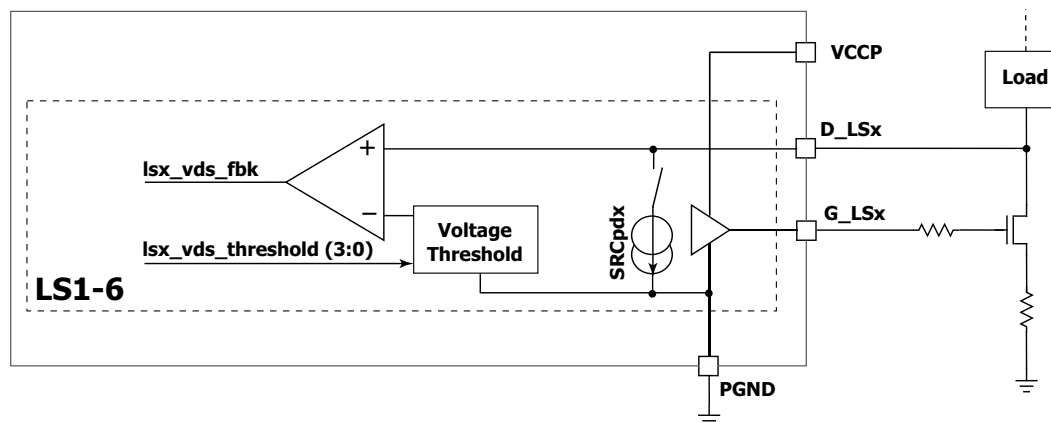


Figure 14.  $V_{DS}$  monitoring LS1 to LS6

For  $V_{DS}$  monitoring of the external low-side MOSFET, a comparator with programmable threshold is provided, sensing the voltage between the drain pin  $D_{LSx}$  and  $PGND$  ( $V_{DS}$  of the low-side MOSFET). If a sense resistor is connected between the low-side MOSFET and ground, the voltage drop on the resistor is included in the measurement.  $V_{DS}$  threshold are selectable by the SPI using  $vds\_threshold\_ls$  registers (refer to [Table 96\\_Vds\\_threshold\\_ls\\_Part 1 \(16Fh\)](#) and [Table 97\\_Vds\\_threshold\\_ls\\_Part 2 \(170h\)](#)). Selectable values are shown in [Table 26](#).

Table 26. Low-side  $V_{DS}$  monitor threshold selection

isx_vds_threshold(3:0)	Threshold Voltage LS $V_{DS}$
0000	0.00
1001	0.10
1010	0.20
1011	0.30
1100	0.40
0001	0.50
0010	1.0
0011	1.5
0100	2.0
0101	2.5
0110	3.0
0111	3.5

## 6.6 Low-side pre-driver for DC/DC converter (LS7 and LS8)

There are two low-side pre-drivers (LS7-8) targeted for DC/DC converter applications. All are to drive external N-channel logic level type power MOSFETs used in low-side configurations. If no DC/DC is required, they can be used as general purpose low-side.

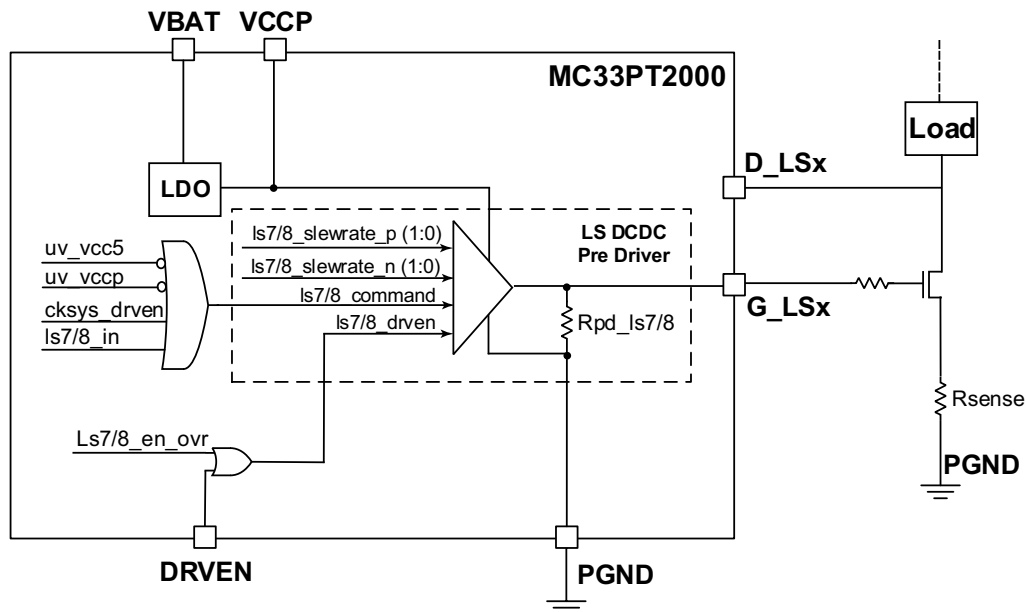


Figure 15. Low-side pre-driver for DC/DC converter (LS7 and LS8)

Internal to the device, a gate to source pull-down resistor  $R_{PD\_LSX}$  holds the external MOSFETs in the off state, while the device is in a power on reset state (RSTB low). The low-side pre-drivers are supplied by  $V_{CCP}$  voltage.

The low-side pre-driver is intended to drive the gate of an external logic level MOSFET in low-side configuration. The logic command  $lsx\_command$ , to switch the external MOSFET, is provided by the digital block. This command is generated, taking into account the following signals:

- Logic command coming from channel logic ( $ls7/8\_in$ ).
- $V_{CCP}$  undervoltage signals ( $uv\_vccp$ ) from VCCP UV monitor: in case of undervoltage, the external MOSFET is switched off.
- $V_{CC5}$  undervoltage signals ( $uv\_vcc5$ ) from VCC5 UV monitor: in case of undervoltage, the external MOSFET is switched off.
- Signal  $cksys\_drven$  coming from the clock monitoring: in case of a missing clock (PLL not locked), the external MOSFET is switched off. This function is disabled by default and can be enabled by setting the  $cksys\_missing\_disable\_driver$  bit high (refer to [Table 152, backup\\_clock\\_status \(1A8h\)](#)).
- For safety purpose  $DRV\_EN$  is added to the control signal for the driver. As long as  $DRV\_EN$  signal is negated (low) the low-side pre-driver is switched off. The low-side pre-driver for the DC/DC converter includes a feature to override the switch off path via signal  $DrvEn$ . As long as the signals  $ls7/8\_en\_ovr$  are high, the pre-driver is not influenced by  $DrvEn$  (refer to [Table 175, LS7\\_output\\_config \(1D4h\)](#) & [LS8\\_output\\_config \(1D7h\)](#)).

The pre-driver is capable of PWM operation up to 400 kHz according to the following table.

Table 27. Low-side pre-driver LS7/8 PWM frequency and load

PWM Frequency / kHz	Gate Charge / nC
400	60
300	75
240	100

A maximum duty cycle of 100% is allowed during PWM operations.

## 6.6.1 Low-side pre-driver slew rate control (LS7 and LS8)

The driver strength can be selected among a set of four values by the SPI registers. The strength for the rising and falling edge can be chosen independently by means of the bits `slew rate_Is7/8_rising(1:0)` and `slew rate_Is7/8_falling(1:0)` (refer to [Table 101](#), [Ls\\_slewrates\\_Part 2 \(173h\)](#))

The slew rate is determined by the PMOS and NMOS  $R_{DS(on)}$  of the push/pull driver circuitry. The typical gate slew rate values are defined in [Table 28](#) and [Table 29](#). These values are given as reference and are impacted by the external circuitry.

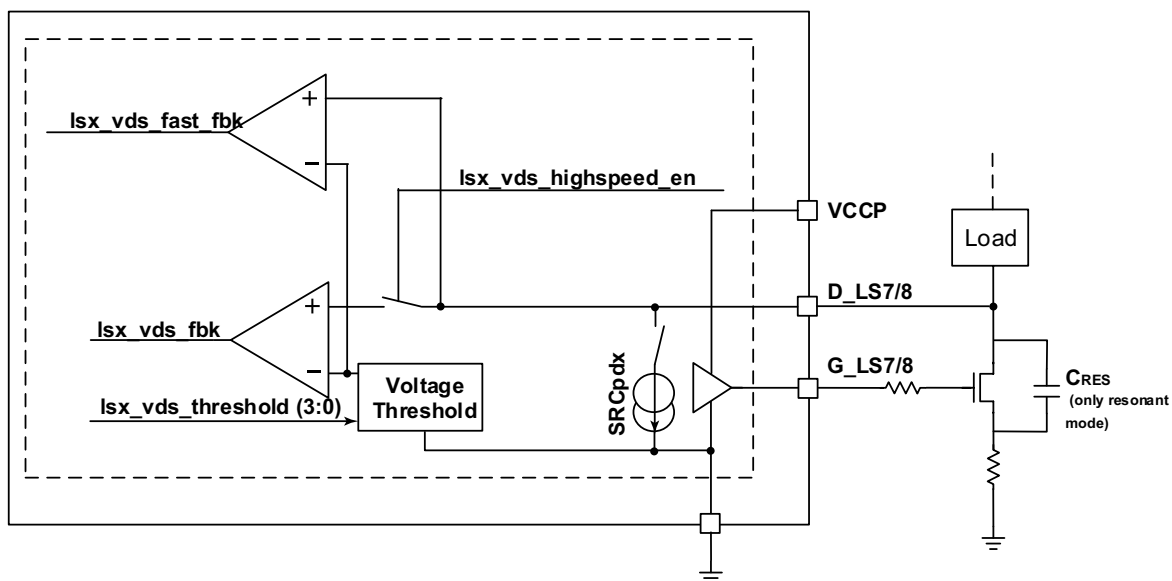
**Table 28. Slew rate settings for LS pre-drivers 7/8 PMOS**

LS7/8_slewrates_p(1:0)	Slew-rate/ V/ $\mu$ S	RDSON_PMOS (switching on)/ $\Omega$
00	1500	5.0
01	300	14.6
10	50	85
11	25	170

**Table 29. Slew rate settings for LS pre-drivers 7/8 NMOS**

LS7/8_slewrates_n(1:0)	Slew-rate/ V/ $\mu$ S	RDSON_PMOS (switching on)/ $\Omega$
00	1500	1.1
01	300	5.9
10	50	35
11	25	69

## 6.6.2 Low-side $V_{DS}$ monitor D\_Is7/D\_Is8 for DC/DC



**Figure 16. Low-side 7 and 8  $V_{DS}$  monitor**



Two comparators are implemented for the  $V_{DS}$  monitoring function of D\_LS7/8. One is used for normal  $V_{DS}$  monitoring (See LS1 - LS6 VDS monitor on page 46), and the other is a high-speed comparator used for the DC/DC resonant converter application.

#### Normal $V_{DS}$ monitoring:

- Input impedance of D\_LSx has to be switched to low speed by setting  $I_{SX\_VDS\_HIGHSPEED\_EN}$  to "0" via the SPI register bit (refer to Table 116,  $Vds7\_dcdc\_config$  (182h) &  $Vds8\_dcdc\_config$  (183h))
- $I_{SX\_VDS\_FBK}$  is used for diagnostics
- Clamp voltage = 3.5 V

#### Fast VDS monitoring (DC/DC resonant converter):

- Input impedance of D\_LSx has to be switched to high speed by setting  $I_{SX\_VDS\_HIGHSPEED\_EN}$  to "1" via SPI register bit (refer to Table 116,  $Vds7\_dcdc\_config$  (182h) &  $Vds8\_dcdc\_config$  (183h))
- $I_{sx\_vds\_dcdc}$  is used for resonant detection
- VDS Threshold needs to be set to 2.5 V

For more details on the DC/DC mode See DC/DC converter control (LS7/8) on page 61.

## 6.7 Current measurement

There are six total input pairs to measure currents with external shunt resistors in a four-wire configuration.

- Four general purpose blocks (#1, 2, 3, and 4).
- Two extended mode block for DC-DC converters (#5 and 6).

The shunt resistors are used in low-side configuration with one of the shunt terminals tied to ground for all the blocks. This means the PT2000 measures a differential voltage over the two input pins.

### 6.7.1 General purpose current measurement block

The actuator current flowing in an external sense resistor is measured to implement a closed loop current control. The current measurement block is comprised of a differential amplifier, sensing the voltage across the sense resistor, a voltage comparator, and an 8-bit current DAC.

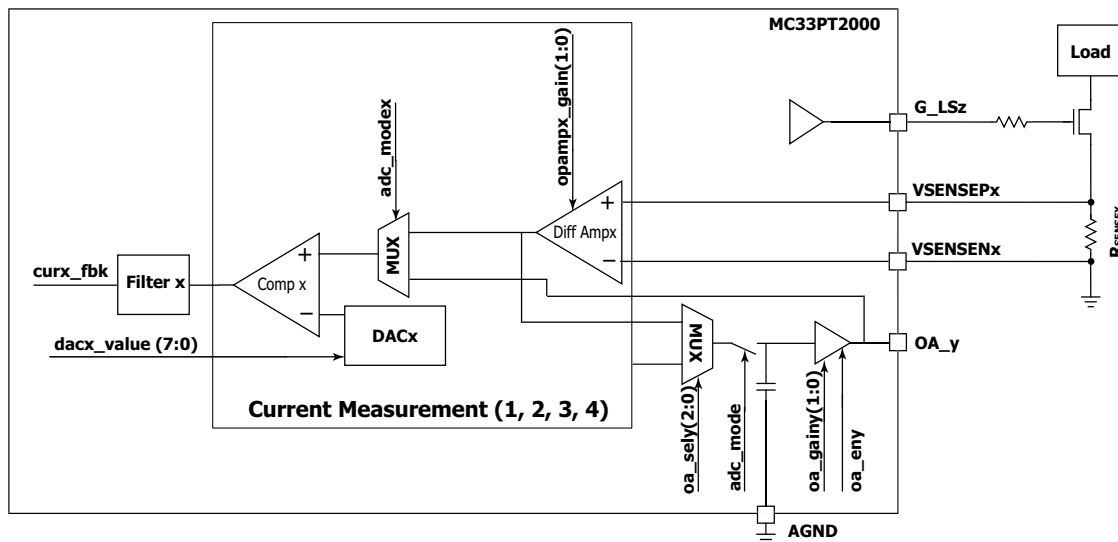


Figure 17. General purpose current measurement block diagram

The differential amplifier gain is selectable among four different values by means of the  $opampx\_gain$  (1:0) signal, to get the suitable signal amplification. The gain can be changed at runtime by the microcore using the  $stgn$  instruction (reference Programming Guide and Instruction Set). The differential amplifier also adds a constant offset to its output. Therefore, the output of the amplifier is always positive. The desired actuator current level can be selected and changed at runtime by the microcore, setting the proper  $dax\_Value$  (7:0) threshold value in the DAC. Each current measurement channel can be used in ADC mode. A track and hold circuit is implemented to keep the voltage at the comparator input stable during the ADC conversion.

The differential amplifier output can be routed to an external pin (OA\_1, OA\_2, and OA\_3). In this configuration, the device output is usually connected to an ADC input of the MCU for safety and test purposes. The output multiplexer block contains an output amplifier with selectable gain by means of the oa\_gainy(1:0) signal, providing full swing output on OA\_y for A/D conversion, if used with 3.3 V or 5.0 V applications. The target for overall accuracy is about 3.7% at 75% DAC range (see Table 12). This includes the external resistor which is assumed to have a worst case error of 2.0%.

### 6.7.1.1 Current sense amplifier

The current sense amplifier provides a voltage which can be monitored through the OA\_x pins. A 250 mV offset is added to monitor negative current, this way output of the amplifier is always positive. The amplifier is fully operational down to an output voltage of typically 100 mV. The current sense amplifier output voltage can be calculated using the following formula:

$$V_{\text{DASENSE}} = (V_{\text{VSENSEPx}} - V_{\text{VSENSENx}}) \times G_{\text{DADIFF}} + V_{\text{DABIAS}}$$

$V_{\text{DABIAS}}$  is fixed value of 250 mV applied to the differential amplifier output.

The  $G_{\text{DADIFF}}$  gain value is configurable at runtime (opampx\_gain(1:0)), this gain can be selected using the instruction stgn. The allowed differential mode input voltages depend on the chosen gain value.

### 6.7.1.2 Current sense DAC

In order to select the proper threshold for current control, an 8-bit current DAC is implemented to provide a threshold to the voltage comparator (dac\_value (7:0)). The current threshold can be calculated using the following formula.

$$I = \frac{((\text{DACVALUE} \times V_{\text{DACLSB}}) - V_{\text{DABIAS}})}{G_{\text{DADIFF}} \cdot R_{\text{SENSE}}}$$

DAC\_VALUE is selected and changed at runtime by the digital microcore by means of the signal dactx\_value (7:0). A DAC\_VALUE below the hexadecimal value 0Ah, must be avoided, as the current sense differential amplifier does not operate with full performance at output voltages below 100 mV.

$V_{\text{DAC_LSB}}$  is the DAC resolution = 9.77 mV.

$V_{\text{DA_BIAS}}$  is the fixed voltage biasing applied to the differential amplifier output = 250 mV.  $G_{\text{DA_DIFF}}$  is the gain value configurable at runtime by the SPI (opampx\_gain(1:0)). This gain can be selected using the instruction stgn.

$R_{\text{SENSEX}}$  is the external sense resistor of the current measurement channel x.

**Table 30. Current sense DAC values with a 10 mΩ shunt**

DAC value		DAC output voltage / mV	Current threshold with 10 mΩ shunt/A			
HEX	BIN		Gain = 5.79	Gain = 8.68	Gain = 12.53	Gain = 19.25
0A	10	98	-2.63	-1.76	-1.22	-0.79
...	...	...	...	...	...	...
19	25	244	-0.10	-0.07	-0.05	-0.03
1A	26	254	0.07	0.05	0.03	0.02
1B	27	264	0.24	0.16	0.11	0.07
...	...	...	...	...	...	...
FF	255	2490	38.69	25.81	17.88	11.64

### 6.7.1.3 Current measurement offset compensation

An analog offset compensation is done which compensates the input offset of the current measurement amplifiers one to six. The offset compensation is started and stopped from the digital microcores using the instruction `stoc`. The offset compensation must be started while there is no current flow in the shunt of the related measurement channel.

The compensation uses a small 6-bit DAC (5-bit plus sign) which injects a current at the input stage of the differential amplifier to compensate the input offset. The offset compensation is finished after a maximum time of  $31 \times 2.0 \mu\text{s} = 62 \mu\text{s}$ . This process is completely automatic and only the start and stop has to be handled by the microcore.

The offset compensation uses the “`ck_ofscmp`” generated from the `ck_sys`. The prescaler can be set using this register (refer to [Table 146](#), [Ck\\_ofscmp\\_Prescaler\(1A4h\)](#)). Because the offset compensation minimum step time can be up to  $2.0 \mu\text{s}$  (refer to [Table 13](#)), it is mandatory to set the `ck_ofscmp` to a maximum of 500 kHz.

Each new offset compensation is started based on the result of the previous offset compensation run for this current measurement channel. If the offset compensation is stopped from the digital microcore when the analog offset compensation is not finished, the procedure is aborted, maintaining the last compensation value reached when the procedure was interrupted. This strategy guarantees each offset compensation decreases the offset of the current measurement amplifier independent of it being finished.

Due to a temperature drift of the differential amplifier input offset, the offset compensation must be performed each time a huge change in device temperature is expected. If this is not possible, an increased residual offset due to the temperature drift has to be taken into account.

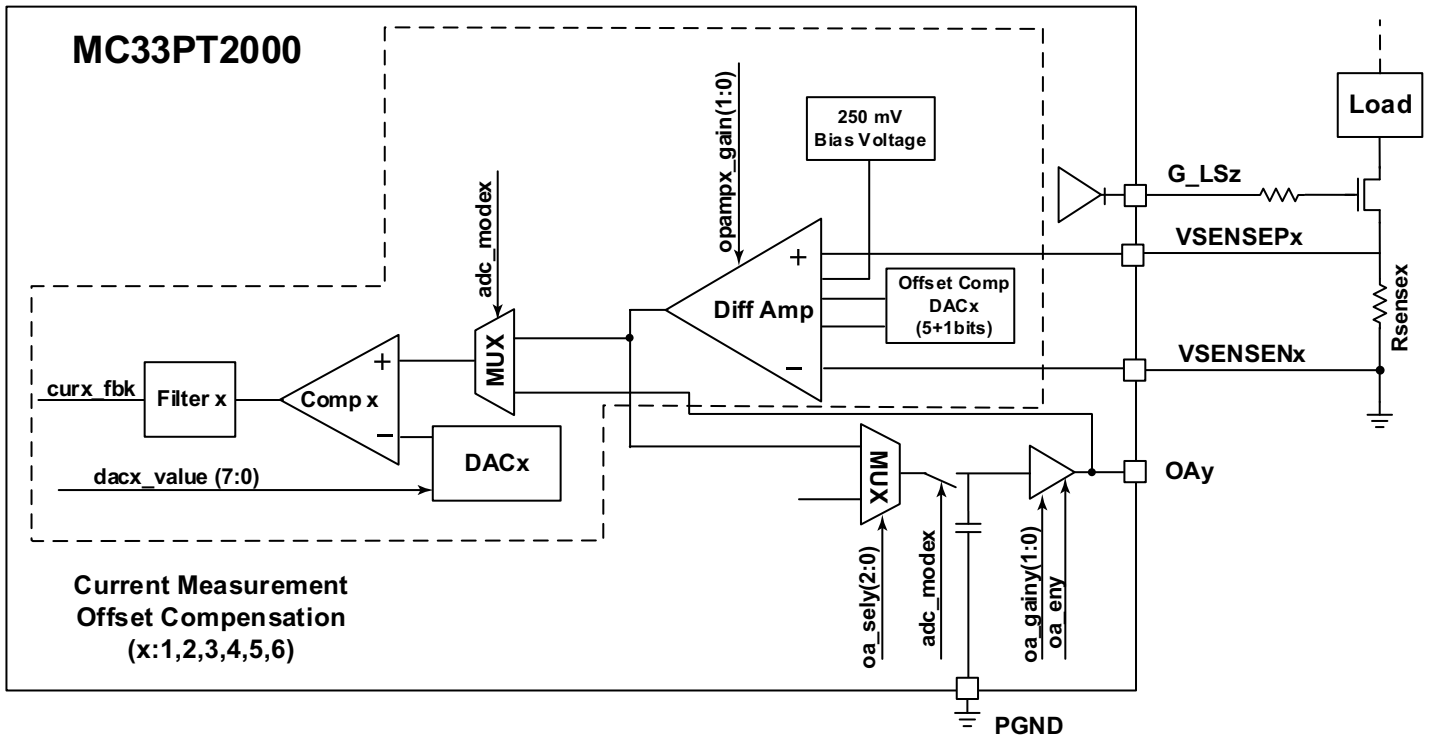


Figure 18. Offset compensation block diagram

## 6.7.2 Current measurement for DC/DC

The inputs of the 5th and 6th current sense need to support a very wide range of applications.

Typical applications use the 5th and 6th current sense e.g.

- Just identical to the other current sense blocks or
- To control a DC/DC converter with a low-side current measurement and concurrently provide an overcurrent supervision at the booster capacitor

The two-point current control of a DC/DC converter results in challenging requirements on latency of the control loop. This means:

- The path from sense input to low-side driver output must achieve a very small delay
- There is no time to change the DAC setting after each switching event.

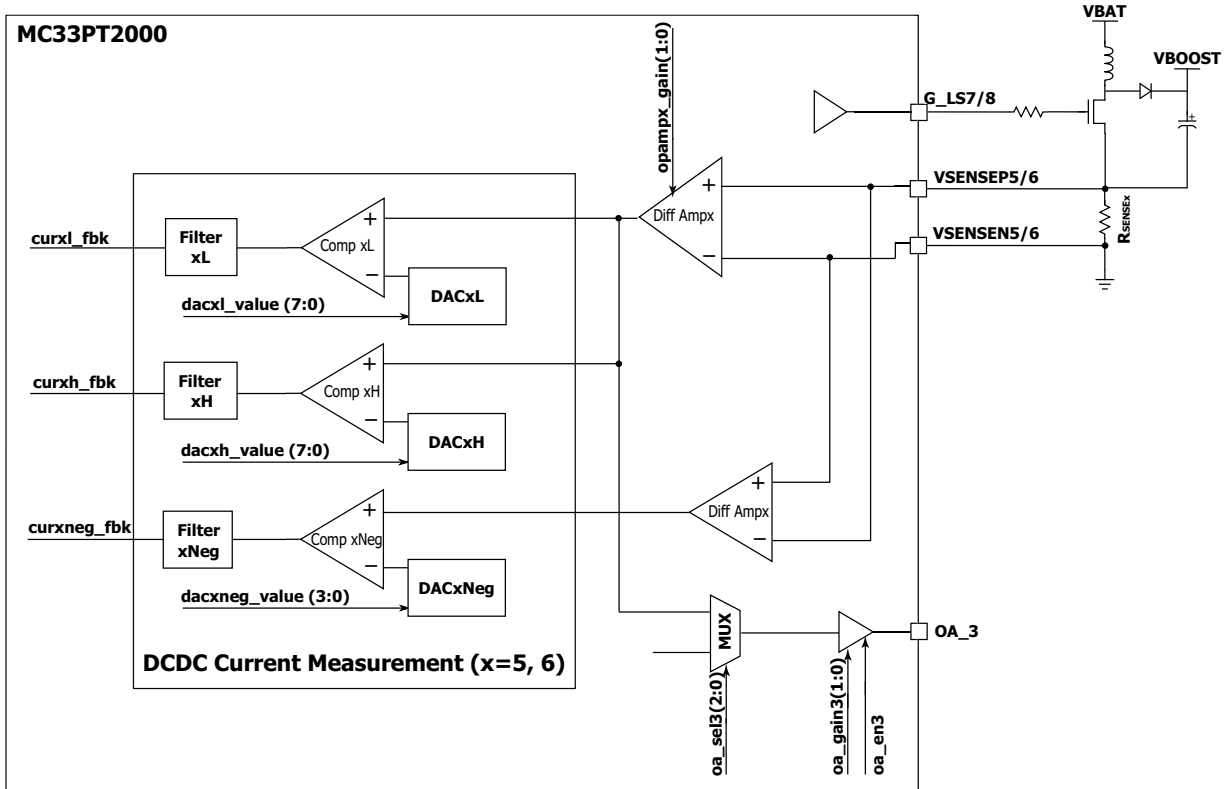


Figure 19. DC/DC current measurement (5 & 6) block diagram

Along with the number of concurrent thresholds to supervise, the key feature of the current measurement block for DC/DC is the ability to provide a short delay from the VSENSE inputs to the G\_LS7/8 output. For this application, the digital core contains a hardwired logic for a two-point current regulation using the cur5/6h\_fbk and cur5/6l\_fbk signals as inputs to directly control the LS7 or LS8 low-side driver. Using the negative comparator it is possible to detect boost overcurrent. During this time the low-side is Off and the current flows from the boost tank capacitor to the load. (See [Negative current differential amplifier on page 53](#))

## 6.7.2.1 Negative current differential amplifier

The key characteristic of the second differential amplifier is it works at negative differential input voltages and therefore has a negative gain. This is used to detect overcurrent when the low-side is Off.

**Table 31. Boost overcurrent sense amplifier overall gain**

Gain value	Normal differential mode input voltage	Full scale range current with 10 mΩ shunt	DAC resolution with 10 mΩ shunt
-2.0	-1047 mV ...0	-104.7 A	-7.81 A

The current threshold can be calculated using the following formula.

$$I = \frac{((\text{DACVALUE} \times V_{\text{DACLSB}}) - 250\text{mV})}{G_{\text{DADIFF}} \cdot R_{\text{SENSE}}}$$

DAC\_VALUE is selected and changed at runtime by the digital microcore by means of the signal dacx\_value (3:0).

V<sub>DAC LSB</sub> is the DAC resolution = 156.25 mV.

V<sub>DA\_BIAS</sub> is the fixed voltage biasing applied to the differential amplifier output = 250 mV.

The Gain Value G<sub>DA\_DIFF</sub> is fixed to -2.0.

R<sub>SENSEX</sub> is the external sense resistor of the current measurement channel x.

## 6.7.2.2 Current measurement offset compensation

The analog offset compensation for the differential amplifier 5 and 6 of channel 5 and 6 is done in the same way for channel 1 to 4. The DAC5/6L and the signal Cur5/6l\_fbk are used for the offset compensation. Since the accuracy is not actually critical to detect the overcurrent, the differential amplifier 5 and 6 negative do not have offset compensation (see [Figure 19](#)).

## 6.8 OA\_x output pins, multiplexer and T & H

### 6.8.1 General features

The output signals of the six current sensing amplifiers are available via three external pins OA\_1, OA\_2, and OA\_3 of the device (refer to [Figure 20](#)). With the OAGainx (1:0) and OAxG1 signal, it is possible to select between five different output gains. This feature is used to adapt the device to an ADC input range of 3.3 V or 5.0 V, and also to add the possibility of amplifying the output signal even higher for some special measurement functions. The maximum output voltage at the OA\_x pins of VCC5 always has to be taken into account.

For the two higher gains of 3.0 and 5.33, the bias voltage of nominal 250 mV of the input signal is removed before amplifying the signal and adding again to the amplified signal afterwards.

**Table 32. OA\_x amplifier gain selection and output voltage**

Oagainx(1:0)	OAxG1	Gain value	Output voltage
d.c.	1	1.0	V <sub>IN</sub> * Gain
00	0	1.33	V <sub>IN</sub> * Gain
01	0	2.0	V <sub>IN</sub> * Gain
10	0	3.0	(V <sub>IN</sub> - 250 mV) * Gain +250 mV
11	0	5.33	(V <sub>IN</sub> - 250 mV) * Gain +250 mV

The OA1/2/3 output pins includes the possibility of switching to hi-impedance mode to enable the direct connection of two of these output pins to one micro-controller ADC input. All OA\_x output multiplexers can also be switched to VCC2P5 as a third option. This is used to check the connection between the PT2000 and the microcontroller ADC on the ECU level.

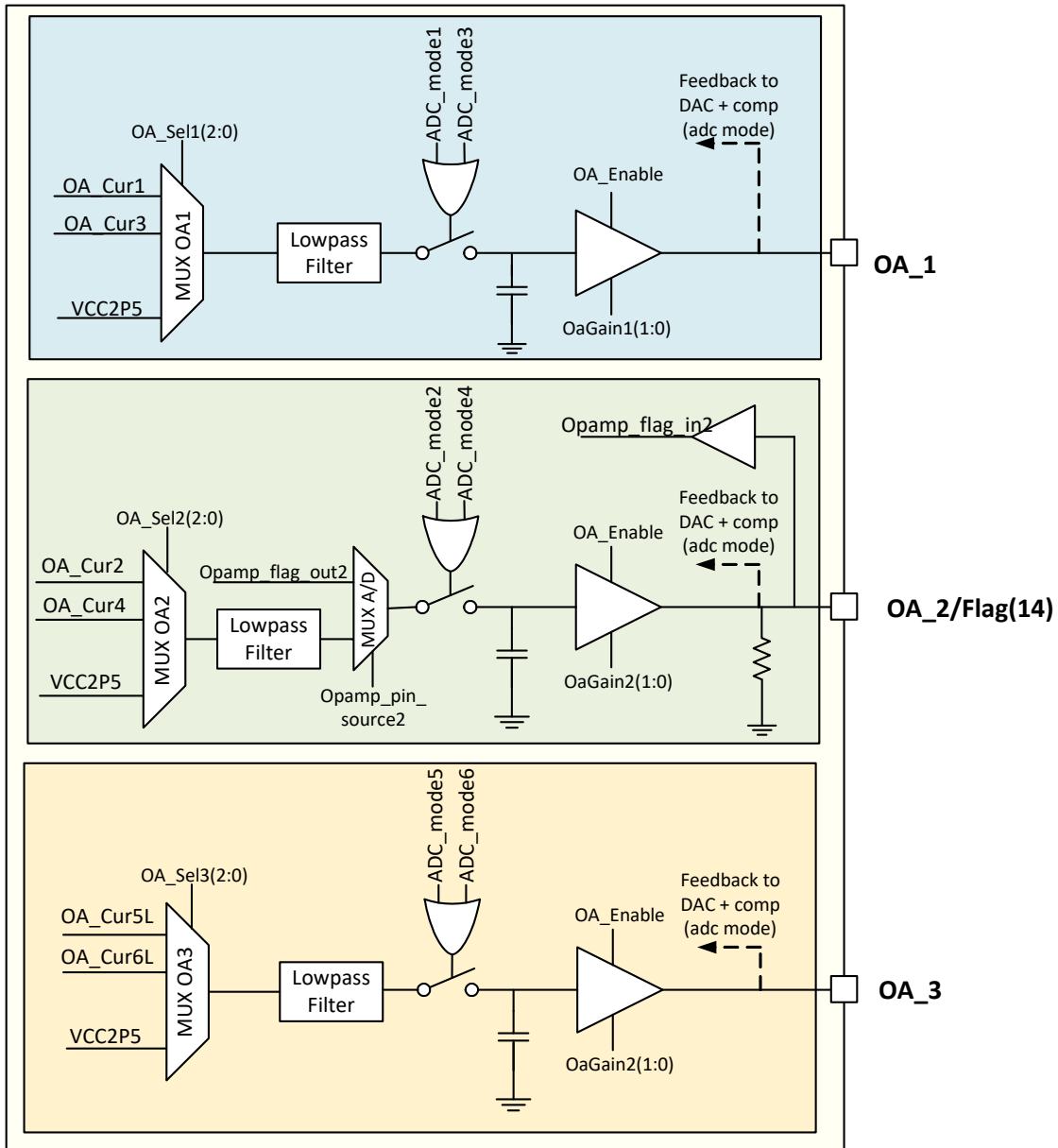


Figure 20. OA\_x multiplexer

Table 33, Table 34, and Table 35 show the output configuration of the OA\_x multiplexers, control is done by register oa\_out\_config (See Analog output (OA<sub>x</sub>) configuration register on page 116).

