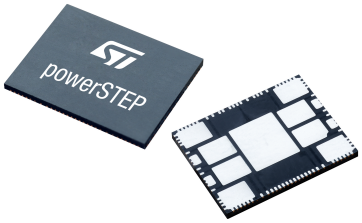
# powerSTEP01

Datasheet

## System-in-package integrating microstepping controller and 10 A power MOSFETs




**VFQFPN 11 x 14 mm**

**Product status link**

powerSTEP01

**Product label**

SUSTAINABLE TECHNOLOGY

## Features

- Power system-in-package integrating a programmable microstepping controller and 8 N-channel power MOSFETs
- Operating voltage: 7.5 V - 85 V
- Dual full-bridge with $R_{DS(on)}$ = 16 mΩ
- 10 $A_{RMS}$ maximum output current
- Adjustable output slew rate
- Programmable speed profile
- Up to 1/128 microstepping
- Sensorless stall detection
- Integrated voltage regulators
- SPI interface
- Low quiescent standby current
- Programmable non-dissipative overcurrent protection
- Overtemperature protection

## Applications

- Industrial process automation
- Textile and sewing machines
- PCB and IC assembling equipment
- Pick and place machines
- ATM and money handling machines
- Medical analysis equipment
- Antenna control
- Robotics
- CCTV, security, and dome cameras

## Description

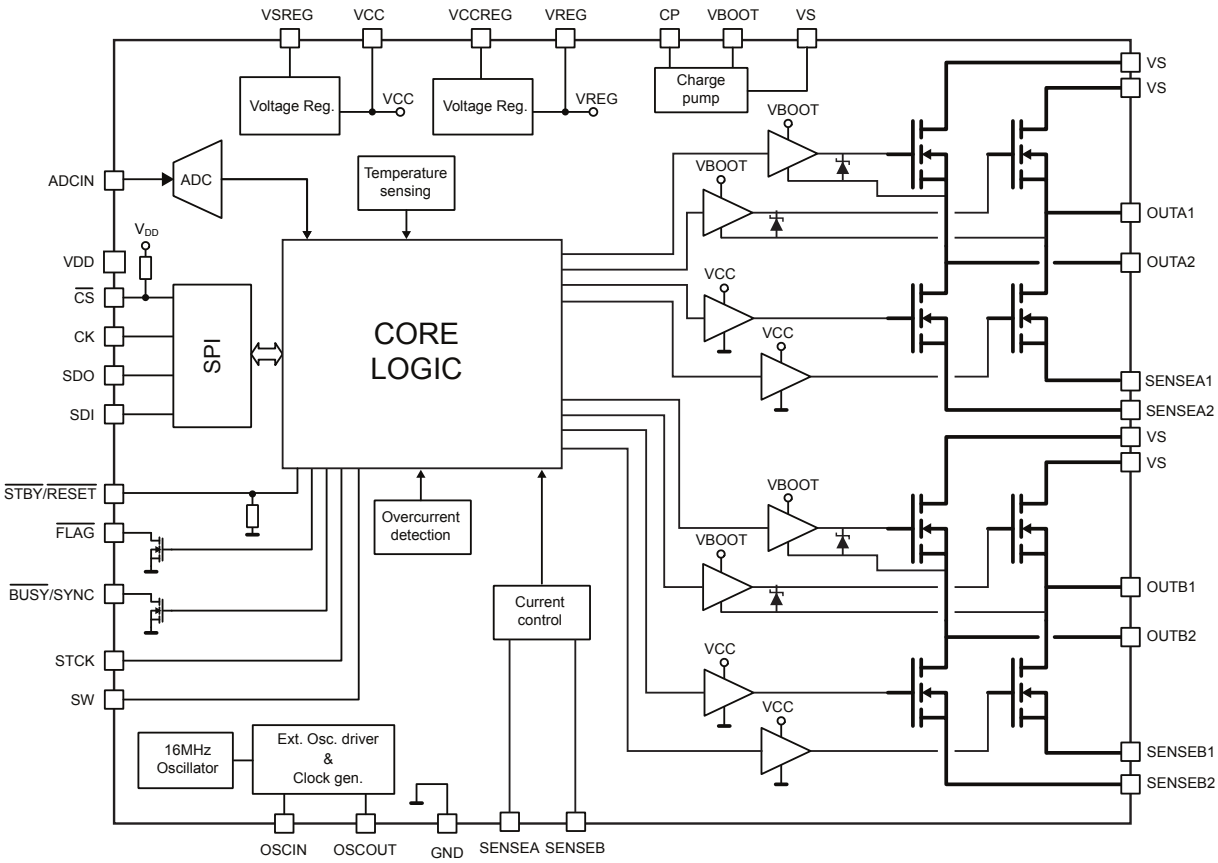
The powerSTEP01 is a system-in-package integrating 8 N-channel 16 mΩ MOSFETs for stepper applications up to 85 V with an SPI programmable controller, providing a full digital control of the motion through a speed profile generation and positioning calculations.

It integrates a dual low $R_{DS(on)}$ full-bridge with embedded non-dissipative overcurrent protection.

The device can operate with both voltage mode driving and advanced current control fitting different application needs. The digital control core can generate the user's defined motion profiles with acceleration, deceleration, speed, or a target position easily programmed through a dedicated register set. All application commands and data registers, including those used to set analog values (that is, current protection trip point, deadtime, PWM frequency, etc.) are sent through a standard 5-Mbit/s SPI. A very rich set of protections (thermal, low bus voltage, overcurrent, and motor stall) make the powerSTEP01 "bullet proof", as required by the most demanding motor control applications.

# 1 Block diagram

Figure 1. **Block diagram**



# 2 Electrical data

## 2.1 Absolute maximum ratings

**Table 1. Absolute maximum ratings**

| Symbol | Parameter | Test conditions | Value | Unit |
|---|---|---|---|---|
| $V_{DD}$ | Logic interface supply voltage | | 5.5 | V |
| $V_{REG}$ | Logic supply voltage | | 3.6 | V |
| $V_S$ | Motor supply voltage | | 95 | V |
| $V_{CC}$ | Low-side gate driver supply voltage | | 18 | V |
| $V_{BOOT}$ | Boot voltage | | 100 | V |
| $\Delta V_{BOOT}$ | High-side gate driver supply voltage ($V_{BOOT}$ - $V_S$) | | 0 to 20 | V |
| $V_{SREG}$ | Internal $V_{CC}$ regulator supply voltage | | 95 | V |
| $V_{CCREG}$ | Internal $V_{REG}$ regulator supply voltage | | 18 | V |
| $V_{OUT1A}$<br>$V_{OUT2A}$<br>$V_{OUT1B}$<br>$V_{OUT2B}$ | Output voltage | DC | -5 to $V_{BOOT}$ | V |
| | | AC | -15 to $V_{BOOT}$ | |
| $I_{OUT1A}$<br>$I_{OUT2A}$<br>$I_{OUT1B}$<br>$I_{OUT2B}$ | Output current | DC | 10 | $A_{RMS}$ |
| $SR_{out}$ | Full-bridge output slew rate (10% - 90%) | | 10 | V/ns |
| $V_{ADCIN}$ | Integrated ADC input voltage range (ADCIN pin) | | -0.3 to 3.6 | V |
| $V_{out\_diff}$ | Differential voltage between VBOOT, VS, OUT1A, OUT2A, PGND and VBOOT, VS, OUT1B, OUT2B, PGND pins | | 100 | V |
| $V_{in}$ | Logic input voltage range | | -0.3 to 5.5 | V |
| $T_{stg}$<br>$T_{OP}$ | Storage and operating junction temperature | | -40 to 150 | °C |

## 2.2 Recommended operating conditions

**Table 2. Recommended operating conditions**

| Symbol | Parameter | Test conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $V_{DD}$ | Logic interface supply voltage | 3.3 V logic outputs | | 3.3 | | V |
| | | 5 V logic outputs | | 5 | | |
| $V_{REG}$ | Logic supply voltage | | | 3.3 | | V |
| $V_S$ | Motor supply voltage | | $V_{SREG}$ | | 85 | V |
| $V_{SREG}$ | Internal $V_{CC}$ voltage regulator supply voltage | $V_{CC}$ voltage internally generated | $V_{CC}$ + 3 | | $V_S$ | V |
| $V_{CC, ext}$ | Gate driver supply voltage | $V_{CC}$ voltage imposed by external source ($V_{SREG}$ = $V_{CC}$) | 7.5 | | 15 | V |
| $V_{CCREG}$ | Internal $V_{REG}$ voltage regulator supply voltage | $V_{REG}$ voltage internally generated | 6.3 | | $V_{CC}$ | V |
| $V_{ADC}$ | Integrated ADC input voltage (ADCIN pin) | | 0 | | $V_{REG}$ | V |

## 2.3 ESD protection ratings

**Table 3. ESD protection ratings**

| Symbol | Parameter | Condition | Class | Value | Unit |
|---|---|---|---|---|---|
| HBM | Human body model | Conforming to ANSI/ESDA/JEDEC JS001-2012 | 2 | 2 | kV |
| CDM | Charge device model | Conforming to ANSI/ESD STM 5.3.1-2009 | C4 | 500 | V |

# 3 Electrical characteristics

**Table 4. Electrical characteristics**

$V_S$ = 48 V; $V_{CC}$ = 7.5 V; $V_{DD}$ = 3.3 V; $T_j$ = 25 °C, unless otherwise specified.

| Symbol | Parameter | Test conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| **General** | | | | | | |
| $V_{CCthOn}$ | $V_{CC}$ UVLO turn-on threshold | UVLO_VAL set high (1) | 9.9 | 10.4 | 10.9 | V |
| | | UVLO_VAL set low (1) | 6.5 | 6.9 | 7.3 | V |
| $V_{CCthOff}$ | $V_{CC}$ UVLO turn-off threshold | UVLO_VAL set high (1) | 9.5 | 10 | 10.5 | V |
| | | UVLO_VAL set low (1) | 5.9 | 6.3 | 6.7 | V |
| $\Delta V_{BOOTthOn}$ | $V_{BOOT}$ - $V_S$ UVLO turn-on threshold | UVLO_VAL set high (1) | 8.6 | 9.2 | 9.94 | V |
| | | UVLO_VAL set low (1) | 5.7 | 6 | 6.35 | V |
| $\Delta V_{BOOTthOff}$ | $V_{BOOT}$ - $V_S$ UVLO turn-off threshold | UVLO_VAL set high (1) | 8.2 | 8.8 | 9.65 | V |
| | | UVLO_VAL set low (1) | 5.3 | 5.5 | 5.9 | V |
| $V_{REGthOn}$ | $V_{REG}$ turn-on threshold | (1) | 2.8 | 3 | 3.18 | V |
| $V_{REGthOff}$ | $V_{REG}$ turn-off threshold | (1) | 2.2 | 2.4 | 2.5 | V |
| $I_{VREGqu}$ | Undervoltage $V_{REG}$ quiescent supply current | $V_{CCREG}$ = $V_{REG}$ < 2.2 V (1) | | 40 | | µA |
| $I_{VREGq}$ | Quiescent $V_{REG}$ supply current | $V_{CCREG}$ = $V_{REG}$ = 3.3 V internal oscillator selected (1) | | 3.8 | | mA |
| $I_{VSREGq}$ | Quiescent $V_{SREG}$ supply current | $V_{CCREG}$ = $V_{CC}$ = 15 V | | 6.5 | | mA |
| **Thermal protection** | | | | | | |
| $T_{j(WRN)Set}$ | Thermal warning temperature | | | 135 | | °C |
| $T_{j(WRN)Rec}$ | Thermal warning recovery temperature | | | 125 | | °C |
| $T_{j(OFF)Set}$ | Thermal bridge shutdown temperature | | | 155 | | °C |
| $T_{j(OFF)Rec}$ | Thermal bridge shutdown recovery temperature | | | 145 | | °C |
| $T_{j(SD)Set}$ | Thermal device shutdown temperature | | | 170 | | °C |
| $T_{j(SD)Rec}$ | Thermal device shutdown recovery temperature | | | 130 | | °C |
| **Charge pump** | | | | | | |
| $V_{pump}$ | Voltage swing for charge pump oscillator | | | $V_{CC}$ | | V |
| $f_{pump,min}$ | Minimum charge pump oscillator frequency (2) | | | 660 | | kHz |
| $f_{pump,max}$ | Maximum charge pump oscillator frequency (2) | | | 800 | | kHz |
| $R_{pumpHS}$ | Charge pump high-side $R_{DS(on)}$ resistance | | | 10 | | Ω |
| $R_{pumpLS}$ | Charge pump low-side $R_{DS(on)}$ resistance | | | 10 | | Ω |
| $I_{boot}$ | Average boot current | | | 2.6 | | mA |
| **Power outputs** | | | | | | |
| $R_{DS(on)}$ | High-side and low-side on-resistance | $V_{CC}$ = 15 V at 25 °C | | 16 | 21 | mΩ |
| | | At 125 °C | | 23 | | |
| $SR_{OUT}$ | Output slew rate | IGATE = 96 mA | | 980 | | V/µs |
| | | IGATE = 32 mA | | 520 | | |
| $I_{DSS}$ | Leakage current | OUT = $V_S$ | | | 0.1 | mA |
| | | OUT = GND | -0.1 | | | mA |

| Symbol | Parameter | Test conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| **Deadtime and blanking** | | | | | | |
| $t_{DT}$ | Programmable deadtime (2) | TDT = '00000' | | 125 | | ns |
| | | TDT = '11111' | | 4000 | | |
| $t_{blank}$ | Programmable blanking time (2) | TBLANK = '000' | | 125 | | ns |
| | | TBLANK = '111' | | 1000 | | |
| **Logic** | | | | | | |
| $V_{IL}$ | Low level logic input voltage | | | | 0.8 | V |
| $V_{IH}$ | High level logic input voltage | | 2 | | | V |
| $I_{IH}$ | High level logic input current | $V_{IN}$ = 5 V, $V_{DD}$ = 5 V | | | 1 | µA |
| $I_{IL}$ | Low level logic input current | $V_{IN}$ = 0 V, $V_{DD}$ = 5 V | -1 | | | µA |
| $V_{OL}$ | Low level logic output voltage (3) | $V_{DD}$ = 3.3 V, $I_{OL}$ = 4 mA | | | 0.3 | V |
| | | $V_{DD}$ = 5 V, $I_{OL}$ = 4 mA | | | 0.3 | |
| $V_{OH}$ | High level logic output voltage | $V_{DD}$ = 3.3 V, $I_{OH}$ = 4 mA | 2.4 | | | V |
| | | $V_{DD}$ = 5 V, $I_{OH}$ = 4 mA | 4.7 | | | |
| $R_{PUCS}$ | $\overline{CS}$ pull-up resistor | | | 430 | | kΩ |
| $R_{PDRST}$ | $\overline{STBY}$/$\overline{RESET}$ pull-down resistor | | | 450 | | |
| $R_{PUSW}$ | SW pull-up resistor | | | 80 | | |
| $t_{high,STCK}$ | Step-clock input high time | | 300 | | | ns |
| $t_{low,STCK}$ | Step-clock input low time | | 300 | | | ns |
| **Internal oscillator and external oscillator driver** | | | | | | |
| $f_{osc,int}$ | Internal oscillator frequency | $T_j$ = 25 °C | -5% | 16 | +5% | MHz |
| $f_{osc,ext}$ | Programmable external oscillator frequency | | 8 | | 32 | MHz |
| $V_{OSCOUTH}$ | OSCOUT clock source high level voltage | Internal oscillator | 2.4 | | | V |
| $V_{OSCOUTL}$ | OSCOUT clock source low level voltage | Internal oscillator | | | 0.3 | V |
| $t_{rOSCOUT}$ $t_{fOSCOUT}$ | OSCOUT clock source rise and fall time | Internal oscillator | | | 10 | ns |
| $t_{high}$ | OSCOUT clock source high time | Internal oscillator | | 31.25 | | ns |
| $t_{extosc}$ | Internal to external oscillator switching delay | | | 3 | | ms |
| $t_{intosc}$ | External to internal oscillator switching delay | | | | 100 | µs |
| **SPI** | | | | | | |
| $f_{CK,MAX}$ | Maximum SPI clock frequency (4) | | 5 | | | MHz |
| $t_{rCK}$ $t_{fCK}$ | SPI clock rise and fall time (4) | | | | 1 | µs |
| $t_{hCK}$ $tl_{CK}$ | SPI clock high and low time (4) | | 90 | | | ns |
| $t_{setCS}$ | Chip select setup time (4) | | 30 | | | ns |
| $t_{holCS}$ | Chip select hold time (4) | | 30 | | | ns |
| $t_{disCS}$ | Deselect time (4) | | 625 | | | ns |
| $t_{setSDI}$ | Data input setup time (4) | | 20 | | | ns |
| $t_{holSDI}$ | Data input hold time (4) | | 30 | | | ns |

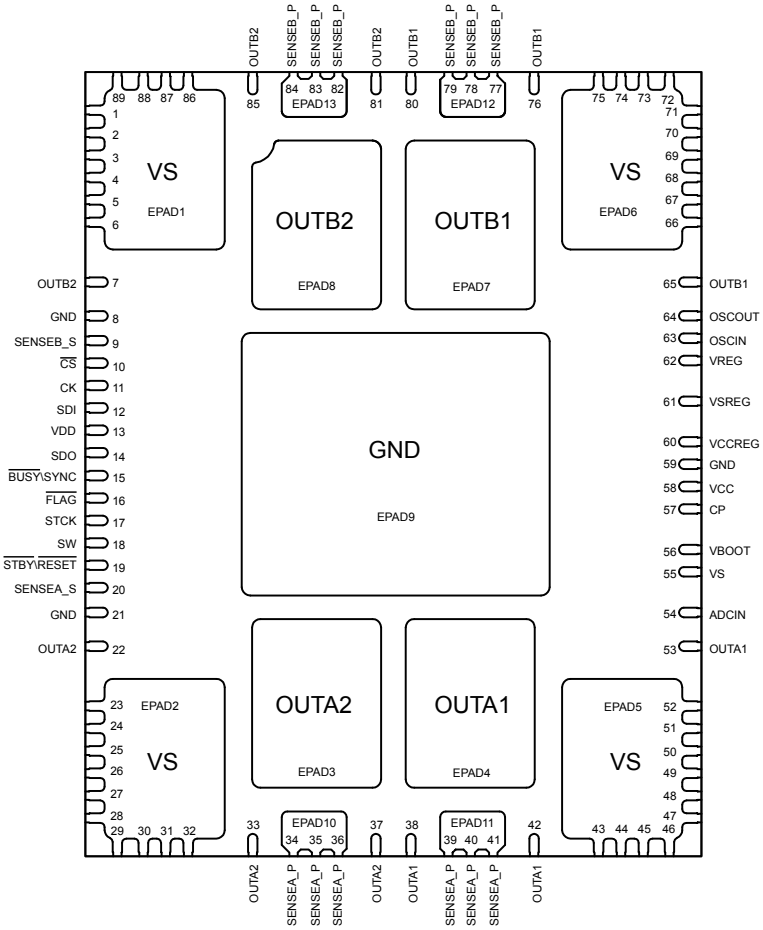
| Symbol | Parameter | Test conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $t_{enSDO}$ | Data output enable time (4) | | | | 95 | ns |
| $t_{disSDO}$ | Data output disable time (4) | | | | 95 | ns |
| $t_{vSDO}$ | Data output valid time (4) | | | | 35 | ns |
| $t_{holSDO}$ | Data output hold time (4) | | 0 | | | ns |
| **PWM modulators** | | | | | | |
| $f_{PWM}$ | Programmable PWM frequency (2) | $f_{osc}$ = 32 MHz<br>F_PWM_INT = '11X'<br>F_PWM_DEC = '000' | | 5.6 | | kHz |
| | | $f_{osc}$ = 32 MHz<br>F_PWM_INT = '000'<br>F_PWM_DEC = '111' | | 125 | | kHz |
| $N_{PWM}$ | PWM resolution | | | 8 | | bit |
| **Current control** | | | | | | |
| $V_{REF, MAX}$ | Maximum reference voltage | | | 1000 | | mV |
| $V_{REF, MIN}$ | Minimum reference voltage | | | 7.8 | | mV |
| **Overcurrent protection** | | | | | | |
| $V_{OCD}$ | Programmable overcurrent detection voltage $V_{DS}$ threshold | OCD_TH = '00000' | 27 | 31 | 35 | mV |
| | | OCD_TH = '01001' | 270 | 312.5 | 344 | mV |
| | | OCD_TH = '10011' | 500 | 625 | 688 | mV |
| | | OCD_TH = '11111' | 800 | 1000 | 1100 | mV |
| $t_{OCD,Comp}$ | OCD comparator delay | | | 100 | 200 | ns |
| $t_{OCD,Flag}$ | OCD to flag signal delay time | | | 230 | 530 | ns |
| $t_{OCD,SD}$ | OCD to shutdown delay time | $I_{gate}$ = 4 mA, $t_{CC}$ = maximum | | 4200 | 6000 | ns |
| **Stall detection** | | | | | | |
| $V_{STALL}$ | Programmable stall detection $V_{DS}$ voltage threshold | STALL_TH = '11111' | | 1000 | | mV |
| | | STALL_TH = '00000' | | 31 | | |
| **Standby** | | | | | | |
| $I_{STBY}$ | Standby mode supply current (VSREG pin) | $V_{CC}$ = $V_{CCREG}$ = 7.5 V, $V_{SREG}$ = 48 V | | 42 | | µA |
| | | $V_{CC}$ = $V_{CCREG}$ = 7.5 V, $V_{SREG}$ = 18 V | | 37.5 | | |
| $I_{STBY,reg}$ | Standby mode supply current (VREG pin) | | | 6 | | µA |
| $t_{STBY,min}$ | Minimum standby time | | | 0.5 | | ms |
| $t_{logicwu}$ | Logic power-on and wake-up time | | | 0.5 | | ms |
| $t_{cpwu}$ | Charge pump power-on and wake-up time | Power bridges disabled,<br>$C_{FLY}$ = 10 nF, $C_{BOOT}$ = 220 nF, $V_{CC}$ = 15 V | | 1 | | ms |
| **Internal voltage regulators** | | | | | | |
| $V_{CCOUT}$ | Internal $V_{CC}$ voltage regulator output voltage | VCCVAL = 0 (default), $I_{CC}$ = 10 mA | 7.3 | 7.5 | | V |
| | | VCCVAL = 1, $I_{CC}$ = 10 mA | 14 | 15 | | |
| $V_{SREG, drop}$ | $V_{SREG}$ to $V_{CC}$ dropout voltage | $I_{CC}$ = 50 mA | | | 3 | V |
| $P_{CC}$ | Internal $V_{CC}$ voltage regulator power dissipation | | | | 2.5 | W |
| $V_{REGOUT}$ | Internal $V_{REG}$ voltage regulator output voltage | $I_{REG}$ = 10 mA | 3.135 | 3.3 | | V |

| Symbol | Parameter | Test conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $V_{CCREG,\ drop}$ | $V_{CCREG}$ to $V_{REG}$ dropout voltage | $I_{REG}$ = 50 mA | | | 3 | V |
| $I_{REGOUT}$ | Internal $V_{REG}$ voltage regulator output current | | | 125 | | mA |
| $I_{REGOUT,STBY}$ | Internal $V_{REG}$ voltage regulator output standby current | | | 55 | | mA |
| $P_{REG}$ | Internal $V_{REG}$ voltage regulator power dissipation | | | | 0.5 | W |
| **Integrated analog-to-digital converter** | | | | | | |
| $N_{ADC}$ | Analog-to-digital converter resolution | | | 5 | | bit |
| $V_{ADC,ref}$ | Analog-to-digital converter reference voltage | | | 3.3 | | V |
| $f_S$ | Analog-to-digital converter sampling frequency | Voltage mode (2) | | $f_{PWM}$ | | kHz |
| | | Current mode (2) | | $f_{OSC}$ / 512 | | kHz |
| $V_{ADC,UVLO}$ | ADCIN UVLO threshold | | 1.05 | 1.16 | 1.35 | V |

