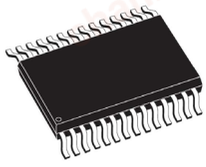
chanhvinh.com

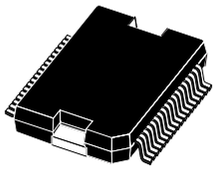
# L6472

## dSPIN™ fully integrated microstepping motor driver

**Datasheet - production data**



HTSSOP28



POWERSO36

## Features

- Operating voltage: 8 - 45 V
- 7.0 A output peak current (3.0 A $_{r.m.s.}$)
- Low $R_{DS(on)}$ Power MOSFETs
- Programmable speed profile
- Programmable Power MOSFET slew rate
- Up to 1/16 microstepping
- Predictive current control with adaptive decay
- Non dissipative current sensing
- SPI interface
- Low quiescent and standby currents
- Programmable non dissipative overcurrent protection on all Power MOSFETs
- Two levels of overtemperature protection

## Applications

- Bipolar stepper motor

## Description

The L6472 device, realized in analog mixed signal technology, is an advanced fully integrated solution suitable for driving two-phase bipolar stepper motors with microstepping. It integrates a dual low $R_{DS(on)}$ DMOS full bridge with all of the power switches equipped with an accurate on-chip current sensing circuitry suitable for non dissipative current control and overcurrent protection. Thanks to a new current control, a 1/16 microstepping is achieved through an adaptive decay mode which outperforms traditional implementations. The digital control core can generate user defined motion profiles with acceleration, deceleration, speed or target position, easily programmed through a dedicated register set.

All application commands and data registers, including those used to set analog values (i.e. current control value, current protection trip point, deadtime, etc.) are sent through a standard 5-Mbit/s SPI.

A very rich set of protections (thermal, low bus voltage, overcurrent) makes the L6472 device “bullet proof”, as required by the most demanding motor control applications.

**Table 1. Device summary**

| Order codes | Package | Packing |
|---|---|---|
| L6472H | HTSSOP28 | Tube |
| L6472HTR | HTSSOP28 | Tape and reel |
| L6472PD | POWERSO36 | Tube |
| L6472PDTR | POWERSO36 | Tape and reel |



This is information on a product in full production. www.st.com

chanhvinh.com



# Contents


1 **Block diagram** . . . . . 8

2 **Electrical data** . . . . . 9

2.1 Absolute maximum ratings . . . . . 9

2.2 Recommended operating conditions . . . . . 9

2.3 Thermal data . . . . . 10

3 **Electrical characteristics** . . . . . 11

4 **Pin connection** . . . . . 16

Pin list . . . . . 17

5 **Typical applications** . . . . . 19

6 **Functional description** . . . . . 20

6.1 Device power-up . . . . . 20

6.2 Logic I/O . . . . . 20

6.3 Charge pump . . . . . 20

6.4 Microstepping . . . . . 21

Automatic full-step mode . . . . . 22

6.5 Absolute position counter . . . . . 23

6.6 Programmable speed profiles . . . . . 23

6.7 Motor control commands . . . . . 23

6.7.1 Constant speed commands . . . . . 24

6.7.2 Positioning commands . . . . . 24

6.7.3 Motion commands . . . . . 25

6.7.4 Stop commands . . . . . 25

6.7.5 Step-clock mode . . . . . 26

6.7.6 GoUntil and ReleaseSW commands . . . . . 26

6.8 Internal oscillator and oscillator driver . . . . . 27

6.8.1 Internal oscillator . . . . . 27

6.8.2 External clock source . . . . . 27

6.9 Overcurrent detection . . . . . 28

chanhvinh.com



6.10 Undervoltage lockout (UVLO) . . . 29
6.11 Thermal warning and thermal shutdown . . . 29
6.12 Reset and standby . . . 29
6.13 External switch (SW pin) . . . 30
6.14 Programmable DMOS slew rate, deadtime and blanking time . . . 30
6.15 Integrated analog-to-digital converter . . . 31
6.16 Internal voltage regulator . . . 31
6.17 BUSY\SYNC pin . . . 32
6.17.1 BUSY operation mode . . . 32
6.17.2 SYNC operation mode . . . 32
6.18 FLAG pin . . . 32

**7 Phase current control . . . 33**
7.1 Predictive current control . . . 33
7.2 Auto-adjusted decay mode . . . 34
7.3 Auto-adjusted fast decay during the falling steps . . . 36
7.4 Torque regulation (output current amplitude regulation) . . . 37