**Table 33. OA\_1 multiplexer logic table**

OaSel1(2:0)	OaEN1	Signal at Output OA_1
000	1	OA_Cur 1 (Feedback of current measurement 1)
001	1	OA_Cur 3 (Feedback of current measurement 3)
010-100	1	Reserved
101	1	VCC2P5
110-111	1	Reserved
xxx	0	HiZ

**Table 34. OA\_2 multiplexer truth table**

OaSel2(2:0)	OaEN2	Signal at Output OA_2
000	1	OA_Cur 2 (Feedback of current measurement 2)
001	1	OA_Cur 4 (Feedback of current measurement 4)
010-100	1	Reserved
101	1	VCC2P5
110-111	1	Reserved
xxx	0	HiZ

**Table 35. OA\_3 multiplexer truth table**

OaSel3(2:0)	OaEN3	Signal at Output OA_3
000	1	OA_Cur 5 (Feedback of current measurement 5)
001	1	OA_Cur 6 (Feedback of current measurement 6)
010-100	1	Reserved
101	1	VCC2P5
110-111	1	Reserved
xxx	0	HiZ

The multiplexer must not be switched to a signal currently processing via another OAx analog output or an internal path. Switching the multiplexer can cause a glitch on the signal being processed.

## 6.8.2 OA\_2 Pin digital I/O function

### 6.8.2.1 General requirements

The OA\_2 pin can also be used as a digital flag bus input or output flag (14). It can be selected by the SPI configuration of the PT2000 using the registers `flags_source` and `flags_direction` (refer to [Table 138, Flag\\_direction \(1A1h\)](#)). The flag pin (14) has a higher rise/fall time of about 3.0  $\mu$ s, compared to the other external flag bus pins of the device. For the digital output functionality, the same output stage is used as with the analog function. As soon as the pin is configured as a digital input, the buffer is switched to hi-impedance. [Table 36](#) shows how the enable signal is created.

**Table 36. OA2 enable truth table**

flags_source	flags_direction	opamp_pin_source (2) (reset=0)	OaEN2 (reset=0)	OA2 Buffer state	Description
0	d.c.	0	0	HiZ	OA2 pin is used as analog output, enable signal is low
0	d.c.	0	1	On	OA2 pin is used as analog output, enable signal is high
1	1	1	-	HiZ	OA2 pin is used as digital input
1	0	1	-	On	OA2 pin is used as digital output

### 6.8.2.2 OA\_2 pin I/O voltage

The I/O voltage of the OA\_2 pins is not automatically set according to the  $V_{CCIO}$  voltage supplied to the device. The OA\_2 output amplifier which is also used for the digital output function is supplied by  $V_{CC5}$ . The digital input signal to the OA\_2 output amplifier is a  $V_{CC2P5}$  based signal. The I/O voltage has to be selected by choosing the right gain of the OA\_2 output amplifier. Table 37 shows how to select the proper I/O voltage.

**Table 37. OA\_2 amplifier gain selection (I/O voltage)**

OaGain2(1:0)	Gain value	Used for
00 (reset value)	1.33	3.3 V I/O
01	2.0	5.0 V I/O
10	3.0	
11	5.33	

There is a weak pull-down resistor at the OA\_2 input/output. This resistor is always present, regardless if the digital or analog functionality is selected.

### 6.8.3 OAx output offset and offset error

It is important to have a close look at the output offset of the OAx pins and the output voltage values corresponding to load current values. The current measurement amplifiers output signal has a fixed offset of 250 mV and also some variable offset of -28.6 mV to +36.4 mV for this path after analog offset compensation. This offset is independent of the current measurement amplifiers gain setting, but is amplified by the OAx amplifier gain. The OAx amplifier also adds some input offset of  $\pm 10 \dots 13.5$  mV to this calculation. So in sum there is a fixed offset of 250 mV and a variable offset at the input of the OAx amplifier.

For the two higher gains of 3.0 and 5.33, the bias voltage of nominal 250 mV of the input signal is removed before amplifying the signal and added again to the amplified signal afterwards. Table 38 gives some examples for load currents, gain settings, and corresponding output voltage ranges. Note that this calculation only takes into account the offset errors. There are also other errors which have to be considered in a full error calculation.

**Table 38. OAx input and output values**

Load current at 10 m $\Omega$ shunt/A	Current measurement amplifier gain setting	OAx amplifier gain setting	OAx output voltage Min./mV	OAx output voltage Typ./mV	OAx output voltage Max./mV
0	8.68	1.33	276	333	399
25.8	8.68	1.33	3256	3312	3378
0	19.25	5.33	45	250	497
1	19.25	5.33	1071	1276	1523
2.5	19.25	5.33	2610	2815	3062



# 7 Functional device operation

## 7.1 Power-up/down sequence

During power-up, the voltage at the VBATT pin can be clearly higher than the voltage at the VBOOST pin. The functionality of the PT2000 within its functional limits and outside its functional limits (no destruction of connected devices) has to be guaranteed independently from the slope of the voltage ramp up of the supply voltages ( $V_{CC5}$ ,  $V_{CCIO}$ ,  $V_{BATT}$ ) and the starting value.

### 7.1.1 Power-up sequence of VCC5, VCC2P5, and reset

After the internal POResetB signal is deactivated, it takes a maximum time of  $t_{DIGIOREADY} = 100 \mu s$  until the digital outputs of the device are functional. CLK can be sent even before this  $t_{DIGIOREADY}$ , but it is not taken into account. Inside the logic core, POResetB is combined with the external reset signal ResetB (active low) and the SPIResetB signal coming from the SPI interface. As long as RSTB is asserted, the SPI module is inactive. After the first RESETB rising edge, it is required to wait  $t_{SPIREADY\_t0} = 100 \mu s$  to allow time for the fuses to load.

Note that the logic core is properly supplied at 2.5 V when 5.0 V is present at the VCC5 pin (thus allowing logic core operations and SPI communication with the microcontroller), even if no voltage is provided at the VBATT pin and by consequence no voltage is present on the VCCP pin.

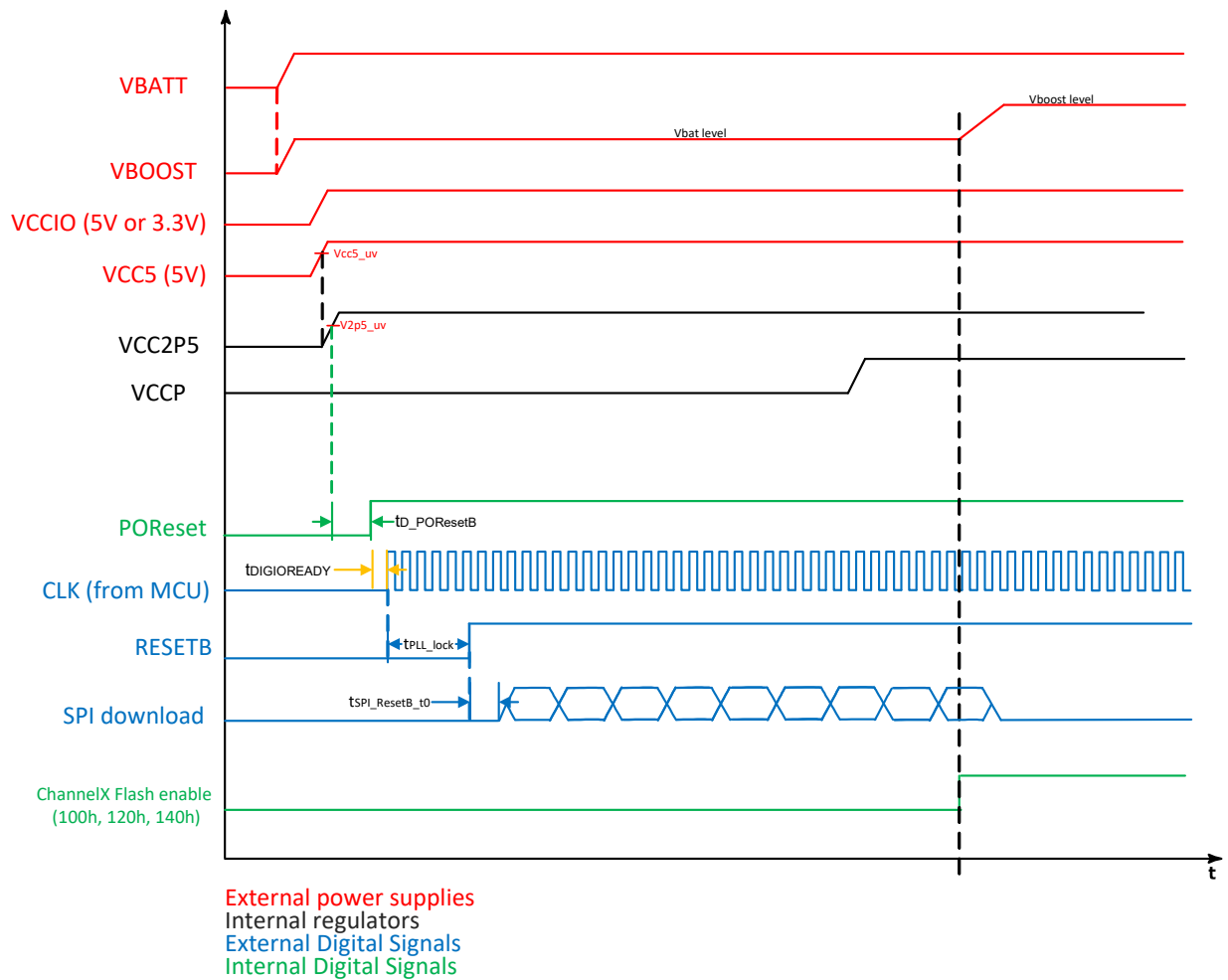


Figure 21. Power-up diagram

## 7.1.2 Power-up sequence VCCP and bootstrap capacitors

### 7.1.2.1 Bootstrap switch control

During initialization phase, the control of the bootstrap switch needs to be carefully controlled. The following device configurations are affected.

- Hsx\_bs\_lowcurrent: the low-current limit (280  $\mu$ A), which is set only during init independently for each HS pre-driver;
- Vsrc\_threshold: the  $V_{SRC}$  thresholds of each HSx, which during init is set first to 0.5 V and after some time to 1.0 V. After init phase is finished the  $V_{SRC}$  threshold returns to the value defined in the appropriate register (refer to [Table 94, Vsrc\\_threshold\\_hs\\_Part2 \(16Eh\)](#)).
- Ls\_bias: all ls\_bias are set active for all LSx outputs during init phase of any HS pre-driver, and then go back to the configuration defined in the appropriate register (refer to register [Table 126, Ls\\_bias\\_config \(18Ch\)](#)) when all HS pre-drivers are out of the init phase.
- Hs\_bias: the hs\_bias is set inactive for the HSx outputs during init and then returns to the configuration defined in the appropriate register (refer to [Table 125, Hs\\_bias\\_config \(18Bh\)](#)).

During the init phase of the bootstrap capacitors, the vccp\_external\_enable signal is affected according to what is defined in [Table 150, VCCP external enable setting](#). In particular, as long as at least one HS pre-driver is in bootstrap init mode, the vccp\_external\_enable setting is set to '0' (internal regulator active), if the value of the DBG pin sampled at reset (POResetB and ResetB) was '1'.

The charging of the bootstrap capacitors starts after reset is deactivated and as soon as the  $V_{CCP}$  voltage is ramped up. As soon as the  $V_{CCP}$  voltage is above the  $V_{CCP}$  undervoltage threshold, a global timer for all hs pre-drivers running on cksys with an end of count value of 36 ms is started. As soon as the timer reaches the end of count value, the Vsrc\_threshold is changed from 0.5 V to 1.0 V for all drivers still in init mode. At the same moment, the hsx\_src\_1V bit is set to '1' for all these drivers.

The bootstrap init for each HS pre-driver ends if one of the following conditions is met:

- The bs ready comparator shows the B\_HSx voltage is close to the  $V_{CCP}$  voltage and at the same time the S\_HSx voltage is below 0.5 V or 1.0 V,
- The clamp is activated and at the same time the S\_HSx voltage is below 0.5 V or 1.0 V;
- An LS pre-driver connected to the same HS pre-driver is switched on and the corresponding hsx\_lsx\_act signal is set to '1';
- The connection between LS pre-drivers and HS pre-driver is disabled (hsx\_ls\_act\_dis signal = '1'); or
- The same HS pre-driver is switched on.

In applications where two HS pre-drivers are connected to the same node by their S\_HSx pin directly or via a diode, care must be taken. It is not allowed in these configurations to turn on the hs\_bias via the SPI register or the microcode command before all HS pre-drivers finished their bootstrap init. Otherwise an active hs\_bias from one pre-driver may block the init of the other. The init mode of each HS pre-driver can be quit by setting the corresponding "hsx\_ls\_act\_dis" bit to '1' (refer to [Table 127, Bootstrap\\_charged \(18Dh\)](#)). This should be done for each HS pre-driver not used in an application.

### 7.1.2.2 Using D\_LSx pull-down sources to charge bootstrap capacitors

After reset release and after the  $V_{CCP}$  voltage is above the UV\_VCCP threshold, the charging of the HS pre-driver's bootstrap capacitors via the D\_LSx pull-down sources starts automatically, as long as there is a current path from S\_HSx of the HS pre-driver to at least one D\_LSx pin or to GND. To make this possible, there is a requirement to switch the current limitation of the bootstrap path from about 95 mA to min. 280  $\mu$ A.

[Table 39](#) shows how long it takes to charge a bootstrap capacitor to 7.0 V using the D\_LSx current sources and a bootstrap path current limitation of 280  $\mu$ A, plus the charge pump current of 20  $\mu$ A.

**Table 39. Charge times bootstrap Cs using D\_LSx sources**

Bootstrap capacitor size (typ.)	Charge time (typ.)/ms
100 nF	2.3
330 nF	7.7
1.0 $\mu$ F	23.3
2.2 $\mu$ F	51.3

### 7.1.2.3 Using the charge pump to charge bootstrap capacitors

If there is no current path from S\_HSx pin to D\_LSx or GND, only the internal CP can be used to charge the bootstrap capacitors during init with a guaranteed current of 20  $\mu$ A per HS pre-driver. This current is only available if there is no leakage current from B\_HSx pin. Any possible leakage current has to be subtracted from this available charge current.

It is not necessary to turn on either the LS MOSFETs or the D\_LSx pull-down current sources to charge the bootstrap capacitors during the init phase, if there is at minimum the current loop via the body diode of the external high-side MOSFET present. In addition, there is some leakage current path from S\_HSx to PGND. The charge pump starts charging the bootstrap capacitors as soon as the device is supplied with V<sub>CC5</sub> and a voltage greater 4.7 V at the VBOOST pin and POResetB is deactivated. Table 40 shows how long it takes to charge a bootstrap capacitor to 7.0 V using the charge pump current of 20  $\mu$ A.

**Table 40. Charge times bootstrap Cs using CP**

Bootstrap capacitor size (typ.)	Charge time (typ.)/ms
100 nF	35
330 nF	116
1.0 $\mu$ F	350
2.2 $\mu$ F	770

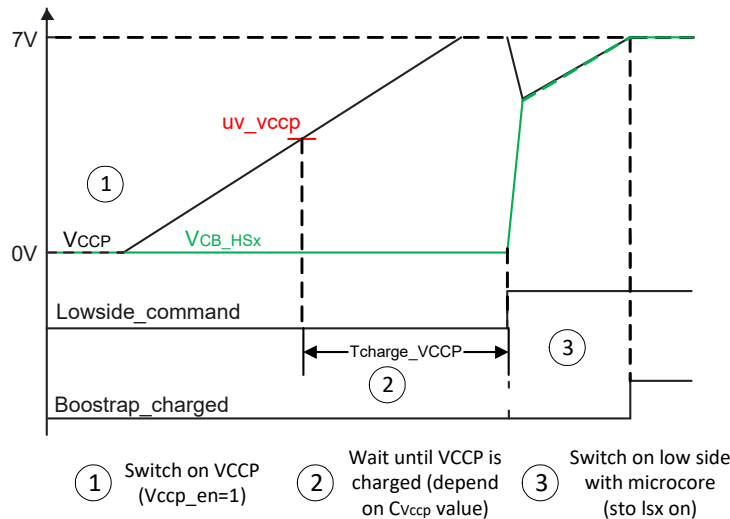
### 7.1.2.4 Using LS MOSFETs to charge bootstrap capacitors

After reset release of the device, the charging of the bootstrap capacitors via the D\_LSx pull-downs and the internal charge pump starts automatically. If there is the requirement to speed up the initial charging of the bootstrap capacitors, it is possible to switch on an LS MOSFET connected to the HS MOSFET. If the device is configured in a proper way (refer Table 131, Hsx\_Is\_act (18Fh - 195h)) the bootstrap switch current limit changes to the high limit of about 95 mA as soon as the LS pre-driver is turned on. Because the bootstrap init mode is left for this HS pre-driver, it must be guaranteed that the internal V<sub>CCP</sub> regulator is already switched on via the SPI bit, if required.

After switching on the internal V<sub>CCP</sub> regulator, the output buffer capacitor at the VCCP pin is charged. The device is still in a V<sub>CCP</sub> undervoltage condition, no external MOSFET (HS or LS) can be switched on, and the bootstrap diodes are also kept in the Off state. During ramp up the voltage on the VCCP pin crosses the V<sub>CCP</sub> undervoltage threshold. After the voltage is above this threshold (plus a minimum 16  $\mu$ s delay), it is possible to switch on the LS MOSFETs.

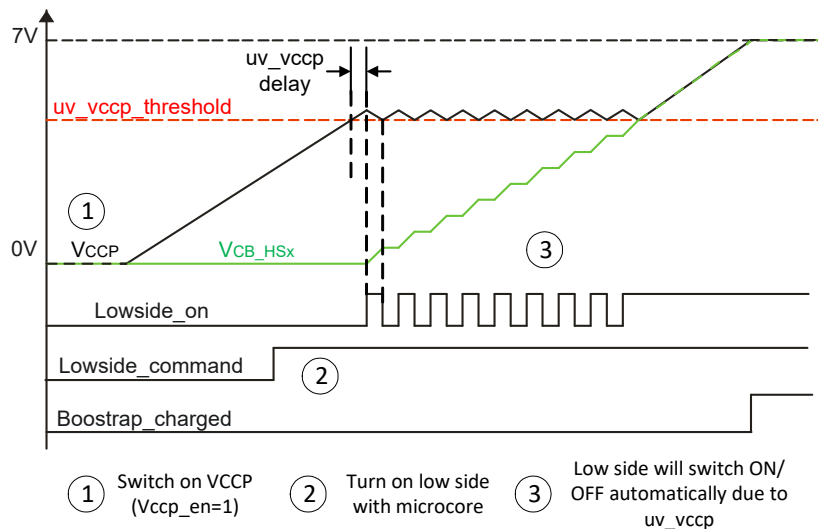
There are two possible solutions to charge the HS pre-driver bootstrap capacitors using the path via the LS MOSFETs:

1. Wait for some specific time after V<sub>CCP</sub> regulator is activated to ensure the V<sub>CCP</sub> output voltage has reached its nominal value of about 7.0 V. This is only possible if the V<sub>BATT</sub> voltage is at a nominal value. Switch on the LS MOSFET(s) to charge the bootstrap capacitors. If the V<sub>CCP</sub> voltage was high enough and the bootstrap capacitors are small compared to the V<sub>CCP</sub> output buffer capacitor, this leads to a transfer of charge without crossing the V<sub>CCP\_UV</sub> threshold again. Figure 22 shows this strategy.



**Figure 22. Power-up: V<sub>CCP</sub> and bootstrap capacitors - after V<sub>CCP</sub> fully charged**

- Switch on the LS MOSFET(s) as soon as possible after the V<sub>CCP</sub> voltage is above the V<sub>CCP\_UV</sub> threshold. This causes the V<sub>CCP</sub> voltage to cross the V<sub>CCP\_UV</sub> threshold again and thereby the LS MOSFET(s) and the bootstrap diode are switched off again. The V<sub>CCP</sub> voltage recovers from undervoltage and after the delay min. of 16 μs, the LS MOSFET(s) and the bootstrap diode are switch on again. This pulsed charging of the bootstrap capacitors with a frequency of below 50 kHz continues until the voltage of the bootstrap capacitors is above the V<sub>CCP\_UV</sub> threshold. [Figure 23](#) shows this strategy.



**Figure 23. Power-up: V<sub>CCP</sub> and bootstrap capacitors - before V<sub>CCP</sub> fully charged**

Strategy 2 is preferred, because it adds less constraints to the size of the bootstrap capacitors and the SW implementation. If strategy 1 is tried, but some of the premises are not fulfilled (V<sub>BATT</sub>/V<sub>CCP</sub> voltage, size of bootstrap capacitors), the result is the same pulsed charging described in strategy 2.

## 7.2 DC/DC converter control (LS7/8)

The two DC/DC converters implemented each support three DC/DC converter control modes. These modes can be chosen via microcode instruction “stdcctl”. This command affects the LS pre-driver which is set as shortcut two of the microcore (see instruction “stdcctl” in the programming guide).

**Table 41. DC/DC converter control modes**

DC/DC mode	Microcode instruction	Short description
1	stdcctl sync	LSx manual mode (LSx controlled by microcode)
2	stdcctl async	LSx hysteretic mode direct controlled by curYh and curYl feedback signals)
3	stdcctl async_vds	LSx resonant converter mode. (direct controlled by VdsX_dcdc and curYh_fbk)

### 7.2.1 General description

The operation modes of the DC/DC converter can be activated by the microcode instruction “stdcctl sync/async/async\_vds”. Every microcore which has access to the LS7/8 pre-driver, according to the crossbar configuration, can activate the DC/DC control modes using this microcode instruction. The LS7/8 outputs can be controlled by standard control method - microcode instruction (mode1) or automatic DC/DC converter control modes (mode 2/3).

As soon as a microcore having access to the LS7/8 is unlocked, the automatic DC/DC control is switched off. Mode 2 and mode 3 can be used to achieve a very fast asynchronous regulation. The current can be changed either through microcode or by writing the DAC Registers via the SPI (See [DAC 1-6 values registers on page 112](#)).

#### 7.2.1.1 Mode 1 (manual mode)

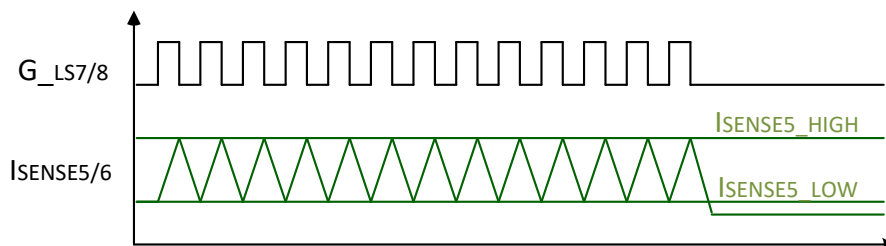
The DC/DC converter control mode 1 is chosen by microcode and is set as the default. The current\_feedback\_5/6 is monitored by microcode and the LS7/8 is controlled completely via microcode. The low-side is controlled directly by the microcore using the “sto” instruction.

#### 7.2.1.2 Mode 2 (hysteretic control)

Mode 2 is intended to be used for standard current controlled ‘asynchronous’ DC/DC converters. A very fast current regulation between the current thresholds 5/6H (higher limit) and 5/6L (lower limit) can be achieved. These two current thresholds can be supplied either from microcode or by writing to the DAC register (refer to [See DAC 1-6 values registers on page 112](#)).

The LSx output is switched on when current\_feedback\_5/6L is low and switched off when current\_feedback\_5/6H is high. The path from the shunt resistor to the LS7/8 output is completely asynchronous to any clock of the device. The current feedback of DAC 5/6H takes priority, so in cases where both feedbacks are active (DAC 5/6H feedback high and DAC 5/6L feedback is low), the output LS7/8 is driven low. This mode is used for standard DC/DC control.

Note that when this mode is used LS7, it should be paired with current sense 5 and LS8 with current sense 6.



**Figure 24. DC/DC mode 2: hysteretic control**

### 7.2.1.3 Mode 3 (LS7/8 resonant mode $V_{DS}$ monitoring)

Mode 3 can be used for controlling resonant converters which require a  $V_{DS}$  threshold based activation of the MOSFET. In this case, a small capacitor ( $\sim 10$  nF)  $C_{RES}$  in parallel with the external MOSFET has to be connected, to avoid oscillation when the low-side is off.

To use Mode 3, the signal  $I_{SX\_VDS\_HIGH\_SPEED\_EN}$  has to be set to a "1" via the control bit in the `vds7/8_dcdc_config` register (see [Table 116, Vds7\\_dcdc\\_config \(182h\)](#) & [Vds8\\_dcdc\\_config \(183h\)](#)).

$V_{DS}$  fast monitoring of D\_LS7/8 uses the same threshold generator as the normal  $V_{DS}$  monitoring, but only the threshold voltages of 2.0 V and 2.5 V can be used.

#### Functionality:

As soon as  $V_{DS}$  7/8 drops below the threshold voltage, the MOSFET activates. To switch Off the MOSFET, the Cur5/6H - Feedback signal is used. The `cur5/6h_dcdc` current feedback takes priority if both feedbacks are high. LS7 or LS8 is driven low. This event is not guaranteed in every condition. When LS is off, there is an oscillation on  $V_{DS}$  (hence the resonant name) whose amplitude depends on the " $V_{BOOST}-V_{BAT}$ " value, so a timeout is used to make sure the LS can be enabled again.

#### Timeout functionality:

If  $V_{BOOST}$  drops below  $V_{BAT} \times 2$ , the  $V_{DS}$  threshold of 2.5 V is not crossed. The MOSFET cannot be activated again, which means the DC/DC converter stops. A timeout must be started each time the lower current threshold is reached (`cur5/6l_fbk` signal changed to low). When  $V_{DS}$  feedback is set low during this timeout period, the timeout monitor stops and is reset. If the  $V_{DS}$  feedback cannot set to a low during this timeout period, the MOSFET activates directly by the timeout logic.

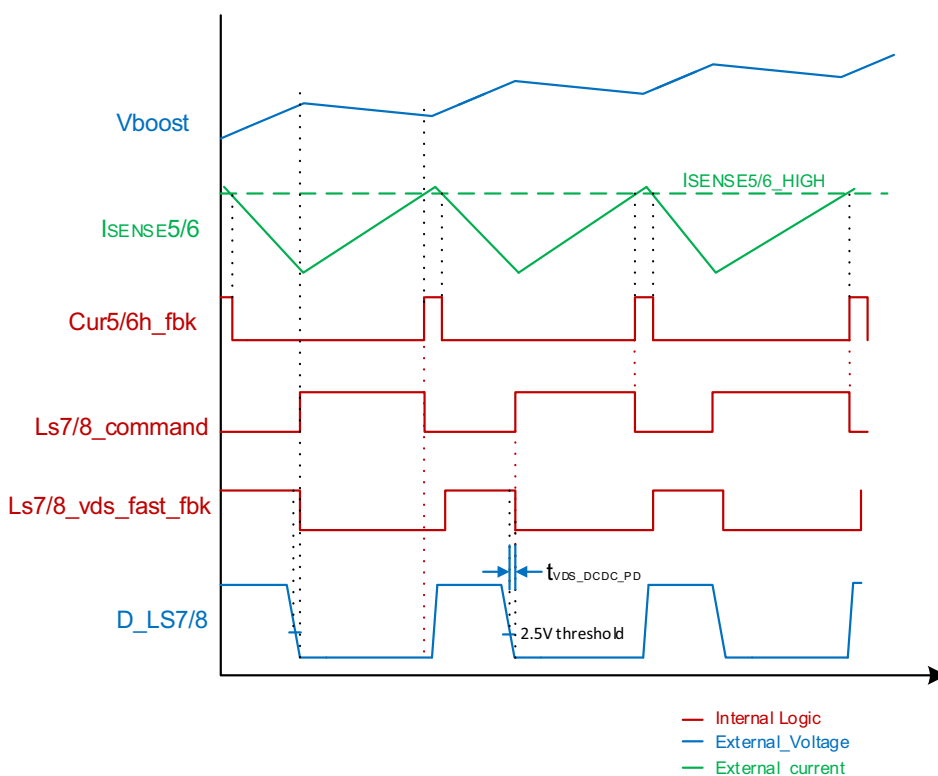


Figure 25. DC/DC mode 3 resonant (threshold 2.5 V used)

## 7.3 Device clock manager and PLL init

After 100  $\mu$ s the clock monitor is enabled, [Table 42](#) shows different strategies to start the device.

The clock manager duty is to detect a low frequency (or missing) input reference or a PLL malfunction. To do so, it monitors the PLL output, to check if the output clock frequency is within acceptable range. This is achieved counting the number of pll\_output\_clock cycles inside six periods of the backup reference clock (six periods of 1.0 MHz clock = 6.0  $\mu$ s). The expected number depends on the selected PLL multiplication factor;

- when the factor is 24, it is detected an invalid clock condition when it is possible to count more than 165 or less than 125 pll\_output\_clock cycles.
- when the factor is 12, it is detected an invalid clock condition when it is possible to count more than 84 or less than 61 pll\_output\_clock cycles.

After requesting the switch back to the external clock reference the device cannot be accessed via the SPI (see [Table 152](#), [backup\\_clock\\_status \(1A8h\)](#)) for about:

- 100  $\mu$ s if there is a valid external clock available
- 290  $\mu$ s (250  $\mu$ s+40  $\mu$ s re-lock time) if there is no valid input clock available and the device has to go back to the backup clock again.

The SPI word transmitted to set the Switch to clock pin bit has to be the last word within a SPI burst.

**Table 42. Device clock manager and PLL init**

Initial state	1. Step	2. Step	3. Step	Result
Device supplied, but in reset (ResetB), no ext. clock	Supply external 1.0 MHz clock at CLK input	Deactivate ResetB	—	Device accessible after 40 $\mu$ s (PLL lock time) and running on ext. clock
Device supplied, but in reset (ResetB), no ext. clock	Supply external 1.0 MHz clock at CLK input	Wait for $t > 20 \mu$ s	Deactivate ResetB	Device accessible immediately and running on ext. clock
Device supplied, but in reset (ResetB), no ext. clock	Deactivate ResetB	—	—	Device accessible after 140 $\mu$ s and running on backup clock
Device supplied, but in reset (ResetB), no ext. clock	Deactivate ResetB	Supply external 1.0 MHz clock at CLK input at $t < 100 \mu$ s	—	Device accessible 40 $\mu$ s (PLL lock time) after CLK available and running on ext. clock
Device running on backup clock, no ext. clock	Supply external 1.0 MHz clock at CLK input	Request switch to external clock	—	Device accessible 100 $\mu$ s after switch request and running on ext. clock
Device running on backup clock, no ext. clock	Request switch to external clock	—	—	Device accessible 290 $\mu$ s after switch request and running on backup clock

## 7.4 SW initialization flow

### 7.4.1 Power supply, reset, and clock

- Supply device
  - The device needs 5.0 V supply voltage on the VCC5 pin and 3.3 V or 5.0 V supply voltage on the VCCIO pin
  - A voltage at the VBATT and/or VBOOST pin is not mandatory for device initialization
  - As soon as the device is properly supplied at the VCC5 pin, VCC2P5 regulator is started
  - When  $V_{CC2P5}$  is above a specific threshold, the internal POResetB signal is deactivated
  - After the internal POResetB signal is deactivated, it takes a maximum time of 100  $\mu$ s until the digital outputs of the device are functional
- Setup external reference clock of 1.0 MHz at the CLK pin
  - The PLL is locked about 25  $\mu$ s after the external CLK is enabled
  - If the external clock signal availability cannot be guaranteed within this period, it is recommended to reset the device via ResetB immediately, or switch to the external clock reference later via the SPI command
- Deactivate ResetB signal
  - The external reset signal ResetB has to be deactivated if it has been active
  - As soon as there is no POResetB and ResetB signal active, the internal reset RSTB is deactivated

## 7.4.2 SPI configuration

The whole SPI configuration can be done while the device is using the internal backup clock reference. The device registers are not locked after device reset.

- Check if device is accessible via the SPI
  - Check if the device is accessible via the SPI by reading the ID/REV register
- Init the main configuration registers
  - Init the main configuration registers including the Clock Prescaler, Flag pin setup,...
  - If the application uses the internal  $V_{CCP}$  regulator to supply the pre-drivers, this regulator must be switched on now
- Set code width
  - If the microcode transmits using one single SPI burst, it is mandatory to write the code width register of each channel used
  - If the microcode is transmitted using multiple bursts which include information about the number of words, the code width register can also be written
  - The code width must be set before setting the `pre_flash_enable` bit, because the checksum calculation information is required
- Download microcode
  - Download microcode via the SPI for each channel used
- Set the CRC32 checksum
  - Set the `checksum_l/h` register (32-bit) of each channel used
- Init diagnostics configuration registers
- Init I/O configuration registers
- Init channel configuration registers
- Init DRAM values for the first time
  - Depending on the application and the microcode, it could be required to set up DRAM parameters of the channels used
- Set the lock bit in the `device_lock` register (optional)

## 7.4.3 Clock monitor, flash enable, and DrvEn

- Check if the device is running on an external reference clock
  - It is recommended to verify the device is running on the external clock reference. This can be checked by reading a bit in the `driver_status` SPI register
- If the device is running on the backup clk, it is possible to switch the clock manager to external reference via a SPI command
  - This is only mandatory if the external clock reference is not available in time and the device is running on the backup oscillator's clock
  - The clock manager is forced to try to switch back to the external reference by a SPI write to a dedicated bit
  - SPI transfers have to be avoided when switching the clock reference. The SPI module is in reset as long as there is no valid clock
  - Do not switch the clock reference while the first checksum calculation is running
- Set the `pre_flash_enable` bit
  - Set the `pre_flash_enable` bit of the used channel(s)
  - This bit "freezes" the CRAM and enables the signature unit to perform the CRC32 check for the first time
  - After the signature unit has finished the first CRC32 check successfully, it sets the `flash_enable` bit to start the microcore(s) of the used channel(s)
  - The microcode should check for the `flash_enable` bit with a timeout ensuring the microcores are running
- Activate the DrvEn signal
  - Depending on the application and the microcode, it could be required to activate the DrvEn signal if deactivated

## 7.5 BIST

The device has a built-in self test (BIST) for the memory (MBIST for CRAM and DRAM) and for the logic core (LBIST). The BIST can be started by the SPI (see [Table 170, Bist\\_interface in write mode \(1BDh\)](#)). A full LBIST check of the device digital core and MBIST check of the device memories can be required accessing the `BIST_register` in Write mode and writing a 16-bit password. This request is accepted only if all three CRAMs are unlocked. It is recommended to run MBIST and LBIST during the initialization phase, since the DRAM and CRAM are erased during BIST.