1. *Guaranteed in the temperature range -25 to 125°C.*
2. *The value accuracy is dependent on oscillator frequency accuracy (see Section 7.8).*
3. *$\overline{FLAG}$ and $\overline{BUSY}$ open-drain outputs included.*
4. *See Figure 25. SPI timings diagram*

# 4 Pin connection

**Figure 2. Pin connection (top view)**




*Note: All VS pins and pads must be connected together.*
*Pins 34, 35, 36, EPAD10 (SENSEA2_P) must be connected to pins 39, 40, 41, EPAD11 (SENSEA1_P)*
*Pins 77, 78, 79, EPAD12 (SENSEB1_P) must be connected to pins 82, 83, 84, EPAD13 (SENSEB2_P)*

# 5 Pin list

**Table 5. Pin description**

| No. | Name | Type | Function |
|---|---|---|---|
| 1, 2, 3, 4, 5, 6, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 43, 44, 45, 46, 47,48, 49, 50, 51, 52, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 86, 87, 88, 89, EPAD1, EPAD2, EPAD5, EPAD6 | VS | Supply | Motor supply voltage (drain of the high-side MOSFETs) (1) |
| 38, 42, 53, EPAD4 | OUTA1 | Power output | Half-bridge A1 output |
| 22, 33, 37, EPAD3 | OUTA2 | Power output | Half-bridge A2 output |
| 65, 76, 80, EPAD7 | OUTB1 | Power output | Half-bridge B1 output |
| 7, 81, 85, EPAD8 | OUTB2 | Power output | Half-bridge B2 output |
| 8, 21, 59, EPAD9 | GND | Ground | Ground |
| 34, 35, 36, EPAD10 | SENSEA2_P | Sense (power) | Tail of half-bridge A2 (source of the respective low-side MOSFET) (2) |
| 39, 40, 41, EPAD11 | SENSEA1_P | Sense (power) | Tail of half-bridge A1 (source of the respective low-side MOSFET) (2) |
| 20 | SENSEA_S | Analog input | Current control comparator input for motor phase A |
| 77, 78, 79, EPAD12 | SENSEB1_P | Sense (power) | Tail of half-bridge B1 (source of the respective low-side MOSFET) (3) |
| 82, 83, 84, EPAD13 | SENSEB2_P | Sense (power) | Tail of half-bridge B2 (source of the respective low-side MOSFET) (3) |
| 9 | SENSEB_S | Analog input | Current control comparator input for motor phase B |
| 60 | VCCREG | Power supply | Internal $V_{REG}$ voltage regulator supply voltage |
| 62 | VREG | Power supply | Logic supply voltage |
| 13 | VDD | Power supply | Logic interface supply voltage |
| 61 | VSREG | Power supply | Internal $V_{CC}$ voltage regulator supply voltage |
| 58 | VCC | Power supply | Gate driver supply voltage |
| 63 | OSCIN | Analog input | Oscillator pin 1. To connect an external oscillator or clock source |
| 64 | OSCOUT | Analog output | Oscillator pin 2. To connect an external oscillator. When the internal oscillator is used, this pin can supply a 2/4/8/16 MHz clock |
| 57 | CP | Output | Charge pump oscillator output |
| 56 | VBOOT | Power supply | Bootstrap voltage needed for driving the high-side power DMOS of both bridges (A and B) |
| 54 | ADCIN | Analog input | Internal analog-to-digital converter input |
| 55 | VS | Power supply | Motor supply voltage |
| 18 | SW | Logical input | External switch input pin |
| 14 | SDO | Logic output | Data output pin for serial interface |
| 12 | SDI | Logic input | Data input pin for serial interface |
| 11 | CK | Logic input | Serial interface clock |
| 10 | $\overline{CS}$ | Logic input | Chip select input pin for serial interface |
| 15 | $\overline{BUSY}$/SYNC | Open-drain output | By default, the $\overline{BUSY}$/SYNC pin is forced low when the device is performing a command.<br>The pin can be programmed in order to generate a synchronization signal |

| No. | Name | Type | Function |
|---|---|---|---|
| 16 | $\overline{\text{FLAG}}$ | Open-drain output | Status flag pin. An internal open-drain transistor can pull the pin to GND when a programmed alarm condition occurs (step loss, OCD, thermal pre-warning or shutdown, UVLO, wrong command, non-performable command) |
| 19 | $\overline{\text{STBY}}$/$\overline{\text{RESET}}$ | Logic input | Standby and reset pin. Low logic level puts the device in standby mode and reset logic.<br>If not used, it should be connected to $V_{REG}$ |
| 17 | STCK | Logic input | Step clock input |

1. *All VS pins and pads must be connected together.*
2. *Pins 34, 35, 36, EPAD10 (SENSEA2_P) must be connected to pins 39, 40, 41, EPAD11 (SENSEA1_P).*
3. *Pins 77, 78, 79, EPAD12 (SENSEB1_P) must be connected to pins 82, 83, 84, EPAD13 (SENSEB2_P).*

# 6 Typical applications

**Table 6. Typical application values**

| Name | Value |
|---|---|
| $C_{VSPOL}$ | 220 µF |
| $C_{VS}$ | 220 nF |
| $C_{BOOT}$ | 470 nF |
| $C_{FLY}$ | 47 nF |
| $C_{VSREG}$ | 100 nF |
| $C_{VCC}$ | 470 nF |
| $C_{VCCREG}$ | 100 nF |
| $C_{VREG}$ | 100 nF |
| $C_{VREGPOL}$ | 22 µF |
| $C_{VDD}$ | 100 nF |
| D1 | Charge pump diodes |
| $R_{PU}$ | 39 kΩ |
| $R_A$ | 1.8 kΩ ($V_S$ = 85 V) |
| $R_B$ | 91 kΩ ($V_S$ = 85 V) |
| $R_{SENSEA}$ | 0.1 Ω (maximum $I_{peak}$ = 10 A) |
| $R_{SENSEB}$ | 0.1 Ω (maximum $I_{peak}$ = 10 A) |

**Figure 3. Typical application schematic - voltage mode**

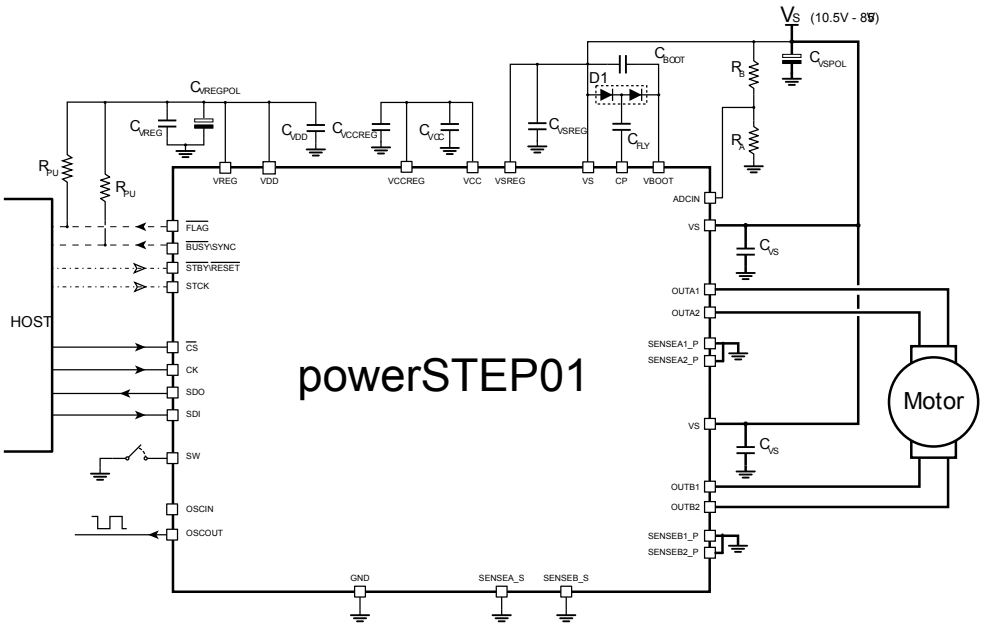
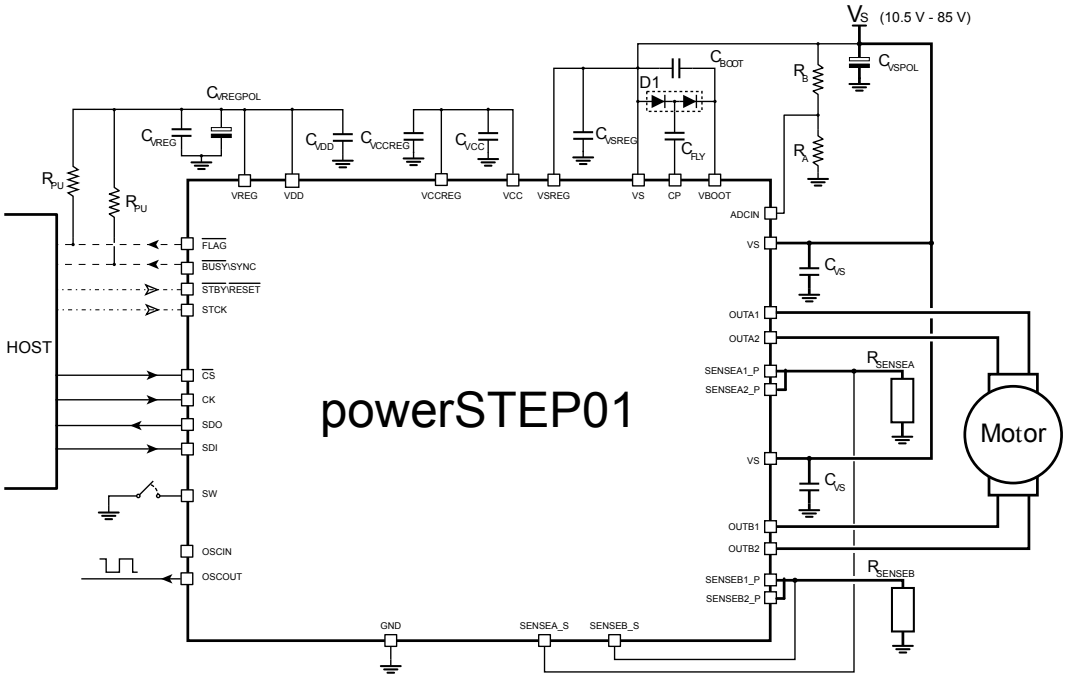


Figure 4. **Typical application schematic - current mode**



# 7 Functional description

## 7.1 Device power-up

During power-up, the device is under reset (all logic IOs disabled and power bridges in high impedance state) until the following conditions are satisfied:

- $V_{REG}$ is greater than $V_{REGthOn}$
- Internal oscillator is operative
- $\overline{STBY}/\overline{RESET}$ input is forced high.

After power-up, the device state is the following:

- Parameters are set to default
- Internal logic is driven by the internal oscillator and a 2 MHz clock is provided by the OSCOUT pin
- Bridges are disabled (high impedance).
- $\overline{FLAG}$ output is forced low (UVLO failure indication).

After power-up, a period of $t_{logicwu}$ must pass before applying a command to allow proper oscillator and logic startup.

Any movement command makes the device exit from the high-Z state (HardStop and SoftStop included).

## 7.2 Logic I/O

Pins $\overline{CS}$, CK, SDI, STCK, SW, and $\overline{STBY}/\overline{RESET}$ are TTL/CMOS 3.3 V to 5 V compatible logic inputs.

Pin SDO is a TTL/CMOS compatible logic output. VDD pin voltage imposes the logical output voltage range.

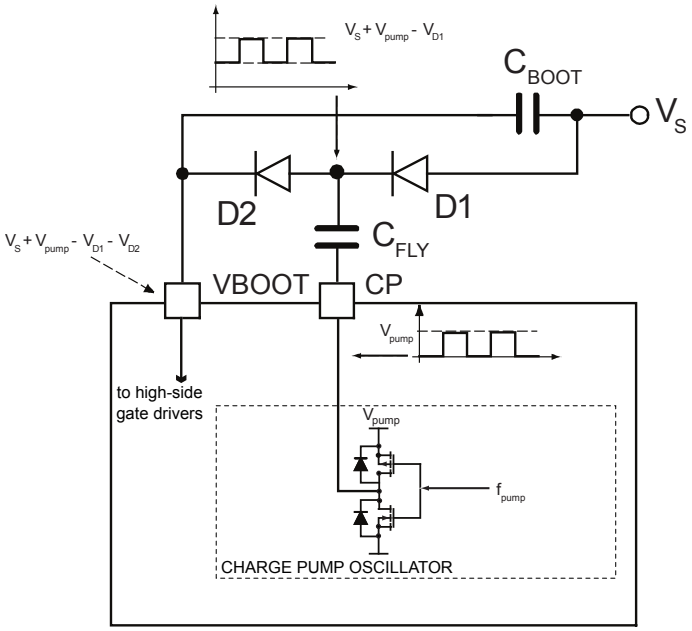
Pins $\overline{FLAG}$ and $\overline{BUSY}$/SYNC are open-drain outputs.

SW and $\overline{CS}$ inputs are internally pulled up to $V_{DD}$ and $\overline{STBY}/\overline{RESET}$ input is internally pulled down to ground.

## 7.3 Charge pump

To ensure the correct driving of the high-side integrated MOSFETs, a voltage higher than the motor power supply voltage needs to be applied to the VBOOT pin. The high-side gate driver supply voltage $V_{BOOT}$ is obtained through an oscillator and a few external components realizing a charge pump (see Figure 5).

**Figure 5. Charge pump circuitry**

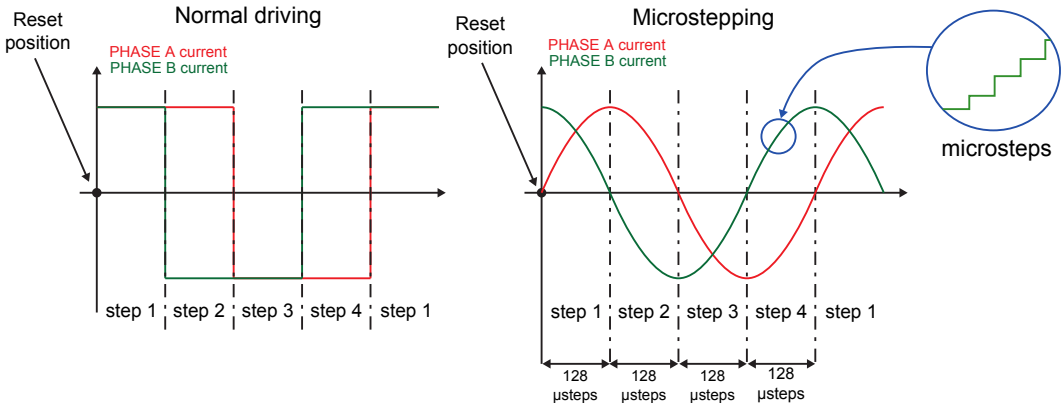


## 7.4 Microstepping

The driver is able to divide the single step up to a maximum of 128 microsteps. Stepping mode can be programmed by the STEP_SEL parameter in the STEP_MODE register (see Table 27). In current mode driving the maximum microstepping resolution is 1/16$^{th}$ of the step.

Step mode can only be changed when bridges are disabled. Every time the step mode is changed, the electrical position (that is, the point of microstepping sine wave that is generated) is reset to zero and the absolute position counter value (see Absolute position counter) becomes meaningless.

**Figure 6. Normal mode and microstepping (128 microsteps)**




### 7.4.1 Automatic full-step and boost modes

When motor speed is greater than a programmable full-step speed threshold, the device switches automatically to full-step mode; the driving mode returns to microstepping when motor speed decreases below the full-step speed threshold.

The switching between the microstepping and full-step mode and vice versa is always performed at an electrical position multiple of π/4 (Figure 7 and Figure 8).

The full-step speed threshold is set through the related parameter in the FS_SPD register (see Section 11.1.9).

When the BOOST_MODE bit of the FS_SPD register is low (default), the amplitude of the voltage squarewave in full-step mode is equal to the peak of the voltage sine wave multiplied by sine (π/4) (see Figure 7). This avoids the current drop between the two driving modes.

When the BOOST_MODE bit of the FS_SPD register is high, the amplitude of the voltage squarewave in full-step mode is equal to the peak of the voltage sine wave (Figure 8). That increases the output current increasing the maximum motor torque.

**Figure 7. Automatic full-step switching in normal mode**

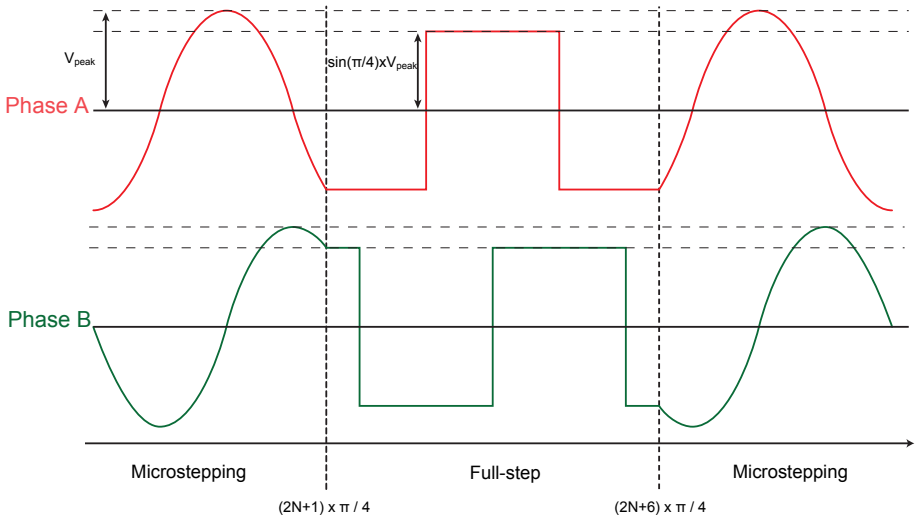
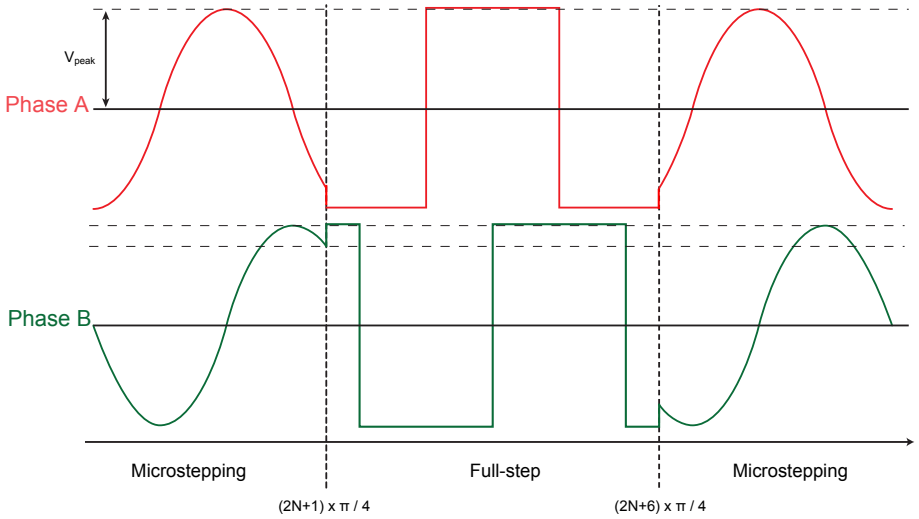


Figure 8. **Automatic full-step switching in boost mode**




## 7.5 Absolute position counter

An internal 22-bit register (ABS_POS) records all the motor motions according to the selected step mode; the stored value unit is equal to the selected step mode (full, half, quarter, etc.). The position range is from $-2^{21}$ to '$+2^{21}$ - 1' steps (see Section 11.1.1).

The reset value of the register is 0, this value is also considered the HOME position for some of the positioning commands (see Section 11.2).

## 7.6 Programmable speed profiles

The user can easily program a customized speed profile independently defining acceleration, deceleration, maximum and minimum speed values by ACC, DEC, MAX_SPEED and MIN_SPEED registers respectively (see Section 11.1.5, Section 11.1.6, Section 11.1.7, and Section 11.1.8).

When a command is sent to the device, the integrated logic generates the microstep frequency profile that performs a motor motion compliant with speed profile boundaries.

All acceleration parameters are expressed in step/tick$^2$ and all speed parameters are expressed in step/tick; the unit of measurement does not depend on the selected step mode.

Acceleration parameter ranges from $2^{-40}$ to '$(2^{12}-2) \cdot 2^{-40}$' step/tick$^2$ (equivalent to 14.55 to 59575.541 step/s$^2$).

Deceleration parameter ranges from $2^{-40}$ to '$(2^{12}-1) \cdot 2^{-40}$' step/tick$^2$ (equivalent to 14.55 to 59590 step/s$^2$).

Minimum speed parameter ranges from 0 to '$(2^{12}-1) \cdot 2^{-24}$' step/tick (equivalent to 0 to 976.3 step/s).

Maximum speed parameter ranges from $2^{-18}$ to '$(2^{10}-1) \cdot 2^{-18}$' step/tick (equivalent to 15.25 to 15610 step/s).

## 7.7 Motor control commands

The powerSTEP01 can accept different types of commands:

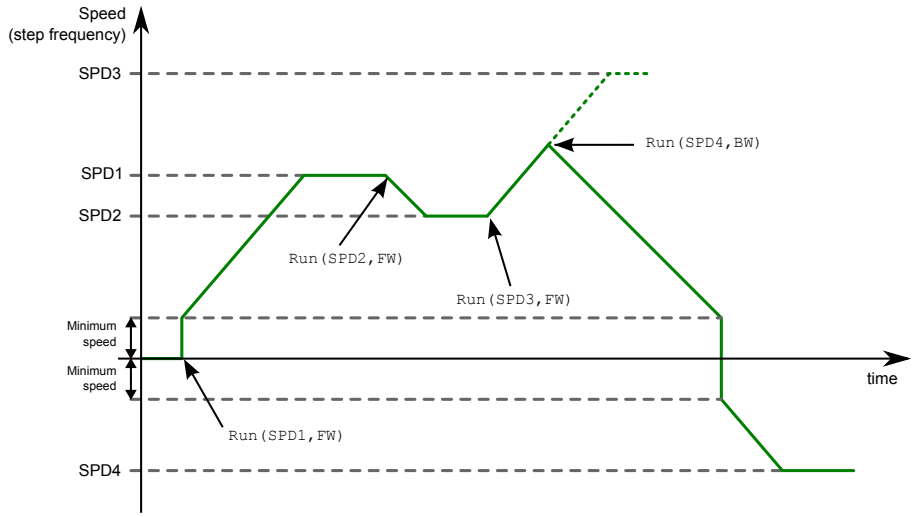
- constant speed commands (Run, GoUntil, ReleaseSW)
- absolute positioning commands (GoTo, GoTo_DIR, GoHome, GoMark)
- motion commands (Move)
- stop commands (SoftStop, HardStop, SoftHiz, HardHiz).

For detailed command descriptions refer to Section 11.2.

### 7.7.1 Constant speed commands

A constant speed command produces a motion in order to reach and maintain a user-defined target speed starting from the programmed minimum speed (set in the MIN_SPEED register) and with the programmed acceleration/deceleration value (set in the ACC and DEC registers). A new constant speed command can be requested anytime.

Figure 9. **Constant speed command examples**



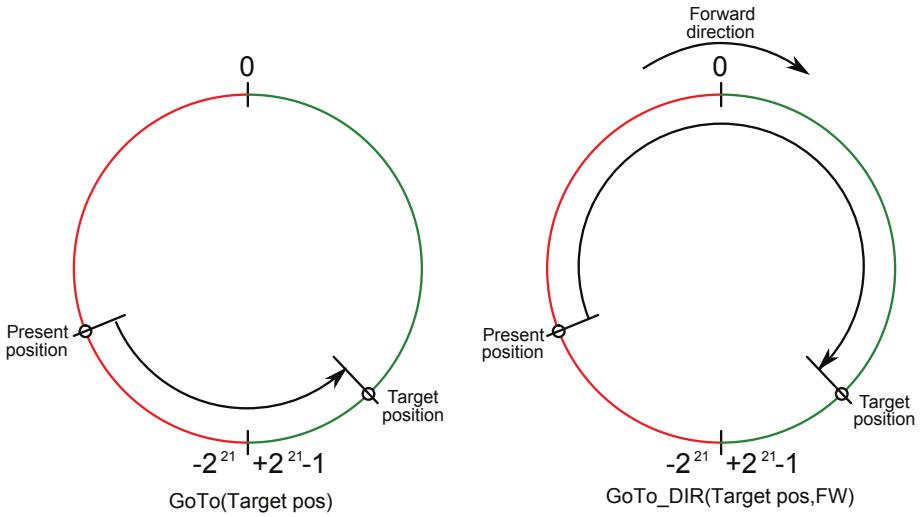

### 7.7.2 Positioning commands

An absolute positioning command produces a motion in order to reach a user-defined position that is sent to the device together with the command. The position can be reached performing the minimum path (minimum physical distance) or forcing a direction (see Figure 10).

Performed motor motion is compliant to programmed speed profile boundaries (acceleration, deceleration, minimum and maximum speed).

Note that with some speed profiles or positioning commands, the deceleration phase can start before the maximum speed is reached.

Figure 10. **Positioning command examples**



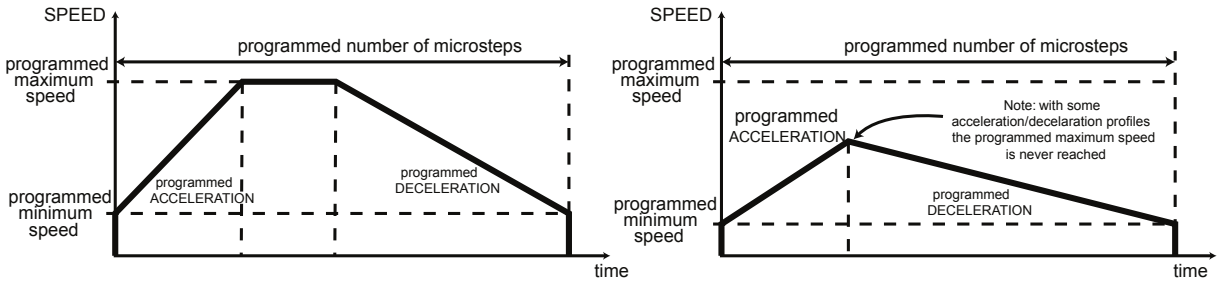

### 7.7.3 Motion commands

Motion commands produce a motion in order to perform a user-defined number of microsteps in a user-defined direction that are sent to the device together with the command (see Figure 11).

Performed motor motion is compliant to programmed speed profile boundaries (acceleration, deceleration, minimum and maximum speed).

Note that with some speed profiles or motion commands, the deceleration phase can start before the maximum speed is reached.

Figure 11. **Motion command examples**




### 7.7.4 Stop commands

A stop command forces the motor to stop. Stop commands can be sent anytime.

The SoftStop command causes the motor to decelerate with a programmed deceleration value until the MIN_SPEED value is reached and then stops the motor keeping the rotor position (a holding torque is applied).

The HardStop command stops the motor instantly, ignoring deceleration constraints and keeping the rotor position (a holding torque is applied).

The SoftHiZ command causes the motor to decelerate with a programmed deceleration value until the MIN_SPEED value is reached and then forces the bridges into a high impedance state (no holding torque is present).

The HardHiZ command instantly forces the bridges into a high impedance state (no holding torque is present).

### 7.7.5 Step-clock mode

In step-clock mode the motor motion is defined by the step-clock signal applied to the STCK pin. At each step-clock rising edge, the motor is moved one microstep in the programmed direction and absolute position is consequently updated.

When the system is in step-clock mode the SCK_MOD flag in the STATUS register is raised, the SPEED register is set to zero, and motor status is considered stopped regardless of the STCK signal frequency (the MOT_STATUS parameter in the STATUS register equal to “00”).

### 7.7.6 GoUntil and ReleaseSW commands

In most applications the power-up position of the stepper motor is undefined, so an initialization algorithm drives the motor to a known position is necessary.

The GoUntil and ReleaseSW commands can be used in combination with external switch input (see Section 7.14) to easily initialize the motor position.

The GoUntil command makes the motor run at the target constant speed until the SW input is forced low (falling edge). When this event occurs, one of the following actions can be performed:

- ABS_POS register is set to zero (home position) and the motor decelerates to zero speed (as a SoftStop command)
- ABS_POS register value is stored in the MARK register and the motor decelerates to zero speed (as a SoftStop command).

If the SW_MODE bit of the CONFIG register is set to ‘0’, the motor does not decelerate but it immediately stops (as a HardStop command).

The ReleaseSW command makes the motor run at a programmed minimum speed until the SW input is forced high (rising edge). When this event occurs, one of the following actions can be performed:

- ABS_POS register is set to zero (home position) and the motor immediately stops (as a HardStop command)
- ABS_POS register value is stored in the MARK register and the motor immediately stops (as a HardStop command).

If the programmed minimum speed is less than 5 step/s, the motor is driven at 5 step/s.

## 7.8 Internal oscillator and oscillator driver

The control logic clock can be supplied by the internal 16-MHz oscillator, an external oscillator (crystal or ceramic resonator) or a direct clock signal.

These working modes can be selected by the EXT_CLK and OSC_SEL parameters in the CONFIG register (see Table 41).

At power-up the device starts using the internal oscillator and provides a 2-MHz clock signal on the OSCOUT pin.

**Attention:** ***In any case, before changing clock source configuration, a hardware reset is mandatory. Switching to different clock configurations during operation may cause unexpected behavior.***

### 7.8.1 Internal oscillator

In this mode, the internal oscillator is activated and OSCIN is unused. If the OSCOUT clock source is enabled, the OSCOUT pin provides a 2, 4, 8, or 16-MHz clock signal (according to OSC_SEL value); otherwise it is unused (see Figure 12).

### 7.8.2 External clock source

Two types of external clock source can be selected: crystal/ceramic resonator or direct clock source. Four programmable clock frequencies are available for each external clock source: 8, 16, 24, and 32 MHz.

When an external crystal/resonator is selected, the OSCIN and OSCOUT pins are used to drive the crystal/resonator (see Figure 12). The crystal/resonator and load capacitors ($C_L$) must be placed as close as possible to the pins. Refer to Table 7 for the choice of the load capacitor value according to the external oscillator frequency.

**Table 7. $C_L$ values according to external oscillator frequency**

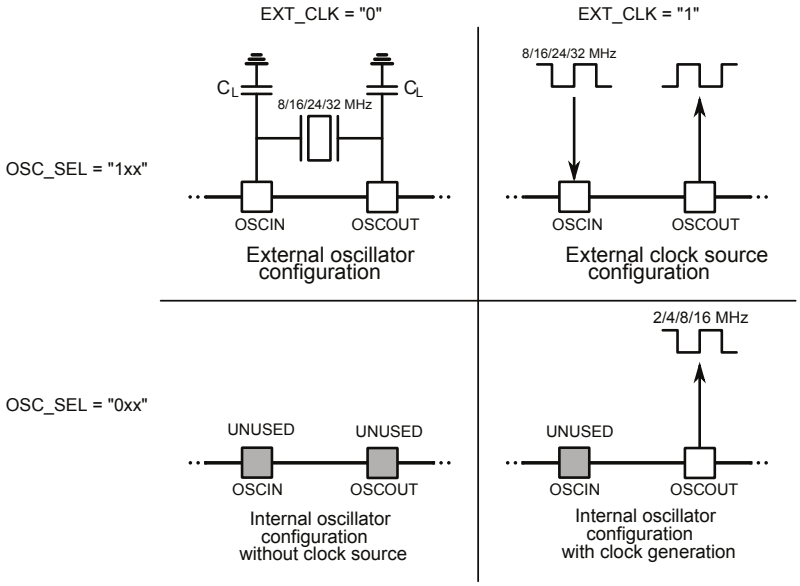
| Crystal/resonator frequency(1) | $C_L$ (2) |
|---|---|
| 8 MHz | 25 pF ($ESR_{max}$ = 80 Ω) |
| 16 MHz | 18 pF ($ESR_{max}$ = 50 Ω) |
| 24 MHz | 15 pF ($ESR_{max}$ = 40 Ω) |
| 32 MHz | 10 pF ($ESR_{max}$ = 40 Ω) |

1. *First harmonic resonance frequency.*
2. *Lower ESR value allows driving greater load capacitors.*

If a direct clock source is used, it must be connected to the OSCIN pin and the OSCOUT pin supplies the inverted OSCIN signal (see Figure 12).

The powerSTEP01 integrates a clock detection system that resets the device in case of the failure of the external clock source (direct or crystal/resonator). The monitoring of the clock source is disabled by default; it can be enabled setting high the WD_EN bit in the GATECFG1 register (Section 11.1.25). When the external clock source is selected, the device continues to work with the integrated oscillator for $t_{extosc}$ milliseconds and then the clock management system switches to the OSCIN input.

**Figure 12. OSCIN and OSCOUT pins configuration**



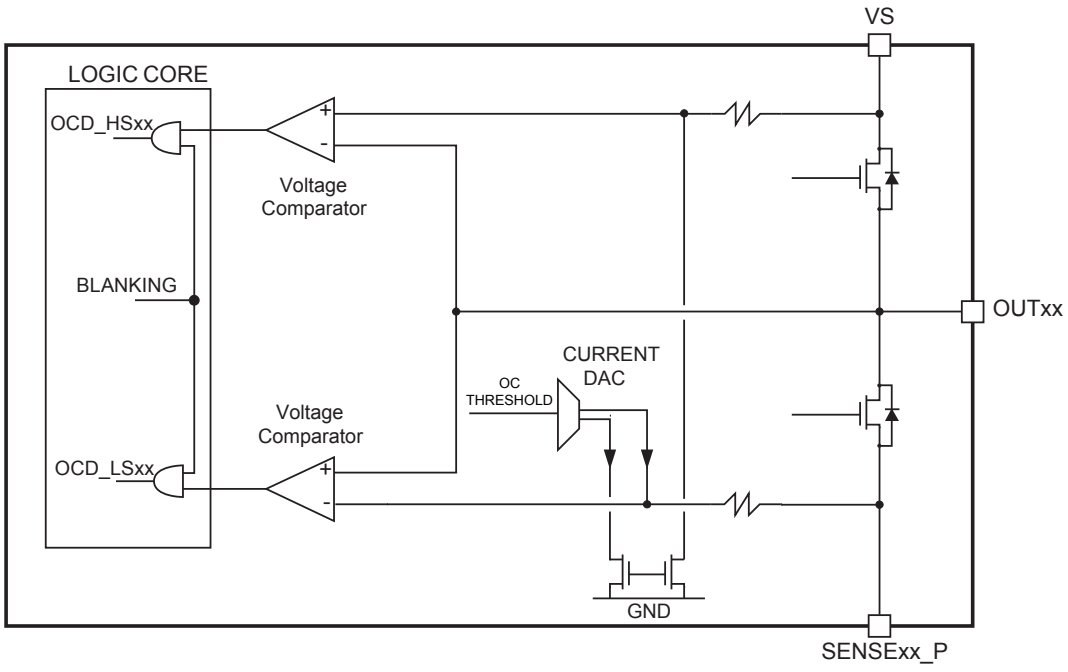

*Note: When OSCIN is UNUSED, it should be left floating.*
*When OSCOUT is UNUSED, it should be left floating.*

## 7.9 Overcurrent detection

The powerSTEP01 measures the load current of each half-bridge sensing the $V_{DS}$ voltage of the integrated MOSFET (see Figure 13). When any of the $V_{DS}$ voltages rise over the programmed threshold, the OCD flag in the STATUS register is forced low until the event expires and a GetStatus command is sent to the device (Section 11.1.28 and Section 11.2.20). The overcurrent event expires when all the MOSFET $V_{DS}$ voltages fall below the programmed threshold.

The overcurrent threshold can be programmed by the OCD_TH register in one of 32 available values listed in Table 25.

**Figure 13. Overcurrent detection - principle scheme**



The overcurrent detection comparators are disabled in order to avoid wrong voltage measurements in the following cases:

- The respective half-bridge is in high impedance state (both MOSFETs forced off)
- The respective half-bridge is commutating
- The respective half-bridge is commutated and the programmed blanking time is not yet elapsed
- The respective gate is turned off

It is possible to set if an overcurrent event causes the bridge to turn off or not through the OC_SD bit in the CONFIG register.

When the power bridges are turned off by an overcurrent event, they cannot be turned on until the OCD flag is released by a GetStatus command.

## 7.10 Undervoltage lockout (UVLO)

The powerSTEP01 provides a programmable gate driver supply voltage UVLO protection. When one of the supply voltages of the gate driver ($V_{CC}$ for the low-sides and $V_{BOOT}$ - $V_S$ for the high-sides) falls below the respective turn-off threshold, an undervoltage event occurs. In this case, all MOSFETs are immediately turned off and the UVLO flag in the STATUS register is forced low.

The UVLO flag is forced low and the MOSFETs are kept off until the gate driver supply voltages return above the respective turn-on threshold; in this case the undervoltage event expires and the UVLO flag can be released through a GetStatus command.

The UVLO thresholds can be selected between two sets according to the UVLOVAL bit value in the CONFIG register.

**Table 8. UVLO thresholds**

| Description | UVLOVAL | |
|---|---|---|
| | 0 | 1 |
| Low-side gate driver supply turn-off threshold ($V_{CCthOff}$) | 6.3 V | 10 V |
| Low-side gate driver supply turn-on threshold ($V_{CCthOn}$) | 6.9 V | 10.4 V |
| High-side gate driver supply turn-off threshold ($\Delta V_{BOOTthOff}$) | 5.5 V | 8.8 V |
| High-side gate driver supply turn-on threshold ($\Delta V_{BOOTthOn}$) | 6 V | 9.2 V |

## 7.11 $V_S$ undervoltage lockout (UVLO_ADC)

The device provides an undervoltage signal of the integrated ADC input voltage (the UVLO_ADC flag in the STATUS register). When $V_{ADCIN}$ falls below the $V_{ADC,UVLO}$ value, the UVLO_ADC flag is forced low and it is kept in this state until the ADCIN voltage is greater than $V_{ADC,UVLO}$ and a GetStatus command is sent to the device.

The ADCIN undervoltage event does not turn off the power bridges.

The motor supply voltage undervoltage detection can be performed by means of this feature, connecting the ADCIN pin to $V_S$ through a voltage divider as described in Section 8.5.

## 7.12 Thermal warning and thermal shutdown

An integrated sensor allows detection of the internal temperature and provides a 3-level protection.

When the $T_{j(WRN)Set}$ threshold is reached, a warning signal is generated. This is the thermal warning condition and it expires when the temperature falls below the $T_{j(WRN)Rec}$ threshold.

When the $T_{j(OFF)Set}$ threshold is reached, all the MOSFETs are turned off and the gate driving circuitry is disabled (Miller clamps are still operative). This condition expires when the temperature falls below the $T_{j(OFF)Rec}$ threshold.

When the $T_{j(SD)Set}$ threshold is reached, all the MOSFETs are turned off using Miller clamps, the internal $V_{CC}$ voltage regulator is disabled, and the current capability of the internal $V_{REG}$ voltage regulator is reduced (thermal shutdown). In this condition logic is still active (if supplied). The thermal shutdown condition only expires when the temperature goes below $T_{j(SD)Rec}$.

The thermal condition of the device is shown by TH_STATUS bits in the STATUS register (see Table 9).

**Table 9. Thermal protection summarizing table**

| State | Set condition | Release condition | Description | TH_STATUS |
|---|---|---|---|---|
| Normal | | | Normal operation state | 00 |
| Warning | $T_j > T_{j(WRN)Set}$ | $T_j < T_{j(WRN)Rec}$ | Temperature warning: operation is not limited | 01 |
| Bridge shutdown | $T_j > T_{j(OFF)Set}$ | $T_j < T_{j(OFF)Rec}$ | High temperature protection: the MOSFETs are turned off and the gate drivers are disabled | 10 |
| Device shutdown | $T_j > T_{j(SD)Set}$ | $T_j < T_{j(SD)Rec}$ | Overtemperature protection: the MOSFETs are turned off, the gate drivers are disabled, the internal $V_{CC}$ voltage regulator is disabled, the current capability of the internal $V_{REG}$ voltage regulator is limited, and the charge pump is disabled | 11 |

## 7.13 Reset and standby

The device can be reset and put into standby mode through the $\overline{STBY}/\overline{RESET}$ pin. When it is forced low, all the MOSFETs are turned off (high impedance state), the charge pump is stopped, the SPI interface and control logic are disabled, and the internal $V_{REG}$ voltage regulator maximum output current is limited; as a result, the powerSTEP01 heavily reduces the power consumption.

At the same time, the register values are reset to their default and all the protection functions are disabled.

The $\overline{STBY}/\overline{RESET}$ input must be forced low at least for $t_{STBY,min}$ in order to ensure the complete switch to standby mode.

When exiting standby mode, as well as for IC power-up, a delay must be given before applying a new command to allow proper oscillator and charge pump startup. Actual delay could vary according to the values of the charge pump external components.

When exiting standby mode all the MOSFETs are off and the HiZ flag is high.

The registers can be reset to the default values without putting the device into standby mode through the ResetDevice command (Section 11.2.14).

## 7.14 External switch (SW pin)

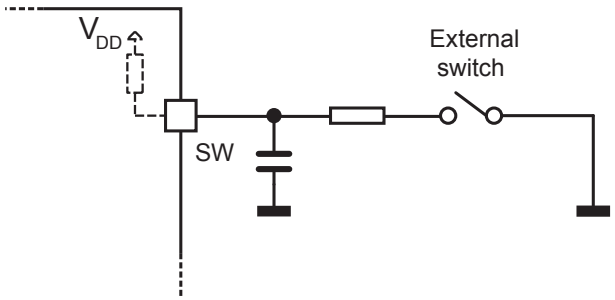
The SW input is internally pulled up to $V_{DD}$ and detects if the pin is open or connected to ground (see Figure 14).

The SW_F bit of the STATUS register indicates if the switch is open (‘0’) or closed (‘1’) (Section 11.1.28 ); the bit value is refreshed at every system clock cycle (125 ns). The SW_EVN flag of the STATUS register is raised when a switch turn-on event (SW input falling edge) is detected (Section 11.1.28). A GetStatus command releases the SW_EVN flag (Section 11.2.20).

By default, a switch turn-on event causes a HardStop interrupt (SW_MODE bit of CONFIG register set to ‘0’). Otherwise (SW_MODE bit of CONFIG register set to ‘1’), switch input events do not cause interrupts and the switch status information is at the user’s disposal (see Table 42).

The switch input can be used by the GoUntil and ReleaseSW commands as described in Section 11.2.10 and Section 11.2.11. If the SW input is not used, it should be connected to $V_{DD}$.

**Figure 14. External switch connection**



## 7.15 Integrated power MOSFETs

The typical characteristics of the 8 N-channel MOSFETs integrated into the system-in-package are listed in Table 10.

**Table 10. Integrated MOSFET characteristics at $T_j$ = 25 °C**

| Symbol | Parameter | Typical value | Unit |
|---|---|---|---|
| $V_{GS(th)}$ | Gate threshold voltage | 3 | V |
| $Q_g$ | Total gate charge at $V_{gs}$ = 10 V | 25 | nC |
| $V_{SD}$ | Source-drain diode forward on voltage at $I_{SD}$ = 10A | 0.8 | V |

## 7.16 Programmable slew rate

The powerSTEP01 integrates eight programmable gate drivers which allow the output slew rate to be set in a wide range of values.

The following parameters can be adjusted:

- Gate sink/source current ($I_{GATE}$)
- Controlled current time ($t_{CC}$)
- Turn-off overboost time ($t_{OB}$)

During the turn-on, the gate driver charges the gate forcing an $I_{GATE}$ current for all the controlled current time period. At the end of the controlled current phase, the gate of the integrated MOSFET should be completely charged.

During the turn-off, the gate driver discharges the gate sinking an $I_{GATE}$ current for all the controlled current time period. At the beginning of the turn-off, an overboost phase can be added; in this case the gate driver sinks an $I_{OB}$ current for the programmed $t_{OB}$ period in order to rapidly reach the plateau region. At the end of the controlled current time, the gate of the integrated MOSFET should be completely discharged.

The gate current can be set to one of the following values: 4, 8, 16, 24, 32, 64, and 96 mA through the $I_{GATE}$ parameter in the GATECFG1 register.

Controlled current time can be programmed within a range from 125 ns to 3.75 µs with a resolution of 125 ns (TCC parameter in GATECFG1 register).

Turn-off overboost time can be set to one of the following values: 0, 62.5, 125, 250 ns (TBOOST parameter in GATECFG1 register). The 62.5 ns value is only available when clock frequency is 16 MHz or 32 MHz; when clock frequency is 8 MHz it is changed to 125 ns and when a 24-MHz clock is used, it is changed to 83.3 ns.

**Table 11. Output slew rate**

| Slew rate ($V_S$ = 48 V) | $I_{gate}$ | $t_{CC}$ | $t_{DT}$ | $t_{blank}$ | $t_{boost}$ |
|---|---|---|---|---|---|
| 980 V/µs | 96 mA | 375 ns | 125 ns | 500 ns | 0 ns |
| 790 V/µs | 64 mA | 500 ns | 125 ns | 375 ns | 0 ns |
| 520 V/µs | 32 mA | 875 ns | 125 ns | 250 ns | 0 ns |
| 400 V/µs | 24 mA | 1000 ns | 125 ns | 250 ns | 0 ns |
| 220 V/µs | 16 mA | 1600 ns | 125 ns | 250 ns | 0 ns |
| 114 V/µs | 8 mA | 3125 ns | 125 ns | 250 ns | 0 ns |

## 7.17 Deadtime and blanking time

During the bridge commutation, a deadtime is added in order to avoid cross conductions. The deadtime can be programmed within a range from 125 ns to 4 µs with a resolution of 125 ns (TDT parameter in GATECFG2 register, see Section 11.1.26).

At the end of each commutation, the overcurrent and stall detection comparators are disabled (blanking) in order to avoid the respective systems detecting body diodes turn-off current peaks.

The duration of blanking time is programmable through theTBLANK parameter in the GATECFG2 register at one of the following values: 125, 250, 375, 500, 625, 750, 875, 1000 ns (see Section 11.1.26).

## 7.18 Integrated analog-to-digital converter

The powerSTEP01 integrates an $N_{ADC}$ bit ramp-compare analog-to-digital converter with a reference voltage equal to $V_{REG}$. The analog-to-digital converter input is available through the ADCIN pin and the conversion result is available in the ADC_OUT register (Section 11.1.20).

The ADC_OUT value can be used for motor supply voltage compensation or can be at the user's disposal.