After this request is performed, the LBIST and MBIST check starts and its evolution can be monitored accessing the same `BIST_register` in read mode.



## 7.5.1 MBIST

The MBIST is started by writing the MBIST password (B157h) to the BIST register (see [Table 170, Bist\\_interface in write mode \(1BDh\)](#)). The overall MBIST operation takes about 2.2 ms (at 24 MHz) to complete. During the memory BIST five different tests are performed using different patterns to test the RAM. The patterns are:

- All 00, All 11
- All 55, All AA
- All 0F, All F0
- All 00, All FF
- All FF, All 00

While the MBIST is running the digital core of the device is functional. The SPI interface can be used. It is not possible to use the CRAM and DRAM.

## 7.5.2 LBIST

The LBIST is started by writing the LBIST password (0666h) to the BIST register (see [Table 170, Bist\\_interface in write mode \(1BDh\)](#)). The overall LBIST operation takes about 32 ms (at 24 MHz) to complete. The coverage of the LBIST is > 92%.

The SPI interface is also covered by the LBIST. During the LBIST the device is no longer accessible via the SPI. When the LBIST is started it takes control of the IRQB pin, which is used as a digital output signal to show the LBIST is busy. The IRQB pin is set low while the LBIST is running. When the LBIST is finished, the IRQB signal goes high again.

The LBIST could fail in a way the IRQB signal never goes high again. In this case, the microcontroller needs to monitor this and reset the device. When LBIST is finished (IRQB signal high), the device can be accessed by the SPI again and the LBIST result is available in the BIST register (see [Table 171, Bist\\_interface in read mode \(1BDh\)](#)). After reading the results, the LBIST needs to be cleared by sending a CLEAR command (refer to [Table 170, Bist\\_interface in write mode \(1BDh\)](#)). This releases the logic and the device returns to its original state. It is recommended to check if the LBIST result is reset to "00" to ensure the LBIST clear command was successful.

The LBIST can be interrupted by a hardware reset via the RESETB pin. There is a second way to stop the LBIST after it is started by writing the LBIST password. This is done by setting the Flag0 input pin to high. There is information in the LBIST result indicating the LBIST was stopped by Flag0. Also in this case, an LBIST clear command (C1A0h) has to be sent or the RESETB pin must be set low to return to normal operation mode. The FLAG0 pin has a weak PD resistor. If the pin is not connected, the LBIST runs and cannot be stopped by Flag0.

## 7.6 Reset sources

The device has three possible reset sources:

- the resetb input reset pin is driven low
- the poresetb (power on reset) signal generated by the internal voltage regulator, incase an undervoltage is detected on VCC2P5
- a SPIresetb request received through SPI, when the appropriate code is written to the "global reset registers" (refer to [See SPIReset global reset register 1 and 2 on page 126](#)). It is kept asserted for a fixed time (333 ns) then it is released.

Reset source can be determined reading the Reset\_source register (refer to [Table 168, Reset\\_Source \(1B7h\)](#)).

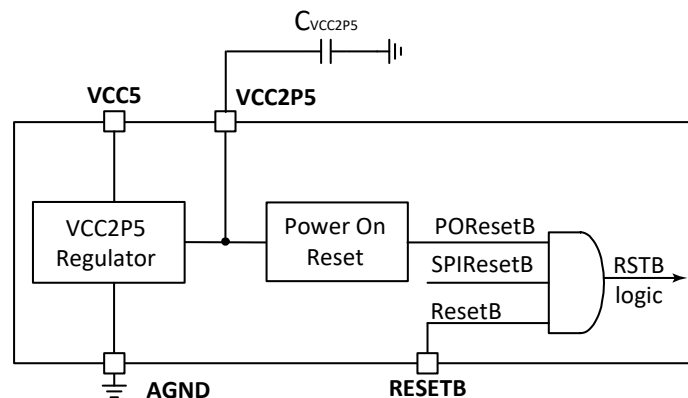


Figure 26. VCC2P5 and reset sources

As long as RSTB is asserted, the SPI module is also inactive. In order to understand when the device has gone out of reset state, the microcontroller should poll the device on the SPI. This can be done by either sending any message to the device and checking for the control pattern (A8h) on the MISO during the command word or by reading out any register with a reset value not equal to zero (e.g. ID register).

## 7.7 Cipher unit

This block has the function to secure the code downloaded by the microcontroller into the code RAM via the SPI. The data loaded at device startup must be encrypted with the suitable cipher. This block receives an encoded SPI stream and decodes it at runtime. The decoded microcode is then stored in the code RAM. This feature cannot be disabled. The cipher algorithm is re-initialized every time the code memory is selected by a write operation to the Selection register (3FFh).

## 7.8 Ground connections

The device integrates three separate ground pins: PGND, DGND, and AGND:

- PGND is the substrate connection and is only connected to the package exposed pad, to guarantee a low-impedance connection and get optimized EMC performances. PGND is the reference ground for the  $V_{CCP}$  regulator, some analog functions, and all of the low-side pre-drivers. It is highly recommended to directly connect PGND to the ECU ground plane.
- DGND is the reference ground for the digital logic core. It is highly recommended to directly connect DGND to the ECU ground plane. The microcontroller as well as other logic devices communicating with the device should share the same reference ground connected to the ground plane to prevent noise.
- AGND is the ground for all the noise sensitive analog blocks integrated into the device. This pin should be connected to the analog ground of the ECU. A star connection is recommended to guarantee a clean analog signal acquisition of the OAX\_x pins from the MCU.

Due to their functionality, some analog functions are referred to PGND:

- VDS monitors the low-side drivers
- VSRC monitors the high-side drivers
- The load biasing S\_HSX regulator and the D\_LsX pull-down

All the ground pins of the device should be connected to the same ground voltage. Even during transient conditions, the voltage difference between PGND, DGND, and AGND must be limited to  $\pm 0.3$  V. The layout of the ground connection of the ECU should be carefully designed to limit the ground noise generated as much as possible, for instance during fast switching of the external power MOSFETs.

The decoupling and filter capacitors at the different supply voltage pins should be implemented as described by the following:

- VCC5 to AGND
- VCCIO to DGND
- VCC2P5 to DGND
- VCCP to PGND
- VBATT to PGND
- VBOOST to AGND or PGND

### 7.8.1 Detection of missing GND connections

The PT2000 can detect any single or multiple missing connection of any ground pin (PGND, DGND, AGND) of the device. At least one ground must remain connected to allow the loss of ground detection. If the ground disconnection is detected, the internal signal  $uv\_vccp$  is asserted and all the pre-drivers are disabled. The ground lost detection is filtered to allow the device to work in a proper way for a time of typically  $t_{FILTER\_UVVCCP}$  via the  $uv\_vccp$  signal.

## 7.9 Shutoff path via the DrvEn pin

The device includes a shutoff path via the DrvEn pin, which is used to safely disable the solenoid injection power stage in a fault condition of the ECU. When the DrvEn pin is negated, all pre-drivers (w/o configuration option for DrvEn) must be switched off. Status of the DRVEN pin can be read back by the SPI (refer to [Table 162](#)).

The shutoff path also works in a defined way (switch off all pre-drivers) when DrvEn is negated, even when the PT2000 is stressed with a voltage of up to 36 V at the supply and/or microcore interface (SPI, Startx,...) pins. Digital interface pins of the PT2000 are self-protected against a voltage of up to 36 V: CLK, IRQB, ResetB, DrvEn, MISO, MOSI, SCLK, CSB, Dbg, Startx (7x), Flagx (4x), OA\_x (3x).

## 7.9.1 DrvEn shutoff path of the high-side pre-driver

The DrvEn path for the HS pre-drivers HS1 to HS4 and HS6 is implemented to ensure a high safety level. It is designed with a high level of independence, because there is a direct wire from this pin to the HS pre-driver input and the failure rate of this functionality is very low. If any of the following failures occur, the shut off path is still functional or the driver must be in a safe off state. These failures are:

- Missing clock signal for the device digital core
- Missing supply voltage for the device digital core
- Missing supply voltage of level shifter
- Missing supply voltage ( $V_{BS}$ ) of HS pre-driver
- Single damaged pre-driver

**Table 43. DrvEn path for HS pre-drivers**

HS pre-driver	Implementation
HS1	Direct wire from the DrvEn pin to the HS pre-driver input. High independence and low FIT rate.
HS2	
HS3	
HS4	
HS6	
HS5	Configuration option for the DrvEn path. The signal is routed via the digital core only.
HS7	

## 7.9.2 DrvEn Shutoff path of the low-side pre-driver

The DrvEn path for the LS pre-drivers LS1 to LS5 is implemented to ensure a high safety level. It is designed with a high level of independence, because there is a direct wire from this pin to the LS pre-driver input and the failure rate of this functionality is very low. If any of the following failures occur, the shut off path is still functional or the driver must be in a safe off state. These failures are:

- Missing clock signal for the device digital core
- Missing supply voltage for the device digital core
- Missing supply voltage VCCP of LS pre-driver
- Single damaged pre-driver

**Table 44. DrvEn path for LS pre-drivers**

LS Pre-driver	Implementation
LS1	Direct wire from the DrvEn pin to the LS pre-driver input. High independence and low FIT rate.
LS2	
LS3	
LS4	
LS6	
LS5	Configuration option for the DrvEn path. The signal is routed via the digital core only.
LS7	

# 8 Digital core

The digital core controls the actuation of the electro-actuators. The digital core serves as the main microcore features for actuator control, structures for HW configuration, and the communication interface with the ECU microcontroller.

There are six total microcores which can run concurrently and independently. The microcores are arranged in channels. Each channel contains two microcores, one instruction RAM (CRAM), and one data RAM (DRAM). This architecture of shared RAM allows efficient use of the RAM when multiple actuators need to be controlled in parallel with identical microcode. Each digital core communicates with the external microcontroller through a SPI bus and by means of a set of single wire I/O. Accessible by the SPI are:

- a). Registers with dedicated function
- b). Data RAM

To avoid access conflicts, a time division multiplex scheme controls the access to the RAMs by the different entities. Thus, microcode runtime of one microcore is not influenced by RAM access from the SPI or the other microcores.

## 8.1 Logic channels description

This section describes the features and operation of the microcores (central processing unit, or CPU, and development support functions) used in the PT2000.

The PT2000 provides a set of three logic channels each channels which include:

- Two 16-bit processing units (microcores) having a specific programming model
- One Code RAM - 1023 x 16-bit. The memory dedicated to microcode storage is shared between the two microcores of logic channel
- One Data RAM - 64 x 16-bit. The memory dedicated to variable storage is shared between the two microcores of a logic channel

[Figure 27](#) describes logic channel 1. The other channels are identical.

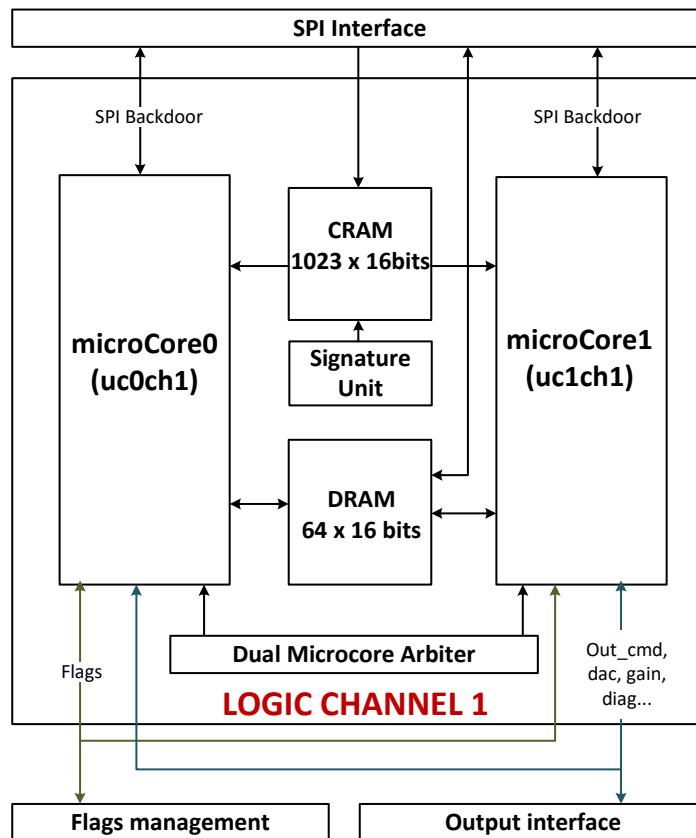


Figure 27. Logic channel 1 diagram

## 8.1.1 Microcores

The microcores are actually small microprocessor cores. These are typically programmed to implement software based finite state machines controlling the actuator operation. The microcores have access to the analog input and output functions, but are also able to communicate with each other by a flag bus.

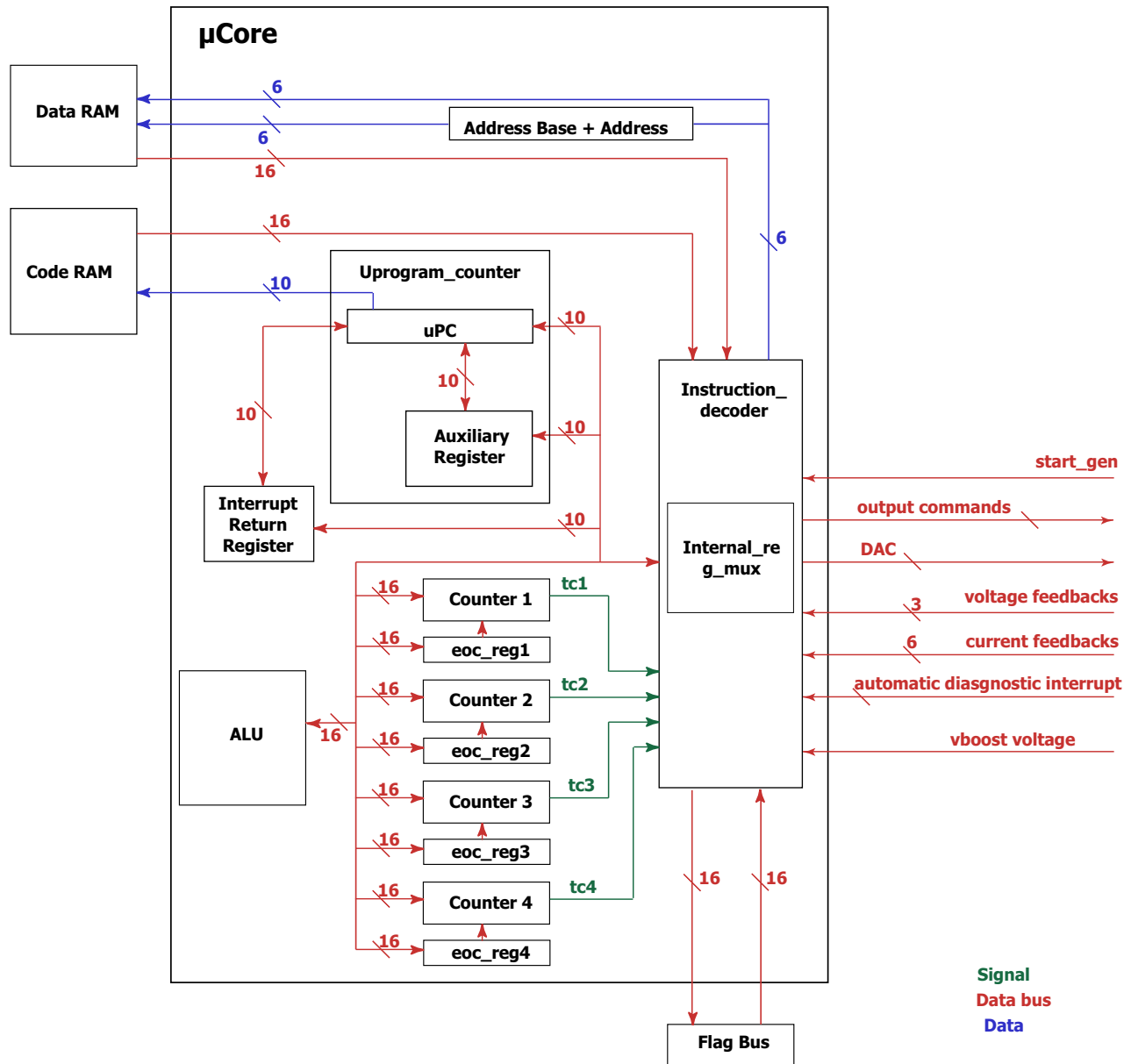


Figure 28. Microcore block diagram

For further detail on how to program the microcores, (reference [Programming Guide and Instruction Set](#)).

## 8.1.2 Dual microcore arbiter

This block handles the access to code RAM and data RAM memories by the different possible users:

- the two microcores
- the signature unit (code RAM only)
- the SPI interface

### 8.1.2.1 Access sequence to code RAM

When the device is operating in single microcore mode, access slots to code RAM are granted according to [Table 45](#).

**Table 45. Code RAM access sequence (single microcore mode)**

Ck_per	flash_enable	T0	T1	T2	T3	Teven	Todd
1	1	uc0	CHKSM	-	-	-	-
1	0	SPI r/w	SPI r/w	-	-	-	-
2	1	uc0	CHKSM	CHKSM	-	-	-
2	0	SPI r/w	SPI r/w	SPI r/w	-	-	-
3	1	uc0	CHKSM	CHKSM	SPI r	-	-
3	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	-	-
4	1	uc0	CHKSM	CHKSM	SPI r	CHKSM	SPI r
4	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w

The clock divider = ck\_per + 1 (refer to [Table 136, Clock\\_Prescaler \(1A0h\)](#)). Teven represents all the time slots with an even number id from T4, and following T4. Todd represents all the time slots with an odd number id from T5 and following T5. When the device is operating in dual microcore mode, access slots to Code RAM are granted according to [Table 46](#).

**Table 46. Code RAM Access Sequence (dual microcore mode)**

Ck_per	flash_enable	T0	T1	T2	T3	Teven	Todd	Cycle stealing when uc0/1 are in wait
1	1	uc0	uc1					CHKSM
1	0	SPI r/w	SPI r/w					
2	1	uc0	uc1	CHKSM				CHKSM
2	0	SPI r/w	SPI r/w	SPI r/w				
3	1	uc0	uc1	CHKSM	SPI r	-	-	CHKSM
3	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	-	-	
4	1	uc0	uc1	CHKSM	SPI r	CHKSM	SPI r	CHKSM
4	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w	

The PT2000 allows using a dual sequencer with a ck\_per = 1, which means with a clock at 12 MHz, however some restrictions apply on the DRAM access ([See Access sequence to data RAM on page 71](#)).

## 8.1.2.2 Access sequence to data RAM

When the device is operating in single microcore mode, access slots to data RAM are granted according to [Table 47](#).

**Table 47. Data RAM access sequence (single microcore mode)**

Ck_per	flash_enable	T0	T1	T2	T3	Tother	Tlast
1	1	SPI r/w	uc0	-	-	-	-
1	0	SPI r/w	SPI r/w	-	-	-	-
2	1	SPI r/w	SPI r/w	uc0	-	-	-
2	0	SPI r/w	SPI r/w	SPI r/w	-	-	-
3	1	SPI r/w	SPI r/w	SPI r/w	uc0	-	-
3	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	-	-
4+	1	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w	uc0
4	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w

The clock divider = ck\_per + 1' (refer to [Table 136, Clock\\_Prescaler \(1A0h\)](#)). Tlast represents the last time slot. Tother represent all time slots (if any) between T3 and Tlast. When the device is operating in dual microcore mode, access slots to data RAM are granted according to [Table 48](#)

**Table 48. Data RAM Access Sequence (dual microcore mode)**

Ck_per	flash_enable	T0	T1	T2	T3	Teven	Todd	Cycle stealing when uc0/1 are in wait
1	1	uc1	uc0					SPI r/w
1	0	SPI r/w	SPI r/w					
2	1	uc1	SPI r/w	uc0				SPI r/w
2	0	SPI r/w	SPI r/w	SPI r/w				
3	1	uc1	SPI r/w	SPI r/w	uc0	-	-	SPI r/w
3	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	-	-	
4	1	uc1	SPI r/w	SPI r/w	SPI r/w	SPI r/w	uc0	SPI r/w
4	0	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w	SPI r/w	

Note that when ck\_per is equal to 1 and dual microcore mode is enabled, there are some limitations for the DRAM access.

As shown in [Table 48](#) when "ck\_per = 1" the SPI has no dedicated access slot to the DRAM. At the same time it is clear for most applications, both microcores do not have to access the DRAM every microcontroller clock cycle. It is possible to use some unused slots from the microcores for the SPI.

The limit with the 12 MHz dual sequencing access to the data RAM: there is a strong limit on how many instructions in a row the data ram can access (load, store instructions). This limit depends on the SPI frequency used in the application.

As a consequence, the microcore 0 and 1 must not block the DRAM access slots for longer than the given number of ck cycles minus 1. If this limit is achieved, it automatically be reported by the NXP IDE (compiler).

Table 49 shows the smallest period in  $\mu\text{s}$  and ck (12 MHz) clock cycles between the address available and first read data bit for SPI read access, which could occur based on a given SPI baud rate.

**Table 49. SPI baud rate and DRAM access sequence for the ck\_prescaler = “1”**

SPI baud rate (MHz)	Minimum SPI read address to first read data bit delay (5 bits)	
	$\mu\text{s}$	ck Cycles (12 MHz)
1.0	5.0	60
2.0	2.5	30
4.0	1.25	15
8.0	0.625	7.5
10	0.5	6.0

### 8.1.3 Signature unit

The task of the signature unit is to compute a checksum of the CRAM to detect possible memory corruption.

The computation is first started when the corresponding CRAM is locked by the pre\_flash\_enable (see Table 58, Flash\_enable (100h, 120h, 140h)). When the computation is complete, the result of the computation is compared to the checksum registers (see Table 66, checksum\_h (108h, 128h, 148h)) and Table 67, checksum\_l (109h, 129h, 149h)). These registers contain the golden checksum, provided during the init phase through the SPI and calculated automatically by the PT2000 IDE.

If the result is correct, the signature unit sets the flash\_enable bit (see Table 58, Flash\_enable (100h, 120h, 140h)). If the result is not correct, an interrupt (optional, refer to Table 58, Flash\_enable (100h, 120h, 140h)) is issued towards the microcontroller and both microcores accessing the same CRAM are disabled.

The signature unit can be disabled by writing the Checksum\_failure\_disable bit in the flash\_enable\_reg (refer to Table 58, Flash\_enable (100h, 120h, 140h)). When the signature is disabled, the flash\_enable bit is set immediately after the pre\_flash\_enable bit and a failed checksum causes only a warning (set the appropriate bit in the flash\_enable register) without disabling code execution.

The computation requires a different time according to the ck\_per value and to the code\_width (refer to Table 65, code\_width (107h, 127h, 147h)). The worst case computation time for a ck\_per of 1 and higher ( $ck = cksys/2+1$  or smaller) is  $20 * t_{CKSYS} * (code\_width-2)$ . For example, for a completely used memory, a ck\_per of 2 and a cksys clock frequency of 24 MHz (ck at 6.0 MHz), the computation takes 851  $\mu\text{s}$ .

The signature unit works only for a code width of 3 or larger. If a shorter code of 1 to 2 words is used, the signature unit has to be disabled.

### 8.1.4 SPI backdoor

It is also possible to access (both to read and to write) to all the registers normally accessible through the SPI by using an SPI backdoor. Both the SPI read and write operations are multi cycle operations. Table 51 shows the number of cycles needed. The registers must not be changed while the operation is in progress.

To read an SPI register, first the 8 LSBs of the address must be provided in the 8 LSBs of the “SPI address” at internal memory map address. A read operation must then be requested with the “rdspi” instruction (refer to programming guide). The result is available at the “SPI data” address of the internal memory map.

To write an SPI register, first the 8 LSBs of the address must be provided in the 8 LSBs of the “SPI address” address and the data to write must be provided at the “SPI data” address. Then a write operation must be requested with the “wrspi” instruction (refer to programming guide).

There are some access limitations when requesting write access to SPI registers via the SPI backdoor:

- First of all, it is only possible to write to SPI registers which are not locked at the moment the write operation “wrspi” is requested.
- For some special registers there are additional limitations dependant on the configuration of the device. Table 51 shows the different limitations.



Both the SPI read and write operations are multi cycle operations. [Table 50](#) shows the number of cycles needed. The registers and the SPI address mode (set using “sisa” instruction) must not be changed while the operation is in progress.

**Table 50. Cycles for SPI backdoor read/write**

ck_prescaler value	Cycles for SPI backdoor r/w
1	4
2	3
3+	2

**Table 51. SPI backdoor access limitation**

SPI_registers	Access rule	Configuration controlling access rule
vds_threshold_hs, vsrc_threshold_hs, vds_threshold_ls	Only microcores which are allowed to control a certain HS or LS pre-driver are allowed to change the corresponding $V_{DS}$ and $V_{SRC}$ threshold. Changes to all other $V_{DS}$ and $V_{SRC}$ values are ignored.	out_acc_seqXchY
hs_slewrate, ls_slewrate	Only microcores which are allowed to control a certain HS or LS pre-driver are allowed to change the corresponding slew rate setting. Changes to all other slew rate settings are ignored.	out_acc_seqXchY
hs_bias_config ls_bias_config	Only microcores which are allowed to control a certain HS or LS pre-driver are allowed to control the corresponding biasing source. Changes to all other biasing sources are ignored	out_acc_seqXchY
vds7_dcdc_config vds8_dcdc_config	Only microcores which are allowed to control the LS7 or LS8 pre-driver are allowed to change the corresponding vds_dcdc_timeout register setting. All other changes are ignored	out_acc_seqXchY
dac1, dac2, dac3, dac4, dac5l, dac5h, dac5neg, dac6l, dac6h, dac6neg, boost_dac	No access is possible through the SPI backdoor	

## 8.1.5 CRAM

The code RAM is a 1024x16 single port RAM memory defined to store the micro-code for the entire channel (both microcore0 and microcore1 if dual microcore mode is enabled).

When enabled, the two microcores can execute either exactly the same code or separate codes, in which case the memory space dedicated to each microcore are a subset of the overall CRAM. This use of the CRAM memory is controlled by configuration registers defining the entry point (meaning the starting address) of each microcore (refer to [Table 68, uc0\\_entry\\_point \(10Ah, 12Ah, 14Ah\)](#) and [Table 69, uc1\\_entry\\_point \(10Bh, 12Bh, 14Bh\)](#)).

## 8.1.6 DRAM

The data RAM is a 64x16-bit RAM which can be used as an interface between microcores and the external microprocessor, and also to store data used only internally by the microcores. The data RAM is accessed as a “flat” memory, where all the 64 memory locations can be accessed by the external microcontroller and both microcores.

## 8.2 Serial peripheral interface

The communication between the PT2000 and the main microcontroller is managed with a 16-bit SPI interface.

This block is the module providing the SPI connection features. The block is full-duplex, so it can receive and transmit at the same time. The device requires a cphase value of 1 and a cpol value of 0. This means the SPI module samples the MOSI signal, during write operations, on the falling edge of the serial clock. Likewise, during read operations, the SPI module always puts the output value available on the MISO signal on the rising edge of the sclk clock.

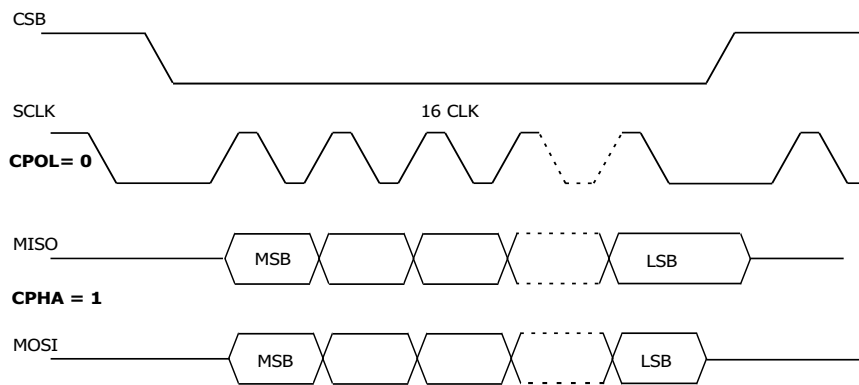


Figure 29. SPI protocol diagram

While the PT2000 component has many memory locations to access, it was necessary to develop a protocol to manage this. The protocol is implemented in such a way, so after a reset occurs and after any burst transaction on the SPI connection, the protocol always waits for a control word. This is a 16-bit word built as follows:

Table 52. SPI protocol

control_word	Meaning
control_word (bit 15)	r_w: read (1)/write (0) operations
control_word (bits 14 to 5)	offset: start address
control_word (bits 4 to 0)	number: number of operations

When this control word is received, the protocol understands the external micro-controller wants to perform a read (when this bit is set to 1) or write (when this bit is set to 0) operation looking at the r\_w bit. Looking at the number value, the protocol then understands how many concurrent read of write operations the external microcontroller wants to perform (max 31). Finally, looking at the offset value, the protocol understands where these operations should start, meaning what is the first address in this burst of operations to be accessed.

## 8.2.1 SPI read access

A SPI burst for read access consists of 2 to 32 16-bit words. The way the number of words is defined depends on the SPI mode used (A or B). Table 53 shows the data transmitted on MOSI and MISO. During the command word the check byte and the SPI error status is transmitted via the MISO line.

**Table 53. SPI read access**

MOSI word 1 (control word) – read access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	r_w	offset										number				
Value	1	0 to 1023										n=0 to 31				
MISO word 1 (control word) – read access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	check byte												cksys missing	frame error	word error	
Values	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0
Value HEX	A				A				A				8			

MOSI word 2 (data) – read access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	empty (don't care)															
Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MISO word 2 (data) – read access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	read data 1															

MOSI word n+1 (data) – read access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	empty (don't care)															
Value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MISO word n+1 (data) – read access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	read data n															

## 8.2.2 SPI write access

A SPI burst for write access consists of 2 to 32 16-bit words. The way the number of words is defined depends on the SPI mode used (A or B). Table 54 shows the data transmitted on MOSI and MISO. During the command word and all data words, the check byte, and the SPI error status is transmitted via the MISO line.

**Table 54. SPI write access**

MOSI word 1 (control word) - write access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	r_w	offset										number				
Value	0	0 to 1023										n=0 to 31				
MISO word 1 (control word) - write access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	check byte													cksys missing	frame error	word error
Values	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0
Value HEX	A				A				A				8			

MOSI word 2 (data) - write access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	write data 1															
MISO word 2 (data) - write access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	check byte													cksys missing	frame error	word error
Values	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0
Value HEX	A				A				A				8			

MOSI word n+1 (data) - write access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	write data n															
MISO word n+1 (data) - write access																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	check byte													cksys missing	frame error	word error
Values	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	0
Value HEX	A				A				A				8			

## 8.2.3 SPI protocol

### 8.2.3.1 Mode A

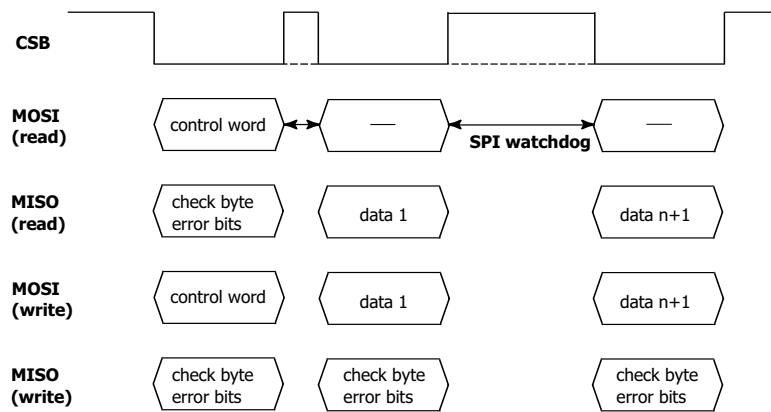


Figure 30. SPI protocol mode A CSB, MOSI, and MISO

The maximum delay between data belonging to the same burst is specified by the `spi_watchdog` parameter (refer to [Table 153, Spi\\_config \(1A9h\)](#)). Between one data transfer and the next, the SPI chip select can be asserted. If the delay exceeds the watchdog time, the SPI interface goes into the error state.