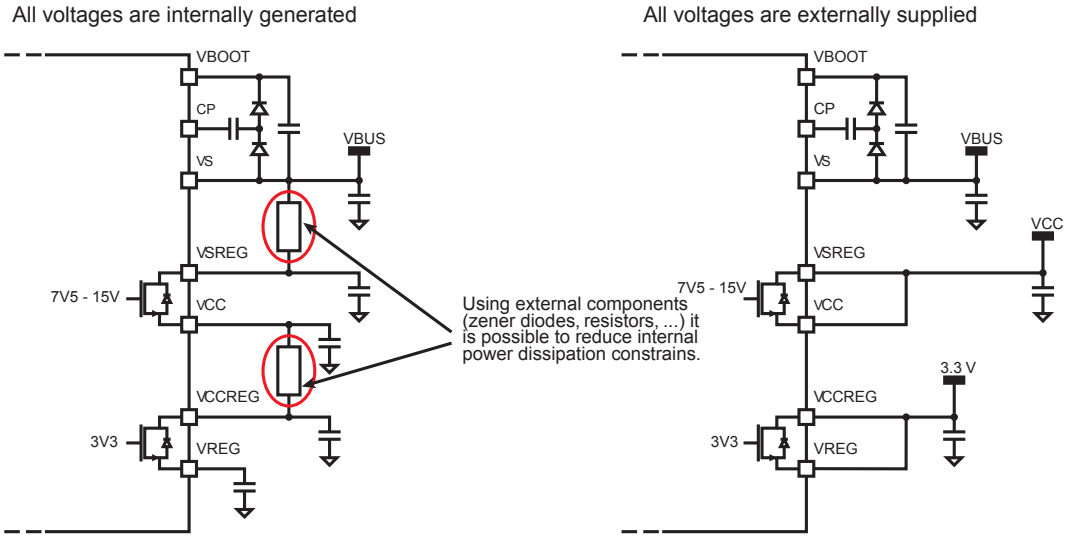
## 7.19 Supply management and internal voltage regulators

The powerSTEP01 integrates two linear voltage regulators: the first one can be used to obtain gate driver supply starting from a higher voltage (for example, the motor supply one). Its output voltage can be set to 7.5 V or 15 V according to the VCCVAL bit value (CONFIG register). The second linear voltage regulator can be used to obtain the 3.3 V logic supply voltage.

The regulators are designed to supply the internal circuitry of the IC and should not be used to supply external components.

The input and output voltages of both regulators are connected to external pins and the regulators are totally independent: in this way a very flexible supply management can be performed using external components or external supply voltages (Figure 15).

**Figure 15. Device supply management**




If $V_{CC}$ is externally supplied, the VSREG and VCC pins must be shorted ($V_{SREG}$ must be compliant with $V_{CC}$ range).

If $V_{REG}$ is externally supplied, the VCCREG and VREG pins must be shorted and equal to 3.3 V.

$V_{SREG}$ must always be less than $V_{BOOT}$ in order to avoid related ESD protection diode turn-on. The device can be protected from this event by adding an external low drop diode between the VSREG and VS pins; charge pump diodes should be low drop too.

$V_{CCREG}$ must always be less than $V_{CC}$ in order to avoid ESD protection diode turn-on. The device can be protected from this event by adding an external low drop diode between the VCCREG and VSREG pins.

Both regulators provide a short-circuit protection limiting the load current within the respective maximum ratings.

## 7.20 $\overline{\text{BUSY}}$/SYNC pin

This pin is an open-drain output that can be used as a busy flag or synchronization signal according to the SYNC_EN bit value (see Section 11.1.23).

## 7.21 FLAG pin

By default, an internal open-drain transistor pulls the $\overline{\text{FLAG}}$ pin to ground when at least one of the following conditions occurs:

- Power-up or standby/reset exit
- Stall detection on bridge A
- Stall detection on bridge B
- Overcurrent detection
- Thermal warning
- Thermal shutdown
- UVLO
- UVLO on ADC input
- Switch turn-on event
- Command error

It is possible to mask one or more alarm conditions by programming the ALARM_EN register (see Table 32). If the corresponding bit of the ALARM_EN register is low, the alarm condition is masked and it does not cause a $\overline{\text{FLAG}}$ pin transition; all other actions imposed by alarm conditions are performed anyway.

In case of daisy chain configuration, FLAG pins of different ICs can be or-wired to save host controller GPIOs.

# 8 Phase current control: voltage mode

When the voltage mode driving is selected (CM_VM bit in STEP_MODE register is set to 0), the powerSTEP01 controls the phase current applying a sinusoidal voltage to motor windings. Phase current amplitude is not directly controlled but depends on phase voltage amplitude, load torque, motor electrical characteristics, and rotation speed. Sine wave amplitude is proportional to the motor supply voltage multiplied by a coefficient ($K_{VAL}$). $K_{VAL}$ ranges from 0 to 100% and the sine wave amplitude can be obtained through the following formula:

$$V_{OUT} = V_S \cdot K_{VAL} \quad (1)$$

Different $K_{VAL}$ values can be programmed for acceleration, deceleration, and constant speed phases, and when the motor is stopped (HOLD phase) through KVAL_ACC, KVAL_DEC, KVAL_RUN, and KVAL_HOLD registers (see Section 11.1.10). $K_{VAL}$ value is calculated according to the following formula:

$$K_{VAL} = \left[\left(K_{VAL\_X} + BEMF_{COMP}\right) \times VSCOMP \times K_{THERM}\right] \times microstep \quad (2)$$

where $K_{VAL\_X}$ is the starting $K_{VAL}$ value programmed for the present motion phase (KVAL_ACC, KVAL_DEC, KVAL_RUN, or KVAL_HOLD), *BEMF_COMP* is the BEMF compensation curve value, *VSCOMP* and *K_THERM* are the motor supply voltage and winding resistance compensation factors and *microstep* is the current microstep value (fraction of target peak current).

The powerSTEP01 offers various methods to ensure a stable current value, allowing the compensation of:

- Low speed distortion (see Section 8.3)
- Back electromotive force (see Section 8.4)
- Motor supply voltage variation (see Section 8.5)
- Winding resistance variation (see Section 8.6)

## 8.1 PWM sine wave generators

The two voltage sine waves applied to the stepper motor phases are generated by two PWM modulators.

The PWM frequency ($f_{PWM}$) is proportional to the oscillator frequency ($f_{OSC}$) and can be obtained through the following formula:

$$f_{PWM} = \frac{f_{OSC}}{512 \cdot N} \cdot m \quad (3)$$

*'N'* is the integer division factor and *'m'* is the multiplication factor. *'N'* and *'m'* values can be programmed by F_PWM_INT and F_PWM_DEC parameters in the CONFIG register (see Table 47 and Table 48).

Available PWM frequencies are listed in Section 11.1.27 from Table 49 to Table 51.

## 8.2 Sensorless stall detection

The powerSTEP01 is able to detect a motor stall caused by an excessive load torque. When the motor is driven using the voltage mode approach, a stall condition corresponds to an unexpected increase of the phase current. Imposing a current threshold slightly above the operative current, it is possible to detect the stall condition without speed or position sensors.

The powerSTEP01 measures the load current of each phase sensing the $V_{DS}$ voltage of the low-side power MOSFETs. When any of the $V_{DS}$ voltages rise over the programmed threshold, the STALL_X flag in the STATUS register of the respective bridge (STALL_A or STALL_B) is forced low. The failure flag is kept low until the $V_{DS}$ voltages fall below the programmed threshold and a GetStatus command is sent to the device (see Section 11.1.28 and Section 11.2.20).

The stall detection threshold can be programmed in one of 32 available values ranging from 31.25 mV to 1 V with steps of 31.25 mV (see Section 11.1.22).

Stall detection comparators are disabled, in order to avoid wrong voltage measurements, in the following cases:

- The respective half-bridge is in a high impedance state (both MOSFETs forced off)
- The respective half-bridge is commutating

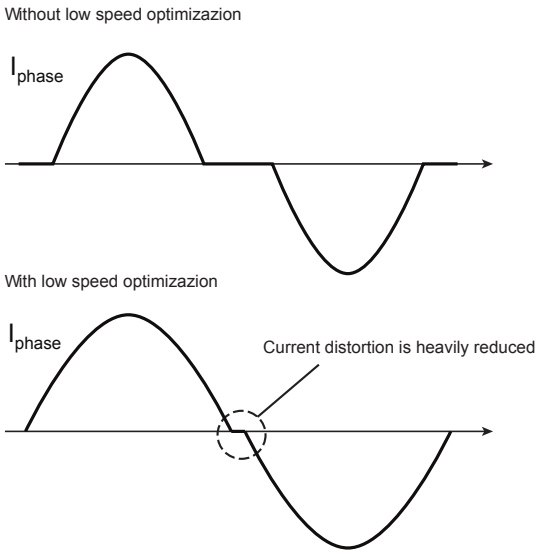
- The respective half-bridge is commutated and the programmed blanking time has not yet elapsed
- The respective low-side gate is turned off.

## 8.3 Low speed optimization

When the motor is driven at a very low speed using a small driving voltage, the resulting phase current can be distorted. As a consequence, the motor position is different from the ideal one (see Figure 16).

The device implements a low speed optimization in order to remove this effect.

**Figure 16. Current distortion and compensation**



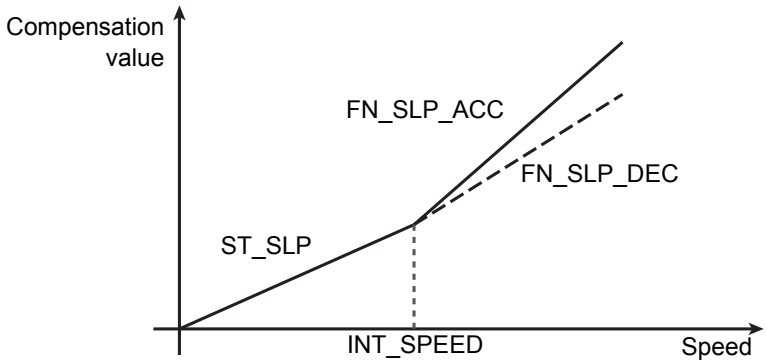

The optimization can be enabled setting high the LSPD_OPT bit in the MIN_SPEED register (Section 11.1.8) and is active in a speed range from zero to MIN_SPEED. When low speed optimization is enabled, the speed profile minimum speed is forced to zero.

## 8.4 BEMF compensation

Using the speed information, a compensation curve is added to the amplitude of the voltage waveform applied to the motor winding in order to compensate the BEMF variations during acceleration and deceleration (see Figure 17).

The compensation curve is approximated by a stacked line with a starting slope (ST_SLP) when speed is lower than a programmable threshold speed (INT_SPEED) and a fine slope (FN_SLP_ACC and FN_SLP_DEC) when speed is greater than the threshold speed (see sections Section 11.1.11, Section 11.1.12, Section 11.1.13 and Section 11.1.14).

**Figure 17. BEMF compensation curve**



To obtain different current values during acceleration and deceleration phase, two different final slope values, and consequently two different compensation curves, can be programmed.

Acceleration compensation curve is applied when the motor runs. No BEMF compensation is applied when the motor is stopped.

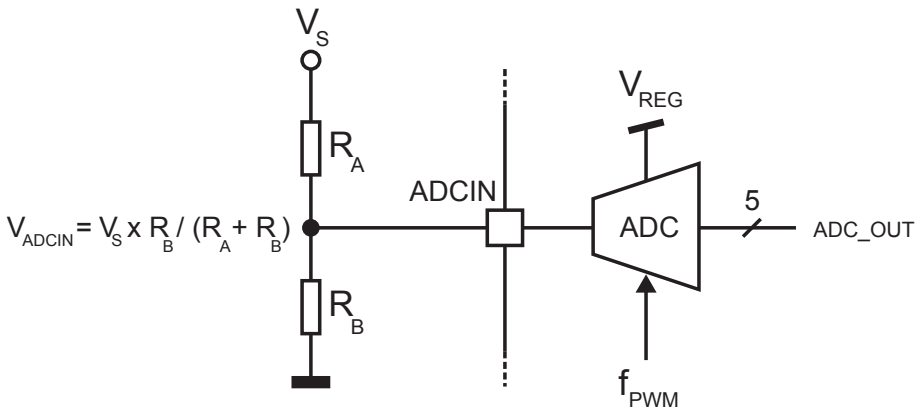
## 8.5 Motor supply voltage compensation

The sine wave amplitude generated by the PWM modulators is directly proportional to the motor supply voltage ($V_S$). When the motor supply voltage is different from its nominal value, the motor phases are driven with an incorrect voltage. The powerSTEP01 can compensate for motor supply voltage variations in order to avoid this effect.

The motor supply voltage should be connected to the integrated ADC input through a resistor divider in order to obtain the $V_{REG}/2$ voltage at the ADCIN pin when $V_S$ is at its nominal value (see Figure 18).

The ADC input is sampled at $f_S$ frequency, which is equal to the PWM frequency.

**Figure 18. Motor supply voltage compensation circuit**




Motor supply voltage compensation can be enabled by setting high the EN_VSCOMP bit of the CONFIG register (see Table 46). If the EN_VSCOMP bit is low, the compensation is disabled and the internal analog-to-digital converter is at the user's disposal; the sampling rate is always equal to PWM frequency.

## 8.6 Winding resistance thermal drift compensation

The higher the winding resistance the greater the voltage to be applied in order to obtain the same phase current.

The powerSTEP01 integrates a register (K_THERM) which can be used to compensate a phase resistance increment due to temperature rising.

The value in the K_THERM register (see Section 11.1.15) multiplies the duty cycle value allowing the higher phase resistance value to be faced.

The compensation algorithm and the eventual motor temperature measurement should be implemented by microcontroller firmware.

# 9 Phase current control: current mode

When the current mode driving is selected (CM_VM bit in STEP_MODE register is set to 1), the powerSTEP01 performs a new current control technique, named predictive current control, allowing the device to obtain the target average phase current. This method is described in detail in Section 9.1. Furthermore, the powerSTEP01 automatically selects the better decay mode in order to follow the current profile.

Current control algorithm parameters can be programmed by the T_FAST, TON_MIN, TOFF_MIN, and CONFIG registers (see Section 11.1.17, Section 11.1.18, Section 11.1.19, and Section 11.1.27 for details).

Different current amplitude can be set for acceleration, deceleration, and constant speed phases and when the motor is stopped through the TVAL_ACC, TVAL_DEC, TVAL_RUN, and TVAL_HOLD registers (see Section 11.1.16). The output current amplitude can also be regulated by the ADCIN voltage value (see Section 9.4).

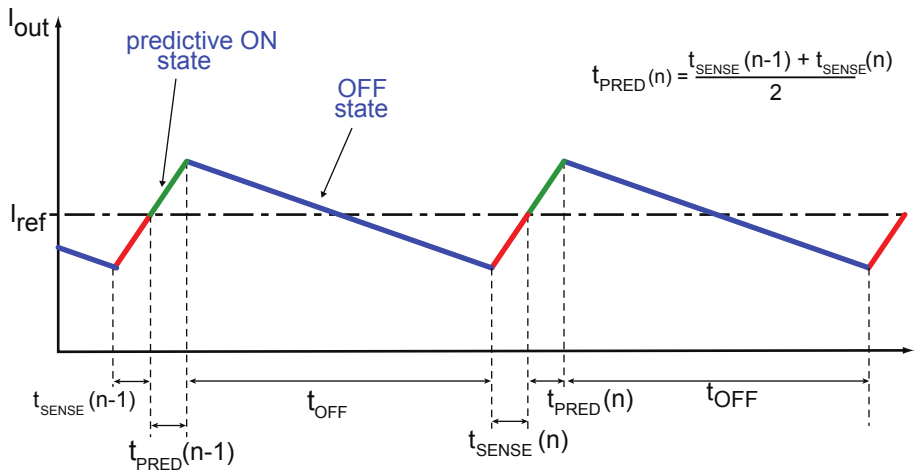
Each bridge is driven by an independent control system that shares with the other bridge the control parameters only.

## 9.1 Predictive current control

Unlike classical peak current control systems, that make the phase current decay when the target value is reached, this new method keeps the power bridge ON for an extra time after reaching the current threshold.

At each cycle the system measures the time required to reach the target current ($t_{SENSE}$). After that the power stage is kept in a “predictive” ON state ($t_{PRED}$) for a time equal to the mean value of $t_{SENSE}$ in the last two control cycles (actual one and previous one), as shown in Figure 19.

**Figure 19. Predictive current control**



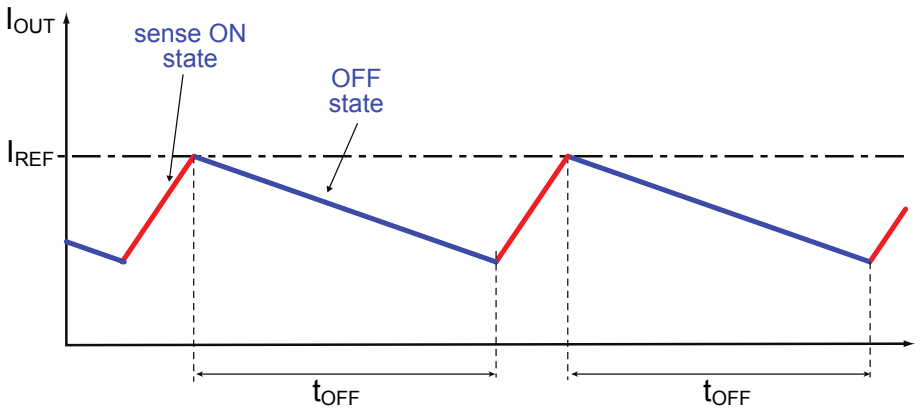

At the end of the predictive ON state the power stage is set in OFF state for a fixed time, as in a constant $t_{OFF}$ current control. During the OFF state both slow and fast decay can be performed; the better decay combination is automatically selected by the powerSTEP01, as described in Section 9.2.

As shown in Figure 19, the system is able to center the triangular wave on the desired reference value, improving dramatically the accuracy of the current control system: in fact the average value of a triangular wave is exactly equal to the middle point of each of its segments and at steady-state the predictive current control tends to equalize the duration of the $t_{SENSE}$ and the $t_{PRED}$ time.

Furthermore, the $t_{OFF}$ value is recalculated each time a new current value is requested (microstep change) in order to keep the PWM frequency as near as possible to the programmed one (TSW parameter in the CONFIG register).

The device can be forced to work using a classic peak current control, setting low the PRED_EN bit in the CONFIG register (default condition). In this case, after the sense phase ($t_{SENSE}$) the power stage is set in OFF state, as shown in Figure 20.

Figure 20. **Non-predictive current control**




## 9.2 Auto-adjusted decay mode

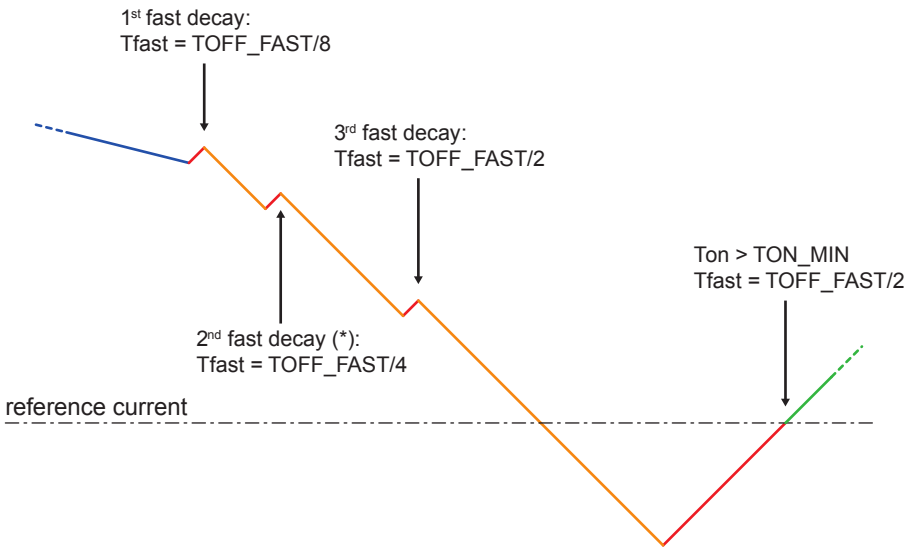
During the current control, the device automatically selects the better decay mode in order to follow the current profile reducing the current ripple.

At reset, the off-time is performed turning on both the low-side MOSFET of the power stage and the current recirculates in the lower half of the bridge (slow decay).

If, during a PWM cycle, the target current threshold is reached in a time shorter than the TON_MIN value, a fast decay of TOFF_FAST/8 (T_FAST register) is immediately performed, turning on the opposite MOS of both half-bridges and the current recirculates back to the supply bus.

After this time, the bridge returns to an ON state: if the time needed to reach the target current value is still less than TON_MIN, a new fast decay is performed with a period twice the previous one. Otherwise, the normal control sequence is followed as described in Section 9.1. The maximum fast decay duration is set by the TOFF_FAST value.

Figure 21. **Adaptive decay - fast decay tuning**

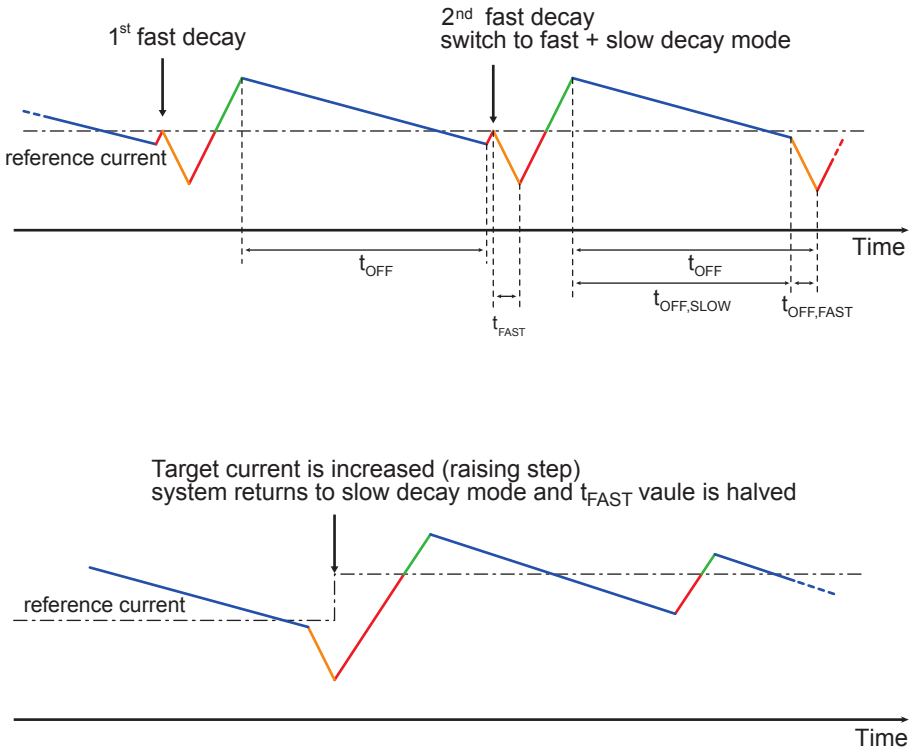



(*)Note: starting from 2nd fast decay the system combines fast and slow decay during the OFF phase.

When two or more fast decays are performed with the present target current, the control system adds a fast decay at the end of every off-time keeping the OFF state duration constant ($t_{OFF}$ is split into $t_{OFF}$, SLOW and $t_{OFF}$, FAST). When the current threshold is increased by a microstep change (rising step), the system returns to normal decay mode (slow decay only) and the $t_{FAST}$ value is halved.

Stopping the motor or reaching the current sine wave zero-crossing causes the current control system to return to the reset state.

**Figure 22. Adaptive decay - switch from normal to slow+ fast decay mode and vice versa**




## 9.3 Auto-adjusted fast decay during the falling steps

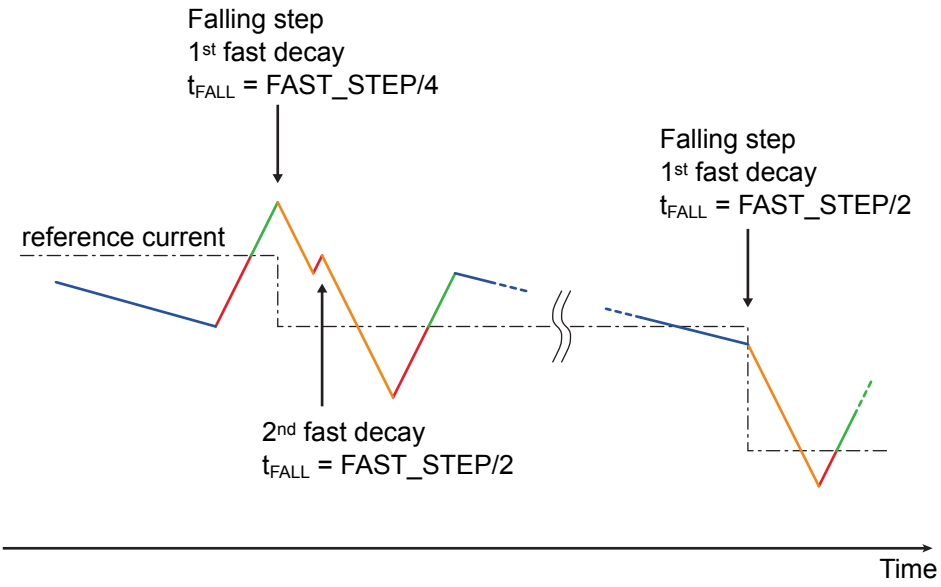
When the target current is decreased by a microstep change (falling step), the device performs a fast decay in order to reach the new value as fast as possible. However, exceeding the fast duration could cause a strong ripple on the step change. The powerSTEP01 automatically adjusts these fast decays reducing the current ripple.

At reset the fast decay value ($t_{FALL}$) is set to FAST_STEP/4 (T_FAST register). The $t_{FALL}$ value is doubled every time, within the same falling step, an extra fast decay is necessary to obtain an on-time greater than TON_MIN (see Section 11.1.18). The maximum $t_{FALL}$ value is equal to FAST_STEP.

At the next falling step, the system uses the last $t_{FALL}$ value of the previous falling step.

Stopping the motor or reaching the current sine wave zero-crossing causes the current control system to return to the reset state.

Figure 23. **Fast decay tuning during the falling steps**




## 9.4 Torque regulation (setting the output current)

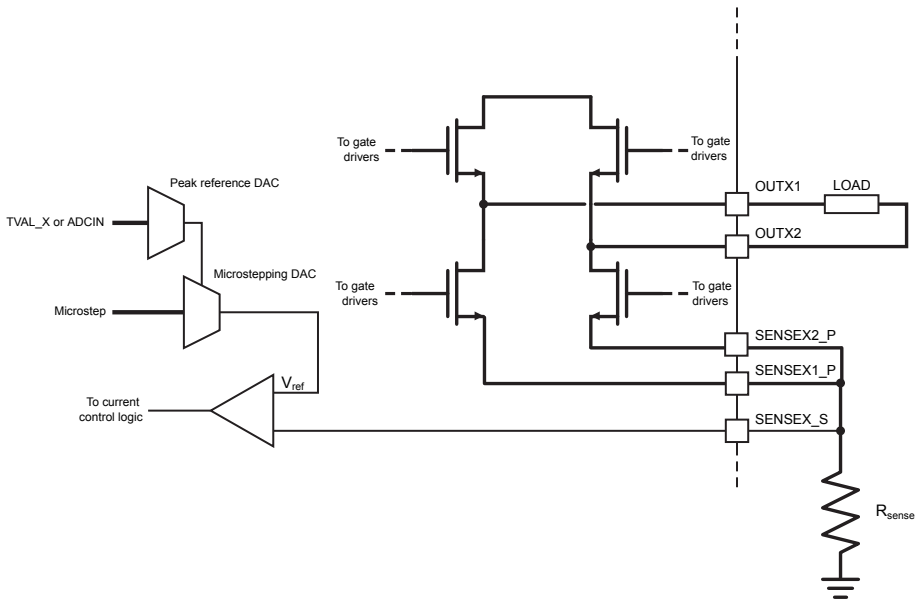
The phase currents are monitored through two shunt resistors (one for each power bridge) connected to the respective sense pin (see Figure 24). The integrated comparator compares the sense resistor voltage with the internal reference generated using the peak value, which is proportional to the output current amplitude, and the microstepping code. The comparison result is provided to the logic in order to implement the current control algorithm as described in previous sections.

The peak reference voltage can be regulated in two ways: writing TVAL_ACC, TVAL_DEC, TVAL_RUN, and TVAL_HOLD registers or varying the ADCIN voltage value.

The EN_TQREG bit (CONFIG register) sets the torque regulation method. If this bit is high, the ADC_OUT prevalue is used to regulate output current amplitude (see Table 24). Otherwise the internal analog-to-digital converter is at the user's disposal and the output current amplitude is managed by the TVAL_HOLD, TVAL_RUN, TVAL_ACC, and TVAL_DEC registers (see Table 18 ).

The voltage applied to the ADCIN pin is sampled at $f_S$ frequency and converted in an NADC bit digital signal. The analog-to-digital conversion result is available in the ADC_OUT register.

Figure 24. **Current sensing and reference voltage generation**



# 10 Serial interface

The integrated 8-bit serial peripheral interface (SPI) is used for a synchronous serial communication between the host microprocessor (always master) and the device (always slave).

The SPI uses chip select ($\overline{CS}$), serial clock (CK), serial data input (SDI), and serial data output (SDO) pins. When $\overline{CS}$ is high, the device is unselected and the SDO line is inactive (high impedance).

The communication starts when $\overline{CS}$ is forced low. The CK line is used for synchronization of data communication.

All commands and data bytes are shifted into the device through the SDI input, most significant bit first. The SDI is sampled on the rising edges of the CK.

All output data bytes are shifted out of the device through the SDO output, most significant bit first. The SDO is latched on the falling edges of the CK. When a return value from the device is not available, an all zero byte is sent.

After each byte transmission, the $\overline{CS}$ input must be raised and be kept high for at least $t_{disCS}$ in order to allow the device to decode the received command and put the return value into the shift register.

All timing requirements are shown in Figure 25 (see Section 3 for values).

Multiple devices can be connected in a daisy chain configuration, as shown in Figure 26.

**Figure 25. SPI timings diagram**

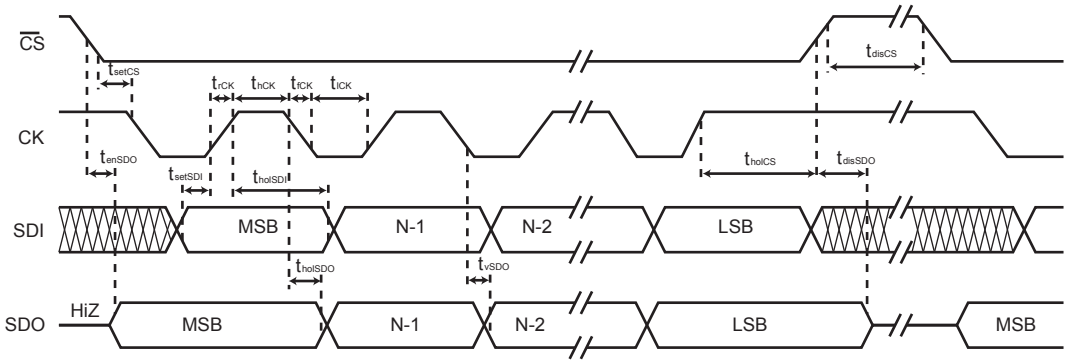
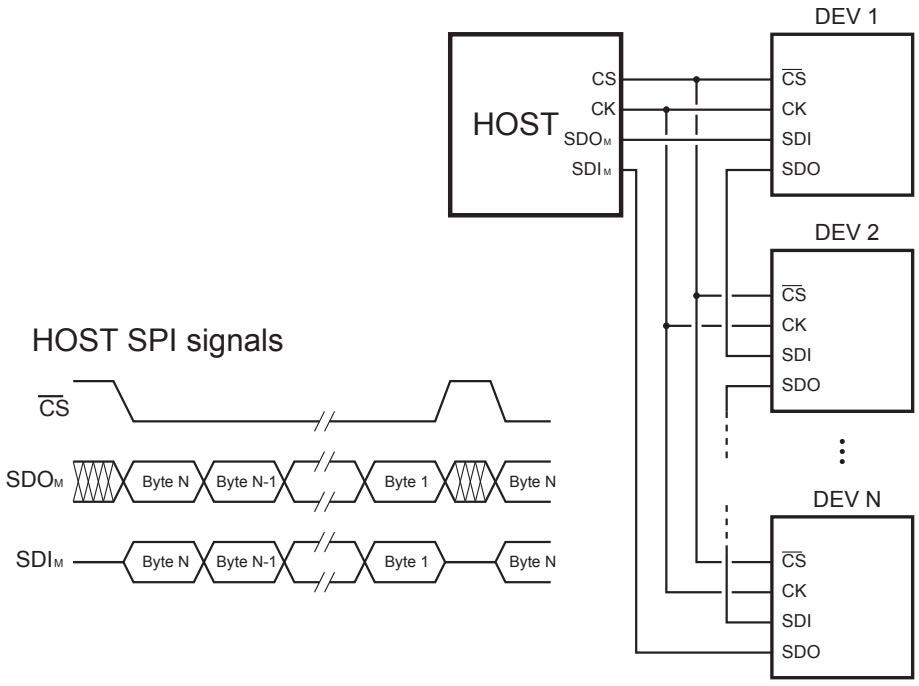



**Figure 26. Daisy chain configuration**



# 11 Programming manual

## 11.1 Register and flag description

Following a map of the user available registers (detailed description in respective paragraphs):

**Table 12. Register map**

| Address [Hex] | Register name | Register function | Length [bit] | Reset [Hex] | Reset value | Remarks (1) |
|---|---|---|---|---|---|---|
| **General configuration** | | | | | | |
| h01 | ABS_POS | Current position | 22 | 000000 | 0 step | R, WS |
| h02 | EL_POS | Electrical position | 9 | 000 | 0 | R, WH |
| h03 | MARK | Mark position | 22 | 000000 | 0 step | R, WR |
| h04 | SPEED | Current speed | 20 | 00000 | 0 step/tick (0 step/s) | R |
| h05 | ACC | Acceleration | 12 | 08A | 125.5e-12 step/tick$^2$ (2008 step/s$^2$) | R, WS |
| h06 | DEC | Deceleration | 12 | 08A | 125.5e-12 step/tick$^2$ (2008step/s$^2$) | R, WS |
| h07 | MAX_SPEED | Maximum speed | 10 | 041 | 248e-6 step/tick (991.8 step/s) | R, WR |
| h08 | MIN_SPEED | Minimum speed | 12 | 000 | 0 step/tick (0 step/s) | R, WS |
| h12 | ADC_OUT | ADC output | 5 | XX (2) | According to ADCIN voltage | R |
| h13 | OCD_TH | OCD threshold | 5 | 8 | 281.25 mV | R, WR |
| h15 | FS_SPD | Full-step speed | 11 | 027 | 150.7e-6 step/tick (602.7 step/s) | R, WR |
| h16 | STEP_MODE | Step mode | 8 | 7 | $\overline{\text{BUSY}}$/SYNC output used as $\overline{\text{BUSY}}$, 128 microsteps, voltage mode | R, WH |
| h17 | ALARM_EN | Alarm enables | 8 | FF | All alarms enabled | R, WS |
| h18 | GATECFG1 | Gate driver configuration | 11 | 0 | $I_{gate}$ = 4 mA, $t_{CC}$ = 125 ns, boost disabled, Clock watchdog disabled | R, WH |
| h19 | GATECFG2 | Gate driver configuration | 8 | 0 | $t_{BLANK}$ = 125 ns, $t_{DT}$ = 125 ns | R, WH |
| h1B | STATUS | Status | 16 | E401 (2) | High impedance state, motor stopped, reverse direction, UVLO/reset flag set, all others fault flags released | R |
| h1A | CONFIG | IC configuration | 16 | 2C88 | Internal 16 MHz oscillator (OSCOUT at 2 MHz), SW event causes HardStop, overcurrent shutdown, $V_{CC}$ = 7.5 V, UVLO threshold low | R, WH |
| **Voltage mode configuration** | | | | | | |
| h09 | KVAL_HOLD | Holding $K_{VAL}$ | 8 | 29 | 0.16 · $V_S$ | R, WR |
| h0A | KVAL_RUN | Constant speed $K_{VAL}$ | 8 | 29 | 0.16 · $V_S$ | R, WR |
| h0B | KVAL_ACC | Acceleration starting $K_{VAL}$ | 8 | 29 | 0.16 · $V_S$ | R, WR |
| h0C | KVAL_DEC | Deceleration starting $K_{VAL}$ | 8 | 29 | 0.16 · $V_S$ | R, WR |
| h0D | INT_SPEED | Intersect speed | 14 | 0408 | 15.4e-6 step/tick (61.5 step/s) | R, WH |
| h0E | ST_SLP | Start slope | 8 | 19 | 0.038 % s/step | R, WH |
| h0F | FN_SLP_ACC | Acceleration final slope | 8 | 29 | 0.063 % s/step | R, WH |
| h10 | FN_SLP_DEC | Deceleration final slope | 8 | 29 | 0.063 % s/step | R, WH |
| h11 | K_THERM | Thermal compensation factor | 4 | 0 | 1.0 | R, WR |

| Address [Hex] | Register name | Register function | Length [bit] | Reset [Hex] | Reset value | Remarks (1) |
|---|---|---|---|---|---|---|
| h14 | STALL_TH | STALL threshold | 5 | 10 | 531.25 mV | R, WR |
| h1A | CONFIG | IC configuration | 16 | 2C88 | Motor supply voltage compensation disabled, $f_{PWM} = f_{OSC}/1024$ | R, WH |
| **Current mode configuration** | | | | | | |
| h09 | TVAL_HOLD | Holding reference voltage | 7 | 29 | 328 mV | R, WR |
| h0A | TVAL_RUN | Constant speed reference voltage | 7 | 29 | 328 mV | R, WR |
| h0B | TVAL_ACC | Acceleration reference voltage | 7 | 29 | 328 mV | R, WR |
| h0C | TVAL_DEC | Deceleration reference voltage | 7 | 29 | 328 mV | R, WR |
| h0E | T_FAST | Fast decay setting | 8 | 19 | 4 µs / 20 µs | R, WH |
| h0F | TON_MIN | Minimum on-time | 7 | 29 | 21 µs | R, WH |
| h10 | TOFF_MIN | Minimum off-time | 7 | 29 | 21 µs | R, WH |
| h1A | CONFIG | IC configuration | 16 | 2C88 | Predictive current control disabled, $t_{SW}$ = 44 µs | R, WH |

1. *R: readable, WH: writable when the outputs are in high impedance only, WS: writable when the motor is stopped or in high impedance only, WR: always writable.*
2. *Actual STATUS value varies according to startup conditions.*

### 11.1.1 ABS_POS

The ABS_POS register contains the current motor absolute position in agreement with the selected step mode; the stored value unit is equal to the selected step mode (full, half, quarter, etc.). The value is in 2 's complement format and it ranges from $-2^{21}$ to $+2^{21} - 1$.

At power-on the register is initialized to “0” (HOME position).

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.2 EL_POS

The EL_POS register contains the current electrical position of the motor. The two MSbits indicate the current step and the other bits indicate the current microstep (expressed in step/128) within the step.

**Table 13. EL_POS register**

| Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| STEP | | MICROSTEP | | | | | | |

When the EL_POS register is written by the user, the new electrical position is instantly imposed. When the EL_POS register is written, its value must be masked in order to match with the step mode selected in the STEP_MODE register in order to avoid a wrong microstep value generation (Section 11.1.23); otherwise the resulting microstep sequence is incorrect. When the device operates in current mode, the bit numbers 0, 1, and 2 of the EL_POS register are meaningless because the maximum microstepping resolution is $1/16^{th}$ of step.

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (Section 11.1.28).

### 11.1.3 MARK

The MARK register contains an absolute position called MARK, according to the selected step mode; the stored value unit is equal to the selected step mode (full, half, quarter, etc.). It is in 2's complement format and it ranges from $-2^{21}$ to $+2^{21}-1$.

### 11.1.4 SPEED

The SPEED register contains the current motor speed, expressed in step/tick (format unsigned fixed point 0.28).

In order to convert the SPEED value in step/s the following formula can be used:

$$\left[step/s\right] = \frac{SPEED \cdot 2^{-28}}{tick} \quad (4)$$

where *SPEED* is the integer number stored in the register and *tick* is 250 ns. The available range is from 0 to 15625 step/s with a resolution of 0.015 step/s.

*Note: The range effectively available to the user is limited by the MAX_SPEED parameter.*

Any attempt to write the register causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28: Status).

### 11.1.5 ACC

The ACC register contains the speed profile acceleration expressed in step/tick$^2$ (format unsigned fixed point 0.40).

In order to convert the ACC value in step/s$^2$ the following formula can be used:

$$\left[step/s^2\right] = \frac{ACC \cdot 2^{-40}}{tick^2} \quad (5)$$

where *ACC* is the integer number stored in the register and *tick* is 250 ns.

The available range is from 14.55 to 59575.541 step/s$^2$ with a resolution of 14.55 step/s$^2$.

The 0xFFF value of the register is reserved and it should never be used. The 0x0 value is considered equivalent to 0x1 (14.55 step/s$^2$).

Any attempt to write to the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.6 DEC

The DEC register contains the speed profile deceleration expressed in step/tick$^2$ (format unsigned fixed point 0.40).

In order to convert the DEC value in step/s$^2$ the following formula can be used:

$$\left[step/s^2\right] = \frac{DEC \cdot 2^{-40}}{tick^2} \quad (6)$$

where *DEC* is the integer number stored in the register and *tick* is 250 ns.

The available range is from 14.55 to 59590 step/s$^2$ with a resolution of 14.55 step/s$^2$. The 0x0 value is considered equivalent to 0x1 (14.55 step/s$^2$).

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.7 MAX_SPEED

The MAX_SPEED register contains the speed profile maximum speed expressed in step/tick (format unsigned fixed point 0.18).

In order to convert it in step/s, the following formula can be used:

$$\left[step/s\right] = \frac{MAX\_SPEED \cdot 2^{-18}}{tick} \quad (7)$$

where *MAX_SPEED* is the integer number stored in the register and *tick* is 250 ns.

The available range is from 15.25 to 15610 step/s with a resolution of 15.25 step/s. The 0x0 value is reserved and it should never be used.

### 11.1.8 MIN_SPEED

The MIN_SPEED register contains the following parameters:

**Table 14. MIN_SPEED register**

| Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LSPD_OPT | MIN_SPEED | | | | | | | | | | | |

The MIN_SPEED parameter contains the speed profile minimum speed. Its value is expressed in step/tick and to convert it in step/s the following formula can be used:

$$\left[step/s\right] = \frac{MIN\_SPEED \cdot 2^{-24}}{tick} \quad (8)$$

where *MIN_SPEED* is the integer number stored in the register and *tick* is 250 ns.

The available range is from 0 to 976.3 step/s with a resolution of 0.238 step/s.

When the LSPD_OPT bit is set high, the low speed optimization feature is enabled (voltage mode driving only) and the MIN_SPEED value indicates the speed threshold below which the compensation works. In this case the minimum speed of the speed profile is set to zero.

Any attempt to write the register when the motor is running causes the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.9 FS_SPD

The FS_SPD register contains the following parameters:

**Table 15. FS_SPD register**

| Bit 10 | Bit 9 | Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| BOOST_MODE | FS_SPD | | | | | | | | | |

The FS_SPD threshold speed value over which the step mode is automatically switched to full-step two-phase on. Its value is expressed in step/tick (format unsigned fixed point 0.18) and to convert it in step/s the following formula can be used:

$$\left[step/s\right] = \frac{\left(FS\_SPD + 0.5\right) \cdot 2^{-18}}{tick} \quad (9)$$

If the FS_SPD value is set to h3FF (max.) the system always works in microstepping mode (SPEED must go over the threshold to switch to full-step mode). Setting FS_SPD to zero does not have the same effect as setting the step mode to full-step two-phase on: the zero FS_SPD value is equivalent to a speed threshold of about 7.63 step/s.

The available range is from 7.63 to 15617 step/s with a resolution of 15.25 step/s.

The BOOST_MODE bit sets the amplitude of the voltage squarewave during the full-step operation (see Section 7.4.1).

### 11.1.10 KVAL_HOLD, KVAL_RUN, KVAL_ACC and KVAL_DEC

These registers are specific for voltage mode driving (see Section 8: Phase current control: voltage mode).

The KVAL_HOLD register contains the $K_{VAL}$ value that is assigned to the PWM modulators when the motor is stopped (compensations excluded).

The KVAL_RUN register contains the $K_{VAL}$ value that is assigned to the PWM modulators when the motor is running at constant speed (compensations excluded).

The KVAL_ACC register contains the starting $K_{VAL}$ value that can be assigned to the PWM modulators during acceleration (compensations excluded).

The KVAL_DEC register contains the starting $K_{VAL}$ value that can be assigned to the PWM modulators during deceleration (compensations excluded).

The available range is from 0 to 0.996 x $V_S$ with a resolution of 0.004 x $V_S$, as shown in Table 16.

**Table 16. Voltage amplitude regulation registers**

| KVAL_X [7…0] | | | | | | | | Output voltage |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | $V_S$ x (1/256) |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | $V_S$ x (254/256) |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | $V_S$ x (255/256) |

### 11.1.11 INT_SPEED

This register is specific for voltage mode driving (see Section 8).

The INT_SPEED register contains the speed value at which the BEMF compensation curve changes slope (Section 8.4 for details). Its value is expressed in step/tick and to convert it in [step/s] the following formula can be used:

$$\left[step/s\right] = \frac{INT\_SPEED \cdot 2^{-26}}{tick} \quad (10)$$

where *INT_SPEED* is the integer number stored in the register and *tick* is 250 ns.

The available range is from 0 to 976.5 step/s with a resolution of 0.0596 step/s.

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.12 ST_SLP

This register is specific for voltage mode driving (see Section 8).

The ST_SLP register contains the BEMF compensation curve slope that is used when the speed is lower than the intersect speed (Section 8.4 for details). Its value is expressed in s/step and the available range is from 0 to 0.004 with a resolution of 0.000015.

When ST_SLP, FN_SLP_ACC, and FN_SLP_DEC parameters are set to zero, no BEMF compensation is performed.

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.13 FN_SLP_ACC

This register is specific for voltage mode driving (see Section 8).

The FN_SLP_ACC register contains the BEMF compensation curve slope that is used when the speed is greater than the intersect speed during acceleration (see Section 8.4 for details). Its value is expressed in s/step and the available range is from 0 to 0.004 with a resolution of 0.000015.

When ST_SLP, FN_SLP_ACC, and FN_SLP_DEC parameters are set to zero, no BEMF compensation is performed.

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.14 FN_SLP_DEC

This register is specific for voltage mode driving (see Section 8).

The FN_SLP_DEC register contains the BEMF compensation curve slope that is used when the speed is greater than the intersect speed during deceleration (Section 8.4 for details). Its value is expressed in s/step and the available range is from 0 to 0.004 with a resolution of 0.000015.

When ST_SLP, FN_SLP_ACC, and FN_SLP_DEC parameters are set to zero, no BEMF compensation is performed.

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.15 K_THERM

This register is specific for voltage mode driving (see Section 8).

The K_THERM register contains the value used by the winding resistance thermal drift compensation system (see Section 8.6).

The available range is from 1 to 1.46875 with a resolution of 0.03125, as shown in Table 17.