It is possible to perform long burst transfers by sending a control word with the parameters number and offset set to zero. The effect varies according to the current value of the channel select register (refer to [Table 56, selection\\_register \(3FFh\)](#)):

- If the value of channel select register is "0xxxx001", the protocol performs a burst of operations starting from address 0; the number of operations is specified by the value of the `code_width` register (refer to [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)) of channel 1. This command is used to write the whole CRAM of channel 1 with only one command word.
- If the value of channel select register is "0xxxx010", the protocol performs a burst of operations starting from address 0; the number of operations is specified by the value of the `code_width` register (refer to [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)) of channel 2. This command is used to write the whole CRAM of channel 2 with only one command word.
- If the value of channel select register is "0xxxx100", the protocol performs a burst of operations starting from address 0; the number of operations is specified by the value of the `code_width` register (refer to [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)) of channel 3. This command is used to write the whole CRAM of channel 3 with only one command word.
- If the value of channel select register is "0xxxx011", the protocol performs a burst of operations starting from address 0; the number of operations is specified by the value of the `code_width` register (refer to [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)) of channel 1. This command is used to fully write the CRAMs of channel 1 and 2 (with exactly the same code) with only one command word.
- If the value of channel select register is "0xxxx101", the protocol performs a burst of operations starting from address 0; the number of operations is specified by the value of the `code_width` register (refer to [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)) of channel 1. This command is used to fully write the CRAMs of channel 1 and 3 (with exactly the same code) with only one command word.
- If the value of channel select register is "0xxxx110", the protocol performs a burst of operations starting from address 0; the number of operations is specified by the value of the `code_width` register (refer to [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)) of channel 2. This command is used to fully write the CRAMs of channel 2 and 3 (with exactly the same code) with only one command word.
- If the value of channel select register is "0xxxx111", the protocol performs a burst of operations starting from address 0; the number of operations is specified by the value of the `code_width` register (refer to [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)) of channel 1. This command is used to fully write the CRAMs of channel 1, 2 and 3 (with exactly the same code) with only one command word.
- If the value of channel select register is "1xxxx000", the protocol performs a burst of operations starting from address 0; the number of operations is 192. This command is used to fully write the DRAMs of all three channels with only one command word.
- For all the other values of channel select register, the command is neglected.

Sending a control word with the parameter number set to zero and the parameter offset greater than zero is not allowed. This leads to data corruption in the registers or DRAM.

For example, to initialize the values of 24 LS pre-driver diagnosis configuration registers, it is necessary to:

- Select the communication interface as the target: this is done by writing the value 0100h at the address 3FFh. First send the data “0\_11111111\_00001” (7FE1h) via the SPI. As the ASIC is in idle conditions, it uses this data as a command word. In particular, this specific command word of the example means: write (because of the initial 0) starting from address 3FFh (the ten bits immediately after) 1 word (the five bits at the end). The next data to send is 0100h. The SPI block is expecting a write to 3FFh, so write 0100h in the location 3FFh. As the number of word expected is arrived, the SPI block returns to the idle state;
- Write the value of the 24 registers: the SPI block is waiting for a command word. The correct data to send is “0\_011100000\_11000”. This means write (0) starting form address 1C0h (011100000) 24 words (11000). The next 24 words are written to the communication interface registers.

### 8.2.3.2 Mode B

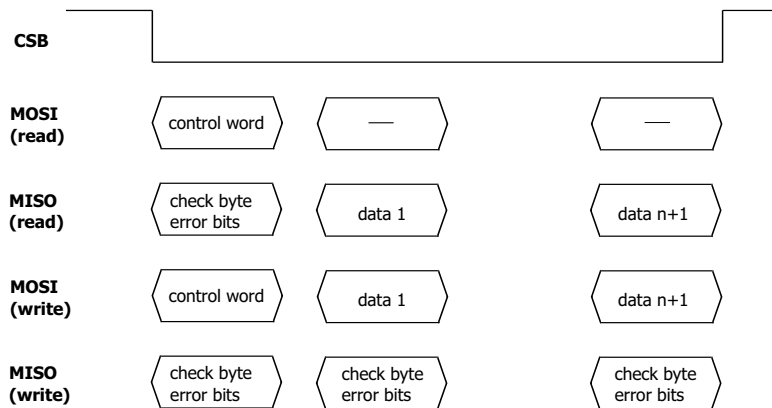


Figure 31. SPI protocol mode B CSB, MOSI, and MISO

For the duration of a burst of data, the SPI chip select must be asserted, even between data transfers. The first word after the chip select assertion is considered to be the command word and the following words are the data. If the number parameter is not zero, the SPI interface goes into error state if:

- the chip select is de-asserted and the number of words transferred is lower than the number parameters required by the command word,
- the number of word transferred is equal to the number parameter + 1.

If the number parameter is zero, there is no check on the number of words transferred, and the length of the burst is decided only by the assertion of SPI chip select. It is not recommended to read from any register which is “reset on read” nor from any register which is located one address before such a register using a mode B burst with the number parameter set to zero. This may lead to the effect of a register reset, which is not a read out via the SPI. This problem does not occur if the number parameter is equal to the number of data words transmitted.

Note: If one additional data word is sent it is detected, but also written. For example:

- SPI write access mode B, parameter number set to 3 and 3 data words written with no error
- SPI write access mode B, parameter number set to 3 and 4 data words written with a SPI frame error after the 4th data word, but the 4th data word is written to RAM or the register.
- SPI write access mode B, parameter number set to 3 and 6 data words written with a SPI frame error after the 4th data word, but the 4th data word is written to RAM or the register. 5th and 6th data words are not written to memory.

## 8.3 SPI address map

Table 55. MC33PT2000 address map

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"0...001" / ch_sel_1(1)	0	yes	Code RAM of channel 1, <a href="#">See CRAM on page 73</a>		Code RAM of channel 1
	...				
	3FE				
"0...010"/ ch_sel_2(1)	0	yes	Code RAM of channel 2, <a href="#">See CRAM on page 73</a>		Code RAM of channel 2
	...				
	3FE				
"0...100"/ ch_sel_3(1)	0	yes	Code RAM of channel 3, <a href="#">See CRAM on page 73</a>		Code RAM of channel 3
	...				
	3FE				
"1...000" / ch_sel_1(0)	0	no	Data RAM of channel 1, <a href="#">See DRAM on page 73</a>		Data RAM of channel 1
	...				
	2F				
	30	yes	Data RAM of channel 1, private area, <a href="#">See DRAM on page 73</a>		
	...				
	3F				
"1...000" / ch_sel_2(0)	40	no	Data RAM of channel 2, <a href="#">See DRAM on page 73</a>		Data RAM of channel 2
	...				
	6F				
	70	yes	Data RAM of channel 2, private area, <a href="#">See DRAM on page 73</a>		
	...				
	7F				
"1...000" / ch_sel_3(0)	80	no	Data RAM of channel 3, <a href="#">See DRAM on page 73</a>		Data RAM of channel 3
	...				
	9F				
	A0	yes	Data RAM of channel 3, private area, <a href="#">See DRAM on page 73</a>		
	...				
	BF				

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1...000" / ch_sel_1(2)	C0		64 free address		Configuration registers of channel 1
	...				
	FF				
	100	yes	flash_enable of channel 1	<a href="#">Table 58</a>	
	101	no	ctrl_reg_uc0 of channel 1	<a href="#">Table 59</a>	
	102	no	ctrl_reg_uc1 of channel 1	<a href="#">Table 60</a>	
	103	yes	start_config_reg_part 1 of channel 1	<a href="#">Table 61</a>	
	104	yes	start_config_reg_part 2 of channel 1	<a href="#">Table 62</a>	
	105	-	status_reg_uc0 of channel 1	<a href="#">Table 63</a>	
	106	-	status_reg_uc1 of channel 1	<a href="#">Table 64</a>	
	107	yes	code_width of channel 1	<a href="#">Table 65</a>	
	108	yes	checksum_h of channel 1	<a href="#">Table 66</a>	
	109	yes	checksum_l of channel 1	<a href="#">Table 67</a>	
	10A	yes	uc0_entry_point of channel 1	<a href="#">Table 68</a>	
	10B	yes	uc1_entry_point of channel 1	<a href="#">Table 69</a>	
	10C	yes	diag_routine_addr of channel 1	<a href="#">Table 70</a>	
	10D	yes	driver_disabled_routine_addr of channel 1	<a href="#">Table 71</a>	
	10E	yes	sw_interrupt_routine_addr of channel 1	<a href="#">Table 72</a>	
	10F	no	uc0_irq_status of channel 1	<a href="#">Table 73</a>	
	110	no	uc1_irq_status of channel 1	<a href="#">Table 74</a>	
	111	yes	counter34_prescaler of channel 1	<a href="#">Table 75</a>	
	112	yes	dac_rtx_cr_config of channel 1	<a href="#">Table 77</a>	
	113	no	unlock_word of channel 1	<a href="#">Table 77</a>	
	114		12 free addresses		
	...				
	11F				



**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed	
"1...000" / ch_sel_2(2)	120	yes	flash_enable of channel 2	<a href="#">Table 58</a>	Configuration registers of channel 2	
	121	no	ctrl_reg_uc0 of channel 2	<a href="#">Table 58</a>		
	122	no	ctrl_reg_uc1 of channel 2	<a href="#">Table 60</a>		
	123	yes	start_config_reg_part1 of channel 2	<a href="#">Table 61</a>		
	124	yes	start_config_reg_part2 of channel 2	<a href="#">Table 62</a>		
	125	-	status_reg_uc0 of channel 2	<a href="#">Table 63</a>		
	126	-	status_reg_uc1 of channel 2	<a href="#">Table 64</a>		
	127	yes	code_width of channel 2	<a href="#">Table 65</a>		
	128	yes	checksum_h of channel 2	<a href="#">Table 66</a>		
	129	yes	checksum_l of channel 2	<a href="#">Table 67</a>		
	12A	yes	uc0_entry_point of channel 2	<a href="#">Table 68</a>		
	12B	yes	uc1_entry_point of channel 2	<a href="#">Table 69</a>		
	12C	yes	diag_routine_addr of channel 2	<a href="#">Table 70</a>		
	12D	yes	driver_disabled_routine_addr of channel 2	<a href="#">Table 71</a>		
	12E	yes	sw_interrupt_routine_addr of channel 2	<a href="#">Table 72</a>		
	12F	no	uc0_irq_status of channel 2	<a href="#">Table 73</a>		
	130	no	uc1_irq_status of channel 2	<a href="#">Table 74</a>		
	131	yes	counter34_prescaler of channel 2	<a href="#">Table 75</a>		
	132	yes	dac_rxtx_cr_config of channel 2	<a href="#">Table 77</a>		
	133	no	unlock_word of channel 2	<a href="#">Table 77</a>		
	134		12 free addresses			
	...					
	13F					

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1...000" / ch_sel_3(2)	140	yes	flash_enable of channel 3	<a href="#">Table 58</a>	Configuration registers of channel 3
	141	no	ctrl_reg_uc0 of channel 3	<a href="#">Table 58</a>	
	142	no	ctrl_reg_uc1 of channel 3	<a href="#">Table 60</a>	
	143	yes	start_config_reg_part1 of channel 3	<a href="#">Table 61</a>	
	144	yes	start_config_reg_part2 of channel 3	<a href="#">Table 62</a>	
	145	-	status_reg_uc0 of channel 3	<a href="#">Table 63</a>	
	146	-	status_reg_uc1 of channel 3	<a href="#">Table 64</a>	
	147	yes	code_width of channel 3	<a href="#">Table 65</a>	
	148	yes	checksum_h of channel 3	<a href="#">Table 66</a>	
	149	yes	checksum_l of channel 3	<a href="#">Table 67</a>	
	14A	yes	uc0_entry_point of channel 3	<a href="#">Table 68</a>	
	14B	yes	uc1_entry_point of channel 3	<a href="#">Table 69</a>	
	14C	yes	diag_routine_addr of channel 3	<a href="#">Table 70</a>	
	14D	yes	driver_disabled_routine_addr of channel 3	<a href="#">Table 71</a>	
	14E	yes	sw_interrupt_routine_addr of channel 3	<a href="#">Table 72</a>	
	14F	no	uc0_irq_status of channel 3	<a href="#">Table 73</a>	
	150	no	uc1_irq_status of channel 3	<a href="#">Table 74</a>	
	151	yes	counter34_prescaler of channel 3	<a href="#">Table 75</a>	
	152	yes	dac_rxtx_cr_config of channel 3	<a href="#">Table 77</a>	
	153	no	unlock_word of channel 3	<a href="#">Table 77</a>	

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1...000" / ext_sel_io	154	yes	fbk_sens_uc0_ch1_part1	<a href="#">Table 81</a>	IO configuration registers
	155	yes	fbk_sens_uc0_ch1_part2	<a href="#">Table 82</a>	
	156	yes	fbk_sens_uc1_ch1_part1	<a href="#">Table 81</a>	
	157	yes	fbk_sens_uc1_ch1_part2	<a href="#">Table 82</a>	
	158	yes	fbk_sens_uc0_ch2_part1	<a href="#">Table 81</a>	
	159	yes	fbk_sens_uc0_ch2_part2	<a href="#">Table 82</a>	
	15A	yes	fbk_sens_uc1_ch2_part1	<a href="#">Table 81</a>	
	15B	yes	fbk_sens_uc1_ch2_part2	<a href="#">Table 82</a>	
	15C	yes	fbk_sens_uc0_ch3_part1	<a href="#">Table 81</a>	
	15D	yes	fbk_sens_uc0_ch3_part2	<a href="#">Table 82</a>	
	15E	yes	fbk_sens_uc1_ch3_part1	<a href="#">Table 81</a>	
	15F	yes	fbk_sens_uc1_ch3_part2	<a href="#">Table 82</a>	
	160	yes	out_acc_uc0_ch1	<a href="#">Table 83</a>	
	161	yes	out_acc_uc1_ch1	<a href="#">Table 83</a>	
	162	yes	out_acc_uc0_ch2	<a href="#">Table 83</a>	
	163	yes	out_acc_uc1_ch2	<a href="#">Table 83</a>	
	164	yes	out_acc_uc0_ch3	<a href="#">Table 83</a>	
	165	yes	out_acc_uc1_ch3	<a href="#">Table 83</a>	
	166	yes	cur_block_access_part1	<a href="#">Table 85</a>	
	167	yes	cur_block_access_part2	<a href="#">Table 86</a>	
	168	yes	cur_block_access_part3	<a href="#">Table 87</a>	
	169	yes	fw_link	<a href="#">Table 88</a>	
	16A	yes	fw_ext_req	<a href="#">Table 90</a>	
	16B	no	vds_thresholds_hs_part1	<a href="#">Table 91</a>	
	16C	no	vds_thresholds_hs_part2	<a href="#">Table 92</a>	
	16D	no	vsrc_thresholds_hs_part1	<a href="#">Table 93</a>	
	16E	no	vsrc_thresholds_hs_part2	<a href="#">Table 94</a>	
	16F	no	vds_thresholds_ls_part1	<a href="#">Table 96</a>	
	170	no	vds_thresholds_ls_part2	<a href="#">Table 97</a>	
	171	no	hs_slewrate	<a href="#">Table 99</a>	

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1...000" / ext_sel_io	172	no	ls_slewrates_part1	<a href="#">Table 100</a>	IO configuration registers
	173	no	ls_slewrates_part2	<a href="#">Table 101</a>	
	174	no	offset_compensation12	<a href="#">Table 102</a>	
	175	no	offset_compensation34	<a href="#">Table 103</a>	
	176	no	offset_compensation56	<a href="#">Table 104</a>	
	177	no	adc12_result	<a href="#">Table 105</a>	
	178	no	adc34_result	<a href="#">Table 106</a>	
	179	no	adc56_result	<a href="#">Table 107</a>	
	17A	yes	current_filter12	<a href="#">Table 108</a>	
	17B	yes	current_filter34	<a href="#">Table 109</a>	
	17C	yes	current_filter5l5h	<a href="#">Table 110</a>	
	17D	yes	current_filter6l6h	<a href="#">Table 111</a>	
	17E	yes	current_filter5neg6neg	<a href="#">Table 112</a>	
	17F	no	boost_dac	<a href="#">Table 113</a>	
	180	yes	boost_dac_access	<a href="#">Table 114</a>	
	181	yes	boost_filter	<a href="#">Table 115</a>	
	182	no	vds7_dcdc_config	<a href="#">Table 116</a>	
	183	no	vds8_dcdc_config	<a href="#">Table 116</a>	
	184	-	batt_result	<a href="#">Table 118</a>	
	185	no	dac12_value	<a href="#">Table 119</a>	
	186	no	dac34_value	<a href="#">Table 120</a>	
	187	no	dac5l5h_value	<a href="#">Table 121</a>	
	188	no	dac5neg_value	<a href="#">Table 122</a>	
	189	no	dac6l6h_value	<a href="#">Table 123</a>	
	18A	no	dac6neg_value	<a href="#">Table 124</a>	
	18B	no	hs_bias_config	<a href="#">Table 125</a>	
	18C	no	ls_bias_config	<a href="#">Table 126</a>	
	18D	-	bootstrap_charged	<a href="#">Table 127</a>	
18E	-	bootstrap_timer	<a href="#">Table 128</a>		
18F	yes	hs1_ls_act	<a href="#">Table 131</a>		

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1...000" / ext_sel_io	190	yes	hs2_ls_act	<a href="#">Table 131</a>	IO configuration registers
	191	yes	hs3_ls_act	<a href="#">Table 131</a>	
	192	yes	hs4_ls_act	<a href="#">Table 131</a>	
	193	yes	hs5_ls_act	<a href="#">Table 131</a>	
	194	yes	hs6_ls_act	<a href="#">Table 131</a>	
	195	yes	hs7_ls_act	<a href="#">Table 131</a>	
	196	yes	dac_settling_time	<a href="#">Table 132</a>	
	197	no	oa_out1_config	<a href="#">Table 133</a>	
	198	no	oa_out2_config	<a href="#">Table 134</a>	
	199	no	oa_out3_config	<a href="#">Table 135</a>	
	19A		6 free addresses		
	...				
	19F				

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1000" / ext_sel_mcr	1A0	yes	Clock_Prescaler	<a href="#">Table 136</a>	Main configuration registers
	1A1	yes	Flags_Direction	<a href="#">Table 138</a>	
	1A2	yes	Flags_Polarity	<a href="#">Table 141</a>	
	1A3	yes	Flags_source	<a href="#">Table 145</a>	
	1A4	yes	Offset_compensation_prescaler	<a href="#">Table 146</a>	
	1A5	yes	Driver_Config_part1	<a href="#">Table 147</a>	
	1A6	yes	Driver_Config_part2	<a href="#">Table 148</a>	
	1A7	yes	PLL_config	<a href="#">Table 151</a>	
	1A8	yes	Backup_Clock_Status	<a href="#">Table 152</a>	
	1A9	yes	SPI_config	<a href="#">Table 153</a>	
	1AA	no	Trace_start	<a href="#">Table 154</a>	
	1AB	no	Trace_stop	<a href="#">Table 155</a>	
	1AC	no	Trace_config	<a href="#">Table 156</a>	
	1AD	yes	Device_lock	<a href="#">Table 157</a>	
	1AE	no	Reset_behavior	<a href="#">Table 158</a>	
	1AF	-	Device_unlock	<a href="#">Table 159</a>	
	1B0	-	Global_reset_code_part1	<a href="#">Table 160</a>	
	1B1	-	Global_reset_code_part2	<a href="#">Table 161</a>	
	1B2	no	Driver_Status	<a href="#">Table 162</a>	
	1B3	-	SPI_error_code	<a href="#">Table 163</a>	
	1B4	no	Interrupt_register_part1	<a href="#">Table 164</a>	
	1B5	no	Interrupt_register_part2	<a href="#">Table 165</a>	
	1B6	-	Device_Identifier	<a href="#">Table 167</a>	
	1B7	-	Reset_source	<a href="#">Table 168</a>	
	1B8		----		
	-		----		
	1BC		----		
	1BD	no	BIST_interface	<a href="#">Table 170</a>	
	1BE	no	----		
1BF	no	----			

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1...000" / ext_sel_diag	1C0	yes	ls1_diag_config1	<a href="#">Table 172</a>	diagnostics configuration registers
	1C1	yes	ls1_diag_config2	<a href="#">Table 173</a>	
	1C2	yes	ls1_output_config	<a href="#">Table 174</a>	
	1C3	yes	ls2_diag_config1	<a href="#">Table 172</a>	
	1C4	yes	ls2_diag_config2	<a href="#">Table 173</a>	
	1C5	yes	ls2_output_config	<a href="#">Table 174</a>	
	1C6	yes	ls3_diag_config1	<a href="#">Table 172</a>	
	1C7	yes	ls3_diag_config2	<a href="#">Table 173</a>	
	1C8	yes	ls3_output_config	<a href="#">Table 174</a>	
	1C9	yes	ls4_diag_config1	<a href="#">Table 172</a>	
	1CA	yes	ls4_diag_config2	<a href="#">Table 173</a>	
	1CB	yes	ls4_output_config	<a href="#">Table 174</a>	
	1CC	yes	ls5_diag_config1	<a href="#">Table 172</a>	
	1CD	yes	ls5_diag_config2	<a href="#">Table 173</a>	
	1CE	yes	ls5_output_config	<a href="#">Table 174</a>	
	1CF	yes	ls6_diag_config1	<a href="#">Table 172</a>	
	1D0	yes	ls6_diag_config2	<a href="#">Table 173</a>	
	1D1	yes	ls6_output_config	<a href="#">Table 174</a>	
	1D2	yes	ls7_diag_config1	<a href="#">Table 172</a>	
	1D3	yes	ls7_diag_config2	<a href="#">Table 173</a>	
	1D4	yes	ls7_output_config	<a href="#">Table 175</a>	
	1D5	yes	ls8_diag_config1	<a href="#">Table 172</a>	
	1D6	yes	ls8_diag_config2	<a href="#">Table 173</a>	
	1D7	yes	ls8_output_config	<a href="#">Table 175</a>	
	1D8	yes	hs1_diag_config1	<a href="#">Table 176</a>	
	1D9	yes	hs1_diag_config2	<a href="#">Table 177</a>	
	1DA	yes	hs1_output_config	<a href="#">Table 178</a>	
	1DB	yes	hs2_diag_config1	<a href="#">Table 176</a>	
	1DC	yes	hs2_diag_config2	<a href="#">Table 177</a>	
	1DD	yes	hs2_output_config	<a href="#">Table 178</a>	
	1DE	yes	hs3_diag_config1	<a href="#">Table 176</a>	
	1DF	yes	hs3_diag_config2	<a href="#">Table 177</a>	
1E0	yes	hs3_output_config	<a href="#">Table 178</a>		
1E1	yes	hs4_diag_config1	<a href="#">Table 176</a>		

**Table 55. MC33PT2000 address map**

Selection register [8:0]/chip select	Address (Hex)	Lock	Description	Table number	Area addressed
"1...000" / ext_sel_diag	1E3	yes	hs4_output_config	<a href="#">Table 178</a>	diagnostics configuration registers
	1E4	yes	hs5_diag_config1	<a href="#">Table 176</a>	
	1E5	yes	hs5_diag_config2	<a href="#">Table 177</a>	
	1E6	yes	hs5_output_config	<a href="#">Table 178</a>	
	1E7	yes	hs6_diag_config1	<a href="#">Table 176</a>	
	1E8	yes	hs6_diag_config2	<a href="#">Table 177</a>	
	1E9	yes	hs6_output_config	<a href="#">Table 178</a>	
	1EA	yes	hs7_diag_config1	<a href="#">Table 176</a>	
	1EB	yes	hs7_diag_config2	<a href="#">Table 177</a>	
	1EC	yes	hs7_output_config	<a href="#">Table 178</a>	
	1ED	-	err_uc0ch1_part1	<a href="#">Table 179</a>	
	1EE	-	err_uc0ch1_part2	<a href="#">Table 180</a>	
	1EF	-	err_uc0ch1_part3	<a href="#">Table 181</a>	
	1F0	-	err_uc1ch1_part1	<a href="#">Table 179</a>	
	1F1	-	err_uc1ch1_part2	<a href="#">Table 180</a>	
	1F2	-	err_uc1ch1_part3	<a href="#">Table 181</a>	
	1F3	-	err_uc0ch2_part1	<a href="#">Table 179</a>	
	1F4	-	err_uc0ch2_part2	<a href="#">Table 180</a>	
	1F5	-	err_uc0ch2_part3	<a href="#">Table 181</a>	
	1F6	-	err_uc1ch2_part1	<a href="#">Table 179</a>	
	1F7	-	err_uc1ch2_part2	<a href="#">Table 180</a>	
	1F8	-	err_uc1ch2_part3	<a href="#">Table 181</a>	
	1F9	-	err_uc0ch3_part1	<a href="#">Table 179</a>	
	1FA	-	err_uc0ch3_part2	<a href="#">Table 180</a>	
	1FB	-	err_uc0ch3_part3	<a href="#">Table 181</a>	
	1FC	-	err_uc1ch3_part1	<a href="#">Table 179</a>	
	1FD	-	err_uc1ch3_part2	<a href="#">Table 180</a>	
	1FE	-	err_uc1ch3_part3	<a href="#">Table 181</a>	
	1FF	yes	diagnostics_option	<a href="#">Table 182</a>	
200		511 free addresses			
-					
3FE					
N/A	3FF	no	Selection_register	<a href="#">Table 56</a>	Selection Register



## 8.3.1 Selection register (3FFh)

The selection register is a 4-bit register aimed to select, before starting the read/write operations toward a given address, which internal code RAM is accessed or to select all the other addresses (including the 3 data RAMs and all the registers).

**Table 56. selection\_register (3FFh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved							comm. page sel	Reserved					CRAM_ ch3_sel	CRAM_ ch2_sel	CRAM_ ch1_sel
R/W	-							r/w	-					r/w	r/w	r/w
Lock	-							no	-					no	no	no
Reset	0000000							0	00000					0	0	0

Table 57 details the meaning of the 4 bits in this register. Not all possible values are allowed for this register.

**Table 57. Selection register**

Selection register comm. page sel	Selection register CRAM chx	Element addressed
'0'	"000"	Nothing selected. Further SPI operation, except for the one concerning this register is ignored.
'0'	"001"	Channel 1 Code RAM selected
'0'	"010"	Channel 2 Code RAM selected
'0'	"100"	Channel 3 Code RAM selected
'0'	"011"	Write operation affects the Code RAM of channel 1 and 2. Read operation is not possible.
'0'	"101"	Write operation affects the Code RAM of channel 1 and 3. Read operation is not possible.
'0'	"110"	Write operation affects the Code RAM of channel 2 and 3. Read operation is not possible.
'0'	"111"	Write operation affects the all three channel's Code RAM. Read operation is not possible.
'1'	d.c.	Common page selected. The lower three bits are ignored. "10000000" is written to the register.

This selection register is accessible at address 3FFh and is the only register accessed anyway from the SPI, whatever the value of the selection register. This means address 3FFh can only be used for this purpose, even if any of the code RAMs is selected, so each 1024x16 code RAM in the MC33PT2000 can actually only store 1023 data. Note that 2 or 3 code RAMs can be written in parallel during normal mode.

## 8.3.2 Configuration register

### 8.3.2.1 Flash\_enable register

Table 58. Flash\_enable (100h, 120h, 140h)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved									checksum_disable	flash_enable	pre_flash_enable	en_dual_uc	Reserved	chksum_irq_en	chksum_failure
R/W	-									r/w	r	r/w	r/w	-	r/w	r
Lock	-									yes, by pre flash enable	-	yes, by itself	yes, by pre flash enable	-	yes	-
Reset	000000000									0	0	0	0	0	0	0

This is a 6 bit configuration register which includes the following parameters:

- Checksum\_disable. If set, this bit disables the effects of a failed checksum, so microcore execution is not stopped
- Pre\_flash\_enable. This bit “freezes” the CRAM so the micro-controller cannot further modify the configuration code unless a specific unlock code is written into register unlock\_reg. It enables the signature\_unit (See Signature unit on page 72)
- Flash\_enable. This bit enables the microcores. It can only be set by the signature\_unit after a successful checksum calculation
- En\_dual\_microcore. This bit is used to enable the dual microcore mode. Note that when using dual microcore ck\_per (refer to Table 136, Clock\_Prescaler (1A0h)) set to lower than three, there are some limitations regarding C/DRAM access.
- Checksum\_irq\_en. If this bit is '1', an interrupt on the\_irq\_device pin is done in case a CRAM corruption detected
- Checksum\_failure. This bit sets to '1' when a mismatch is found between the calculated checksum and the checksum code stored in the appropriate registers (refer to Table 66, checksum\_h (108h, 128h, 148h) and Table 67, checksum\_l (109h, 129h, 149h)). This bit sets when a checksum calculation fails, even if the checksum is disabled. This bit resets each time the pre\_flash\_enable bit sets to '1' to lock the memory.

### 8.3.2.2 Control register microcore0

Table 59. Ctrl\_reg\_uc0 (101h, 121h, 141h)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Control_register_Shared								Control_register							
R/W	configurable r or r/w								r/w							
Lock	no								no							
Reset	00000000								00000000							

- Control\_register: these 8 bits can be used to control the execution of the micro-program of uc0, providing control bits which can be read by the micro-program. For instance one bit could be used to enable/disable recharge pulses on the channel or to re-enable the actuation after an error condition has been detected.
- Control\_register\_shared: according to a configuration bit stored in the “control\_register\_split” register (refer to Table 77, Dac\_rtx\_cr\_config (112h, 132h, 152h)), these 8 bits can be used either as control (like the other 8 bits) or like status (like the status register, refer to Table 63, status\_reg\_uc0 (105h, 125h, 145h) and Table 64, status\_reg\_uc1 (106h, 126h, 146h)). In this case, they can only be read through the SPI, while they can be set by the “set control register bit” instruction.

### 8.3.2.3 Control register microcore1

**Table 60. ctrl\_reg\_uc1 (102h, 122h, 142h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Control_register_Shared								Control_register							
R/W	configurable r or r/w								r/w							
Lock	no								no							
Reset	00000000								00000000							

- Control register: these 8 bits can be used to control the execution of the micro-program of uc0, providing control bits which can be read by the micro-program itself. For instance one bit could be used to enable/disable recharge pulses on the channel or to re-enable the actuation after an error condition has been detected.
- Control register shared: according to a configuration bit stored in the “control register split” register (refer to [Table 77, Dac\\_rxtx\\_cr\\_config \(112h, 132h, 152h\)](#)), these 8 bits can be used either as control (like the other 8 bits) or like status (like the status register, refer to [Table 63, status\\_reg\\_uc0 \(105h, 125h, 145h\)](#) and [Table 64, status\\_reg\\_uc1 \(106h, 126h, 146h\)](#)). In this case they can only be read through the SPI, while they can be set by the “set control register bit” instruction.

### 8.3.2.4 Start configuration register

These are two configuration registers where it is possible to configure the sensitivity of each microcore to the start signals. It is also possible to enable a smart start mode for each microcore (reference [Programming Guide and Instruction Set](#)).

**Table 61. start\_config\_reg\_Part1 (103h, 123h, 143h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	start8_sens_uc1	start7_sens_uc1	start6_sens_uc1	start5_sens_uc1	start4_sens_uc1	start3_sens_uc1	start2_sens_uc1	start1_sens_uc1	start8_sens_uc0	start7_sens_uc0	start6_sens_uc0	start5_sens_uc0	start4_sens_uc0	start3_sens_uc0	start2_sens_uc0	start1_sens_uc0
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 62. start\_config\_reg\_Part2 (104h, 124h, 144h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	reserved														smart_start_uc1	smart_start_uc0
R/W	-														r/w	r/w
Lock	-														yes	yes
Reset	00000000														0	0

- start1\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start1, '0' otherwise
- start2\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start2, '0' otherwise
- start3\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start3, '0' otherwise
- start4\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start4, '0' otherwise
- start5\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start5, '0' otherwise
- start6\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start6, '0' otherwise
- start7\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start7, '0' otherwise
- start8\_sens\_uc0: This bit is '1' if the uc0 is sensitive to start8, '0' otherwise
- start1\_sens\_uc1: This bit is '1' if the uc1 is sensitive to start1, '0' otherwise
- start2\_sens\_uc1: This bit is '1' if the uc1 is sensitive to start2, '0' otherwise

- start3\_sens\_uc1: This bit is '1' if the uc1\_is sensitive to start3, '0' otherwise
- start4\_sens\_uc1: This bit is '1' if the uc1\_is sensitive to start4, '0' otherwise
- start5\_sens\_uc1: This bit is '1' if the uc1\_is sensitive to start5, '0' otherwise
- start6\_sens\_uc1: This bit is '1' if the uc1\_is sensitive to start6, '0' otherwise
- start7\_sens\_uc1: This bit is '1' if the uc1\_is sensitive to start7, '0' otherwise
- start8\_sens\_uc1: This bit is '1' if the uc1\_is sensitive to start8, '0' otherwise
- smart\_start\_uc0: This bit is '1' if the smart start mode is enabled for uc0, '0' otherwise (reference [Programming Guide and Instruction Set](#)).
- smart\_start\_uc1: This bit is '1' if the smart start mode is enabled for uc1, '0' otherwise (reference [Programming Guide and Instruction Set](#)).

### 8.3.2.5 Status register microcore0

This 16-bit register is a read-only register and only provides information about the uc0\_status to the external microcontroller. The register can be used to exchange application dependent information (status bits, for instance regarding the execution phase of the micro-program, diagnostics results) between the microcore and the main microcontroller according to the way the micro-program is designed. The registers can be configured so they reset after a SPI read operation to the register (see [Table 158, Reset\\_Behavior \(1AEh\)](#)).

**Table 63. status\_reg\_uc0 (105h, 125h, 145h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	status_register															
R/W	r															
Lock	-															
Reset on read	configurable															
Reset	0000000000000000															

### 8.3.2.6 Status register microcore1

This 16-bit register is a read-only register and only provides information about the uc1\_status to the external microcontroller. The register can be used to exchange application dependent information (status bits, for instance regarding the execution phase of the micro-program) between the microcore and the main microcontroller according to the way the micro-program is designed. The registers can be configured so they reset after a SPI read operation to the register (see [Table 158, Reset\\_Behavior \(1AEh\)](#)).

**Table 64. status\_reg\_uc1 (106h, 126h, 146h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	status_register															
R/W	r															
Lock	-															
Reset on read	configurable															
Reset	0000000000000000															

### 8.3.2.7 Code\_width register

This 10-bit register provides the length of the section of the CRAM used to store the code. This information has two uses:

- It is used by the SPI interface to determine the length of the special burst transfer used for CRAM initialization (refer to [See SPI protocol on page 77](#)).
- The signature unit computes the checksum only of the used part of the CRAM. This signature unit only works if the code with is bigger than 3 lines, if it is not the case signature unit has to be disabled (refer to [Table 58, Flash\\_enable \(100h, 120h, 140h\)](#))

This allows the application not to write all the CRAM, but only the part which is really used.

**Table 65. code\_width (107h, 127h, 147h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved						code width									
R/W	-						r/w									
Lock	-						yes									
Reset	000000						0000000000									

### 8.3.2.8 Checksum high register

This 16-bit register contains the 16 MSBs of the checksum of the code contained in the CRAM. The signature\_unit (refer to [See Signature unit on page 72](#)) compares the result of its computation to this register and checksum\_l.

**Table 66. checksum\_h (108h, 128h, 148h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	checksum_high															
R/W	r/w															
Lock	yes															
Reset	0000000000000000															

### 8.3.2.9 Checksum low register

This 16-bit register contains the 16 LSBs of the checksum of the code contained in the CRAM. The signature\_unit (refer to section [See Signature unit on page 72](#)) compares the result of its computation to checksum\_h and this register.

**Table 67. checksum\_l (109h, 129h, 149h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Checksum_low															
R/W	r/w															
Lock	yes															
Reset	0000000000000000															

### 8.3.2.10 Microcore0 entry point address register

This 10-bit register contains the CRAM address of the first instruction to be executed by microcontroller0.

**Table 68. uc0\_entry\_point (10Ah, 12Ah, 14Ah)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved						entry_point_address									
R/W	-						r/w									
Lock	-						yes									
Reset	000000						0000001000									

### 8.3.2.11 Microcore1 entry point address register

This 10-bit register contains the CRAM address of the first instruction to be executed by uc1. This is done to allow the two microcores to execute completely independent microcodes (when the two entry point differ), while still having the capability to execute the same program in case the two entry points coincide.

**Table 69. uc1\_entry\_point (10Bh, 12Bh, 14Bh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved						entry_point_address									
R/W	-						r/w									
Lock	-						yes									
Reset	000000						0000001000									

### 8.3.2.12 Diagnostics interrupt routine address register

**Table 70. Diag\_routine\_addr (10Ch, 12Ch, 14Ch)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved				diagnostics_routine_address_uc1						diagnostics_routine_address_uc0					
R/W	-				r/w						r/w					
Lock	-				yes						yes					
Reset	0000				000000						000000					

- `diagnostics_routine_address_uc0`. The complete address is "0000" & "diagnostics routine address uc0": this is the CRAM address of the first instruction of the interrupt routine to be executed by uc0 when an automatic diagnostics exception is raised.
- `diagnostics_routine_address_uc1`. The complete address is "0000" & "diagnostics routine address uc1": this is the CRAM address of the first instruction of the interrupt routine to be executed by uc1 when an automatic diagnostics exception is raised.

### 8.3.2.13 Driver disabled interrupt routine address register

**Table 71. Driver\_disable\_routine\_addr (10Dh, 12Dh, 14Dh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved				driver_disable_routine_address_uc1						driver_disable_routine_address_uc0					
R/W	-				r/w						r/w					
Lock	-				yes						yes					
Reset	0000				000000						000000					

- driver\_disable\_routine\_address\_uc0. The complete address is “0000” & “driver disable routine address uc0”: This is the CRAM address of the first instruction of the interrupt routine to be executed by uc0\_when a disabled driver or cksys missing exception is raised.
- driver\_disable\_routine\_address\_uc1. The complete address is “0000” & “driver disable routine address uc1”: This is the CRAM address of the first instruction of the interrupt routine to be executed by uc1\_when a disabled driver or cksys missing exception is raised.

The following events can trigger this interrupt (all configurable):

- DrvEn pin going low
- UV\_VCCP
- UV\_VCC5
- UV\_VBOOST
- cksys missing
- Overtemperature

### 8.3.2.14 Software interrupt routine address register

**Table 72. Sw\_interrupt\_routine\_addr (10Eh, 12Eh, 14Eh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	sw_irq_falling_edge_start_uc1	sw_irq_rising_edge_start_uc1	sw_irq_falling_edge_start_uc0	sw_irq_rising_edge_start_uc0	software_interrupt_routine_address_uc1						software_interrupt_routine_address_uc0					
R/W	r/w	r/w	r/w	r/w	r/w						r/w					
Lock	yes	yes	yes	yes	yes						yes					
Reset	0	0	0	0	000000						000000					

- software\_interrupt\_routine\_address\_uc0. The complete address is “0000” & “software interrupt routine address uc0”: This is the CRAM address of the first instruction of the interrupt routine to be executed by uc0\_when a software interrupt is requested.
- software\_interrupt\_routine\_address\_uc1. The complete address is “0000” & “software interrupt routine address uc1”: This is the CRAM address of the first instruction of the interrupt routine to be executed by uc1\_when a software interrupt is requested.
- sw\_irq\_rising\_edge\_start\_uc0. When this bit is set to '1', the software interrupt 0 is generated towards microcore 0 if a rising edge is detected on the gen\_start signal. When set to '0', no software interrupt is required.
- sw\_irq\_falling\_edge\_start\_uc0. When this bit is set to '1', the software interrupt 0 is generated towards microcore 0 if a falling edge is detected on the gen\_start signal. When set to '0', no software interrupt is required.
- sw\_irq\_rising\_edge\_start\_uc1. When this bit is set to '1', the software interrupt 1 is generated towards microcore 1 if a rising edge is detected on the gen\_start signal. When set to '0', no software interrupt is required.
- sw\_irq\_falling\_edge\_start\_uc1. When this bit is set to '1', the software interrupt 1 is generated towards microcore 1 if a falling edge is detected on the gen\_start signal. When set to '0', no software interrupt is required.

### 8.3.2.15 Microcore0 interrupt status register

This 13-bit register stores the information about the interrupt currently being served by uc0. If no interrupt is being served, this register is cleared.

**Table 73. uc0\_irq\_status (10Fh, 12Fh, 14Fh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		interrupt_routine_in_progress	irq_source			iret_address									
R/W	-		r	r			r									
Lock	-		no	no			no									
Reset	00		0	000			1111111111									

- Interrupt\_routine\_in\_progress: '1' when an interrupt is being served.
- Irq\_source:
  - "000": serving start rising edge interrupt
  - "001": serving driver disable interrupt request
  - "010": serving automatic diagnostics interrupt request
  - "011": serving start falling edge interrupt
  - "100": serving software interrupt request 0
  - "101": serving software interrupt request 1
  - "110": serving software interrupt request 2
  - "111": serving software interrupt request 3
- Iret\_address: the value of the return address after the interrupt is served.

The return address after an interrupt is always the address where the code execution would continue if no interrupt had occurred. For wait and conditional jump instructions, the address is defined considering the status of the feedback at the moment the interrupt request took place.

### 8.3.2.16 Microcore1 interrupt status register

This 13-bit register stores the information about the interrupt currently being served by uc1. If no interrupt is being served, this register is cleared.

**Table 74. uc1\_irq\_status (110h, 130h, 150h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		interrupt_routine_in_progress	irq_source			iret_address									
R/W	-		r	r			r									
Lock	-		no	no			no									
Reset	00		0	000			1111111111									

- Interrupt\_routine\_in\_progress: '1' when an interrupt is being served.
- Irq\_source:
  - "000": serving start rising edge interrupt
  - "001": serving driver disable interrupt request
  - "010": serving automatic diagnostics interrupt request
  - "011": serving start falling edge interrupt
  - "100": serving software interrupt request 0
  - "101": serving software interrupt request 1
  - "110": serving software interrupt request 2
  - "111": serving software interrupt request 3
- Iret\_address: the value of the return address after the interrupt is served.



The return address after an interrupt is always the address where the code execution would continue if no interrupt had occurred. For wait and conditional jump instructions, the address is defined considering the status of the feedback at the moment the interrupt request took place.

### 8.3.2.17 Counter 3 and 4 prescaler register

**Table 75. Counter\_34\_prescaler (111h, 131h, 151h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	counter_4_per_uc1				counter_3_per_uc1				counter_4_per_uc0				counter_3_per_uc0			
R/W	r/w				r/w				r/w				r/w			
Lock	yes				yes				yes				yes			
Reset	0000				0000				0000				0000			

The counter 3 and 4 of both microcores uses a clock whose period can be programmed to be a multiple of the ck period. Note that the actual ratio is according to [Table 76, Counter prescaler](#).

Example:

cksys set to 24 MHz (default) with ck\_per = 3, resulting with ck at 6.0 MHz

counter\_4\_per\_uc1 = 0001b -> prescaler of 3

counter4\_freq = ck / counter\_Y\_per\_ucx = 6.0 MHz / 3 = 2.0 MHz

**Table 76. Counter prescaler**

counter_3/4_per_ucx	Prescaler
0000	1
0001	2
0010	3
0011	4
0100	5
0101	6
0110	7
0111	8
1000	9
1001	10
1010	11
1011	12
1100	14
1101	16
1110	32
1111	64

### 8.3.2.18 DAC Rxtx configuration register

Each microcore can only access four out of six current measurement channel DACs via the internal address map. In addition, either the two special DACs of current measurement channel five or six can be used. The two DACs linked to their own channel can always be addressed via sssc and ossc. In addition, the two DACs of one other channel can be addressed via the ssoc and osoc parameter. It can be decided by register bits or microcode instruction sloddac for each microcore, which of the two other channels (next channel or previous channel), DACs are accessible and if DACs H and NEG of channel five or six are accessible.

Only one microcore can be addressed at the same time in the channel communication register (rxtx register in the memory map). This can be decided by register bits or microcode instruction sl56dac.

**Table 77. Dac\_rxtx\_cr\_config (112h, 132h, 152h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		rxtx_link_sel_uc1			rxtx_link_sel_uc0			Reserved		dac56_sel_uc1	dac56_sel_uc0	oc_dac_sel_uc1	oc_dac_sel_uc0	CR_shared_uc1	CR_shared_uc0
R/W	-		r/w			r/w			-		r/w	r/w	r/w	r/w	r/w	r/w
Lock	-		no			no			-		no	no	no	no	yes	yes
Reset	00		000			000			00		0	0	0	0	0	0

- CR\_shared\_uc0: if set to '0', all 16 of the bits of the control register uc0\_ are used as control bits. If set to '1', the 8 MSBs of the control register (control register shared) are used as status bits.
- CR\_shared\_uc1: if set to '0', all 16 of the bits of the control register uc1\_ are used as control bits. If set to '1', the 8 MSBs of the control register (control register shared) are used as status bits.
- oc\_dac\_sel\_uc0: selects the other dac for microcore 0.
  - '0', channel 1: ssoc refers to dac3, osoc refers to dac4 (next channel = 2)
  - '0', channel 2: ssoc refers to dac5, osoc refers to dac6 (next channel = 3)
  - '0', channel 3: ssoc refers to dac1, osoc refers to dac2 (next channel = 1)
  - '1', channel 1: ssoc refers to dac5, osoc refers to dac6 (prev. channel = 3)
  - '1', channel 2: ssoc refers to dac1, osoc refers to dac2 (prev. channel = 1)
  - '1', channel 3: ssoc refers to dac3, osoc refers to dac4 (prev. channel = 2)
- oc\_dac\_sel\_uc1: selects the other dac for microcore 1.
  - '0', channel 1: ssoc refers to dac4, osoc refers to dac3 (next channel = 2)
  - '0', channel 2: ssoc refers to dac6, osoc refers to dac5 (next channel = 3)
  - '0', channel 3: ssoc refers to dac2, osoc refers to dac1 (next channel = 1)
  - '1', channel 1: ssoc refers to dac6, osoc refers to dac5 (prev. channel = 3)
  - '1', channel 2: ssoc refers to dac2, osoc refers to dac1 (prev. channel = 1)
  - '1', channel 3: ssoc refers to dac4, osoc refers to dac3 (prev. channel = 2)
- dac56\_sel\_ucX: selects the dac5/6 for microcore X.
  - '0': dac56h56n refers to dac5
  - '1': dac56h56n refers to dac6

**Table 78. Other dac configuration**

Microcore	dac sssc	dac ossc	oc_dac_sel_ucX = '0' (next channel)		oc_dac_sel_ucX = '1' (previous channel)	
			dac ssoc	dac osoc	dac ssoc	dac osoc
uc0, channel 1	dac1	dac2	dac3	dac4	dac5	dac6
uc1, channel 1	dac2	dac1	dac4	dac3	dac6	dac5
uc0, channel 2	dac3	dac4	dac5	dac6	dac1	dac2
uc1, channel 2	dac4	dac3	dac6	dac5	dac2	dac1
uc0, channel 3	dac5	dac6	dac1	dac2	dac3	dac4
uc1, channel 3	dac6	dac5	dac2	dac1	dac4	dac3

- `rxtx_link_sel_ucX`: selects the target for the channel communication register (`rxtx`) in the internal memory map for microcore X (can also be done using the 'stcrt instruction').
  - '0': same uc same channel (`sssc`)
  - '1': other uc same channel (`ossc`)
  - '2': same uc next channel (`ssnc`)
  - '3': other uc next channel (`osnc`)
  - '4': sum of highest 4 bits of all `rxtx` registers (`sumh`)
  - '5': sum of second highest 4 bits of all `rxtx` registers (`suml`)
  - '6': same uc previous channel (`sspc`)
  - '7': other uc previous channel (`ospc`)

**Table 79. Rxtx register link configuration**

Microcore	sssc	ossc	ssnc	osnc	sspc	ospc
uc0, channel 1	uc0Ch1	uc1Ch1	uc0Ch2	uc1Ch2	uc0Ch3	uc1Ch3
uc1, channel 1	uc1Ch1	uc0Ch1	uc1Ch2	uc0Ch2	uc1Ch3	uc0Ch3
uc0, channel 2	uc0Ch2	uc1Ch2	uc0Ch3	uc1Ch3	uc0Ch1	uc1Ch1
uc1, channel 2	uc1Ch2	uc0Ch2	uc1Ch3	uc0Ch3	uc1Ch1	uc0Ch1
uc0, channel 3	uc0Ch3	uc1Ch3	uc0Ch1	uc1Ch1	uc0Ch2	uc1Ch2
uc1, channel 3	uc1Ch3	uc0Ch3	uc1Ch1	uc0Ch1	uc1Ch2	uc0Ch2

### 8.3.2.19 Unlock word register

The actuation channel execution can be stopped by writing the unlock code at this SPI address. The unlock code is "1011111011101111" (binary) or "BEEF" (hexadecimal). As this is not a register, no SPI read operations can be performed at this address.

**Table 80. Unlock\_word (113h, 133h, 153h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Unlock_word															
R/W	w															
Lock	no															
Reset	-															

## 8.3.3 IO configuration registers

### 8.3.3.1 Feedback microcore sensitivities registers

These 12 registers (two for each microcore). Select the feedback to which each microcore is sensitive (e.g. configures if ucX chY is sensitive to  $V_{DS}$  errors on HS1).

**Table 81. Fbk\_sens\_ucxchy\_part1 (154h, 156h, 158h, 15Ah, 15Ch, 15Eh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved	Hs7_Vsrc_sens	Hs6_Vsrc_sens	Hs5_Vsrc_sens	Hs4_Vsrc_sens	Hs3_Vsrc_sens	Hs2_Vsrc_sens	Hs1_Vsrc_sens	Reserved	Hs7_Vsd_sens	Hs6_Vsd_sens	Hs5_Vsd_sens	Hs4_Vsd_sens	Hs3_Vsd_sens	Hs2_Vsd_sens	Hs1_Vsd_sens
R/W	-	r/w	r/w	r/w	r/w	r/w	r/w	r/w	-	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	-	yes	yes	yes	yes	yes	yes	yes	-	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 82. Fbk\_sens\_ucxchy\_part2 (155h, 157h, 159h, 15Bh, 15Dh, 15Fh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								Ls8_Vds_sens	Ls7_Vds_sens	Ls6_Vds_sens	Ls5_Vds_sens	Ls4_Vds_sens	Ls3_Vds_sens	Ls2_Vds_sens	Ls1_Vds_sens
R/W	-								r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	-								yes	yes	yes	yes	yes	yes	yes	yes
Reset	00000000								0	0	0	0	0	0	0	0

### 8.3.3.2 Microcores output access registers

These six registers are designed to provide access rights to manage the control signals (output\_command,  $V_{DS}$  threshold,  $V_{SRC}$  threshold, fw auto, en\_halt\_x) of each output block to the required microcores. Each bit controls the access from one microcore to manage the control signals of an output: if the value is set to 1, the microcore can drive the control signals (output\_command,  $V_{DS}$  threshold,  $V_{SRC}$  threshold, fw auto, en\_halt\_x), otherwise access is denied.

**Table 83. Out\_acc\_ucx\_chy (160h, 161h, 162h, 163h, 164h, 165h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Acc_ucX_chY_ls8	Acc_ucX_chY_ls7	Acc_ucX_chY_ls6	Acc_ucX_chY_ls5	Acc_ucX_chY_ls4	Acc_ucX_chY_ls3	Acc_ucX_chY_ls2	Acc_ucX_chY_ls1	Reserved	Acc_ucX_chY_hs7	Acc_ucX_chY_hs6	Acc_ucX_chY_hs5	Acc_ucX_chY_hs4	Acc_ucX_chY_hs3	Acc_ucX_chY_hs2	Acc_ucX_chY_hs1
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	-	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	-	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The requests coming from the microcores are not continuous; instead the microcores perform a request each time a change to a control signal (output\_command,  $V_{DS}$  threshold,  $V_{SRC}$  threshold, fw auto, en\_halt\_x) is required. If multiple microcores have access to the same output block, as a shared resource, this block is able to handle the collision: if more than one microcore wants to change one of the control signals in the same ck cycle, priorities are used as defined in [Table 84, Out\\_acc\\_ucxchy collision handling](#).

If one of the microcores which have access to a pre-driver not locked, the other microcore can switch on the pre-driver for only one ck cycle maximum. After one ck cycle the output is switched off again, because this request comes from the disabled microcore. This is a safety feature of the device. If requests to change a control signal are received from different microcores (assuming both have access rights) in different ck cycles, all the requested changes are applied in sequence.

**Table 84. Out\_acc\_ucxchy collision handling**

Microcore	Priority
uc0-ch1	1 (highest)
uc1-ch1	2
uc0-ch2	3
uc1-ch2	4
uc0-ch3	5
uc1-ch3	6 (lowest)

### 8.3.3.3 Microcore current sense access registers

This register is designed to provide access rights to manage the control signals (DAC value, Opamp gain, Ofscmp request) of each current measure block to the required microcores. Each bit controls the access from one microcore to manage the control signals of a current measure block: if the value is set to 1, the microcore can drive those input signals, otherwise access is denied.

The pins of the current measurement channel 4 are multiplexed with digital IO pins. The current measurement function is activated via the flags\_source (refer to Table 145, Flag\_source (1A3h)) and flags\_direction register (refer to Table 138, Flag\_direction (1A1h)). Current measure blocks 5 and 6 are different, as it requires 3 DAC values instead of just one as in the others. The “acc ucX chY curr 5/6L” bits grants access to all the control signals (DAC value 5/6L, ofscmp request, opamp gain) save for the DAC values 5/6H and 5/6Neg, which are controlled by the “acc ucX chY curr 5/6H 5/6Neg” bits.

**Table 85. Cur\_block\_access\_part1 channel1 (166h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	acc_uc1_ch1_curr6H_6Neg	acc_uc1_ch1_curr6L	acc_uc1_ch1_curr5H_5Neg	acc_uc1_ch1_curr5L	acc_uc1_ch1_curr4	acc_uc1_ch1_curr3	acc_uc1_ch1_curr2	acc_uc1_ch1_curr1	acc_uc0_ch1_curr6H_6Neg	acc_uc0_ch1_curr6L	acc_uc0_ch1_curr5H_5Neg	acc_uc0_ch1_curr5L	acc_uc0_ch1_curr4	acc_uc0_ch1_curr3	acc_uc0_ch1_curr2	acc_uc0_ch1_curr1
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 86. Cur\_block\_access\_part2 channel2 (167h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	acc_uc1_ch2_curr6H_6Neg	acc_uc1_ch2_curr6L	acc_uc1_ch2_curr5H_5Neg	acc_uc1_ch2_curr5L	acc_uc1_ch2_curr4	acc_uc1_ch2_curr3	acc_uc1_ch2_curr2	acc_uc1_ch2_curr1	acc_uc0_ch2_curr6H_6Neg	acc_uc0_ch2_curr6L	acc_uc0_ch2_curr5H_5Neg	acc_uc0_ch2_curr5L	acc_uc0_ch2_curr4	acc_uc0_ch2_curr3	acc_uc0_ch2_curr2	acc_uc0_ch2_curr1
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 87. Cur\_block\_access\_part3 channel3 (168h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	acc_uc1_ch3_curr6H_6Neg	acc_uc1_ch3_curr6L	acc_uc1_ch3_curr5H_5Neg	acc_uc1_ch3_curr5L	acc_uc1_ch3_curr4	acc_uc1_ch3_curr3	acc_uc1_ch3_curr2	acc_uc1_ch3_curr1	acc_uc0_ch3_curr6H_6Neg	acc_uc0_ch3_curr6L	acc_uc0_ch3_curr5H_5Neg	acc_uc0_ch3_curr5L	acc_uc0_ch3_curr4	acc_uc0_ch3_curr3	acc_uc0_ch3_curr2	acc_uc0_ch3_curr1
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The requests coming from the microcores are not continuous, but they perform a request each time a change to a control signal (Dac, opamp gain, ofscmp request) is required. If multiple microcores have access to the same current measure block, as a shared resource, this block is able to handle the collision: if more than one microcore wants to change one of the control signals in the same moment, only one of the requested values is applied. If requests to change a control signal are received from different microcores (assuming both have access rights) in different ck cycles, all the requested changes are applied in sequence.

### 8.3.3.4 Freewheeling link register

Due to some HS pre-driver having two possible fw slaves, there is the need to select which of the links is affected by the stfw instruction (refer to programming guide for more details on this instruction). This configuration is done in the fw\_link register.

**Table 88. Fw\_link (169h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved	Flag3_fw_link	Flag2_fw_link	Flag1_fw_link	Flag0_fw_link	Reserved		Hs7_fw_link	Reserved	Ls7_fw_link	Ls6_fw_link	Ls5_fw_link	Ls4_fw_link	Ls3_fw_link	Ls2_fw_link	Ls1_fw_link
R/W	-	r/w	r/w	r/w	r/w	-	-	r/w	-	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	-	yes	yes	yes	yes	-	-	yes	-	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	00		0	0	0	0	0	0	0	0	0

- Ls1\_fw\_link: if set, the Ls1 is driven as a fw relative to Hs1, when activated via stfw instruction.
- Ls2\_fw\_link: if set, the Ls2 is driven as a fw relative to Hs2, when activated via stfw instruction.
- Ls3\_fw\_link: if set, the Ls3 is driven as a fw relative to Hs3, when activated via stfw instruction.
- Ls4\_fw\_link: if set, the Ls4 is driven as a fw relative to Hs4, when activated via stfw instruction.
- Ls5\_fw\_link: if set, the Ls5 is driven as a fw relative to Hs5, when activated via stfw instruction.
- Ls6\_fw\_link: if set, the Ls6 is driven as a fw relative to Hs6, when activated via stfw instruction.
- Ls7\_fw\_link: if set, the Ls7 is driven as a fw relative to Hs7, when activated via stfw instruction.
- Hs7\_fw\_link: if set, the Hs7 is driven as a fw relative to Hs1, when activated via stfw instruction.
- Flag0\_fw\_link: if set, the Flag0 is driven as a fw relative to Hs4, when activated via stfw instruction.
- Flag1\_fw\_link: if set, the Flag1 is driven as a fw relative to Hs5, when activated via stfw instruction.
- Flag2\_fw\_link: if set, the Flag2 is driven as a fw relative to Hs6, when activated via stfw instruction.
- Flag3\_fw\_link: if set, the Flag3 is driven as a fw relative to Hs7, when activated via stfw instruction.

**Table 89. Freewheeling link register**

Freewheeling pre-driver output	Related pre-driver high-side
LS1	HS1
LS2	HS2
LS3	HS3
LS4	HS4
LS5	HS5
LS6	HS6
LS7	HS7
Flag0	HS4
Flag1	HS5
Flag2	HS6
Flag3	HS7

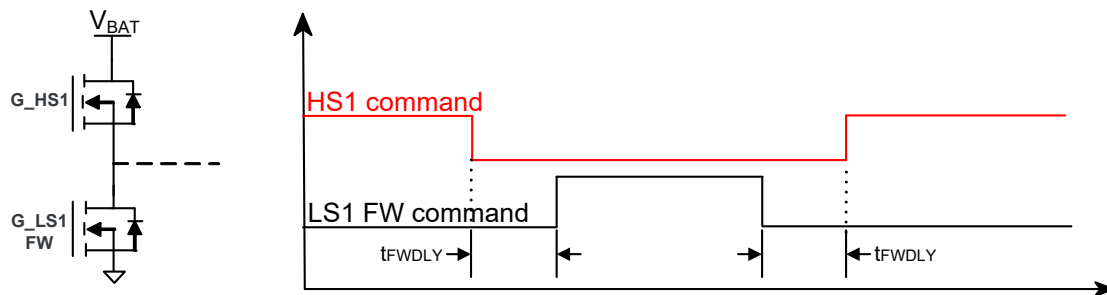
### 8.3.3.5 Freewheeling external request configuration register

After the freewheeling configuration (see [Table 89, Freewheeling link register](#)) it is possible to activate automatic freewheeling by writing the corresponding bit of this register even when the microcode is not running. This can also be enabled by the stfw instruction.

**Table 90. Fw\_external\_request (16ah)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Name	Reserved										Hs7_fw_en	Hs6_fw_en	Hs5_fw_en	Hs4_fw_en	Hs3_fw_en	Hs2_fw_en	Hs1_fw_en
R/W	-										r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	-										yes	yes	yes	yes	yes	yes	yes
Reset	000000000										0	0	0	0	0	0	0

- Hs1\_fw\_en: if set, the fw relative to Hs1 is activated, otherwise the status is defined by the microcore request (see stfw instruction).
- Hs2\_fw\_en: if set, the fw relative to Hs2 is activated, otherwise the status is defined by the microcore request (see stfw instruction).
- Hs3\_fw\_en: if set, the fw relative to Hs3 is activated, otherwise the status is defined by the microcore request (see stfw instruction).
- Hs4\_fw\_en: if set, the fw relative to Hs4 is activated, otherwise the status is defined by the microcore request (see stfw instruction).
- Hs5\_fw\_en: if set, the fw relative to Hs5 is activated, otherwise the status is defined by the microcore request (see stfw instruction).
- Hs6\_fw\_en: if set, the fw relative to Hs6 is activated, otherwise the status is defined by the microcore request (see stfw instruction).
- Hs7\_fw\_en: if set, the fw relative to Hs7 is activated, otherwise the status is defined by the microcore request (see stfw instruction).



**Figure 32. Automatic freewheeling example**

### 8.3.3.6 V<sub>DS</sub> and V<sub>SRC</sub> threshold selection

Each comparator threshold is expressed on 4 bits. These registers can be written through the SPI. The microcores can change the value of each field at runtime, provided they have the access right to control the related output (refer to [Table 83, Out\\_acc\\_ucx\\_chy \(160h, 161h, 162h, 163h, 164h, 165h\)](#)).

**Table 91. Vds\_threshold\_hs\_Part1 (16Bh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Vds_thr_Hs4				Vds_thr_Hs3				Vds_thr_Hs2				Vds_thr_Hs1			
R/W	r/w				r/w				r/w				r/w			
Lock	no				no				no				no			
Reset	0000				0000				0000				0000			

**Table 92. Vds\_threshold\_hs\_Part2 (16Ch)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved				Vds_thr_Hs7				Vds_thr_Hs6				Vds_thr_Hs5			
R/W	-				r/w				r/w				r/w			
Lock	-				no				no				no			
Reset	0000				0000				0000				0000			

**Table 93. Vsrc\_threshold\_hs\_Part1 (16Dh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Vsrc_thr_Hs4				Vsrc_thr_Hs3				Vsrc_thr_Hs2				Vsrc_thr_Hs1			
R/W	r/w				r/w				r/w				r/w			
Lock	no				no				no				no			
Reset	0000				0000				0000				0000			

**Table 94. Vsrc\_threshold\_hs\_Part2 (16Eh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved				Vsrc_thr_Hs7				Vsrc_thr_Hs6				Vsrc_thr_Hs5			
R/W	-				r/w				r/w				r/w			
Lock	-				no				no				no			
Reset	0000				0000				0000				0000			

Note that when reading back this register, what is actually read from the SPI is not the content of the register, but the real configuration of the thresholds, in particular the HSx Vsrc thresholds. (see [See Bootstrap switch control on page 58](#)).



**Table 95. Vsrc\_threshold\_hs and vds\_threshold\_hs value**

hsx_vds/src_threshold(3:0)	Threshold voltage HS VDS / HS VSRC (V)
0000	0.00
1001	0.10
1010	0.20
1011	0.30
1100	0.40
0001	0.50
0010	1.0
0011	1.5
0100	2.0
0101	2.5
0110	3.0
0111	3.5

**Table 96. Vds\_threshold\_ls\_Part 1 (16Fh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Vds thr Ls4				Vds thr Ls3				Vds thr Ls2				Vds thr Ls1			
R/W	r/w				r/w				r/w				r/w			
Lock	no				no				no				no			
Reset	0000				0000				0000				0000			

**Table 97. Vds\_threshold\_ls\_Part 2 (170h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Vds thr Ls8				Vds thr Ls7				Vds thr Ls6				Vds thr Ls5			
R/W	r/w				r/w				r/w				r/w			
Lock	no				no				no				no			
Reset	0000				0000				0000				0000			

**Table 98. Vds\_threshold\_Is value**

Isx_vds_threshold(3:0)	Threshold Voltage LS V <sub>DS</sub> (V)
0000	0.00
1001	0.10
1010	0.20
1011	0.30
1100	0.40
0001	0.50
0010	1.0
0011	1.5
0100	2.0
0101	2.5
0110	3.0
0111	3.5

### 8.3.3.7 Slew rate high-side and low-side selection register

These registers store the slew rate configuration value for each output driver. The microcores can change the value of each field at runtime, provided they have the access right to control the related output (refer to [Table 83, Out\\_acc\\_ucx\\_chy \(160h, 161h, 162h, 163h, 164h, 165h\)](#)). Each output has the same slew rate for the rising and falling edge, save for the low-side 7 and 8.

**Table 99. Hs\_slewrates (171h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		slewrates_hs7		slewrates_hs6		slewrates_hs5		slewrates_hs4		slewrates_hs3		slewrates_hs2		slewrates_hs1	
R/W	-		r/w		r/w		r/w		r/w		r/w		r/w		r/w	
Lock	-		no		no		no		no		no		no		no	
Reset	00		00		00		00		00		00		00		00	

Refer to [Table 22](#) for the slew rates values.

**Table 100. Ls\_slewrates\_Part 1 (172h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved				slewrates_ls6		slewrates_ls5		slewrates_ls4		slewrates_ls3		slewrates_ls2		slewrates_ls1	
R/W	-				r/w		r/w		r/w		r/w		r/w		r/w	
Lock	-				no		no		no		no		no		no	
Reset	0000				00		00		00		00		00		00	

Refer to [Table 25](#) for slew rates values.

**Table 101. Ls\_slewrates\_Part 2 (173h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								slewrates_LS8_ri sing	slewrates_LS8_fa lling	slewrates_LS7_ri sing	slewrates_LS7_f alling				
R/W	-								r/w	r/w	r/w	r/w				
Lock	-								no	no	no	no				
Reset	00000000								00	00	00	00				

Refer to [Table 28](#) and [Table 29](#) for slewrates values.

### 8.3.3.8 Offset compensation results registers

It is possible to measure the offset of each current measure block, including opamp, DAC and comparator; for current measure block 5 and 6, only comparator 5L and 6L, the relative opamp, and the DAC 5L and 6L are considered. The measured offset is automatically compensated during normal operation. The compensation must be enabled by the microcores (when the input current to the current measurement block is 0) with the “stoc” instruction (refer to programming user guide).

The offset can be read through the SPI registers. It is also possible to change the value compensated by writing these registers. If the offset compensation is requested by microcores, it starts from the precedent result (or from the data forced through the SPI). As the offset can be both positive and negative, all the values in these registers are represented as two's complement.

**Table 102. Offset\_compensation12 (174h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		Sign_offset_cur2	Offset_cur2				Reserved		Sign_offset_cur1	Offset_cur1					
R/W	-		r/w	r/w				-		r/w	r/w					
Lock	-		no	no				-		no	no					
Reset	00		0	00000				00		0	00000					

**Table 103. Offset\_compensation34 (175h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		Sign_offset_cur4	Offset_cur4				Reserved		Sign_offset_cur3	Offset_cur3					
R/W	-		r/w	r/w				-		r/w	r/w					
Lock	-		no	no				-		no	no					
Reset	00		0	00000				00		0	00000					

**Table 104. Offset\_compensation56 (176h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		Sign_offset_cur6L	Offset_cur6L				Reserved		Sign_offset_cur5L	Offset_cur5L					
R/W	-		r/w	r/w				-		r/w	r/w					
Lock	-		no	no				-		no	no					
Reset	00		0	00000				00		0	00000					

The offset values are stored in registers using the two's complement notation. The values can range from -32 to 31. The value is then converted to sign-module notation before being transferred to the analog section.

### 8.3.3.9 ADC conversion results registers

It is possible to use the current measure block to perform an ADC conversion. A conversion is performed when requested by a microcore (reference [Programming Guide and Instruction Set](#)) with the correct access rights (refer to [Table 83, Out\\_acc\\_ucx\\_chy \(160h, 161h, 162h, 163h, 164h, 165h\)](#)). The DAC5L and 6L is used when performing an ADC conversion using current measurement channel 5 and 6.