**Table 17. Winding resistance thermal drift compensation coefficient**

| K_THERM [3 … 0] | | | | Compensation coefficient |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1.03125 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 0 | 1.4375 |
| 1 | 1 | 1 | 1 | 1.46875 |

### 11.1.16 TVAL_HOLD, TVAL_RUN, TVAL_ACC, and TVAL_DEC

These registers are specific for current mode driving (see Section 9).

The TVAL_HOLD register contains the reference voltage that is assigned to the torque regulation DAC when the motor is stopped.

The TVAL_RUN register contains the reference voltage that is assigned to the torque regulation DAC when the motor is running at constant speed.

The TVAL_ACC register contains the reference voltage that is assigned to the torque regulation DAC during acceleration.

The TVAL_DEC register contains the reference voltage that is assigned to the torque regulation DAC during deceleration.

The available range is from 7.8 mV to 1 V with a resolution of 7.8 mV, as shown in Table 18.

**Table 18. Torque regulation by TVAL_HOLD, TVAL_ACC, TVAL_DEC, and TVAL_RUN registers**

| TVAL_X [6…0] | | | | | | | Peak reference voltage |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 7.8 mV |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 15.6 mV |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 992.2 mV |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 V |

### 11.1.17 T_FAST

This register is specific for current mode driving (see Section 9).

The T_FAST register contains the maximum fast decay time (TOFF_FAST) and the maximum fall step time (FAST_STEP) used by the current control system (see Section 9.2 and Section 9.3 for details):

**Table 19. T_FAST register**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| TOFF_FAST | | | | FAST_STEP | | | |

The available range for both parameters is from 2 µs to 32 µs.

**Table 20. Maximum fast decay times**

| TOFF_FAST [3 … 0] / FAST_STEP [3 … 0] | | | | Fast decay time |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 2 µs |
| 0 | 0 | 0 | 1 | 4 µs |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 0 | 28 µs |
| 1 | 1 | 1 | 1 | 32 µs |

Any attempt to write to the register when the motor is running causes the command to be ignored and CMD_ERROR to rise (see Section 11.1.28).

### 11.1.18 TON_MIN

This register is specific for current mode driving (see Section 9).

This parameter is used by the current control system when current mode operation is selected.

The TON_MIN register contains the minimum on-time value used by the current control system (see Section 9.2).

The available range is from 0.5 µs to 64 µs.

**Table 21. Minimum on-time**

| TON MIN [6 … 0] | | | | | | | Time |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 µs |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 µs |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 63.5 µs |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 64 µs |

Any attempt to write to the register when the motor is running causes the command to be ignored and CMD_ERROR to rise (see Section 11.1.28).

### 11.1.19 TOFF_MIN

This register is specific for current mode driving (see Section 9).

This parameter is used by the current control system when current mode operation is selected.

The TOFF_MIN register contains the minimum off-time value used by the current control system (see Section 9.1).

The available range is from 0.5 µs to 64 µs.

Table 22. **Minimum off-time**

| TOFF MIN [6 … 0] | | | | | | | Time |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.5 µs |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 µs |
| … | … | … | … | … | … | … | … |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 63.5 µs |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 64 µs |

Any attempt to write to the register when the motor is running causes the command to be ignored and CMD_ERROR to rise (see Section 11.1.28).

### 11.1.20 ADC_OUT

The ADC_OUT register contains the result of the analog-to-digital conversion of the ADCIN pin voltage.

Any attempt to write to the register causes the command to be ignored and CMD_ERROR to rise (see Section 11.1.28: Status).

Table 23. **ADC_OUT value and motor supply voltage compensation feature**

| $V_S$ | $V_{ADCIN}/V_{REG}$ | ADC_OUT [4 … 0] | | | | | Compensation coefficient |
|---|---|---|---|---|---|---|---|
| Greater than $V_{S,nom}$ + 50% | > 24/32 | 1 | 1 | X | X | X | 0.65625 |
| $V_{S,nom}$ + 50% | 24/32 | 1 | 1 | 0 | 0 | 0 | 0.65625 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| $V_{S,nom}$ | 16/32 | 1 | 0 | 0 | 0 | 0 | 1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| $V_{S,nom}$ – 50% | 8/32 | 0 | 1 | 0 | 0 | 0 | 1.968875 |
| Lesser than $V_{S,nom}$ – 50% | < 8/32 | 0 | 0 | X | X | X | 1.968875 |

Table 24. **ADC_OUT value and torque regulation feature**

| $V_{ADCIN}/V_{REG}$ | ADC_OUT [4 … 0] | | | | | Reference voltage |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 31.25 mV |
| 1/32 | 0 | 0 | 0 | 0 | 1 | 62.5 mV |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 30/32 | 1 | 1 | 1 | 1 | 0 | 968.8 mV |
| 31/32 | 1 | 1 | 1 | 1 | 1 | 1000 mV |

### 11.1.21 OCD_TH

The OCD_TH register contains the overcurrent threshold value (see Section 7.9 for details).
The available range is from 31.25 mV to 1 V, with steps of 31.25 mV, as shown in Table 25.

**Table 25. Overcurrent detection threshold**

| OCD_TH [4 … 0] | | | | | Overcurrent detection threshold |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 31.25 mV |
| 0 | 0 | 0 | 0 | 1 | 62.5 mV |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 1 | 0 | 968.75 mV |
| 1 | 1 | 1 | 1 | 1 | 1 V |

### 11.1.22 STALL_TH

This register is specific for voltage mode driving (see Section 8).
The STALL_TH register contains the stall detection threshold value.
The available range is from 31.25 mV to 1 V with a resolution of 31.25 mV.

**Table 26. Stall detection threshold**

| STALL_TH [4 … 0] | | | | | Stall detection threshold |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 31.25 mV |
| 0 | 0 | 0 | 0 | 1 | 62.5 mV |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 1 | 0 | 968.75 mV |
| 1 | 1 | 1 | 1 | 1 | 1 V |

### 11.1.23 STEP_MODE

The STEP_MODE register has the following structure:

**Table 27. STEP_MODE register**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| SYNC_EN | SYNC_SEL | | | CM_VM | STEP_SEL | | |

The CM_VM bit sets the current control method between voltage and current mode:

**Table 28. Control mode selection**

| CM_VM | Control mode |
|---|---|
| 0 | Voltage mode |
| 1 | Advanced current control (current mode) |

STEP_SEL selects one of eight possible stepping modes:

**Table 29. Step mode selection**

| STEP_SEL [2 … 0] | | | Step mode (CM_VM = 0) | Step mode (CM_VM = 1) |
|---|---|---|---|---|
| 0 | 0 | 0 | Full-step | Full-step |
| 0 | 0 | 1 | Half-step | Half-step |
| 0 | 1 | 0 | 1/4 microstep | 1/4 microstep |
| 0 | 1 | 1 | 1/8 microstep | 1/8 microstep |
| 1 | 0 | 0 | 1/16 microstep | 1/16 microstep |
| 1 | 0 | 1 | 1/32 microstep | 1/16 microstep |
| 1 | 1 | 0 | 1/64 microstep | 1/16 microstep |
| 1 | 1 | 1 | 1/128 microstep | 1/16 microstep |

Every time the step mode changes, the electrical position (that is, the point of microstepping sine wave that is generated) resets at the first microstep.

It is important to note that every time STEP_SEL changes, the value in the ABS_POS register loses meaning and it should be then reset.

When sync clock is disabled, $\overline{\text{BUSY}}$/SYNC output is used as BUSY (command state machine busy signaling), otherwise $\overline{\text{BUSY}}$/SYNC output provides a clock signal according to the SYNC_SEL parameter.

**Table 30. Sync clock enable**

| SYNC_EN | Sync clock |
|---|---|
| 0 | Disabled |
| 1 | Enabled |

**Table 31. Sync clock selection**

| SYNC_SEL [2 … 0] | | | Step information (CM_VM = 0) | Step information (CM_VM = 1) |
|---|---|---|---|---|
| 0 | 0 | 0 | Full-step | Full-step |
| 0 | 0 | 1 | Half-step | Half-step |
| 0 | 1 | 0 | 1/4 microstep | 1/4 microstep |
| 0 | 1 | 1 | 1/8 microstep | 1/8 microstep |
| 1 | 0 | 0 | 1/16 microstep | 1/16 microstep |
| 1 | 0 | 1 | 1/32 microstep | Always low |
| 1 | 1 | 0 | 1/64 microstep | Always low |
| 1 | 1 | 1 | 1/128 microstep | Always low |

Any attempt to write to the register, when the motor is running, causes the command to be ignored and CMD_ERROR to rise (see Section 11.1.28).

### 11.1.24 ALARM_EN

The ALARM_EN register allows the selection of which alarm signals are used to generate the $\overline{\text{FLAG}}$ output. If the respective bit of the ALARM_EN register is set high, the alarm condition forces the $\overline{\text{FLAG}}$ pin output down.

**Table 32. ALARM_EN register**

| ALARM_EN bit | Alarm condition |
|---|---|
| 0 (LSB) | Overcurrent |
| 1 | Thermal shutdown |
| 2 | Thermal warning |
| 3 | UVLO |
| 4 | ADC UVLO |
| 5 | Stall detection (voltage mode only) |
| 6 | Switch turn-on event |
| 7 (MSB) | Command error |

### 11.1.25 GATECFG1

The GATECFG1 register has the following structure:

**Table 33. GATECFG1 register**

| Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 |
|---|---|---|---|---|---|---|---|
| | | | | WD_EN | TBOOST | | |
| **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| IGATE | | | TCC | | | | |

The IGATE parameter selects the sink/source current used by gate driving circuitry to charge/discharge the respective gate during commutations. Seven possible values ranging from 4 mA to 96 mA are available, as shown in Table 34.

**Table 34. IGATE parameter**

| IGATE [2 … 0] | | | Gate current [mA] |
|---|---|---|---|
| 0 | 0 | 0 | 4 |
| 0 | 0 | 1 | 4 |
| 0 | 1 | 0 | 8 |
| 0 | 1 | 1 | 16 |
| 1 | 0 | 0 | 24 |
| 1 | 0 | 1 | 32 |
| 1 | 1 | 0 | 64 |
| 1 | 1 | 1 | 96 |

The TCC parameter defines the duration of the constant current phase during gate turn-on and turn-off sequences (see Section 7.16).

**Table 35. TCC parameter**

| TCC [4 … 0] | | | | | Constant current time [ns] |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 125 |
| 0 | 0 | 0 | 0 | 1 | 250 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 0 | 0 | 3625 |
| 1 | 1 | 1 | 0 | 1 | 3750 |
| 1 | 1 | 1 | 1 | 0 | 3750 |
| 1 | 1 | 1 | 1 | 1 | 3750 |

The TBOOST parameter defines the duration of the overboost phase during gate turn-off (see Section 7.16).

**Table 36. TBOOST parameter**

| TBOOST [2 … 0] | | | Turn-off boost time[ns] |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 62.5 (1) / 83.3 (2) / 125 (3) |
| 0 | 1 | 0 | 125 |
| 0 | 1 | 1 | 250 |
| 1 | 0 | 0 | 375 |
| 1 | 0 | 1 | 500 |
| 1 | 1 | 0 | 750 |
| 1 | 1 | 1 | 1000 |

1. *Clock frequency equal to 16 MHz or 32 MHz.*
2. *Clock frequency equal to 24 MHz.*
3. *Clock frequency equal to 8 MHz.*

The WD_EN bit enables the clock source monitoring (see Section 7.8.2).

### 11.1.26 GATECFG2

The GATECFG2 register has the following structure:

**Table 37. GATECFG2 register**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|
| TBLANK | | | TDT | | | | |

The TDT parameter defines the deadtime duration between the gate turn-off and the opposite gate turn-on sequences (see Section 7.17).

**Table 38. TDT parameter**

| TDT [4 … 0] | | | | | Deadtime [ns] |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 125 |
| 0 | 0 | 0 | 0 | 1 | 250 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 1 | 0 | 3875 |
| 1 | 1 | 1 | 1 | 1 | 4000 |

The TBLANK parameter defines the duration of the blanking of the current sensing comparators (stall detection and overcurrent) after each commutation (see Section 7.17).

**Table 39. TBLANK parameter**

| TBLANK [2 … 0] | | | Blanking time [ns] |
|---|---|---|---|
| 0 | 0 | 0 | 125 |
| 0 | 0 | 1 | 250 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 0 | 875 |
| 1 | 1 | 1 | 1000 |

### 11.1.27 CONFIG

The CONFIG register has the following structure:

**Table 40. CONFIG register**

| CM_VM | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 |
|---|---|---|---|---|---|---|---|---|
| 0 | F_PWM_INT | | | F_PWM_DEC | | | VCCVAL | UVLOVAL |
| 1 | PRED_EN | TSW | | | | | VCCVAL | UVLOVAL |

| CM_VM | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|
| 0 | OC_SD | | EN_VSCOMP | SW_MODE | EXT_CLK | OSC_SEL | | |
| 1 | OC_SD | | EN_TQREG | SW_MODE | EXT_CLK | OSC_SEL | | |

The OSC_SEL and EXT_CLK bits set the system clock source:

**Table 41. Oscillator management**

| EXT_CLK | OSC_SEL [2 … 0] | | | Clock source | OSCIN | OSCOUT |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | Internal oscillator: 16 MHz | Unused | Unused |
| 0 | 0 | 0 | 1 | | | |
| 0 | 0 | 1 | 0 | | | |
| 0 | 0 | 1 | 1 | | | |
| 1 | 0 | 0 | 0 | Internal oscillator: 16 MHz | Unused | Supplies a 2 MHz clock |
| 1 | 0 | 0 | 1 | Internal oscillator: 16 MHz | Unused | Supplies a 4 MHz clock |
| 1 | 0 | 1 | 0 | Internal oscillator: 16 MHz | Unused | Supplies an 8 MHz clock |
| 1 | 0 | 1 | 1 | Internal oscillator: 16 MHz | Unused | Supplies a 16 MHz clock |
| 0 | 1 | 0 | 0 | External crystal or resonator: 8 MHz | Crystal/resonator driving | Crystal/resonator driving |
| 0 | 1 | 0 | 1 | External crystal or resonator: 16 MHz | Crystal/resonator driving | Crystal/resonator driving |
| 0 | 1 | 1 | 0 | External crystal or resonator: 24 MHz | Crystal/resonator driving | Crystal/resonator driving |
| 0 | 1 | 1 | 1 | External crystal or resonator: 32 MHz | Crystal/resonator driving | Crystal/resonator driving |
| 1 | 1 | 0 | 0 | External clock source: 8 MHz (crystal/resonator driver disabled) | Clock source | Supplies inverted OSCIN signal |
| 1 | 1 | 0 | 1 | External clock source: 16 MHz (crystal/resonator driver disabled) | Clock source | Supplies inverted OSCIN signal |
| 1 | 1 | 1 | 0 | External clock source: 24 MHz (crystal/resonator driver disabled) | Clock source | Supplies inverted OSCIN signal |
| 1 | 1 | 1 | 1 | External clock source: 32 MHz (crystal/resonator driver disabled) | Clock source | Supplies inverted OSCIN signal |

The SW_MODE bit sets the external switch to act as HardStop interrupt or not:

**Table 42. External switch hard stop interrupt mode**

| SW_MODE | Switch mode |
|---|---|
| 0 | HardStop interrupt |
| 1 | User disposal |

The OC_SD bit sets if an overcurrent event causes or not the bridges to turn off; the OCD flag in the status register is forced low anyway:

**Table 43. Overcurrent event**

| OC_SD | Overcurrent event |
|---|---|
| 1 | Bridges shut down |
| 0 | Bridges do not shut down |

The VCCVAL bit sets the internal $V_{CC}$ regulator output voltage:

**Table 44. Programmable $V_{CC}$ voltage regulator output**

| VCCVAL | $V_{CC}$ voltage |
|---|---|
| 0 | 7.5 V |
| 1 | 15 V |

The UVLOVAL bit sets the UVLO protection thresholds:

**Table 45. Programmable UVLO thresholds**

| UVLOVAL | $V_{ccthOn}$ | $V_{ccthOff}$ | $DV_{BOOTThOn}$ | $DV_{BOOTThOff}$ |
|---|---|---|---|---|
| 0 | 6.9 V | 6.3 V | 6 V | 5.5 V |
| 1 | 10.4 V | 10 V | 9.2 V | 8.8 V |

When the device operates in voltage mode, the EN_VSCOMP bit sets if the motor supply voltage compensation is enabled or not.

**Table 46. Motor supply voltage compensation enable**

| EN_VSCOMP | Motor supply voltage compensation |
|---|---|
| 0 | Disabled |
| 1 | Enabled |

When the device operates in voltage mode, the F_PWM_INT bits set the integer division factor of PWM frequency generation:

**Table 47. PWM frequency: integer division factor**

| F_PWM_INT [2 … 0] | | | Integer division factor |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 2 |
| 0 | 1 | 0 | 3 |
| 0 | 1 | 1 | 4 |
| 1 | 0 | 0 | 5 |
| 1 | 0 | 1 | 6 |
| 1 | 1 | 0 | 7 |
| 1 | 1 | 1 | |

When the device operates in voltage mode, the F_PWM_DEC bits set the multiplication factor of PWM frequency generation:

**Table 48. PWM frequency: multiplication factor**

| F_PWM_DEC [2 … 0] | | | Multiplication factor |
|---|---|---|---|
| 0 | 0 | 0 | 0.625 |
| 0 | 0 | 1 | 0.75 |
| 0 | 1 | 0 | 0.875 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1.25 |
| 1 | 0 | 1 | 1.5 |
| 1 | 1 | 0 | 1.75 |
| 1 | 1 | 1 | 2 |

In the following tables all available PWM frequencies are listed according to oscillator frequency, F_PWM_INT and F_PWM_DEC values (CONFIG register OSC_SEL parameter has to be correctly programmed).

**Table 49. Available PWM frequencies [kHz]: 16 MHz oscillator frequency**

| | F_PWM_DEC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| F_PWM_INT | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 000 | 19.5 | 23.4 | 27.3 | 31.3 | 39.1 | 46.9 | 54.7 | 62.5 |
| 001 | 9.8 | 11.7 | 13.7 | 15.6 | 19.5 | 23.4 | 27.3 | 31.3 |
| 010 | 6.5 | 7.8 | 9.1 | 10.4 | 13.0 | 15.6 | 18.2 | 20.8 |
| 011 | 4.9 | 5.9 | 6.8 | 7.8 | 9.8 | 11.7 | 13.7 | 15.6 |
| 100 | 3.9 | 4.7 | 5.5 | 6.3 | 7.8 | 9.4 | 10.9 | 12.5 |
| 101 | 3.3 | 3.9 | 4.6 | 5.2 | 6.5 | 7.8 | 9.1 | 10.4 |
| 110 | 2.8 | 3.3 | 3.9 | 4.5 | 5.6 | 6.7 | 7.8 | 8.9 |

**Table 50. Available PWM frequencies [kHz]: 24 MHz oscillator frequency**

| | F_PWM_DEC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| F_PWM_INT | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 000 | 29.3 | 35.2 | 41.0 | 46.9 | 58.6 | 70.3 | 82.0 | 93.8 |
| 001 | 14.6 | 17.6 | 20.5 | 23.4 | 29.3 | 35.2 | 41.0 | 46.9 |
| 010 | 9.8 | 11.7 | 13.7 | 15.6 | 19.5 | 23.4 | 27.3 | 31.3 |
| 011 | 7.3 | 8.8 | 10.3 | 11.7 | 14.6 | 17.6 | 20.5 | 23.4 |
| 100 | 5.9 | 7.0 | 8.2 | 9.4 | 11.7 | 14.1 | 16.4 | 18.8 |
| 101 | 4.9 | 5.9 | 6.8 | 7.8 | 9.8 | 11.7 | 13.7 | 15.6 |
| 110 | 4.2 | 5.0 | 5.9 | 6.7 | 8.4 | 10.0 | 11.7 | 13.4 |

Table 51. **Available PWM frequencies [kHz]: 32 MHz oscillator frequency**

| | F_PWM_DEC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| F_PWM_INT | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
| 000 | 39.1 | 46.9 | 54.7 | 62.5 | 78.1 | 93.8 | 109.4 | 125.0 |
| 001 | 19.5 | 23.4 | 27.3 | 31.3 | 39.1 | 46.9 | 54.7 | 62.5 |
| 010 | 13.0 | 15.6 | 18.2 | 20.8 | 26.0 | 31.3 | 36.5 | 41.7 |
| 011 | 9.8 | 11.7 | 13.7 | 15.6 | 19.5 | 23.4 | 27.3 | 31.3 |
| 100 | 7.8 | 9.4 | 10.9 | 12.5 | 15.6 | 18.8 | 21.9 | 25.0 |
| 101 | 6.5 | 7.8 | 9.1 | 10.4 | 13.0 | 15.6 | 18.2 | 20.8 |
| 110 | 5.6 | 6.7 | 7.8 | 8.9 | 11.2 | 13.4 | 15.6 | 17.9 |

When the device operates in current mode, the EN_TQREG bit sets if the peak current is adjusted through the ADCIN input or not.

Table 52. **External torque regulation enable**

| EN_TQREG | Peak current adjust through ADCIN |
|---|---|
| 0 | Disabled |
| 1 | Enabled |

When the device operates in current mode, the PRED_EN bit sets if the predictive current control method is enabled or not.

Table 53. **Motor supply voltage compensation enable**

| PRED_EN | Predictive current control |
|---|---|
| 0 | Disabled |
| 1 | Enabled |

When the device operates in current mode, the TSW bits set the target switching period of the current control algorithm:

Table 54. **Switching period**

| TSW [4 … 0] | | | | | Switching period |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 4 µs (250 kHz) |
| 0 | 0 | 0 | 0 | 1 | 4 µs (250 kHz) |
| 0 | 0 | 0 | 1 | 0 | 8 µs (125 kHz) |
| … | | | | | … |
| 1 | 1 | 1 | 1 | 1 | 124 µs (8 kHz) |

Any attempt to write the CONFIG register when the outputs are enabled causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.1.28 Status

The STATUS register has the following structure:

**Table 55. Status register**

| Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 |
|---|---|---|---|---|---|---|---|
| STALL_A | STALL_B | OCD | TH_STATUS | | UVLO_ADC | UVLO | STCK_MOD |
| **Bit 7** | **Bit 6** | **Bit 5** | **Bit 4** | **Bit 3** | **Bit 2** | **Bit 1** | **Bit 0** |
| CMD_ERROR | MOT_STATUS | | DIR | SW_EVN | SW_F | BUSY | HiZ |

When the HiZ flag is high it indicates that the bridges are in high impedance state. Whichever motion command makes the device to exit from high-Z state (HardStop and SoftStop included), unless error flags forcing a high-Z state are active.

The UVLO flag is active low and is set by an undervoltage lockout or reset events (power-up included).

The UVLO_ADC flag is active low and indicates an ADC undervoltage event.

The OCD flag is active low and indicates an overcurrent detection event.

The STALL_A and STALL_B flags are forced low when a stall condition is detected on bridge A or bridge B respectively. The stall detection is operative only when the voltage mode control is selected.

The CMD_ERROR flag is active high and indicates that the command received by SPI cannot be performed or does not exist at all.

The SW_F reports the SW input status (low for open and high for closed).

The SW_EVN flag is active high and indicates a switch turn-on event (SW input falling edge).

TH_STATUS bits indicate the current device thermal status (see Section 7.12):

**Table 56. Status register TH_STATUS bits**

| TH_STATUS | | Status |
|---|---|---|
| 0 | 0 | Normal |
| 0 | 1 | Warning |
| 1 | 0 | Bridge shutdown |
| 1 | 1 | Device shutdown |

UVLO, UVLO_ADC, OCD, STALL_A, STALL_B, CMD_ERROR, SW_EVN, and TH_STATUS bits are latched: when the respective conditions make them active (low or high), they remain in that state until a GetStatus command is sent to the IC.