A signal path via the OAx multiplexer, a track and hold circuit, the OAx amplifier, and the DACfb multiplexer is used for ADC mode. While using ADC mode on current measurement channel 1 and 3, the OA1 output is blocked. While using ADC mode on channel 2 or 4, the OA2 output is blocked, and while using ADC mode on channel 5 or 6, the OA3 output is blocked (see [Figure 20](#)). The OAx multiplexer must be set to the right input and the OAx output must be enabled manually. The OAx amplifier is set to a gain of 1.0 automatically. It is not possible to do ADC conversion at the same time at channel 1 and 3, on channel 2 and 4, or on channel 5 and 6.

The conversion takes 11 ck\_ofscmp clock cycles (refer to [Table 146, Ck\\_ofscmp\\_Prescaler\(1A4h\)](#)). 4 ck\_ofscmp clock cycles are needed for the first bit, because the OAx amplifier output has to settle first after changing the gain. After the first bit 1, a clock cycle is needed for every of the 7 following bits. The PT2000 has a “track and hold” circuit for the ADC mode. The switch of the track and hold circuit is opened before the ADC conversion starts and is closed again when ADC mode is switched off. The result of the conversion is stored in the corresponding adc register after the conversion is finished. It is available to the microcore as long as the ADC mode is on and available via the SPI register until the next ADC conversion is started.

To trigger a new conversion, switch off the ADC mode and switch it on again, note that a minimum of 1 ck\_ofscmp clock cycle is needed between “stadc on” and “stadc off” (reference [Programming Guide and Instruction Set](#)). The result can be read via the SPI registers ([Table 105, Adc12\\_results \(177h\)](#), [Table 106, Adc34\\_results \(178h\)](#), and [Table 107, Adc56\\_results \(179h\)](#)).

**Table 105. Adc12\_results (177h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	conversion_2_value								conversion_1_value							
R/W	r								r							
Lock	no								no							
Reset	10000000								10000000							

**Table 106. Adc34\_results (178h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	conversion_4_value								conversion_3_value							
R/W	r								r							
Lock	no								no							
Reset	10000000								10000000							

**Table 107. Adc56\_results (179h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	conversion_6_value								conversion_5_value							
R/W	r								r							
Lock	no								no							
Reset	10000000								10000000							

### 8.3.3.10 Current filters configuration registers

The 10 current feedback are filtered before feeding them to the microcores. The filters of all the current feedback are independent.

**Table 108. Current\_filter12 (17Ah)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Filter_Type2	Reserved		Filter_length_2				Filter_Type1	Reserved		Filter_length_1					
R/W	r/w	-		r/w				r/w	-		r/w					
Lock	yes	-		yes				yes	-		yes					
Reset	0	00		00001				0	00		00001					

**Table 109. Current\_filter34 (17Bh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Filter_Type4	Reserved		Filter_length_4				Filter_Type3	Reserved		Filter_length_3					
R/W	r/w	-		r/w				r/w	-		r/w					
Lock	yes	-		yes				yes	-		yes					
Reset	0	00		00001				0	00		00001					

**Table 110. Current\_filter5I5h (17Ch)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Filter_Type5H	Reserved		Filter_length_5H				Filter_Type5L	Reserved		Filter_length_5L					
R/W	r/w	-		r/w				r/w	-		r/w					
Lock	yes	-		yes				yes	-		yes					
Reset	0	00		00001				0	00		00001					

**Table 111. Current\_filter6I6h (17Dh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Filter_Type6H	Reserved		Filter_length_6H				Filter_Type6L	Reserved		Filter_length_6L					
R/W	r/w	-		r/w				r/w	-		r/w					
Lock	yes	-		yes				yes	-		yes					
Reset	0	00		00001				0	00		00001					

**Table 112. Current\_filter5neg6neg (17Eh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Filter Type 6Neg	Reserved		Filter length 6Neg				Filter Type 5Neg	Reserved		Filter length 5Neg					
R/W	r/w	-		r/w				r/w	-		r/w					
Lock	yes	-		yes				yes	-		yes					
Reset	0	00		00001				0	00		00001					

- Filter\_type. This 1 bit parameter selects the type of filter used for the relative current feedback:
  - if 0 - Any different sample resets the filter counter
  - if 1 - Any different sample decreases the filter counter
- Filter\_lenght. This 5-bit parameter set the filtering time for the current feedback signal.  
 $t_{FTN} = t_{CK} \times (\text{Filter\_length} + 1)$

### 8.3.3.11 Boost DAC configuration registers

This block contains the threshold for the boost DAC. This register can be set either from the SPI interface or from a microcore. It is possible to limit the microcore access to the boost\_dac register by setting access rights. End of line offset compensation is provided for the boost monitoring, requiring no microcode operation.

**Table 113. Boost\_dac (17Fh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								boost_threshold							
R/W	-								r/w							
Lock	-								no							
Reset	00000000								00000000							

**Table 114. Boost\_dac\_access (180h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved										uc1_ch 3 acc	uc0_ch 3 acc	uc1_ch 2 acc	uc0_ch 2 acc	uc1_ch 1 acc	uc0_ch 1 acc
R/W	-										r/w	r/w	r/w	r/w	r/w	r/w
Lock	-										yes	yes	yes	yes	yes	yes
Reset	0000000000										0	0	0	0	0	0

- Boost\_threshold. This 8-bit parameter is the threshold used for boost voltage monitoring.
- ucX chY acc. This 1-bit parameter (active high) grants access to the dac\_boost register.

**Table 115. Boost\_filter (181h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved			filter_ty pe	Boost_fbk_filter											
R/W	-			r/w	r/w											
Lock	-			yes	yes											
Reset	000			0	000000000001											

- Filter\_type: This 1 bit parameter selects the type of filter used:
  - if 0 - Any different sample resets the filter counter
  - if 1 - Any different sample decreases the filter counter
- Boost\_fbk\_filter: This 12-bit parameter sets the filtering time for the output of the vboost comparator.  
 The filtering time is:  $t_{FTN} = t_{CK} \times (x\_filter + 1)$

### 8.3.3.12 V<sub>DS</sub> low-side 7/8 configuration register

These two registers are used for configuring the timeout for V<sub>DS</sub> monitoring of DC/DC resonant converter. If V<sub>BOOST</sub> drops below 2 x V<sub>BAT</sub>, V<sub>DS</sub> would not fall below the V<sub>DS</sub> threshold of 2.5 V, which would not activate the MOSFET in resonant DC/DC converter mode again. Therefore this timeout is required to force the MOSFET to be switched on. The access for this register is only via the SPI.

**Table 116. Vds7\_dcdc\_config (182h) & Vds8\_dcdc\_config (183h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dcdc mode		Reserved			lsx_vds_highspeed_en	Cur_dcdc_x_fbk_sel	vdsx_to_en	vdsx_dcdc_timeout							
R/W	r		-			r/w	r/w	r/w	r/w							
Lock	-		-			no	no	no	no							
Reset	00		000			0	0	0	00000000							

- Vdsx\_dcdc timeout: This 8 bit parameter defines the time duration of the DC/DC converter (V<sub>DS7/8</sub> monitoring Ls7/8). It is needed only if the async\_vds mode is used. Timeout is used if the VDS threshold 2.5 V is not reached.
  - The timeout is:  $t_{FTN} = t_{CK} \times (x_{filter} + 1)$
- Vdsx\_to\_en: V<sub>DS7/8</sub> timeout is enabled. MOSFET is activated automatically when the timeout has been exceeded
- Cur\_dcdc\_x\_fbk\_sel for async\_vds and async mode:
  - For LS7:
    - 0: selects cur5h\_dcdc feedback signal
    - 1: selects cur6h\_dcdc feedback signal
  - For LS8:
    - 0: selects cur6h\_dcdc feedback signal
    - 1: selects cur5h\_dcdc feedback signal
- Lsx\_vds\_highspeed\_en: Enable high speed V<sub>DS</sub> comparator for DC/DC control
  - 0: standard low speed V<sub>DS</sub> monitor enabled
  - 1: high speed V<sub>DS</sub> monitor for DC/DC control enabled
- dcdc\_mode. This bits shows when the automatic DC/DC control feature for LS7/8 is enabled. Refer to [See DC/DC converter control \(LS7/8\) on page 61](#) for the behavior of LS7/8 during this mode.

**Table 117. DC/DC mode**

DC/DC Mode	Description
00, 10	Sync mode enabled. LS7/8 controlled by microcore command
01	Async mode enabled with two current thresholds.
11	Async mode enabled with one current threshold and V <sub>DS</sub> monitor.

### 8.3.3.13 Battery monitoring results

This block contains the result of the batt DAC plus comparator in ADC mode. This register can be read either from the SPI interface or from a microcore.

**Table 118. batt\_results (184h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								batt_result							
R/W	-								r							
Lock	-								no							
Reset	00000000								000000							

The ofs\_comp\_prescaler is used to define the step length of the ADC conversion. The ADC mode is running continuously. One conversion needs 7 cycles.

### 8.3.3.14 DAC 1-6 values registers

Other than from microcores, it is possible to set the DAC for the current measure blocks by writing to these registers.

**Table 119. Dac12\_value (185h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAC 2 value								DAC 1 value							
R/W	r/w								r/w							
Lock	no								no							
Reset	00000000								00000000							

**Table 120. Dac34\_value (186h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAC 4 value								DAC 3 value							
R/W	r/w								r/w							
Lock	no								no							
Reset	00000000								00000000							

**Table 121. Dac515h\_value (187h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAC 5H value								DAC 5L value							
R/W	r/w								r/w							
Lock	no								no							
Reset	00000000								00000000							

**Table 122. Dac5neg\_value (188h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved											DAC 5Neg value				
R/W	-											r/w				
Lock	-											no				
Reset	000000000000											0000				

**Table 123. Dac616h\_value (189h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAC 6H value								DAC 6L value							
R/W	r/w								r/w							
Lock	no								no							
Reset	00000000								00000000							



**Table 124. Dac6neg\_value (18Ah)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved												DAC 6Neg value			
R/W	-												r/w			
Lock	-												no			
Reset	000000000000												0000			

### 8.3.3.15 Load bias configuration register

These two registers configure the biasing for each output. The microcores can change the value of each field at runtime, provided they have the access right to control the related output (refer to [Table 83, Out\\_acc\\_ucx\\_chy \(160h, 161h, 162h, 163h, 164h, 165h\)](#)). High-side 2 and high-side 4 have two biasing structures, one identical (hsx\_en\_pu) to the other high-sides and one stronger (hsx\_en\_s\_pu).

Note that when reading back this register, what is actually read from the SPI is not the content of the register, but the real configuration of the HS and LS bias, therefore after the masks imposed by the init phase of the bootstrap switch (see [See Bootstrap switch control on page 58](#)). Low-side bias is enabled by default, and they stay ON even after bootstrap charge.

**Table 125. Hs\_bias\_config (18Bh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Name	Reserved				hs4_en_s_pu	Re-served	hs2_en_s_pu	Reserved			hs7_en_pu	hs6_en_pu	hs5_en_pu	hs4_en_pu	hs3_en_pu	hs2_en_pu	hs1_en_pu
R/W	-				r/w	-	r/w	-			r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	-				no	-	no	-			no	no	no	no	no	no	no
Reset	0000				0	0	0	00			0	0	0	0	0	0	0

**Table 126. Ls\_bias\_config (18Ch)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								ls8_en_pd	ls7_en_pd	ls6_en_pd	ls5_en_pd	ls4_en_pd	ls3_en_pd	ls2_en_pd	ls1_en_pd
R/W	-								r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	-								no	no	no	no	no	no	no	no
Reset	00000000								1	1	1	1	1	1	1	1

### 8.3.3.16 Bootstrap charged/timer status registers

This register reads the charge status of the HS bootstrap capacitors during initialization phase. ([See Bootstrap switch control on page 58](#)).

**Table 127. Bootstrap\_charged (18Dh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserv ed	hs7_sr c_1V	hs6_sr c_1V	hs5_sr c_1V	hs4_sr c_1V	hs3_sr c_1V	hs2_sr c_1V	hs1_sr c_1V	Reserv ed	hs7_bs _charg ed	hs6_bs _charg ed	hs5_bs _charg ed	hs4_bs _charg ed	hs3_bs _charg ed	hs2_bs _charg ed	hs1_bs _charg ed
R/W	-	r	r	r	r	r	r	r	-	r	r	r	r	r	r	r
Lock	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Reset	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1

**Table 128. Bootstrap\_timer (18Eh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	bootstrap_init_timer						Reserved									
R/W	r						-									
Lock	-						-									
Reset	000000						0000000000									

- hsx\_bs\_charged: when '0', the bootstrap capacitor for HSx is charged
- hsx\_src\_1V: when '1' it was necessary for this pre-driver to switch the  $V_{SRC}$  threshold to 1.0 V to finish the bootstrap init
- bootstrap init timer: this shows the current value of the six MSBs of the bootstrap init timer. The value is '110100' when the timer expires

Table 129, Bootstrap\_charged Bits shows the exact meaning of the bits hsx\_bs\_charged and hsx\_src\_1V.

**Table 129. Bootstrap\_charged Bits**

hsx_bs_charged	hsx_src_1V	Description
1	0	Bootstrap capacitor not charged, bootstrap init timer not lapsed, $V_{SRC} = 0.5$ V (reset value)
1	1	Bootstrap capacitor not charged, timer lapsed, $V_{SRC} = 1.0$ V
0	0	Bootstrap capacitor charged, $V_{SRC} = 0.5$ V when charging finished
0	1	Bootstrap capacitor charged, $V_{SRC} = 1.0$ V when charging finished

The bootstrap init timer value can be used, together with the other bits of the register, to identify in detail how much time has passed since  $V_{CCP}$  voltage was stable and which threshold is used to detect the charge of the bootstrap capacitor.

**Table 130. Bootstrap init timer**

Bit value	Description
000000	$V_{CCP}$ voltage is not stable (undervoltage)
000001	$V_{CCP}$ voltage is stable since 0.68 ms ( $2^{14} / 24$ MHz <sup>(52)</sup> ). Source HS voltage threshold used to detect bootstrap charge is 0.5 V
...	...
100000	$V_{CCP}$ voltage is stable since 21.8 ms <sup>(52)</sup> . Source HS voltage threshold used to detect bootstrap charge is 0.5 V
...	...
110011	$V_{CCP}$ voltage is stable since 34.8 ms <sup>(52)</sup> . Source HS voltage threshold used to detect bootstrap charge is 0.5 V
110100 (final value)	$V_{CCP}$ voltage is stable since at least 35.5 ms <sup>(52)</sup> . Source HS voltage threshold used to detect bootstrap charge is 1.0 V.

Notes

52. PLL factor set to 1 means 24 MHz cksys. This calculation will be different if PLL factor is set to 0.

### 8.3.3.17 High-side 1-7 ground reference configuration registers

**Table 131. Hsx\_ls\_act (18Fh - 195h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	hsx_ls_act_dis	Reserved							hsx_ls_act_dis	hsx_ls7_act	hsx_ls6_act	hsx_ls5_act	hsx_ls4_act	hsx_ls3_act	hsx_ls2_act	hsx_ls1_act
R/W	r/w	-							r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	-							yes	yes	yes	yes	yes	yes	yes	yes
Reset	0	0000000							1	1	1	1	1	1	1	

These seven registers are used to configure the ground reference of the hs1 to hs7 source pins.

The hsx\_ls\_act signal is high if any of the LSX pins connected to the HSX source pin is switched on or if the function is disabled by the hsx\_ls\_act\_dis bit.

- hsx\_ls1\_act: must be set to '1' if ls1 is connected to the same load as hsx.
- hsx\_ls2\_act: must be set to '1' if ls2 is connected to the same load as hsx.
- hsx\_ls3\_act: must be set to '1' if ls3 is connected to the same load as hsx.
- hsx\_ls4\_act: must be set to '1' if ls4 is connected to the same load as hsx.
- hsx\_ls5\_act: must be set to '1' if ls5 is connected to the same load as hsx.
- hsx\_ls6\_act: must be set to '1' if ls6 is connected to the same load as hsx.
- hsx\_ls7\_act: must be set to '1' if ls7 is connected to the same load as hsx.
- hsx\_ls8\_act: must be set to '1' if ls8 is connected to the same load as hsx.
- hsx\_ls\_act\_dis: set this bit to disable the link between hsx and ls pre-drivers. If this bit is set the hsx\_ls\_act signal is forced to '0' regardless if an ls is active or not.

### 8.3.3.18 DAC settling time register

This register is used to set the DAC settling time: while this time is being counted, no micro-code checks on the related current feedback is true. This is not applicable to the VBoost Dac.

**Table 132. Dac\_settling\_time (196h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved										dac_settling_time					
R/W	-										r/w					
Lock	-										yes					
Reset	0000000000										000000					

Every time the value of related DAC register is written, the current feedback is marked as invalid for

$$t_x = t_{CK} \times (\text{dac\_settling\_time} + \text{Filter\_length} + 4)$$

The filter\_length value can be set in the current filter registers (refer to [See Current filters configuration registers on page 109](#) and [See Boost DAC configuration registers on page 110](#)). Since filter configuration can be different for each DAC, the settling time is also different.

### 8.3.3.19 Analog output (OAx) configuration register

These three registers configure the function of the three OA\_OUTx pins.

**Table 133. Oa\_out1\_config (197h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved									oa1_g1	oa_sel1			oa1_gain	oa1_en	
R/W	-									r/w	r/w			r/w	r/w	
Lock	-									no	no			no	no	
Reset	000000000									0	000			00	0	

- oa1 en: when '1', the selected source is sent to the OA\_OUT1 pin, otherwise it is put in high-impedance to connect all OAx pins to the same MCU ADC.
- oa1 gain: select the gain to apply to the signal
  - "00": gain 1.33
  - "01": gain 2.0
  - "10": gain 3.0
  - "11": gain 5.33
- oa1 g1: select the gain to apply to the signal
  - "0": gain according to oa1 gain
  - "1": gain forced to 1.0
- oa\_sel1: select the signal to send to the OA\_OUT1 pin.
  - "000": output from current measurement block 1
  - "001": output from current measurement block 3
  - "101": 2.5 Volt

**Table 134. Oa\_out2\_config (198h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved									oa2_g1	oa_sel2			oa2_gain	oa2_en	
R/W	-									r/w	r/w			r/w	r/w	
Lock	-									no	no			no	no	
Reset	000000000									0	000			00	0	

- oa2 en: when '1' the selected source is sent to OA\_OUT2.
- oa2 gain: select the gain to apply to the signal.
  - "00": gain 1.33
  - "01": gain 2.0
  - "10": gain 3.0
  - "11": gain 5.33
- oa2 g1: select the gain to apply to the signal
  - "0": gain according to oa2 gain
  - "1": gain forced to 1.0
- oa\_sel2: select the signal to send to the OA\_OUT2 pin.
  - "000": output from current measurement block 2
  - "001": output from current measurement block 4
  - "101": 2.5 Volt

**Table 135. Oa\_out3\_config (199h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved									oa3_g1	oa_sel3			oa3_gain	oa3_en	
R/W	-									r/w	r/w			r/w	r/w	
Lock	-									no	no			no	no	
Reset	000000000									0	000			00	0	

- oa3 en: when '1', the selected source is sent to the OA\_OUT3 pin, otherwise it is put in high-impedance to connect all OAx pins to the same MCU ADC.
- oa3 gain: select the gain to apply to the signal
  - "00": gain 1.33
  - "01": gain 2.0
  - "10": gain 3.0
  - "11": gain 5.33
- oa3 g1: select the gain to apply to the signal.
  - "0": gain according to oa3 gain
  - "1": gain forced to 1.0
- oa\_sel3: select the signal to send to the OA\_OUT3 pin.
  - "000": output from current measurement block 5
  - "001": output from current measurement block 6
  - "101": 2.5 Volt
  - "111": V<sub>CCA</sub>

## 8.3.4 Main configuration registers

### 8.3.4.1 "Ck" clock prescaler configuration register

This is a 6-bit register setting the divider ratio to generate the ck clock starting from cksys (24 MHz or 12 MHz refer to [Table 151, PLL\\_Config \(1A7h\)](#)). The ck\_per register must not be changed again after the microcores are running. During operation the register should be locked. Note that the actual divider ratio is  $ck = cksys/(ck\_per+1)$ . Therefore setting ck\_per to "000100" sets  $ck = cksys/(4+1)$ .

**Table 136. Clock\_Prescaler (1A0h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved										ck_per					
R/W	-										r/w					
Lock	-										yes					
Reset	0000000000										000000					

Depending on the ck\_per setting, different device/channel operation modes is possible according to [Table 137, Ck\\_per and device modes](#) and [See Dual microcore arbiter on page 70](#).

**Table 137. Ck\_per and device modes**

ck_per	Clock divider	microcore frequency at 24 MHz cksys (PLL factor = 1)	SPI access (r/w) to registers and DRAM	SPI access (r/w) to CRAM after flash enable	Single/dual microcore	Drive outputs from flag pins
0	0+1=1	-	yes (r/w)	no	no	no
1	1+1=2	12 MHz <sup>(53)</sup>	yes (r/w)	no	yes	yes
2	2+1=3	8.0 MHz	yes (r/w)	yes (r)	yes	yes
≥3	≥4	≤6.0 MHz	yes (r/w)	yes (r)	yes	yes

Notes

53. Some limitations apply on the number of consecutive DRAM access and SPI frequency (See [Dual microcore arbiter on page 70](#)).

### 8.3.4.2 Flags direction configuration register

This is a 16-bit register where each bit sets the direction of the corresponding flag as shown in [Table 139, Flags\\_source & Flags\\_direction registers](#). The value of this register is used only for the flags which drive or can be driven by a device pin as specified in the flag\_source register.

**Table 138. Flag\_direction (1A1h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	flag_dir 15	flag_dir 14	flag_dir 13	flag_dir 12	flag_dir 11	flag_dir 10	flag_dir 9	flag_dir 8	flag_dir 7	flag_dir 6	flag_dir 5	flag_dir 4	flag_dir 3	flag_dir 2	flag_dir 1	flag_dir 0
	Dbg	OA2	Irq	Start8	Start7	Start6	Start5	Start4	Start3	Start2	Start1	Flag4	Flag3	Flag2	Flag1	Flag0
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

**Table 139. Flags\_source & Flags\_direction registers**

flags_source(x)	flags_direction(x)	flag_bus(x) source
0	0/1	The corresponding pin is used for its non-flag function (start_irq, analog OA2, extFWx, etc.). Flag_bus(x) is driven by int_flags(x).
1	0	The corresponding pin is used as an output flag. The device pin is driven by int_flags(x). Flag_bus(x) is driven by int_flags(x).
1	1 (reset value)	The corresponding pin is used as an input flag. The Flag_bus(x) is driven by the device pin.

**Table 140. Flags\_source & Flags\_direction registers for Flag 4 and 12**

flags_source (12)	flags_source (4)	flags_direction (12)	flags_direction (4)	flag_bus (12) source	flag_bus (4) source
-	0	-	-	The pin is used for current measurement channel VsenseN4.	The pin is used for current measurement channel VsenseP4.
0	1	-	0	The pin is used for its non-flag function start8.	The corresponding pin is used as an output flag.
0 (reset value)	1 (reset value)	-	1 (reset value)	The pin is used for its non-flag function start8.	The corresponding pin is used as an input flag.
1	1	0	0	The corresponding pin is used as an output flag.	The corresponding pin is used as an output flag.
1	1	0	1	The corresponding pin is used as an input flag.	The corresponding pin is used as an input flag.

**Table 140. Flags\_source & Flags\_direction registers for Flag 4 and 12**

flags_source (12)	flags_source (4)	flags_direction (12)	flags_direction (4)	flag_bus (12) source	flag_bus (4) source
1	1	1	0	The corresponding pin is used as an input flag.	The corresponding pin is used as an output flag.
1	1	1	1	The corresponding pin is used as an input flag.	The corresponding pin is used as an input flag.

### 8.3.4.3 Flags polarity register

This is a 16-bit register where each bit sets the polarity of the corresponding flag as shown in [Table 142, Flags\\_polarity](#). If a 1 is set, the corresponding flag inverts. The value of this register is only used for the flags driven by or drive a device pin, as specified in the flag\_source register.

**Table 141. Flag\_polarity (1A2h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	flag_po_l15	flag_po_l14	flag_po_l13	flag_po_l12	flag_po_l11	flag_po_l10	flag_po_l9	flag_po_l8	flag_po_l7	flag_po_l6	flag_po_l5	flag_po_l4	flag_po_l3	flag_po_l2	flag_po_l1	flag_po_l0
	Dbg	OA2	Irq	Start8	Start7	Start6	Start5	Start4	Start3	Start2	Start1	Flag4	Flag3	Flag2	Flag1	Flag0
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 142. Flags\_polarity**

flags_polarity(x)	flag_bus(x) condition
0	as it is
1	inverted

Some bits of this register are also used to set the polarity of the start pins when they are not used as flag I/O.

**Table 143. Flag\_polarity register (1A2h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	-	-	-	start_pol8	start_pol7	start_pol6	start_pol5	start_pol4	start_pol3	start_pol2	start_pol1	-	-	-	-	-
R/W	-	-	-	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	-	-	-	-	-
Lock	-	-	-	yes	yes	yes	yes	yes	yes	yes	yes	-	-	-	-	-
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 144. Start\_polarity**

flags_polarity(x)	flag_bus(x) condition
0	Start active high
1	Start active Low

### 8.3.4.4 Flags source register

All of the 16 flags have a configurable source. This is a 16-bit register where each bit sets the source of the corresponding flag as shown in Table 139, [Flags\\_source & Flags\\_direction registers](#). The flag 12 pin can even be used for three different functions (refer to Table 140, [Flags\\_source & Flags\\_direction registers for Flag 4 and 12](#))

**Table 145. Flag\_source (1A3h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	flag_sr_c15	flag_sr_c14	flag_sr_c13	flag_sr_c12	flag_sr_c11	flag_sr_c10	flag_sr_c9	flag_sr_c8	flag_sr_c7	flag_sr_c6	flag_sr_c5	flag_sr_c4	flag_sr_c3	flag_sr_c2	flag_sr_c1	flag_sr_c0
	Dbg	OA2	Irq	Start8	Start7	Start6	Start5	Start4	Start3	Start2	Start1	Vsense <sub>4</sub>	Flag3	Flag2	Flag1	Flag0
R/W	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	1	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1

### 8.3.4.5 Offset compensation and ADC clock (ck\_ofscmp) prescaler

This is an 8-bit register setting the divider ratio to generate the ck\_ofscmp clock starting from cksys. This clock feeds the following blocks:

- Current measurement block ([See Current measurement offset compensation on page 51](#))
- Battery voltage measurement ([See Battery voltage monitor on page 36](#))
- ADC conversion ([See ADC conversion results registers on page 108](#))

Note that the actual divider ratio is  $ck\_ofscmp = ck\_ofscmp\_per + 1$ . Therefore setting ck\_ofscmp\_per to “00001000” ck\_ofscmp is cksys/9. The reset value is 2.0 μs (cksys at 24 MHz).

**Table 146. Ck\_ofscmp\_Prescaler(1A4h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								ck_ofscmp_per							
R/W	-								r/w							
Lock	-								yes							
Reset	00000000								00101111							

### 8.3.4.6 Driver interrupt configuration registers

These two registers can configure which conditions concur to the driver disable, which conditions concur to the interrupt generation, and if the interrupt request must be generated toward the microcontroller and the microcores.

**Table 147. Driver\_config\_Part 1 (1A5h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved			Overtemp_irq_en	Drv_en_irq_en	Vboost_irq_en	Vcc5_irq_en	Vccp_irq_en	Iret_en	Irq_uc1_ch3_en	Irq_uc0_ch3_en	Irq_uc1_ch2_en	Irq_uc0_ch2_en	Irq_uc1_ch1_en	Irq_uc0_ch1_en	Irq_uc_en
R/W	-			r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w
Lock	-			yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Reset	000			0	0	0	0	0	0	0	0	0	0	0	0	0



**Table 148. Driver\_config\_Part 2 (1A6h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	vccp_ext_en	Reserved													Vboost_mon_en	Vboost_disable_en
R/W	r/w	-													r/w	r/w
Lock	yes	-													yes	yes
Reset	1	0000000000000													0	0

The driver\_enable block generates a local interrupt masking the five possible disable conditions with the following bits:

- Drv\_en\_irq\_en: if set, the drv\_en generates the local interrupt
- Vboost\_irq\_en: if set, an undervoltage on V<sub>BOOST</sub> generates the local interrupt
- Vcc5\_irq\_en: if set, an undervoltage on V<sub>CC5</sub> generates the local interrupt
- Vccp\_irq\_en: if set, an undervoltage on V<sub>CCP</sub> generates the local interrupt
- Overtemp\_irq\_en: if set, the over temperature condition generates the local interrupt

If a local interrupt is generated, it is possible to propagate it to an external device (micro controller) and to the six microcores. This is done when the following bits are set:

- Irq\_uc\_en, for the external device using the IRQB pin
- Irq\_uc0\_ch1\_en, for the microcore 0 of channel 1
- Irq\_uc1\_ch1\_en, for the microcore 1 of channel 1
- Irq\_uc0\_ch2\_en, for the microcore 0 of channel 2
- Irq\_uc1\_ch2\_en, for the microcore 1 of channel 2
- Irq\_uc0\_ch3\_en, for the microcore 0 of channel 3
- Irq\_uc1\_ch3\_en, for the microcore 1 of channel 3

**Table 149. Truth table for propagation of UV Vboost\_irq**

Configuration				Resulting behavior		
Vboost_irq_en	Vboost_disable_en	Irq_ucx_chy_en	Irq_uc_en	UV Vboost bit set	irq_to microcore	irq_to μC (Irqb pin)
0	0	d.c.	d.c.	no	no	no
0	1	d.c.	d.c.	yes	no	no
1	0	0	d.c.	no	no	no
1	0	1	d.c.	no	no	no
1	1	0	0	yes	no	no
1	1	0	1	yes	no	yes
1	1	1	0	yes	yes	no
1	1	1	1	yes	yes	yes

This register also contains some other configuration bit concerning the output drivers:

- iret\_en: the driver\_enable block automatically generates an iret request toward all the microcores (this request can be filtered by microcode if not required). No iret request is generated if the interrupt was triggered by a loss of clock. It is possible to select two types of iret:
  - If iret\_en is set to '0', an iret request is sent to the microcores when the drivers are re-enabled after a disable condition
  - If iret\_en is set to '1', an iret request is sent to the microcores when the drivers\_status register is cleared. For the iret to happen, either write the driver status register or to read it while the reset on read configuration is active
- Vboost\_disable\_en: if set, an undervoltage of V<sub>BOOST</sub> disables the output drivers
- Vboost\_mon\_en: this signal configures the divider on the V<sub>BOOST</sub> voltage
  - If Vboost\_mon\_en is set to '0', V<sub>BOOST</sub> is divided by 32 and then compared with a threshold.
  - If Vboost\_mon\_en is set to '1', V<sub>BOOST</sub> is divided by 4 and then compared with a threshold.

- vccp\_ext\_enable: if set to '0', the internal voltage regulator is enabled and the corresponding pin is used only to connect a bypass capacitor. If set to '1', the internal voltage regulator is disabled and the V<sub>CCP</sub> voltage must be supplied externally through the corresponding pin. During bootstrap switch init (See [Bootstrap switch control on page 58](#)), this setting is bypassed and the value of the vccp\_ext\_enable signal is set to the inverted value of the DBG pin sampled at reset (PORresetB and ResetB). This is better defined in [Table 150, VCCP external enable setting](#). This means the DBG pin, at reset, needs to be configured as an input whose value is latched at the rising edge of the PORresetB and ResetB signal, and used to set the configuration of the V<sub>CCP</sub> internal regulator during the init phase of the bootstrap switch. A SPI reset leaves the latched information unchanged. The DBG pin has an internal weak pull-up resistor so its value is '1' when not connected (n.c.).

**Table 150. V<sub>CCP</sub> external enable setting**

dbg pin (latched)	SPI bit	BS Init (min. 1 HS)	VCCP external enable
1 (n.c.)	-	1	0 (internal reg ON)
1 (n.c.)	1	0	1 (internal reg OFF)
1 (n.c.)	0	0	0 (internal reg ON)
0	1	-	1 (internal reg OFF)
0	0	-	0 (internal reg ON)

### 8.3.4.7 PLL factor and speed configuration register

**Table 151. PLL\_Config (1A7h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved														PLL_spre ad_disabl e	PLL_fact or
R/W															r/w	r/w
Lock															yes	yes
Reset	00000000000000														0	1

- PLL\_factor: if set to '0', the PLL multiplication factor is 12, otherwise it is 24
- PLL\_spread\_disable: if set to '0' the PLL output clock has a spread, otherwise it has no spread

The PLL factor is changed synchronously with clock monitor cycle to avoid a clock monitor alert when changing between 12 MHz and 24 MHz. This register is reset by PORreset, ResetB, and the SPI reset.

### 8.3.4.8 Backup clock status register

**Table 152. backup\_clock\_status (1A8h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved	cksys_mi ssing_dis able_driv er	Reserved				uc1_ch 3_irq_e n	uc0_ch 3_irq_e n	uc1_ch 2_irq_e n	uc0_ch 2_irq_e n	uc1_ch 1_irq_e n	uc0_ch 1_irq_e n	uc_irq_ en	switch_to _clock_pi n	loss_of _clock	
R/W	-	r/w	-				r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r
Lock	-	yes	-				yes	yes	yes	yes	yes	yes	yes	no	no	
Reset	0	0	00000				0	0	0	0	0	0	0	0	0	

This 11-bit register is meant to provide the external microcontroller a way to monitor the status of the clock signal. This is done by latching the information occurring on a switch to the backup\_reference signal:

- Loss\_of\_clock: this read\_only bit (loss\_of\_clock) latches the condition when the input reference is missing

- `Switch_to_clock_pin`: this bit (active on rising edge) is used to provide a way to reset the loss of clock condition. If this bit is set during a loss of clock condition it is reset as soon as the clock manager switches the PLL input to the external reference. If this bit sets while there is no loss of clock, the bit resets immediately without any effect.
- `uc_irq_en`: enable the generation of an interrupt request to the external micro controller when `cksys` missing is detected. This interrupt is active until this register is read
- `uc0_ch1_irq_en`: enable the generation of an interrupt request to microcore 0 channel 1 when `cksys` missing is detected
- `uc1_ch1_irq_en`: enable the generation of an interrupt request to microcore 1 channel 1 when `cksys` missing is detected
- `uc0_ch2_irq_en`: enable the generation of an interrupt request to microcore 0 channel 2 when `cksys` missing is detected
- `uc1_ch2_irq_en`: enable the generation of an interrupt request to microcore 1 channel 2 when `cksys` missing is detected
- `uc0_ch3_irq_en`: enable the generation of an interrupt request to microcore 0 channel 3 when `cksys` missing is detected
- `uc1_ch3_irq_en`: enable the generation of an interrupt request to microcore 1 channel 3 when `cksys` missing is detected
- `cksys_missing_disable_driver`: if set, the output drivers are disabled via the signal `cksys_drven` as long as the `cksys_missing` signal is '1'.

Once the `loss_of_clock` bit sets, it can be reset only by completing a “switch to clock” pin. For this operation to complete, it must be requested when the main clock input pin again provides a valid clock frequency.

The interrupt to the external micro and to the microcores is triggered as long as the `cksys_missing` signal is set. The microcore is able to process the interrupt as soon as there is a clock available. It is triggered every time the clock manager switches to the internal clock reference and when the clock manager tries to switch back to the external clock reference, due to a SPI request. The interrupt can even occur multiple times during `cksys_missing` state.

### 8.3.4.9 SPI configuration register

The `spi_config` register (address 1A9h) is an 8-bit register storing the SPI protocol configuration and SPI diagnostics.

**Table 153. Spi\_config (1A9h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								MISO_sle wrate	protocol_ mode	irq_en	watchdog				
R/W	-								r/w	r/w	r/w	r/w				
Lock	-								yes	yes	yes	yes				
Reset	00000000								0	0	0	01010				

- `Miso_slewrates`: selects one of the two possible values for the slew rate of the MISO pin.
  - 0 slow slew rate
  - 1 fast slew rate
- `Protocol_mode`: select the type of burst transmission accepted by the protocol, '0' means mode A, '1' means mode B.
- `irq_enable`: enable the SPI interface to request an interrupt toward the microcontroller if an incorrect SPI transmission is received.
- `Watchdog`: when using mode A, the maximum time the SPI chip select can be inactive during a burst is expressed as follows:

$$t_{WATCHDOG} = t_{CKSYS} * ((watchdog + 1) * 32768)$$

where  $t_{CKSYS}$  is the period of the `cksys` internal clock.

When changing the SPI protocol mode by a write access to this register, it has to be done using a SPI transmission which is compatible to mode A and B (see [See Mode A on page 77](#) and [See Mode B on page 78](#)). This means it must not use '0' as number of operations for the SPI transmission, and must not deassert the chip select during the transmission. Changing the protocol mode can be done as often as required.

### 8.3.4.10 Tracer start/stop registers

**Table 154. Trace\_start (1AAh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved						start_address									
R/W	-						r/w									
Lock	-						no									
Reset	000000						0000000000									

The trigger\_address field contains the address used to synchronize the PT2000 trace\_unit with the external tracer. If the trace operation is enabled and the trace unit is in the idle state, when the uPC value of the selected microcore reaches this address, the sync code is transmitted on the DBG pin. Then the trace\_unit goes to the next phase.trace\_stop Register (1ABh)

**Table 155. Trace\_stop (1ABh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved						stop_address									
R/W	-						r/w									
Lock	-						no									
Reset	000000						0000000000									

The stop\_address field contains the address used to finalize the trace operation. If the trace operation is ongoing (trace phase), when the uPC value of the selected microcore reaches this address, the trace\_unit goes to the next phase (post trigger phase).

### 8.3.4.11 Tracer configuration register

**Table 156. Trace\_config (1ACh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Trace enable	Reserved				uc select			Post trigger length							
R/W	r/w	-				r/w			r/w							
Lock	no	-				no			no							
Reset	0	0000				000			00000000							

- Trace enable. When this bit is set to '1', the trace\_unit starts the first phase of the trace operation. This bit can be set to '0' by the user, to immediately stop the PT2000 trace unit transmission. This bit is automatically reset after the trace operation is complete (all the four phases are finished).
- uc select. Select which is the microcore target of the trace operation:
  - "000": microcore 0, channel 1
  - "001": microcore 1, channel 1
  - "010": microcore 0, channel 2
  - "011": microcore 1, channel 2
  - "100": microcore 0, channel 3
  - "101": microcore 1, channel 3
- Post trigger length. This field selects the duration of the post trigger phase, expressed as number of ck clock cycles. Writing 255 in the post\_trigger\_length field of the trace\_config register causes the trace unit to output a continuous stream after the stop point. With this, run the trace operation for an unlimited amount of time and simply deactivate it by writing zero in the trace\_enable bit of the trace config register.

### 8.3.4.12 Lock device register

It is possible to lock some registers of the PT2000. Locked registers can be read but cannot be written. The lock is not mandatory for the correct working of the device; it is only a safety feature. It is possible to reset this lock by writing an unlock password. It is also possible to independently lock a section of each Data RAM, the last 16 addresses.

**Table 157. Device\_Lock (1ADh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved												Dram3_private_are a_lock	Dram2_private_are a_lock	Dram1_private_are a_lock	Dev_lock
R/W	-												r/w	r/w	r/w	r/w
Lock	no												yes, by itself	yes, by itself	yes, by itself	yes
Reset	000000000000												0	0	0	0

The dev\_lock bit can be set by writing the device lock register. It cannot be reset by writing the device\_lock register, but only by writing the correct password in “unlock password”. If the device lock bit is set, all the register bits marked as “lock = yes” in this documentation can no longer be changed by the SPI. Note that there are some bits not locked by the device lock bit, but locked by other mechanism. Each of the lock bits for a DRAM is locked by itself and not by the dev lock bit as all the other registers in the device. When the correct “unlock password” is provided, these three bit are also reset (refer to [Table 80, Unlock\\_word \(113h, 133h, 153h\)](#)).

### 8.3.4.13 Reset register behavior

Some registers of the device can be configured to reset when read by an external device through the SPI; read accesses by microcores using the SPI backdoor never reset those registers.

**Table 158. Reset\_Behavior (1AEh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Driver_enable_RB	Reserved	SR_uc1_ch3_RB	SR_uc0_ch3_RB	SR_uc1_ch2_RB	SR_uc0_ch2_RB	SR_uc1_ch1_RB	SR_uc0_ch1_B	Reserved		diag_uc1_ch3_RB	diag_uc0_ch3_RB	diag_uc1_ch2_RB	diag_uc0_ch2_RB	diag_uc1_ch1_RB	diag_uc0_ch1_RB
R/W	r/w	-	r/w	r/w	r/w	r/w	r/w	r/w	-	-	r/w	r/w	r/w	r/w	r/w	r/w
Lock	no	-	no	no	no	no	no	no	-	-	no	no	no	no	no	no
Reset	0	0	0	0	0											

- Driver enable reset behavior: if set to '1' driver\_status register is reset on read (refer to [Table 162, driver\\_status \(1B2h\)](#))
- Automatic diagnostics uc0\_ch1 reset behavior: if set to '1' err\_uc register of microcore 0 of channel 1 is reset on read ([See Automatic diagnostics error status register on page 137](#)). The three registers are reset when the err\_ucXchY\_3 register is read
- Automatic diagnostics uc1\_ch1 reset behavior: if set to '1' err\_uc register of microcore 1 of channel 1 is reset on read ([See Automatic diagnostics error status register on page 137](#)). The three registers are reset when the err\_ucXchY\_3 register is read
- Automatic diagnostics uc0\_ch2 reset behavior: if set to '1' err\_uc register of microcore 0 of channel 2 is reset on read ([See Automatic diagnostics error status register on page 137](#)). The three registers are reset when the err\_ucXchY\_3 register is read
- Automatic diagnostics uc1\_ch2 reset behavior: if set to '1' err\_uc register of microcore 1 of channel 2 is reset on read ([See Automatic diagnostics error status register on page 137](#)). The three registers are reset when the err\_ucXchY\_3 register is read
- Automatic diagnostics uc0\_ch3 reset behavior: if set to '1' err\_uc register of microcore 0 of channel 3 is reset on read ([See Automatic diagnostics error status register on page 137](#)). The three registers are reset when the err\_ucXchY\_3 register is read
- Automatic diagnostics uc1\_ch3 reset behavior: if set to '1' err\_uc register of microcore 1 of channel 3 is reset on read ([See Automatic diagnostics error status register on page 137](#)). The three registers are reset when the err\_ucXchY\_3 register is read
- Status register uc0\_ch1 reset behavior: if set to '1' the status register of microcore 0 of channel 1 is reset on read ([See Status register microcore0 on page 92](#) and [See Status register microcore1 on page 92](#))
- Status register uc1\_ch1 reset behavior: if set to '1' the status register of microcore 1 of channel 1 is reset on read ([See Status register microcore0 on page 92](#) and [See Status register microcore1 on page 92](#))
- Status register uc0\_ch2 reset behavior: if set to '1' the status register of microcore 0 of channel 2 is reset on read ([See Status register microcore0 on page 92](#) and [See Status register microcore1 on page 92](#))

- Status register uc1\_ch2 reset behavior: if set to '1' the status register of microcore 1 of channel 2 is reset on read (See Status register microcore0 on page 92 and See Status register microcore1 on page 92)
- Status register uc0\_ch3 reset behavior: if set to '1' the status register of microcore 0 of channel 3 is reset on read (See Status register microcore0 on page 92 and See Status register microcore1 on page 92)
- Status register uc1\_ch3 reset behavior: if set to '1' the status register of microcore 1 of channel 3 is reset on read (See Status register microcore0 on page 92 and See Status register microcore1 on page 92)

The reset on read feature is implemented so no data is lost when the reset of the register is requested and at the same time there is a request to set a bit in the register from the microcode. The bit sets and transmits via the SPI the next time the register is read.

### 8.3.4.14 Unlock device register

**Table 159. Device\_Unlock (1AFh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Unlock_password															
R/W	w															
Lock	-															
Reset	-															

Writing the password **1337h** in the unlock password field, resets the device\_lock register (refer to Table 157, Device\_Lock (1ADh)).

### 8.3.4.15 SPIReset global reset register 1 and 2

This 32-bit register is divided into two 16-bit slices. When the correct “global reset code” is written in this register, a SPIreset is generated. This reset lasts for eight cksys clock cycles, then the global reset registers are reset. The global reset code is “**F473h**” for Global reset register 1 and “**57A1h**” for Global reset register 2.

**Table 160. Global\_Reset\_code\_part1 (1B0h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Global_Reset_Register_code_1															
R/W	w															
Lock	no															
Reset	0000h															

**Table 161. Global\_Reset\_code\_part2 (1B1h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Global_Reset_Register_code_2															
R/W	w															
Lock	no															
Reset	0000h															

### 8.3.4.16 Driver disable status register

This 7-bit register is meant to provide the external micro-controller a way to monitor the status of the output drivers. This is done latching any error condition which disables the output drivers. Some of these error conditions must be enabled (refer to [Table 152, backup\\_clock\\_status \(1A8h\)](#) and [Table 147, Driver\\_config\\_Part 1 \(1A5h\)](#)).

**Table 162. driver\_status (1B2h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved									cksys_missing	DrvEn_latch	DrvEn_value	Overtemp	uv_vboost	uv_vcc5	uv_vccp
R/W										r	r	r	r	r	r	r
Lock										-	-	-	-	-	-	-
Reset	0000000000000									0	1	-	0	0	0	0

- cksys\_missing: this bit is set if the cksys missing condition it disables the drivers. This condition can be configured (refer to [Table 152, backup\\_clock\\_status \(1A8h\)](#)).
- DrvEn\_latch: this bit latches the condition when the DRVEN input pin is inactive. The bit is reset to 1.
  - 1: DRVEN pin was NOT low since last reset of the driver\_status register
  - 0: DRVEN pin was low since the last reset of the driver\_status register
- DrvEn\_value: this bit is not an error condition; it is only a “living copy” of the drv\_en pin.
  - 1: DRVEN pin is high
  - 0: DRVEN pin is low
- Overtemperature: this bit latches the condition where an over temperature is present. It is not used to disable the drivers.
- Uv\_vboost: this bit is set if the undervoltage on the vboost disables the high-side drivers. This condition can be configured (refer to [Table 147, Driver\\_config\\_Part 1 \(1A5h\)](#)).
- Uv\_Vcc5: this bit latches the undervoltage condition on Vcc5.
- Uv\_vccp: this bit latches the undervoltage condition on Vccp and the error from GND loss detection.

Once an error bit has been set, it can only be reset by an SPI write operation in this register (if the corresponding error is no more present). The same error bits are reset even upon SPI read operations but only when a proper enable bit is set (refer to [Table 158, Reset\\_Behavior \(1AEh\)](#)).

### 8.3.4.17 SPI error status register

**Table 163. Spi\_error (1B3h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved												cksys_missing	frame_error	word_error	
R/W													r	r	r	
Lock													-	-	-	
Reset	0000000000000												0	0	0	

The spi\_error register is a 3-bit register split in this way:

- spi\_error(2): cksys\_missing error condition
- spi\_error(1): frame incomplete error condition
- spi\_error(0): word incomplete error condition

The duty of this block is to monitor the spi\_protocol and the spi\_interface to find errors during the communication with the microcontroller. If an error is detected, the corresponding code is stored in the spi\_error\_code register. To warn the microcontroller, during the write transfer (from microcontroller to asic), the MISO signal transfers a diagnostic word: the first 13 bits of this word are constant (“1010101010101”) and are used to detect short-circuits on the MISO line. The last three bits copy the three LSBs of the spi\_error register.

After an error code writes in this register, the register becomes write-protected to latch the error condition and is blind to other occurring errors. This is because after one error, others are often generated: for example for an incomplete word, which can cause an incorrect interpretation of a command word and lead to a frame incomplete error.

By latching only the first occurring error, it is possible to read the cause of the failure in the communication, without having to see all the side effects. Furthermore, the PT2000 has the possibility to generate an interrupt request toward the microcontroller by setting the appropriate bit in the spi\_config register (refer to [Table 153, Spi\\_config \(1A9h\)](#)).

When an error is met in the SPI connection, the SPI protocol inside the PT2000 moves to a state where it only accepts a specific two word SPI transmission with command word (B661h) meant to read the spi\_error register (refer to [Table 163, Spi\\_error \(1B3h\)](#)). This command causes the device to transmit a single word to the SPI master and then reset the protocol (along with the interrupt if enabled). If the value of the selection register was set to the value 0100h (selecting the generic configuration registers) before the error, the word transmitted is the error code and the error code register is reset immediately, otherwise a random word is sent and the error state of the SPI protocol is reset. In this second case, the error code register can be read (and thus reset) in a following burst. The following are the possible kind of errors and their relative codes (during correct operations the value of the register is 0000h).

- cksys\_missing: this error is set if a SPI transfer is required (which means if the SPI chip select CSB is pulled low) while the cksys clock is missing.
- frame\_error: this error is set if the number of data words in a burst is not the expected number programmed in the command word:
  - Mode A is selected, the slave\_protocol block received a control word specifying n word transfers, but the microcontroller performs less operations and then ends the communication. In this case, this module provides a watchdog function if during a programmed transfer, the communication with the microcontroller is inactive for a time longer than a prefixed limit, so the transfer is considered aborted and an error is detected.
  - Mode B is selected, the number parameter is not zero in the command word and the number of transferred words is different from the one programmed in the command word.
  - A frame error can also occur when the access limitations to DRAM in dual sequencer mode at maximum ck are violated (refer to [Table 49](#))
- word\_error: during the transfer of a word long data, the device receives or sends an incorrect number of bits (15 or 17 instead of 16, for example). If multiple words are being transferred in a row with the chip select always active (the fastest way), the error is detected at the end of the sequence and it is not possible to say which is the incorrect word. To be sure of the incorrect data, the chip select must be deactivated and reactivated between each word transfer.

### 8.3.4.18 Interruption status registers

These two registers latch the status of all the interrupt requests toward the external microcontroller. They also latch the halt signal generated by the automatic diagnostics toward the six microcores.

**Table 164. Interrupt\_Register\_Part1 (1B4h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		irq_uc1_ch3	irq_uc0_ch3	irq_uc1_ch2	irq_uc0_ch2	irq_uc1_ch1	irq_uc0_ch1	Reserved		halt_uc1_ch3	halt_uc0_ch3	halt_uc1_ch2	halt_uc0_ch2	halt_uc1_ch1	halt_uc0_ch1
R/W	-		r	r	r	r	r	r	-		r	r	r	r	r	r
Lock	no		no	no	no	no	no	no	no		no	no	no	no	no	no
Reset	00		0	0	0	0	0	0	00		0	0	0	0	0	0

**Table 165. Interrupt\_Register\_Part2 (1B5h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		checks_u <sub>m</sub> _ch3	checks_u <sub>m</sub> _ch2	checks_u <sub>m</sub> _ch1	cksys_mis <sub>s</sub> ing	SPI_irq	drv_irq	Reserved							
R/W	-		r	r	r	r	r	r	-							
Lock	no		no	no	no	no	no	no	no							
Reset	00		0	0	0	0	0	0	00000000							



**Table 166. Interrupt register bit description**

Bit name	Function
halt uc0_ch1	'1' if the automatic diagnostics has detected a short-circuit on uc0_ch1
halt uc1_ch1	'1' if the automatic diagnostics has detected a short-circuit on uc1_ch1
halt uc0_ch2	'1' if the automatic diagnostics has detected a short-circuit on uc0_ch2
halt uc1_ch2	'1' if the automatic diagnostics has detected a short-circuit on uc1_ch2
halt uc0_ch3	'1' if the automatic diagnostics has detected a short-circuit on uc0_ch3
halt uc1_ch3	'1' if the automatic diagnostics has detected a short-circuit on uc1_ch3
irq_uc0_ch1	'1' if the microcode of uc0_ch1 has asserted its interrupt request
irq_uc1_ch1	'1' if the microcode of uc1_ch1 has asserted its interrupt request
irq_uc0_ch2	'1' if the microcode of uc0_ch2 has asserted its interrupt request
irq_uc1_ch2	'1' if the microcode of uc1_ch2 has asserted its interrupt request
irq_uc0_ch3	'1' if the microcode of uc0_ch3 has asserted its interrupt request
irq_uc1_ch3	'1' if the microcode of uc1_ch3 has asserted its interrupt request
Drv_irq	'1' if the driver status block has disabled the output drivers
SPI_irq	'1' if the SPI interface has detected an error on the SPI communication
cksys_missing	'1' if the clock monitor has detected a cksys missing condition
Checksum_ch1	'1' if the checksum of the code RAM of ch1 is wrong
Checksum_ch2	'1' if the checksum of the code RAM of ch2 is wrong
Checksum_ch3	'1' if the checksum of the code RAM of ch3 is wrong

### 8.3.4.19 Interrupt subsystem overview

Figure 33 gives an overview over the handling and configuration of the different interrupt sources of the device.

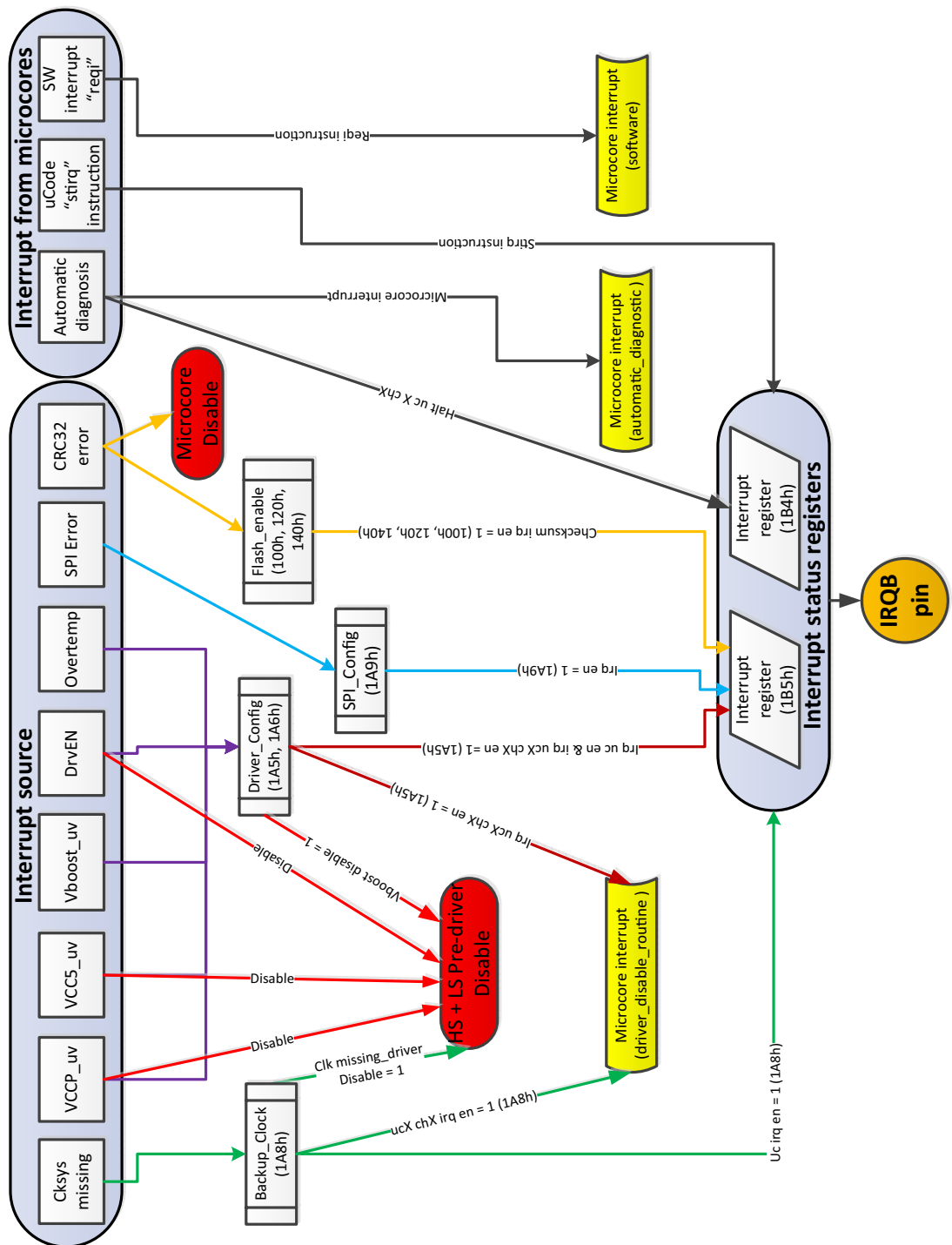


Figure 33. Interrupt subsystem

### 8.3.4.20 Device identifier register

**Table 167. Device\_Identifier (1B6h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Device_id								Mask_id				Sw_id			
R/W	r								r				r			
Lock	-								-				-			
Reset	10101110								0011				0010			

This register provide a device identifier for the component. The three fields are:

- Device id is a constant. It identifies the PT2000 device.
- Mask id is a version number of the mask set used for the device. Different mask sets must have a different mask id.
- Sw id is a version number related to the mask set.

### 8.3.4.21 Reset source status register

This 3-bit register identifies which resets were asserted since the last time this register was read. Each time this register is accessed, it is also reset.

**Table 168. Reset\_Source (1B7h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved												SPI reset	Po resetb	resetb	
R/W	-												r	r	r	
Lock	-												-	-	-	
Reseton read	-												yes	yes	yes	
Reset	00000000000000												*	*	*	

**Table 169. Reset Source Register Bits**

Bit Name	function
SPI reset	'1' if the global SPI reset was asserted since the last time this register was read
Poresetb	'1' if the power on reset was asserted since the last time this register was read
resetb	'1' if the reset pin was asserted since the last time this register was read. After POResetB this bit is in an unknown state.

### 8.3.4.22 BIST configuration register

The BIST register is used in write mode to trigger the BIST execution.

**Table 170. Bist\_interface in write mode (1BDh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BIST activation password															
R/W	w															
Lock	no															
Reset	-															

The BIST passwords are:

- **B157h**: MBIST
- **0666h**: LBIST
- **C1A0**: Clear LBIST (need to be sent after LBIST is done)

Note that after a LBIST is done a **clear BIST command** needs to be sent to reenale the logic.

In Read mode the register shows the BIST result.

**Table 171. Bist\_interface in read mode (1BDh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved												LBIST_result	MBIST_result		
R/W	-												r	r		
Lock	-												no	no		
Reset	000000000000												00	00		

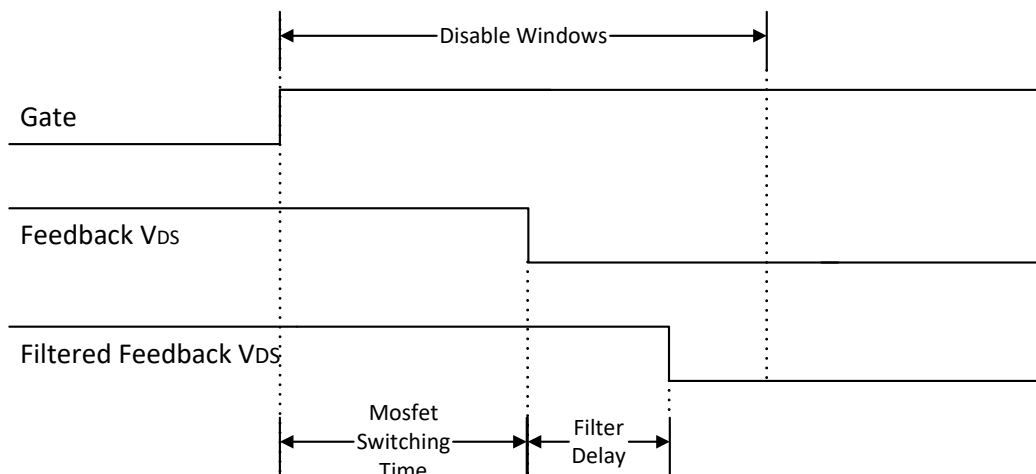
- MBIST result: set to “00” if the memory BIST was never requested
- MBIST result: set to “01” if the memory BIST operation is in progress
- MBIST result: set to “10” if the memory BIST operation was successfully completed
- MBIST result: set to “11” if the memory BIST operation has failed
- LBIST result: set to “00” if the logic BIST was never requested or a clear command has been sent
- LBIST result: set to “10” if the logic BIST operation was successfully completed
- LBIST result: set to “11” if the logic BIST operation has failed
- LBIST result: set to “01” if the logic BIST was stopped by Flag0

## 8.3.5 Diagnostics configuration registers

### 8.3.5.1 Automatic diagnostics reaction time

If the disable window is exceeded and the automatic diagnostics detects an error between the HSx\_in and the filtered HSx\_Vds\_fbk signal, an interrupt is generated toward the microcore. This interrupts the program counter of the microcore and sets to the first instruction of the error routine (refer to [Table 70, Diag\\_routine\\_addr \(10Ch, 12Ch, 14Ch\)](#)).

It takes four clock cycles (666 ns at 6.0 MHz) until the execution of the first microcode operation of the error routine is completed. This means if the first microcode command is used to switch off all pre-drivers, this action is delayed by four clock cycles. In more detail, it takes one clock cycle to detect the error, one clock cycle to generate the interrupt, one clock cycle to move the program counter to the error routine, and one clock cycle to execute the first instruction.



**Figure 34. Example of V<sub>DS</sub> automatic diagnostics filtering and disable windows**

### 8.3.5.2 LS1 to LS6 output/filter/diag configuration register

These registers define the automatic diagnostics parameter and output routing option from the low-side 1-6 output.

**Table 172. LSx\_diag\_config1 (1C0h, 1C3h, 1C6h, 1C9h, 1CCh, 1CFh, 1D2h, 1D5h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		Filter_type	Filter_length						Disable_window						
R/W	-		r/w	r/w						r/w						
Lock	-		yes	yes						yes						
Reset	00		0	000000						0000000						

**Table 173. LSx\_diag\_config2 (1C1h, 1C4h, 1C7h, 1CAh, 1CDh, 1D0h, 1D3h, 1D6h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved												Error_table			
R/W	-												r/w			
Lock	-												yes			
Reset	000000000000												0000			

**Table 174. LSx\_output\_config (1C2h, 1C5h, 1C8h, 1CBh, 1CEh, 1D1h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LSx_ovr	Reserved										output_routing				inv
R/W	r/w	-										r/w				r/w
Lock	yes	-										yes				yes
Reset	0	0000000000										11111				0

- Filter\_type. This 1-bit parameter selects the type of filter used:
  - if 0 - Any different sample resets the filter counter
  - if 1 - Any different sample decreases the filter counter
- Filter\_length. This 6 bits parameter set the filtering time for the input feedback signal.  
 $t_{FTN} = t_{CK} \times (\text{Filter\_length} + 1)$
- Error\_table. This 4-bit parameter defines the logical value of an error signal, starting from the output and the related  $V_{DS}$  feedback signal. This table defines the output of the coherency check between the driven output and the acquired feedback. A logic 1 value means there is no coherency in the check and an error signal towards the micro-microcore should be generated.

	output_command = 0 (pre-driver switched off)	output_command = 1 (pre-driver switched on)
LSx_Vds_feed = 0 ( $V_{DS}$ below threshold)	error_table(0)	error_table (2)
LSx_Vds_feed = 1 ( $V_{DS}$ above threshold)	error_table (1)	error_table (3)

- Disable\_window. This 7-bit parameter configures a time period during which any check on the LSx\_Vds\_feed signal is disabled after any change on the output\_command signal.  
 $t_{DTL} = t_{CK} \times (\text{Disable\_window} + 4)$
- Output\_routing. This 4-bit parameter defines if the LSx output is controlled by the microcores or by an input flag pin. When an input flag pin is selected, the signal from the flag pin and the control signal from the microcores are combined by a logic OR. When a flag pin is selected to drive the output, it is possible to control low-sides without programming the microcore.

output_routing	flag <sup>(54)</sup>	output_command
0	Flag0	driven from flag0
1	Flag1	driven from flag1
2	Flag2	driven from flag2
3	Flag3	driven from flag3
4	Vsense4	driven from flag4
5	Start1	driven from flag5
6	Start2	driven from flag6
7	Start3	driven from flag7
8	Start4	driven from flag8
9	Start5	driven from flag9
10	Start6	driven from flag10
11	Start7	driven from flag11
12	Start8	driven from flag12
13	Irq	driven from flag13
14	OA2	driven from flag14
15	Dbg	driven from flag15
16-31		driven from the microcores

Notes

54. Configuration is linked to the value of flag source register (see [Table 145](#)).

- Invert: This parameter inverts the polarity of the LSx output signal, with respect to the polarity defined by the microcore. This affects the output command toward the pre-drivers, but the error\_table of the associated feedback is not affected since diagnostics already takes into account the pre-driver status (even when the invert bit is set). The invert bit doesn't affect the polarity of the pre-driver when it is driven from a flag pin.
  - LSx\_ovr: if set to '1', the low-side x output driver is not influenced by the drv\_en.
- This bit is only writeable for LS6. For all the other pre-drivers, this feature is not available. The drv\_en path is always active (hard wired).

### 8.3.5.3 LS7 and LS8 output/filter/diag configuration register

For LS7 and LS8, the LSx\_diag\_config1/2 registers have the same layout as for LS1-6 (refer to [Table 172](#), [LSx\\_diag\\_config1 \(1C0h, 1C3h, 1C6h, 1C9h, 1CCh, 1CFh, 1D2h, 1D5h\)](#) and [Table 173](#), [LSx\\_diag\\_config2 \(1C1h, 1C4h, 1C7h, 1CAh, 1CDh, 1D0h, 1D3h, 1D6h\)](#). [Table 175](#), [LS7\\_output\\_config \(1D4h\)](#) & [LS8\\_output\\_config \(1D7h\)](#) shows the layout of the LS7/8\_output\_config register.

**Table 175. LS7\_output\_config (1D4h) & LS8\_output\_config (1D7h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LSx_ovr	Reserved									output_routing				inv	
R/W	r/w	-									r/w				r/w	
Lock	yes	-									yes				yes	
Reset	0	00000000									11111				0	

- Output\_routing: This 4-bit parameter defines if the LSx output is controlled by the microcores or by an input flag pin. When an input flag pin is selected, the signal from the flag pin and the control signal from the microcores are combined by a logic OR. When a flag pin is selected to drive the output, it is possible to control low-sides without programming the microcore.

output_routing	flag <sup>(55)</sup>	output_command
0	Flag0	driven from flag0
1	Flag1	driven from flag1
2	Flag2	driven from flag2
3	Flag3	driven from flag3
4	Vsense4	driven from flag4
5	Start1	driven from flag5
6	Start2	driven from flag6
7	Start3	driven from flag7
8	Start4	driven from flag8
9	Start5	driven from flag9
10	Start6	driven from flag10
11	Start7	driven from flag11
12	Start8	driven from flag12
13	Irq	driven from flag13
14	OA2	driven from flag14
15	Dbg	driven from flag15
16-31		driven from the microcores

Notes

55. Configuration is linked to the value of flag source register (see [Table 145](#)).

- Invert: This parameter inverts the polarity of the LSx output signal, with respect to the polarity defined by the microcore. The invert bit doesn't affect the polarity of the pre-driver when it is driven from a flag pin.
- LSx\_ovr: if set to '1', the low-side x output driver is not influenced by the drv\_en.

### 8.3.5.4 HSx output/filter/diag configuration register

These registers define the automatic diagnostics parameter and output routing option fro the high-side X output.