The BUSY bit reflects the $\overline{\text{BUSY}}$ pin status. The BUSY flag is low when a constant speed, positioning, or motion command is under execution and is released (high) after the command has been completed.

The STCK_MOD bit is an active high flag indicating that the device is working in step-clock mode. In this case the step clock signal should be provided through the STCK input pin.

The DIR bit indicates the current motor direction:

**Table 57. Status register DIR bit**

| DIR | Motor direction |
|---|---|
| 1 | Forward |
| 0 | Reverse |

MOT_STATUS indicates the current motor status:

**Table 58. Status register MOT_STATUS bits**

| MOT_STATUS | | Motor status |
|---|---|---|
| 0 | 0 | Stopped |
| 0 | 1 | Acceleration |
| 1 | 0 | Deceleration |
| 1 | 1 | Constant speed |

Any attempt to write to the register causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28: Status).

## 11.2 Application commands

The command summary is given in Table 59.

**Table 59. Application commands**

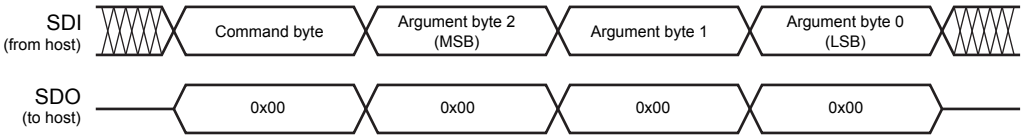
| Command mnemonic | Command binary code | | | | | Action |
|---|---|---|---|---|---|---|
| | [7…5] | [4] | [3] | [2…1] | [0] | |
| NOP | 000 | 0 | 0 | 00 | 0 | Nothing |
| SetParam(PARAM,VALUE) | 000 | [PARAM] | | | | Writes VALUE in PARAM register |
| GetParam(PARAM) | 001 | [PARAM] | | | | Returns the stored value in PARAM register |
| Run(DIR,SPD) | 010 | 1 | 0 | 00 | DIR | Sets the target speed and the motor direction |
| StepClock(DIR) | 010 | 1 | 1 | 00 | DIR | Puts the device in step-clock mode and imposes DIR direction |
| Move(DIR,N_STEP) | 010 | 0 | 0 | 00 | DIR | Makes N_STEP (micro)steps in DIR direction (not performable when motor is running) |
| GoTo(ABS_POS) | 011 | 0 | 0 | 00 | 0 | Brings motor in ABS_POS position (minimum path) |
| GoTo_DIR(DIR,ABS_POS) | 011 | 0 | 1 | 00 | DIR | Brings motor in ABS_POS position forcing DIR direction |
| GoUntil(ACT,DIR,SPD) | 100 | 0 | ACT | 01 | DIR | Performs a motion in DIR direction with speed SPD until SW is closed, the ACT action is executed then a SoftStop takes place |
| ReleseSW(ACT, DIR) | 100 | 1 | ACT | 01 | DIR | Performs a motion in DIR direction at minimum speed until the SW is released (open), the ACT action is executed then a HardStop takes place |
| GoHome | 011 | 1 | 0 | 00 | 0 | Brings the motor in HOME position |
| GoMark | 011 | 1 | 1 | 00 | 0 | Brings the motor in MARK position |
| ResetPos | 110 | 1 | 1 | 00 | 0 | Resets the ABS_POS register (sets HOME position) |
| ResetDevice | 110 | 0 | 0 | 00 | 0 | Device is reset to power-up conditions |
| SoftStop | 101 | 1 | 0 | 00 | 0 | Stops motor with a deceleration phase |
| HardStop | 101 | 1 | 1 | 00 | 0 | Stops motor immediately |
| SoftHiZ | 101 | 0 | 0 | 00 | 0 | Puts the bridges in high impedance status after a deceleration phase |
| HardHiZ | 101 | 0 | 1 | 00 | 0 | Puts the bridges in high impedance status immediately |
| GetStatus | 110 | 1 | 0 | 00 | 0 | Returns the status register value |
| RESERVED | 111 | 0 | 1 | 01 | 1 | RESERVED COMMAND |
| RESERVED | 111 | 1 | 1 | 00 | 0 | RESERVED COMMAND |

### 11.2.1 Command management

The host microcontroller can control motor's motion and configure the powerSTEP01 through a complete set of commands.

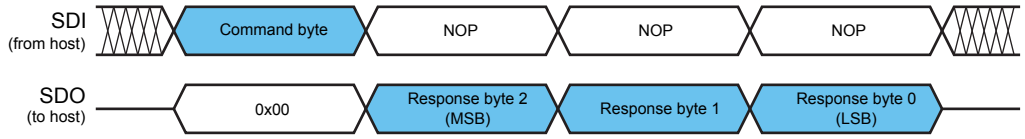
All commands are composed by a single byte. After the command byte, some bytes of argument should be needed (see Figure 27). Argument length can vary from 1 to 3 bytes.

**Figure 27. Command with 3-byte argument**



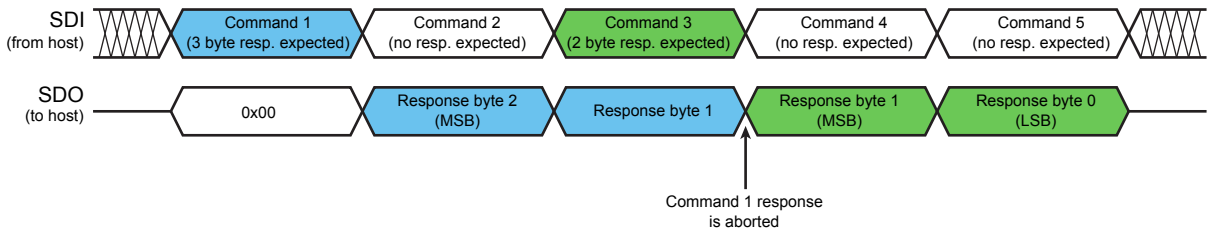

By default, the device returns an all zero response for any received byte; the only exceptions are GetParam and GetStatus commands. When one of these commands is received, the following response bytes represent the related register value (see Figure 28). Response length can vary from 1 to 3 bytes.

**Figure 28. Command with 3-byte response**




During response transmission, new commands can be sent. If a command requiring a response is sent before the previous response is completed, the response transmission is aborted and the new response is loaded into the output communication buffer (see Figure 29).

**Figure 29. Command response aborted**




When a byte that does not correspond to a command is sent to the IC it is ignored and the CMD_ERROR flag in the STATUS register is raised (see paragraph Section 11.1.28).

### 11.2.2 Nop

**Table 60. Nop command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | From host |

Nothing is performed.

### 11.2.3 SetParam (PARAM,VALUE)

**Table 61. SetParam command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | PARAM | | | | | From host |
| VALUE byte 2 (if needed) | | | | | | | | From host |
| VALUE byte 1 (if needed) | | | | | | | | From host |
| VALUE byte 0 | | | | | | | | From host |

The SetParam command sets the PARAM register value equal to VALUE; PARAM is the respective register address listed in Table 12.

The command should be followed by the new register VALUE (most significant byte first). The number of bytes composing the VALUE argument depends on the length of the target register (see Table 12).

Some registers cannot be written (see Table 12); any attempt to write one of those registers causes the command to be ignored and the CMD_ERROR flag to rise at the end of the command byte, as if an unknown command code were sent (see Section 11.1.28).

Some registers can only be written in particular conditions (see Table 12); any attempt to write one of those registers when the conditions are not satisfied causes the command to be ignored and the CMD_ERROR flag to rise at the end of the last argument byte (see Section 11.1.28).

Any attempt to set a nonexistent register (wrong address value) causes the command to be ignored and the CMD_ERROR flag to rise at the end of the command byte as if an unknown command code were sent.

### 11.2.4 GetParam (PARAM)

**Table 62. GetParam command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | PARAM | | | | | From host |
| | | | | | | | | |
| ANS byte 2 (if needed) | | | | | | | | To host |
| ANS byte 1 (if needed) | | | | | | | | To host |
| ANS byte 0 | | | | | | | | To host |

This command reads the current value of PARAM register; PARAM is the respective register address listed in Table 12.

The command response is the current value of the register (most significant byte first). The number of bytes composing the command response depends on the length of the target register (see Table 12).

The returned value is the one available at the moment of GetParam command decoding. If register values change after this moment, the response is not accordingly updated.

All registers can be read anytime.

Any attempt to read a nonexistent register (wrong address value) causes the command to be ignored and the CMD_ERROR flag to rise at the end of the command byte as if an unknown command code were sent.

### 11.2.5 Run (DIR, SPD)

**Table 63. Run command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 | 0 | 0 | DIR | From host |
| X | X | X | X | SPD (byte 2) | | | | From host |
| SPD (byte 1) | | | | | | | | From host |
| SPD (byte 0) | | | | | | | | From host |

The Run command produces a motion at SPD speed; the direction is selected by the DIR bit: '1' forward or '0' reverse. The SPD value is expressed in step/tick (format unsigned fixed point 0.28) that is the same format as the SPEED register (see Section 11.1.4).

*Note: The SPD value should be lower than MAX_SPEED and greater than MIN_SPEED, otherwise the Run command is executed at MAX_SPEED or MIN_SPEED respectively.*

This command keeps the BUSY flag low until the target speed is reached.

This command can be given anytime and is immediately executed.

### 11.2.6 StepClock (DIR)

**Table 64. StepClock command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 1 | 0 | 0 | DIR | From host |

The StepClock command switches the device in step-clock mode (see Section 7.7.5) and imposes the forward (DIR = '1') or reverse (DIR = '0') direction.

When the device is in step-clock mode, the SCK_MOD flag in the STATUS register is raised and the motor is always considered stopped (see Section 7.7.5 and Section 11.1.28).

The device exits step-clock mode when a constant speed, absolute positioning, or motion command is sent through SPI. Motion direction is imposed by the respective StepClock command argument and can be changed by a new StepClock command without exiting step-clock mode.

Events that cause bridges to be forced into a high impedance state (overtemperature, overcurrent, etc.) do not cause the device to leave step-clock mode.

The StepClock command does not force the BUSY flag low. This command can only be given when the motor is stopped. If a motion is in progress, the motor should be stopped and it is then possible to send a StepClock command.

Any attempt to perform a StepClock command when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.2.7 Move (DIR, N_STEP)

**Table 65. Move command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | DIR | From host |
| X | X | N_STEP (byte 2) | | | | | | From host |
| N_STEP (byte 1) | | | | | | | | From host |
| N_STEP (byte 0) | | | | | | | | From host |

The Move command produces a motion of N_STEP microsteps; the direction is selected by the DIR bit ('1' forward or '0' reverse).

The N_STEP value is always in agreement with the selected step mode; the parameter value unit is equal to the selected step mode (full, half, quarter, etc.).

This command keeps the BUSY flag low until the target number of steps is performed. This command can only be performed when the motor is stopped. If a motion is in progress, the motor must be stopped and it is then possible to perform a move command.

Any attempt to perform a move command when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.2.8 GoTo (ABS_POS)

**Table 66. GoTo_DIR command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | From host |
| X | X | ABS_POS (byte 2) | | | | | | From host |
| ABS_POS (byte 1) | | | | | | | | From host |
| ABS_POS (byte 0) | | | | | | | | From host |

The GoTo command produces a motion to ABS_POS absolute position through the shortest path. The ABS_POS value is always in agreement with the selected step mode (full, half, quarter, etc.).

The GoTo command keeps the BUSY flag low until the target position is reached.

This command can be given only when the previous motion command has been completed (BUSY flag released).

Any attempt to perform a GoTo command when a previous command is under execution (BUSY low) causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.2.9 GoTo_DIR (DIR, ABS_POS)

**Table 67. GoTo_DIR command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | DIR | From host |
| X | X | ABS_POS (byte 2) | | | | | | From host |
| ABS_POS (byte 1) | | | | | | | | From host |
| ABS_POS (byte 0) | | | | | | | | From host |

The GoTo_DIR command produces a motion to ABS_POS absolute position imposing a forward (DIR = '1') or a reverse (DIR = '0') rotation. The ABS_POS value is always equal to the selected step mode (full, half, quarter, etc.).

The GoTo_DIR command keeps the BUSY flag low until the target position is reached. This command can be given only when the previous motion command has been completed (BUSY flag released).

Any attempt to perform a GoTo_DIR command when a previous command is under execution (BUSY low) causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.2.10 GoUntil (ACT, DIR, SPD)

**Table 68. GoUntil command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | ACT | 0 | 1 | DIR | From host |
| X | X | X | X | SPD (byte 2) | | | | From host |
| SPD (byte 1) | | | | | | | | From host |
| SPD (byte 0) | | | | | | | | From host |

The GoUntil command produces a motion at SPD speed imposing a forward (DIR = '1') or a reverse (DIR = '0') direction. When an external switch turn-on event occurs (see Section 7.14) the ABS_POS register is reset (if ACT = '0') or the ABS_POS register value is copied into the MARK register (if ACT = '1'); the system then performs a SoftStop command.

The SPD value is expressed in step/tick (format unsigned fixed point 0.28) that is the same format as the SPEED register (see Section 11.1.4).

The SPD value should be lower than MAX_SPEED and greater than MIN_SPEED, otherwise the target speed is imposed at MAX_SPEED or MIN_SPEED respectively.

If the SW_MODE bit of the CONFIG register is set low, the external switch turn-on event causes a HardStop interrupt instead of the SoftStop one (see Section 7.14 and Section 11.1.27).

This command keeps the BUSY flag low until the switch turn-on event occurs and the motor is stopped. This command can be given anytime and is immediately executed.

### 11.2.11 ReleaseSW (ACT, DIR)

**Table 69. ReleaseSW command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 1 | ACT | 0 | 1 | DIR | From host |

The ReleaseSW command produces a motion at minimum speed imposing a forward (DIR = '1') or reverse (DIR = '0') rotation. When SW is released (opened) the ABS_POS register is reset (ACT = '0') or the ABS_POS register value is copied into the MARK register (ACT = '1'); the system then performs a HardStop command.

Note that, resetting the ABS_POS register is equivalent to setting the HOME position.

If the minimum speed value is less than 5 step/s or low speed optimization is enabled, the motion is performed at 5 step/s.

The ReleaseSW command keeps the BUSY flag low until the switch input is released and the motor is stopped.

### 11.2.12 GoHome

**Table 70. GoHome command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | From host |

The GoHome command produces a motion to the HOME position (zero position) via the shortest path.

Note that, this command is equivalent to the “GoTo(0…0)” command. If a motor direction is mandatory, the GoTo_DIR command must be used (see Section 11.2.9).

The GoHome command keeps the BUSY flag low until the home position is reached. This command can be given only when the previous motion command has been completed. Any attempt to perform a GoHome command when a previous command is under execution (BUSY low) causes the command to be ignored and CMD_ERROR to rise (see Section 11.1.28).

### 11.2.13 GoMark

Table 71. **GoMark command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | From host |

The GoMark command produces a motion to the MARK position performing the minimum path.

Note that, this command is equivalent to the “GoTo (MARK)” command. If a motor direction is mandatory, the GoTo_DIR command must be used.

The GoMark command keeps the BUSY flag low until the MARK position is reached. This command can be given only when the previous motion command has been completed (BUSY flag released).

Any attempt to perform a GoMark command when a previous command is under execution (BUSY low) causes the command to be ignored and the CMD_ERROR flag to rise (see Section 11.1.28).

### 11.2.14 ResetPos

Table 72. **ResetPos command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | From host |

The ResetPos command resets the ABS_POS register to zero. The zero position is also defined as the HOME position (see Section 7.5).

### 11.2.15 ResetDevice

Table 73. **ResetDevice command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | From host |

The ResetDevice command resets the device to power-up conditions (see Section 7.1).

*Note: At power-up the power bridges are disabled.*

### 11.2.16 SoftStop

Table 74. **SoftStop command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | From host |

The SoftStop command causes an immediate deceleration to zero speed and a consequent motor stop; the deceleration value used is the one stored in the DEC register (see Section 11.1.6).

When the motor is in a high impedance state, a SoftStop command forces the bridges to exit from the high impedance state; no motion is performed.

This command can be given anytime and is immediately executed. This command keeps the BUSY flag low until the motor is stopped.

### 11.2.17 HardStop

**Table 75. HardStop command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | From host |

The HardStop command causes an immediate motor stop with infinite deceleration.

When the motor is in a high impedance state, a HardStop command forces the bridges to exit the high impedance state; no motion is performed.

This command can be given anytime and is immediately executed.

### 11.2.18 SoftHiZ

**Table 76. SoftHiZ command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | From host |

The SoftHiZ command disables the power bridges (high impedance state) after a deceleration to zero speed; the deceleration value used is the one stored in the DEC register (see Section 11.1.6). When bridges are disabled, the HiZ flag is raised.

When the motor is stopped, a SoftHiZ command forces the bridges to enter a high impedance state.

This command can be given anytime and is immediately executed.

### 11.2.19 HardHiZ

**Table 77. HardHiZ command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | From host |

The HardHiZ command immediately disables the power bridges (high impedance state) and raises the HiZ flag.

When the motor is stopped, a HardHiZ command forces the bridges to enter a high impedance state.

This command can be given anytime and is immediately executed. This command keeps the BUSY flag low until the motor is stopped.

### 11.2.20 GetStatus

**Table 78. GetStatus command structure**

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | From host |
| STATUS MSbyte | | | | | | | | To host |
| STATUS LSbyte | | | | | | | | To host |

The GetStatus command returns the status register value.

The GetStatus command resets the STATUS register warning flags. The command forces the system to exit from any error state. The GetStatus command does not reset the HiZ flag.

# 12 Package information

To meet environmental requirements, ST offers these devices in different grades of ECOPACK packages, depending on their level of environmental compliance. ECOPACK specifications, grade definitions, and product status are available at: www.st.com. ECOPACK is an ST trademark.