**Table 176. HSx\_diag\_config1 (1D8h, 1DBh, 1DEh, 1E0h, 1E4h, 1E7h, 1EAh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved		Filter_t ype	Filter_length						Disable_window						
R/W	-		r/w	r/w						r/w						
Lock	-		yes	yes						yes						
Reset	00		0	000000						0000000						

**Table 177. HSx\_diag\_config2 (1D9h, 1DCh, 1DFh, 1E2h, 1E5h, 1E8h, 1EBh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved								Error_table_src				Error_table_vds			
R/W	-								r/w				r/w			
Lock	-								yes				yes			
Reset	00000000								0000				0000			

**Table 178. HSx\_output\_config (1DA, 1DDh, 1E0h, 1E3h, 1E6h, 1E9h)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HSx ovr	reserved				dead_time					output_routing				inv	
R/W	r/w	-				r/w					r/w				r/w	
Lock	yes	-				yes					yes				yes	
Reset	0	000				00000					11111				0	

- **Error\_table\_vds:** This 4-bit parameter defines the logical value of an error signal, starting from the output and the related  $V_{DS}$  feedback signal. This table defines the output of the coherency check between the driven output and the acquired feedback. A logic 1 value means there is no coherency in the check and then an error signal towards the micro-microcore should be generated.

	<b>output_command = 0 (Pre-driver switched off)</b>	<b>output_command = 1 (Pre-driver switched on)</b>
HSx_Vds_feed = 0 ( $V_{DS}$ below threshold)	error_table(0)	error_table (2)
HSx_Vds_feed = 1 ( $V_{DS}$ above threshold)	error_table (1)	error_table (3)

- **Disable\_window:** This 7-bit parameter configures a time period during which any check on the HSx\_Vds\_feed and HSx\_Vsrc\_feed signals is disabled after any change on the output\_command signal.  
 $t_{DTL} = t_{CK} \times (\text{Disable\_window} + 4)$
- **Error\_table\_src:** This 4-bit parameter defines the logical value of an error signal, starting from the output and the related  $V_{SRC}$  feedback signal. This table defines the output of the coherency check between the driven output and the acquired feedback. A logic 1 value means there is no coherency in the check and then an error signal towards the micro-microcore should be generated.

	<b>output_command = 0 (pre-driver switched off)</b>	<b>output_command = 1 (pre-driver switched on)</b>
HSx_Vsrc_feed = 0 ( $V_{SRC}$ below threshold)	error_table(0)	error_table (2)
HSx_Vsrc_feed = 1 ( $V_{SRC}$ above threshold)	error_table (1)	error_table (3)

- **Filter\_type.** This 1 bit parameter selects the type of filter used:
  - if 0 - Any different sample resets the filter counter
  - if 1 - Any different sample decreases the filter counter
- **Dead\_time:** This 5-bit register is used to store the value of the dead\_time end of count used in the generation of the free wheeling output (delay between the high-side output and the free wheeling output). The FW command goes high after a programmable time ( $t_{FWDLY}$ ) with respect to the high-side falling edge. In this mode the high-side command rising edge is always delayed of the same programmable time ( $t_{FWDLY}$ ) with respect to the rising edge requested by the microcores.  
 $t_{FWDLY} = T_{ck} \times (\text{Dead\_time} + 1)$
- **Output\_routing:** This 4-bit parameter defines if the HSx output is controlled by the microcores or by an input flag pin. When an input flag pin is selected, the signal from the flag pin and the control signal from the microcores are combined by a logic OR. When a flag pin is selected to drive the output, it is possible to control low-sides without programming the microcore.

<b>output_routing</b>	<b>flag <sup>(56)</sup></b>	<b>output_command</b>
0	Flag0	driven from flag0
1	Flag1	driven from flag1
2	Flag2	driven from flag2
3	Flag3	driven from flag3
4	Vsense4	driven from flag4
5	Start1	driven from flag5



output_routing	flag <sup>(56)</sup>	output_command
6	Start2	driven from flag6
7	Start3	driven from flag7
8	Start4	driven from flag8
9	Start5	driven from flag9
10	Start6	driven from flag10
11	Start7	driven from flag11
12	Start8	driven from flag12
13	Irq	driven from flag13
14	OA2	driven from flag14
15	Dbg	driven from flag15
16-31		driven from the microcores

Notes

56. Configuration is linked to the value of flag source register (see [Table 145](#)).

- Invert: This parameter inverts the polarity of the HSx output signal, with respect to the polarity defined by the microcore. This affects the output command towards the pre-drivers, but the error\_table of the associated feedback is not affected since diagnostics already takes into account the pre-driver status (even when the invert bit is set). The invert bit doesn't affect the polarity of the pre-driver when it is driven from a flag pin.
- Filter\_lenght. This 6-bit parameter sets the filtering time for the input feedback signal.
  - $t_{FTN} = t_{CK} \times (\text{Filter\_length} + 1)$
- HSx ovr: if set to '1', the high-side x output driver is not influenced by the drv\_en. This bit is only writeable for HS5 and HS7. For all the other pre-drivers this feature is not available, the drv\_en path is always active (hard wired).

### 8.3.5.5 Automatic diagnostics error status register

This is the status register controlled by automatic diagnostics: one for each microcore. This register stores all meaningful information whenever an error condition is detected on any of the pairs (output / feedback) to which the microcore is sensitive. The information stored in the register with regards to the output commands and the related voltage ( $V_{DS}$  and  $V_{SRC}$ ) feedback.

A cksys\_missing (PLL output clock not valid) condition does not trigger the err\_ucXchY to be latched. If the register is latched, the cksys\_missing bit shows the cksys status at the same moment when the automatic diagnostics error occurred. The cksys\_missing bit sets when the PLL output clock was not valid at the time the automatic diagnostics error occurred. The registers can be configured as reset on read. Refer also to section [Table 158, Reset\\_Behavior \(1AEh\)](#). Those three registers are reset when the err\_ucXchY\_3 register is read. Note that automatic diagnostics are enabled using the endiag or endiaga instructions (reference [Programming Guide and Instruction Set](#)).

**Table 179. Err\_ucxchy\_part1 (1EDh, 1F0h, 1F3h, 1F6h, 1F9h, 1FCh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserv ed	cmd_H s5	Vsrc_H s5	Vds_H s5	cmd_H s4	Vsrc_H s4	Vds_H s4	cmd_H s3	Vsrc_H s3	Vds_H s3	cmd_H s2	Vsrc_H s2	Vds_H s2	cmd_H s1	Vsrc_H s1	Vds_H s1
R/W	-	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
Lock	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Reset on read	-	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Table 180. Err\_ucxchy\_part2 (1EEh, 1F1h, 1F4h, 1F7h, 1FAh, 1FDh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Name	cksys_mi ssing	Reserved										cmd_ Hs7	Vsrc_ Hs7	Vds_ Hs7	cmd_ Hs6	Vsrc_ Hs6	Vds_ Hs6
R/W	r											r	r	r	r	r	r
Lock	-											-	-	-	-	-	-
Reset on read	conf.											conf.	conf.	conf.	conf.	conf.	conf.
Reset	0	000000000										0	0	0	0	0	0

**Table 181. Err\_ucxchy\_part3 (1EFh, 1F2h, 1F5h, 1F8h, 1FBh, 1FEh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	cmd_L S8	Vds_Ls 8	cmd_L S7	Vds_Ls 7	cmd_L s6	Vds_Ls 6	cmd_L s5	Vds_Ls 5	cmd_L s4	Vds_Ls 4	cmd_L s3	Vds_Ls 3	cmd_L s2	Vds_Ls 2	cmd_L s1	Vds_Ls 1
R/W	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
Lock	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Reset on read	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.	conf.
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

### 8.3.5.6 Automatic diagnostics command/feedback coherency

The PT2000 performs automatic diagnostics by checking the coherency between the commands it sends to the MOSFETs and the  $V_{DS}/V_{SRC}$  feedback it gets. The following register configure if the check is between:

- $diag\_option = 0 \Rightarrow$  coherency check between what the microcore wants to drive, using instructions like `sto/stos/ldca/ldcd` and the feedback from the MOSFETS
- $diag\_option = 1 \Rightarrow$  coherency check between what the device is really driving, using instructions like `sto/stos/ldca/ldcd`, but also including the status of overtemperature/drven pin/undervoltages/etc. and the feedback from the MOSFETS

For example, in case of an overtemperature/drven pin/undervoltages/etc., drivers are disabled:

In case of  $diag\_option = 0 \Rightarrow$ , automatic diagnosis coherency check fails and detects an error (microcores are trying to drive but the output does not move due to a disabled driver,  $V_{DS}/V_{SRC}$  feedback incoherent with driving request  $\Rightarrow$  fail)

In case of  $diag\_option = 1 \Rightarrow$ , automatic diagnosis coherency check does not detect anything (microcores are trying to drive, but the output does not move due to a disabled driver,  $V_{DS}/V_{SRC}$  feedback is in this case first compared to “no driving”  $\Rightarrow$  ok)

To summarize if the  $diag\_option = 0$  then automatic diagnostics is able to cover all kind of faults.

**Table 182. diagnostics\_option register (1FFh)**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Reserved															Diag_opt ion
R/W																r/w
Lock																yes
Reset	000000000000000															1

# 9 Typical applications

The PT2000 can be configured in several applications. [Figure 35](#) and [Figure 36](#) shows the PT2000 in a typical application.

## 9.1 Application diagram: 3 bank, 6 cylinder with DC/DC

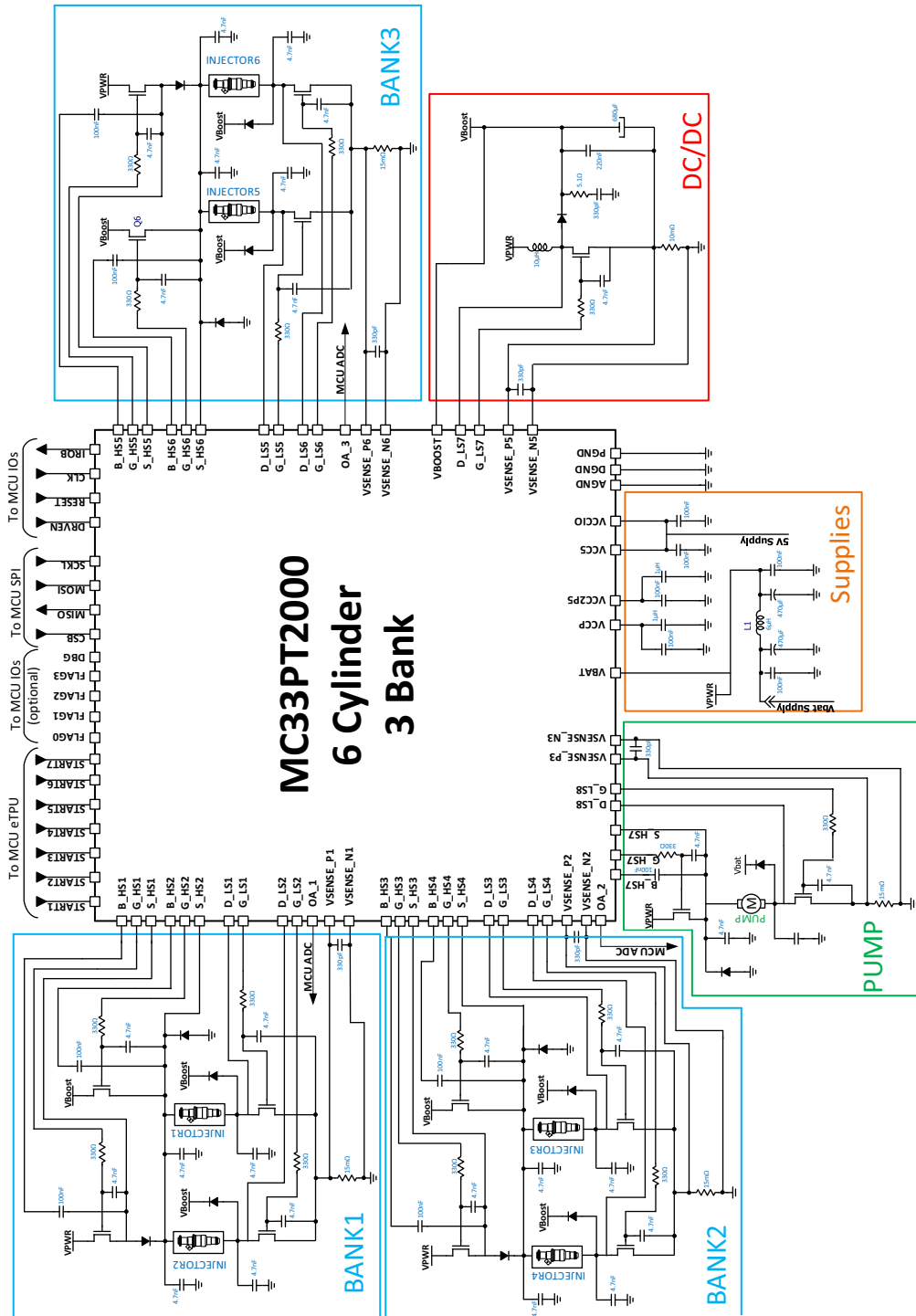


Figure 35. Example of application circuit (6 cylinder 3 bank with DC/DC)

# 9.2 Application diagram: 3 bank, 3 cylinder (full overlap) with DC/DC

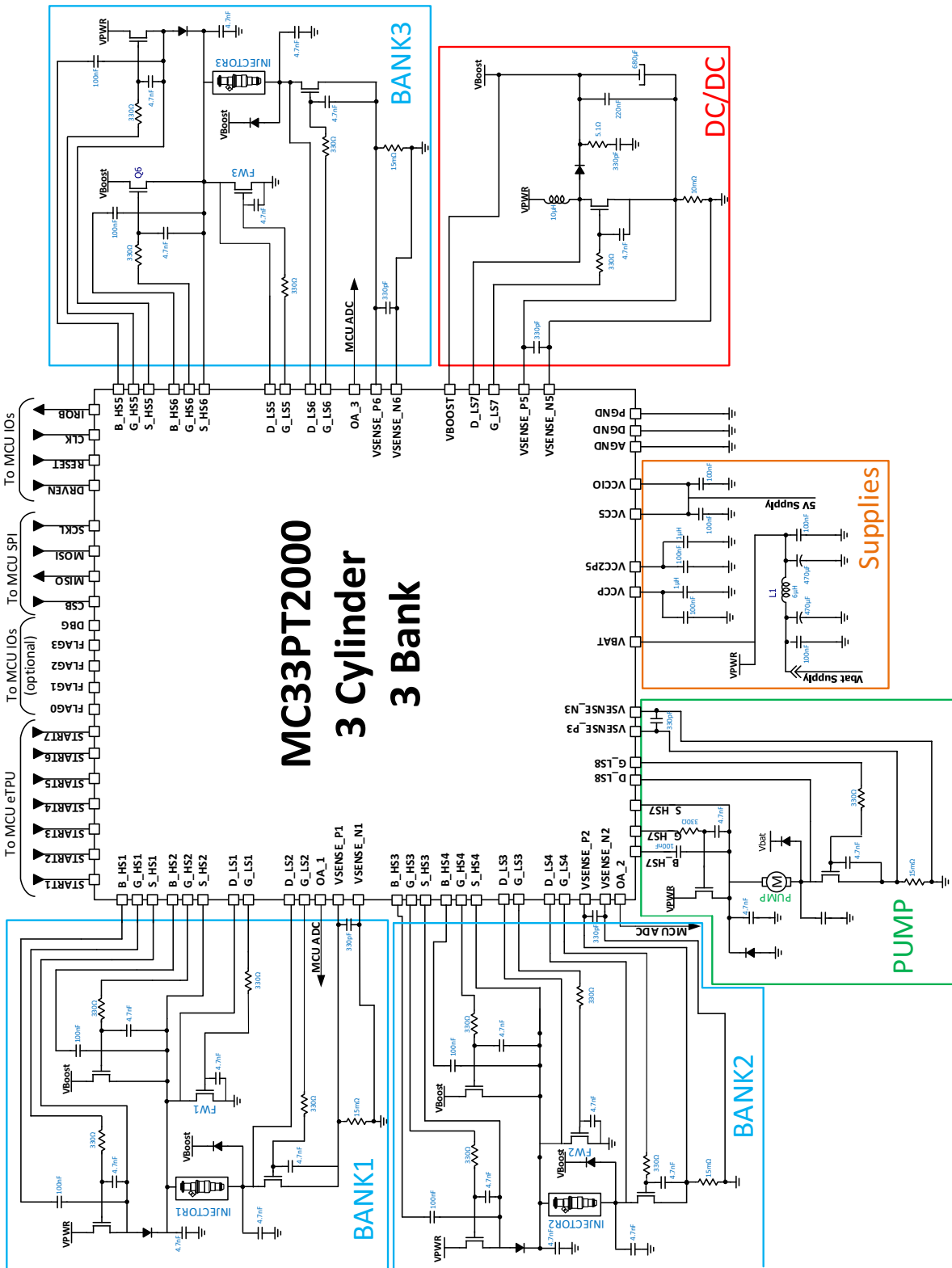


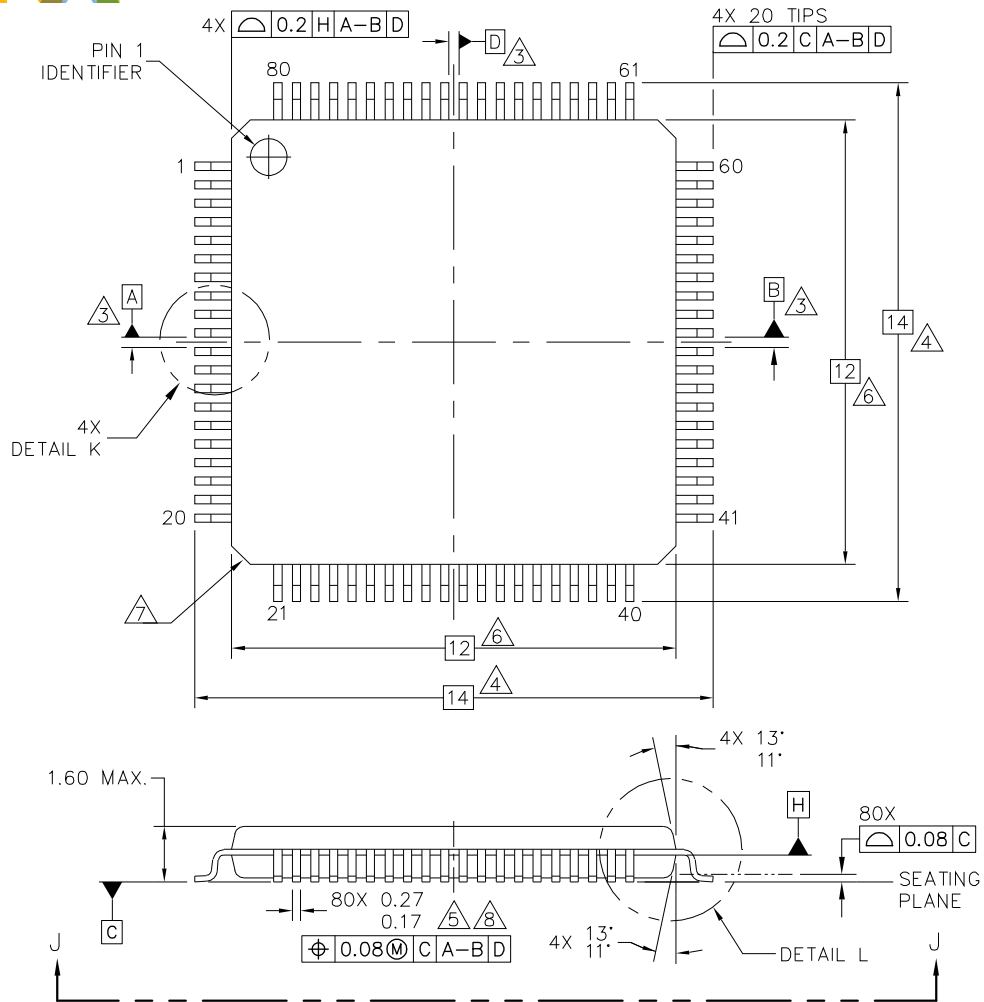
Figure 36. Application diagram 3 bank, 3 cylinder with synchronous rectification and DC/DC

# 10 Packaging

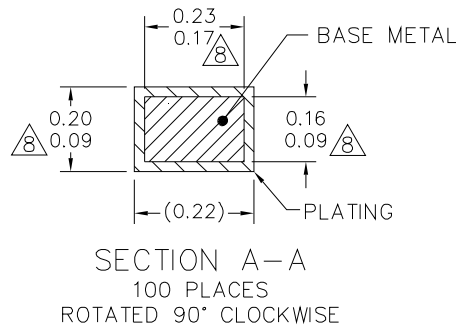
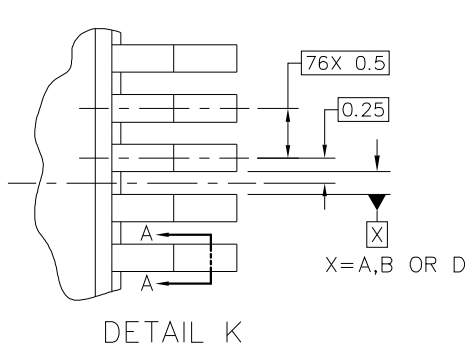
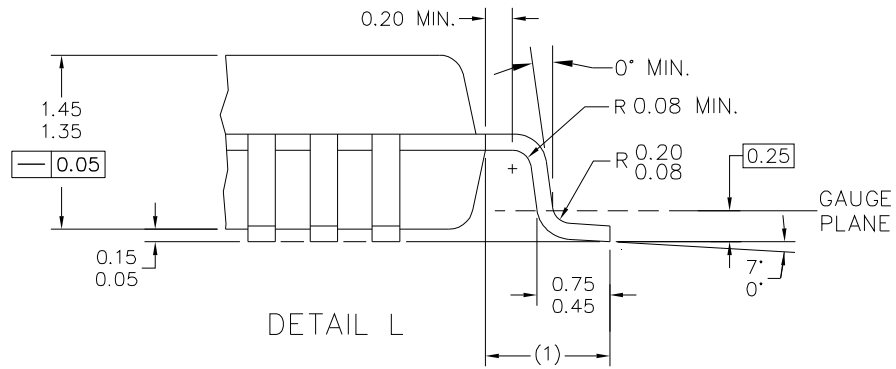
## 10.1 Package mechanical dimensions

Package dimensions are provided in package drawings. To find the most current package outline drawing, go to [www.nxp.com](http://www.nxp.com) and perform a keyword search for the drawing's document number.

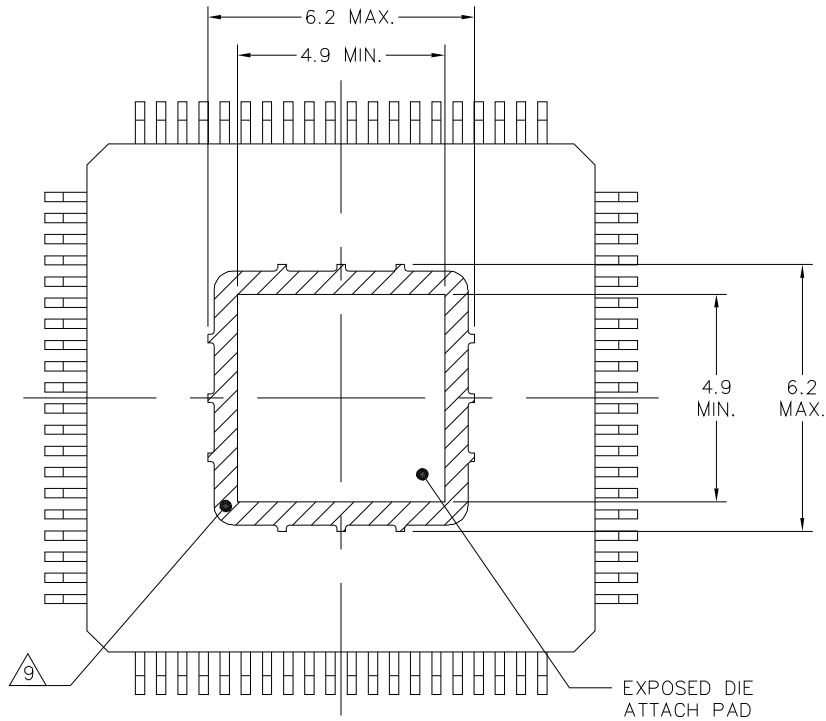
Package	Suffix	Package outline drawing number
80-Pin LQFP	AF	98ASA00505D



© NXP SEMICONDUCTORS N.V. ALL RIGHTS RESERVED	MECHANICAL OUTLINE	PRINT VERSION NOT TO SCALE
TITLE: LQFP, 12 X 12 X 1.4 PKG, 0.5 PITCH, 80 I/O, 5.6 X 5.6 EXPOSED PAD	DOCUMENT NO: 98ASA00505D	REV: A
	STANDARD: NON-JEDEC	
	SOT1572-1	11 JAN 2016



© NXP SEMICONDUCTORS N.V. ALL RIGHTS RESERVED	<b>MECHANICAL OUTLINE</b>	PRINT VERSION NOT TO SCALE
TITLE: LQFP, 12 X 12 X 1.4 PKG, 0.5 PITCH, 80 I/O, 5.6 X 5.6 EXPOSED PAD	DOCUMENT NO: 98ASA00505D	REV: A
	STANDARD: NON-JEDEC	
	SOT1572-1	11 JAN 2016



VIEW J-J

© NXP SEMICONDUCTORS N.V. ALL RIGHTS RESERVED	<b>MECHANICAL OUTLINE</b>	PRINT VERSION NOT TO SCALE	
TITLE: LQFP, 12 X 12 X 1.4 PKG, 0.5 PITCH, 80 I/O, 5.6 X 5.6 EXPOSED PAD		DOCUMENT NO: 98ASA00505D	REV: A
		STANDARD: NON-JEDEC	
		SOT1572-1	11 JAN 2016



NOTES:

1. DIMENSIONS ARE IN MILLIMETERS.
2. INTERPRET DIMENSIONS AND TOLERANCES PER ASME Y14.5M-1994.
3. DATUMS A, B AND D TO BE DETERMINED AT DATUM PLANE H.
4. DIMENSIONS TO BE DETERMINED AT SEATING PLANE C.
5. DIMENSION DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE LEAD WIDTH TO EXCEED THE MAXIMUM DIMENSION BY MORE THAN 0.08 MM. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSION AND ADJACENT LEAD OR PROTRUSION 0.07 MM.
6. DIMENSIONS DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 MM PER SIDE. DIMENSIONS ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.
7. EXACT SHAPE OF EACH CORNER IS OPTIONAL.
8. THESE DIMENSIONS APPLY TO THE FLAT SECTION OF THE LEAD BETWEEN 0.10 MM AND 0.25 MM FROM THE LEAD TIP.
9. HATCHED AREA TO BE KEEP-OUT ZONE FOR PCB ROUTING.

© NXP SEMICONDUCTORS N.V. ALL RIGHTS RESERVED	<b>MECHANICAL OUTLINE</b>	PRINT VERSION NOT TO SCALE	
TITLE: LQFP, 12 X 12 X 1.4 PKG, 0.5 PITCH, 80 I/O, 5.6 X 5.6 EXPOSED PAD	DOCUMENT NO: 98ASA00505D		REV: A
	STANDARD: NON-JEDEC		
	SOT1572-1		11 JAN 2016



# 11 Reference section

**Table 183. PT2000 reference**

Description	URL
Programming guide and instruction set	<a href="http://www.nxp.com/files/analog/doc/user_guide/PT2000SWUG.pdf">http://www.nxp.com/files/analog/doc/user_guide/PT2000SWUG.pdf</a>
Product summary page	<a href="http://www.nxp.com/PT2000">http://www.nxp.com/PT2000</a>

# 12 Revision history

Revision	Date	Description of changes
1.0	3/2015	<ul style="list-style-type: none"> <li>Initial release</li> </ul>
2.0	3/2015	<ul style="list-style-type: none"> <li>Made minor corrections</li> </ul>
	4/2015	<ul style="list-style-type: none"> <li>Corrected typo error on High-side <math>V_{DS}</math> Threshold in Table 8</li> <li>Corrected typo error on Low-side <math>V_{DS}</math> Threshold in Table 9</li> <li>Changed Current Measurement for DC/DC title to heading 3</li> <li>Updated reset value in Table 122 (low-side pull-down disable by default)</li> <li>Updated figure 35 and 36 to use current sense measurement 6 for Bank 3, recommended in order to use OA3 for Bank3</li> </ul>
3.0	4/2015	<ul style="list-style-type: none"> <li>Added <a href="#">Shutoff path via the DrvEn pin</a> section.</li> <li><math>I_{VBATT\_OPER}</math> value updated to fit with maximum current allowed on VCCP in <a href="#">Table 5</a></li> <li><math>I_{VCC5}</math> updated with 6 microcores ON and biasing enabled in <a href="#">Table 5</a></li> <li>Total Error <math>\epsilon V_{BOOST\_DAC}</math> updated in <a href="#">Table 5</a></li> <li>OAx Input impedance updated in <a href="#">Table 15</a></li> <li>LBIST clear command added in <a href="#">LBIST</a></li> </ul>
	6/2015	<ul style="list-style-type: none"> <li>Corrected <a href="#">Figure 20</a></li> <li>Added description detail to <a href="#">section 7.2.1.3</a></li> </ul>
4.0	11/2015	<ul style="list-style-type: none"> <li>Updated <math>I_{VCC5}</math> characteristic in <a href="#">Table 5</a></li> <li>Added <a href="#">Table 50</a></li> <li>Added note <sup>(18)</sup></li> <li>Updated <a href="#">section 6.1.4.1</a></li> <li>Updated <a href="#">section 8.2.3.2</a></li> <li>Updated <a href="#">section 8.3.3.12</a></li> <li>Updated the example in <a href="#">section 8.3.2.17</a></li> </ul>
	12/2015	<ul style="list-style-type: none"> <li>Corrected package suffix</li> <li>Added the <math>VCS\_OFF\_GD</math> parameter</li> <li>Corrected <a href="#">Table 167</a></li> <li>Corrected table titles for <a href="#">Table 172</a>, <a href="#">Table 173</a>, <a href="#">Table 174</a>, <a href="#">Table 176</a>, <a href="#">Table 177</a>, <a href="#">Table 178</a>, <a href="#">Table 179</a>, <a href="#">Table 180</a>, and <a href="#">Table 181</a></li> </ul>
5.0	4/2016	<ul style="list-style-type: none"> <li>Updated the formula for calculating current threshold</li> <li>Update <a href="#">Figure 33</a></li> <li>Updated package drawing</li> <li>Updated form and style</li> </ul>
6.0	6/2016	<ul style="list-style-type: none"> <li>Added SPI timings to <a href="#">Table 18</a></li> <li>added clarification to <a href="#">Section 7.1.1. Power-up sequence of VCC5, VCC2P5, and reset, page 57</a></li> <li>Updated <a href="#">Figure 21</a></li> </ul>
7.0	9/2016	<ul style="list-style-type: none"> <li>Added <a href="#">Table 3. Resistor types</a></li> <li>Added Pull resistor type column to <a href="#">Table 2. PT2000 pin definitions (2), (3), (4)</a></li> <li>Added notes <sup>(3)</sup> and <sup>(4)</sup>.</li> <li>Made minor correction to <a href="#">Section 7.2.1.3. Mode 3 (LS7/8 resonant mode VDS monitoring), page 62</a></li> </ul>
8.0	4/2017	<ul style="list-style-type: none"> <li>Updated <a href="#">Figure 3</a> (replaced VSENSEN8 by VSENSEN2)</li> <li>Updated min. value for <math>V_{BATT}</math> and <math>S_{RVCC5}</math> in <a href="#">Table 5</a></li> <li>Updated typical value for <math>I_{VCCIO}</math> in <a href="#">Table 5</a></li> <li>Added note <sup>(16)</sup> and <sup>(52)</sup></li> <li>Made minor corrections to <a href="#">Application without boost voltage</a>, <a href="#">Low-side VDS monitor D_Is7/D_Is8 for DC/DC</a>, <a href="#">VDS low-side 7/8 configuration register</a>, and <a href="#">DAC settling time register</a></li> <li>Updated <a href="#">Table 145</a></li> </ul>
9.0	4/2017	<ul style="list-style-type: none"> <li>Deleted <math>S_{RVCC5}</math> characteristic from <a href="#">Table 5</a></li> </ul>

Revision	Date	Description of changes
10.0	9/2017	<ul style="list-style-type: none"> <li>Updated <a href="#">Figure 1</a></li> <li>Updated <a href="#">Table 126</a> (changed all reset values to 1)</li> <li>Updated <a href="#">Table 152</a></li> </ul>
11.0	6/2018	<ul style="list-style-type: none"> <li>Updated <a href="#">Table 5</a> (added the <math>V_{cc5\_BGmin}</math> parameter)</li> <li>Minor typo corrections in <a href="#">Table 11</a>, <a href="#">Table 13</a> and <a href="#">Table 16</a></li> <li>Updated bit 5 and bit 13 values in <a href="#">Table 102</a>, <a href="#">Table 103</a> and <a href="#">Table 104</a></li> <li>Updated <a href="#">Figure 33</a> (replaced 1B4h by 1B5h and 1B5h by 1B4h)</li> </ul>
12.0	1/2019	<ul style="list-style-type: none"> <li>Changed document status from Advance information to Technical data</li> </ul>

**How to Reach Us:****Home Page:**[NXP.com](http://www.nxp.com)**Web Support:**<http://www.nxp.com/support>

Information in this document is provided solely to enable system and software implementers to use NXP products.

There are no expressed or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

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

<http://www.nxp.com/terms-of-use.html>.

NXP, the NXP logo, Freescale, the Freescale logo and SMARTMOS are trademarks of NXP B.V. All other product or service names are the property of their respective owners. All rights reserved.

© NXP B.V. 2019.

Document Number: MC33PT2000

Rev. 12

1/2019