## 12.1 VFQFPN 11x14x10 package information

**Figure 30. VFQFPN 11x14x1 package outline - top and side view**

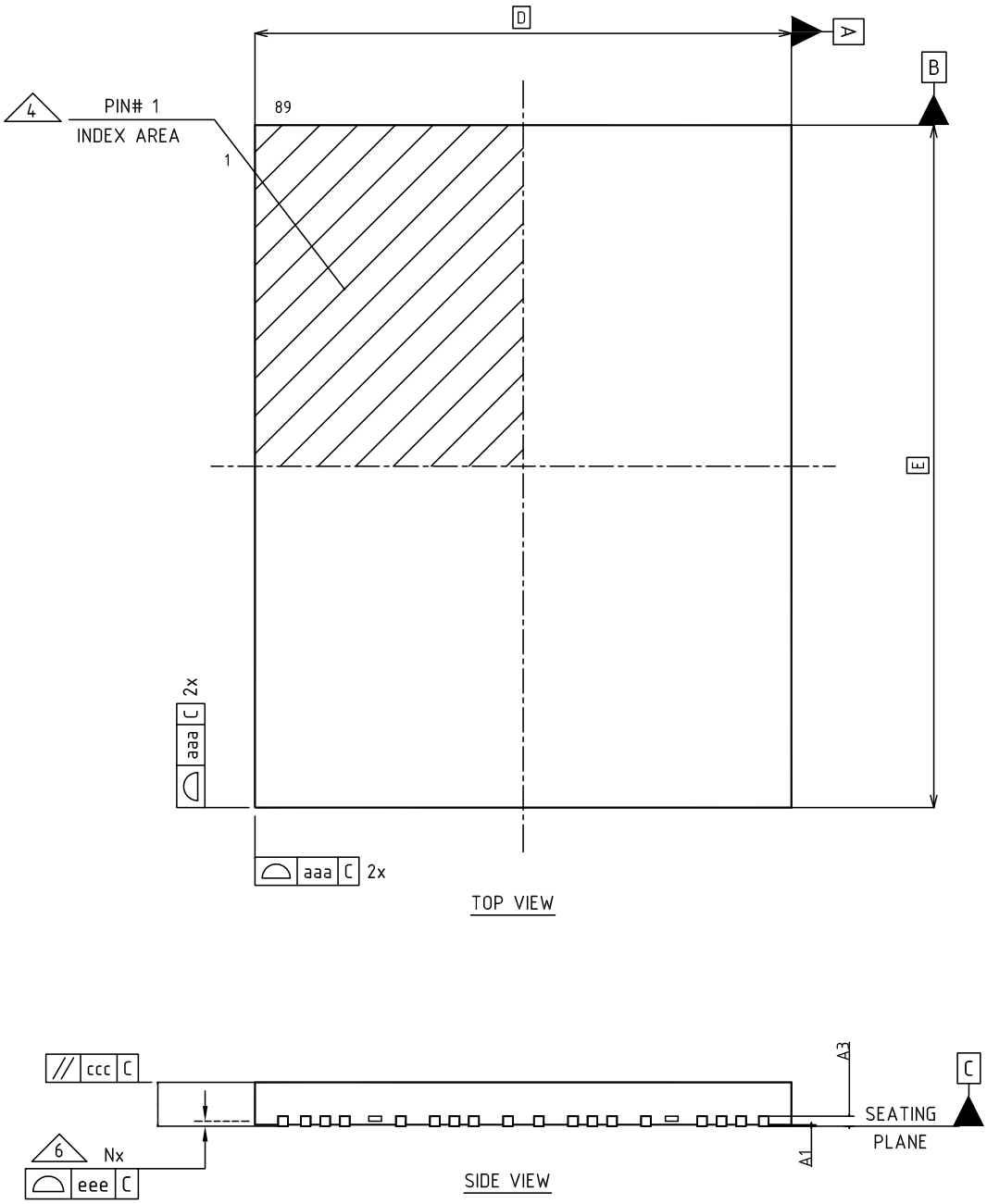


Figure 31. **VFQFPN 11x14x1 package outline - bottom view**

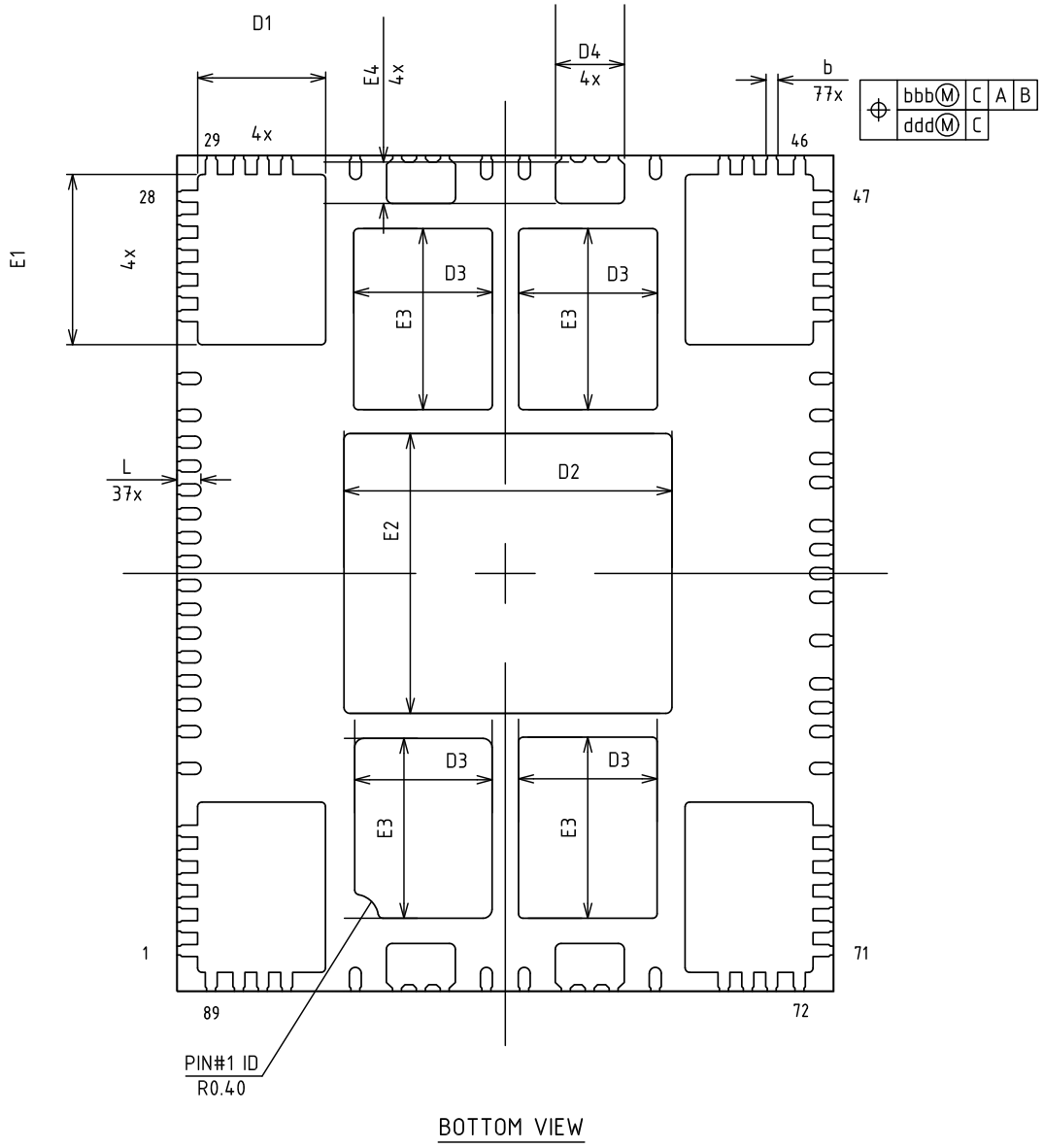


Figure 32. **VFQFPN 11x14x1 package outline - pins identifier**

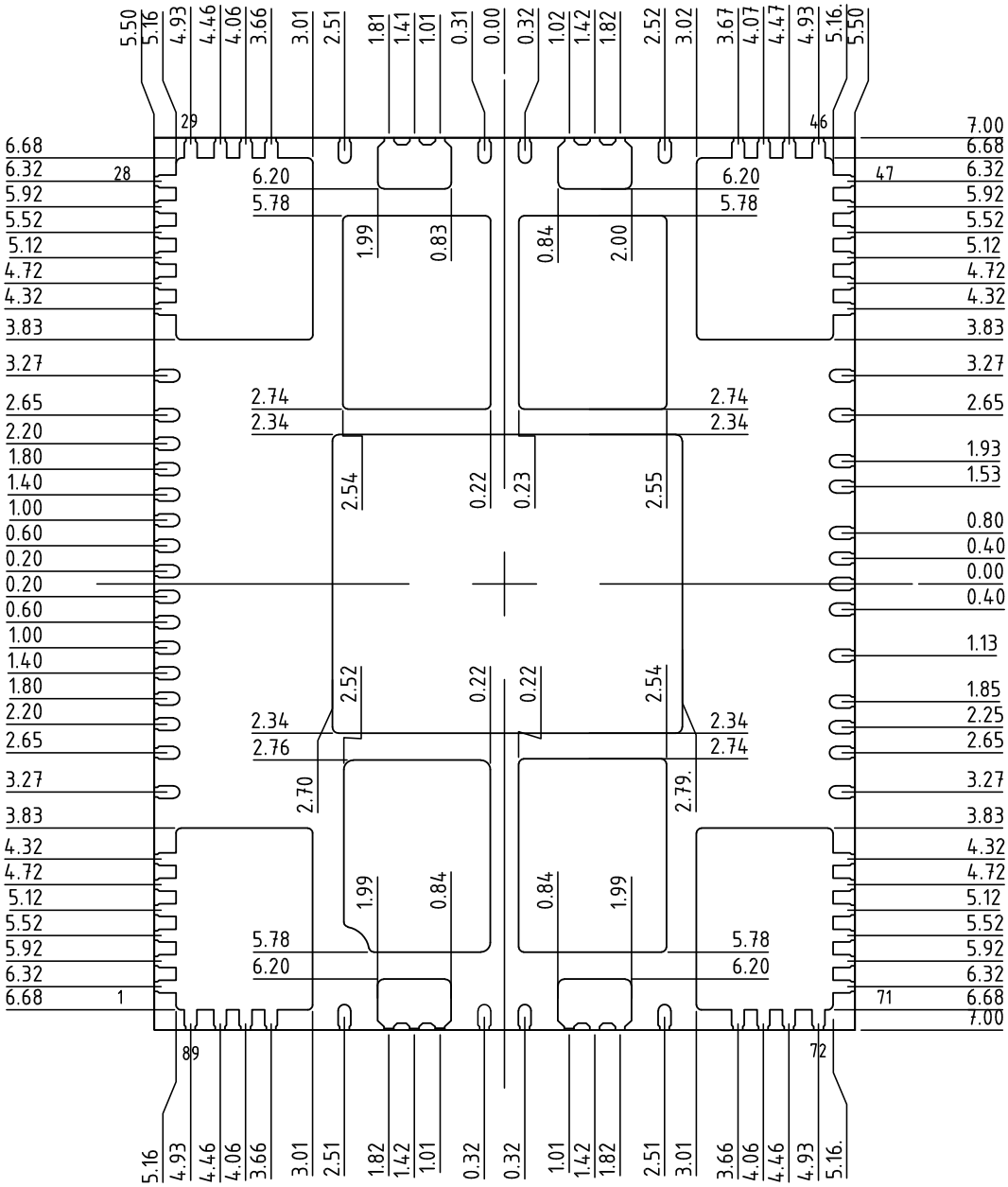


**Table 79. VFQFPN 11x14x1 package mechanical data**

(1)

| Symbol | Dimensions (mm) max. | | | Note |
|---|---|---|---|---|
| | Min. | Typ. | Max. | |
| A | 0.80 | 0.90 | 1.00 | (2) |
| A1 | 0.00 | 0.02 | 0.05 | |
| A3 | --- | 0.20 ref. | --- | |
| b | 0.15 | 0.2 | 0.25 | (3) |
| D | 10.90 | 11.00 | 11.10 | |
| E | 13.90 | 14.00 | 14.10 | |
| D1 | 2.00 | 2.15 | 2.25 | |
| E1 | 2.70 | 2.85 | 2.95 | |
| D2 | 5.35 | 5.50 | 5.60 | |
| E2 | 4.54 | 4.69 | 4.79 | |
| D3 | 2.17 | 2.32 | 2.42 | |
| E3 | 2.88 | 3.04 | 3.13 | |
| D4 | 1.01 | 1.16 | 1.26 | |
| E4 | 0.54 | 0.69 | 0.79 | |
| L | 0.30 | 0.40 | 0.50 | |
| aaa | | 0.10 | | |
| bbb | | 0.10 | | |
| ccc | | 0.10 | | |
| ddd | | 0.05 | | |
| eee | | 0.08 | | |
| N | | 89 | | (4) |

1. *The pin #1 identifier must be existed on the top surface of the package by using an indentation mark or other feature of the package body. Exact shape and size of this feature is optional.*
2. *VFQFPN stands for thermally enhanced very thin fine pitch quad flat packages no lead. Very thin: 0.80 mm < A ≤1.00 mm / fine pitch: e < 1.00 mm. Dimensioning and tolerances conform to ASME Y14.5-2009. All dimensions are in millimeters. The location of the marked terminal #1 identifier is within the hatched area.*
   *Coplanarity applies to the terminals and all other bottom surface metalization.*
3. *Dimension b applies to the metalized terminal and is measured between 0.15 mm and 0.30 mm from the terminal tip. If the terminal has a radius on the other end of it, dimension b should not be measured in that radius area.*
4. *N is the total number of terminals.*

Figure 33. **VFQFPN 11x14x1 suggested footprint**

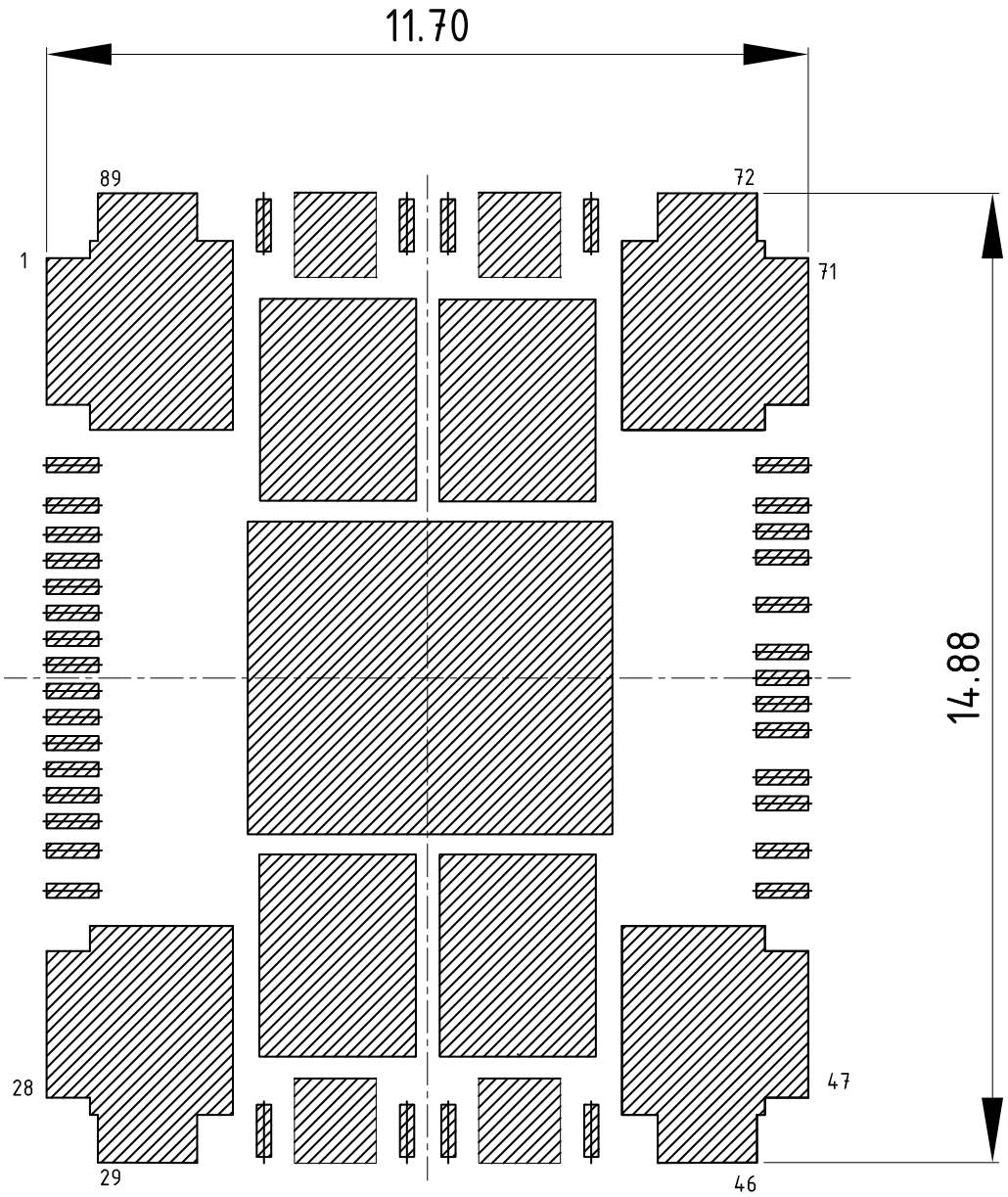


Figure 34. **VFQFPN 11x14x1 suggested footprint - lead land positioning**

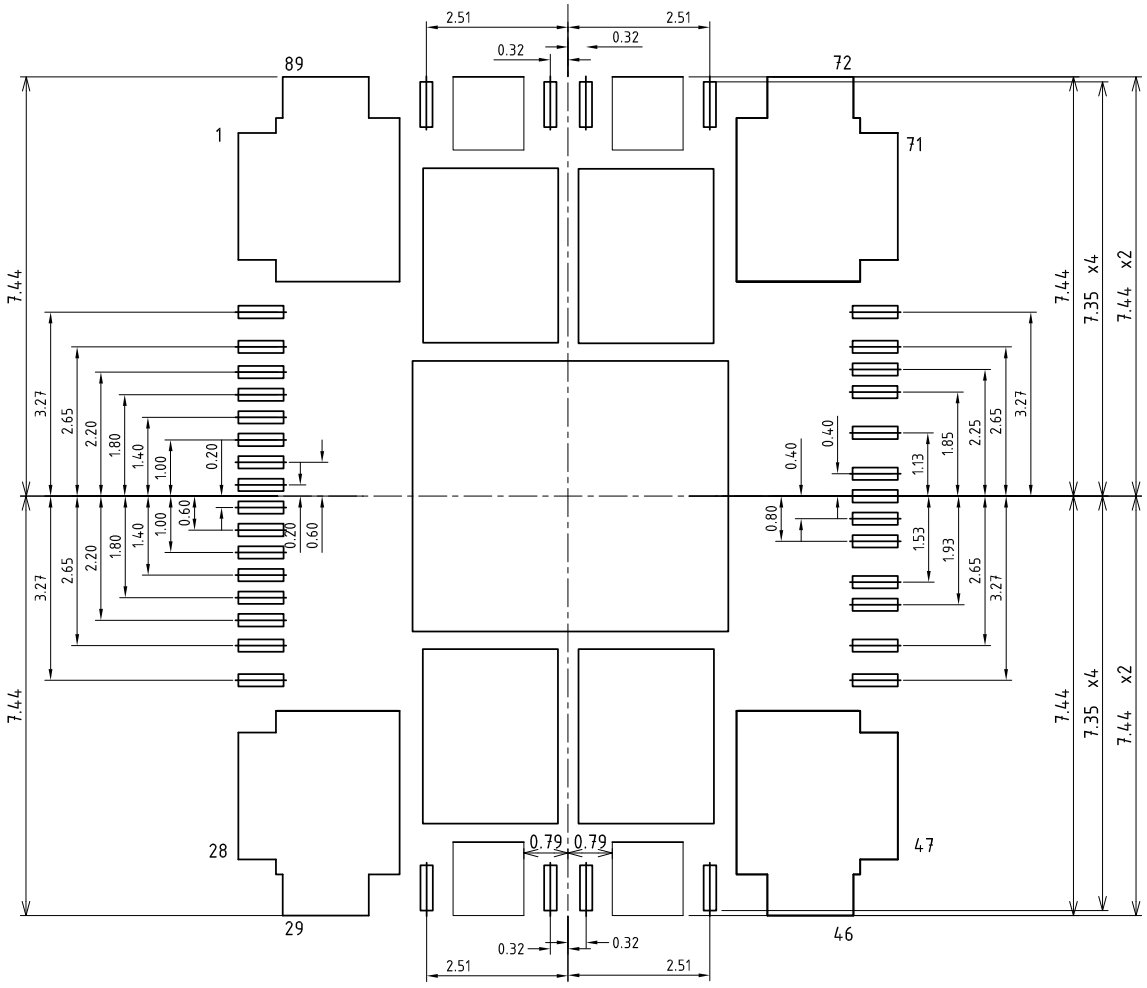


Figure 35. **VFQFPN 11x14x1 suggested footprint - land size and exposed pad positioning**

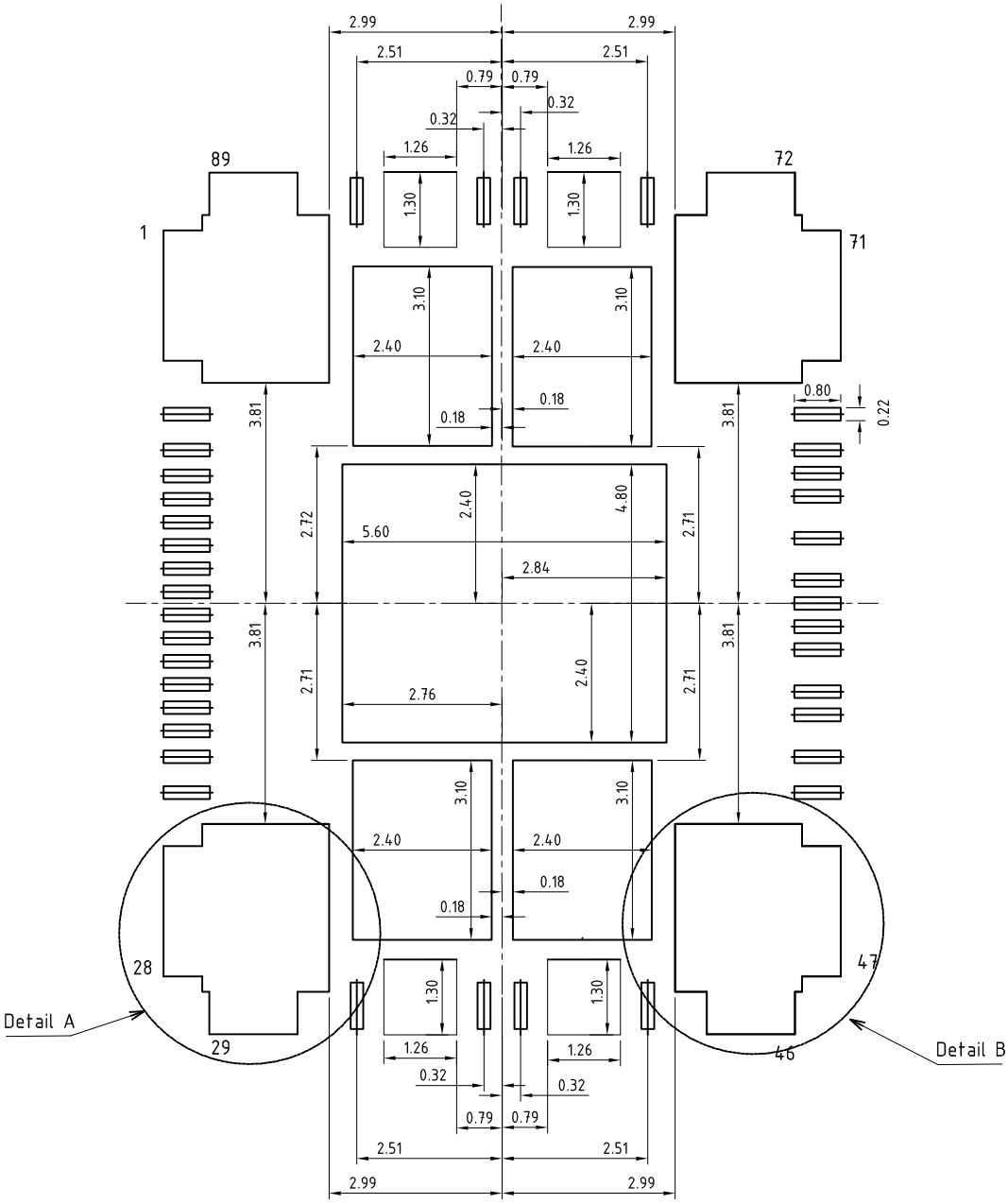
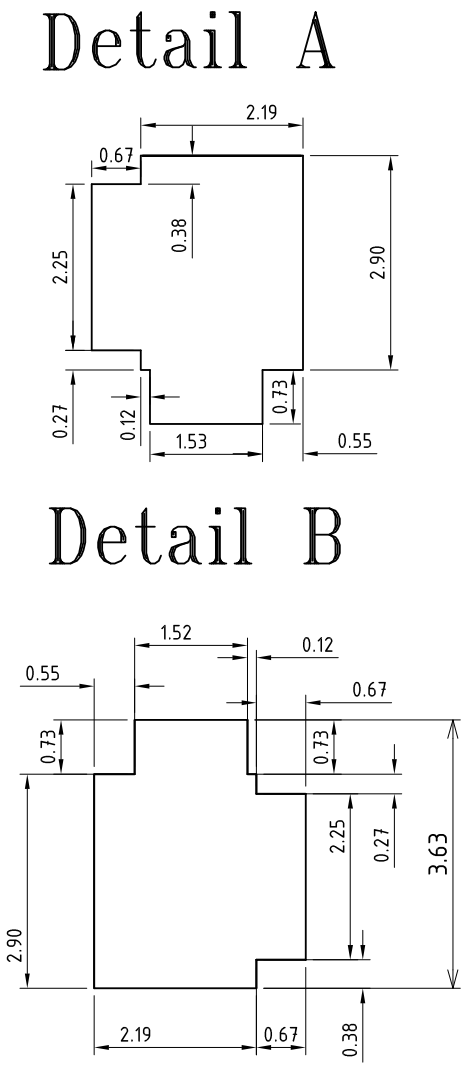


Figure 36. **VFQFPN 11x14x1 suggested footprint**



# 13 Ordering information

Table 80. **Device summary**

| Order code | Package | Packing |
|---|---|---|
| POWERSTEP01 | VFQFPN 11 x 14 x1 mm | Tray |
| POWERSTEP01TR | VFQFPN 11 x 14 x1 mm | Tape and reel |

# Revision history

Table 81. **Document revision history**

| Date | Version | Changes |
|---|---|---|
| 22-Jul-2013 | 1 | Initial release. |
| 16-Jun-2014 | 2 | Several modifications. |
| 03-Oct-2014 | 3 | Updated main title on page 1 (replaced “8-N channel” by “10A”).<br>Updated package silhouette on page 1 (replaced by new figure).<br>Updated Section Applications (added sub-list).<br>Updated Table 2 (removed “$P_{tot}$”, “Peak” and unit of $I_{OUT1x}$ symbol).<br>Updated Table 4 (updated “Test conditions” and values of $R_{DS(on)}$, $SR_{OUT}$, $I_{DSS}$ and $t_{OCD, SD}$ symbols).<br>Added Section 7.15. Minor modifications throughout document. |
| 22-Apr-2015 | 4 | Document updated from “preliminary” to “production data”.<br>Updated Table 4 (updated max. values).<br>Updated Table 5 (updated pin no. of OUTA1).<br>Updated Table 8 (replaced “Of” by “On” in High-side gate driver supply turn-on threshold $\Delta V_{BOOTthon}$).<br>Updated Table 12 (updated length column of STEP_MODE and FS_SPD).<br>Updated Section 11.1.9 (removed Bit 11 column).<br>Updated Section 12.1 (replaced Figure 30 to Figure 32 and Table 79 by new figures/table).<br>Minor modifications throughout document. |
| 10-Jun-2015 | 5 | Updated Table 4 (updated max. values).<br>Updated Table 45 (updated values).<br>Added Figure 33 to Figure 36.<br>Minor modifications throughout document. |
| 09-Nov-2017 | 6 | Updated Section 12.1.<br>Minor text changes. |
| 28-Jan-2020 | 7 | Updated Figure 1, Figure 2, Figure 1, Figure 4, Figure 5 and Figure 24.<br>Table 5 amended and Footnotes added.<br>Table 12 updated. Change title of Table 37. |
| 23-Oct-2024 | 8 | Added Section 2.3.<br>Updated Eq. (10), Section 7.5, Section 7.6, Section 7.12, Section 8.2, Section 9.3, Section 11.1.5, Section 11.1.6, Section 11.1.7, Section 11.1.9, Section 11.1.17, Section 11.1.28, Section 11.2.8, Section 11.2.9, Section 11.2.17 and Section 11.2.18.<br>Updated Table 4 and Table 12, corrected typo in Table 55.<br>Updated Figure 1, Figure 2, Figure 3, Figure 4, Figure 5, Figure 7, Figure 8, Figure 13, Figure 14.<br>Updated figure labels in Section 12.1 and Figure 35. |

# Contents

# List of tables

# List of figures

**IMPORTANT NOTICE – READ CAREFULLY**

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgment.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.