**8 Serial interface . . . 38**

**9 Programming manual . . . 40**
9.1 Register and flag description . . . 40
9.1.1 ABS_POS . . . 41
9.1.2 EL_POS . . . 41
9.1.3 MARK . . . 42
9.1.4 SPEED . . . 42
9.1.5 ACC . . . 42
9.1.6 DEC . . . 42
9.1.7 MAX_SPEED . . . 43
9.1.8 MIN_SPEED . . . 43
9.1.9 FS_SPD . . . 43
9.1.10 TVAL_HOLD, TVAL_RUN, TVAL_ACC and TVAL_DEC . . . 44
9.1.11 T_FAST . . . 45
9.1.12 TON_MIN . . . 45
9.1.13 TOFF_MIN . . . 46
9.1.14 ADC_OUT . . . 46

chanhvinh.com



9.1.15 OCD_TH . . . 47
9.1.16 STEP_MODE . . . 47
9.1.17 ALARM_EN . . . 49
9.1.18 CONFIG . . . 49
9.1.19 STATUS . . . 52
9.2 Application commands . . . 54
9.2.1 Command management . . . 55
9.2.2 Nop . . . 56
9.2.3 SetParam (PARAM, VALUE) . . . 56
9.2.4 GetParam (PARAM) . . . 56
9.2.5 Run (DIR, SPD) . . . 57
9.2.6 StepClock (DIR) . . . 57
9.2.7 Move (DIR, N_STEP) . . . 58
9.2.8 GoTo (ABS_POS) . . . 58
9.2.9 GoTo_DIR (DIR, ABS_POS) . . . 59
9.2.10 GoUntil (ACT, DIR, SPD) . . . 59
9.2.11 ReleaseSW (ACT, DIR) . . . 60
9.2.12 GoHome . . . 60
9.2.13 GoMark . . . 61
9.2.14 ResetPos . . . 61
9.2.15 ResetDevice . . . 61
9.2.16 SoftStop . . . 62
9.2.17 HardStop . . . 62
9.2.18 SoftHiZ . . . 62
9.2.19 HardHiZ . . . 63
9.2.20 GetStatus . . . 63

**10 Package information . . . 64**

**11 Revision history . . . 69**

chanhvinh.com



# List of tables


Table 1. Device summary . . . 1
Table 2. Absolute maximum ratings . . . 9
Table 3. Recommended operating conditions . . . 9
Table 4. Thermal data . . . 10
Table 5. Electrical characteristics . . . 11
Table 6. Pin description . . . 17
Table 7. Typical application values . . . 19
Table 8. CL values according to external oscillator frequency . . . 27
Table 9. Register map . . . 40
Table 10. EL_POS register . . . 41
Table 11. MIN_SPEED register . . . 43
Table 12. Torque regulation by TVAL_HOLD, TVAL_ACC, TVAL_DEC and TVAL_RUN registers . . . 44
Table 13. T_FAST register . . . 45
Table 14. Maximum fast decay times . . . 45
Table 15. Minimum ON time . . . 45
Table 16. Minimum OFF time . . . 46
Table 17. ADC_OUT value and torque regulation feature . . . 46
Table 18. Overcurrent detection threshold . . . 47
Table 19. STEP_MODE register . . . 47
Table 20. Step mode selection . . . 47
Table 21. SYNC output frequency . . . 48
Table 22. SYNC signal source . . . 48
Table 23. ALARM_EN register . . . 49
Table 24. CONFIG register . . . 49
Table 25. Oscillator management . . . 50
Table 26. External switch hard stop interrupt mode . . . 50
Table 27. Overcurrent event . . . 51
Table 28. Programmable power bridge output slew rate values . . . 51
Table 29. External torque regulation enable . . . 51
Table 30. Switching period . . . 51
Table 31. STATUS register . . . 52
Table 32. STATUS register DIR bit . . . 52
Table 33. STATUS register MOT_STATUS bits . . . 53
Table 34. Application commands . . . 54
Table 35. NOP command structure . . . 56
Table 36. SetParam command structure . . . 56
Table 37. GetParam command structure . . . 56
Table 38. Run command structure . . . 57
Table 39. StepClock command structure . . . 57
Table 40. Move command structure . . . 58
Table 41. GoTo command structure . . . 58
Table 42. GoTo_DIR command structure . . . 59
Table 43. GoUntil command structure . . . 59
Table 44. ReleaseSW command structure . . . 60
Table 45. GoHome command structure . . . 60
Table 46. GoMark command structure . . . 61
Table 47. ResetPos command structure . . . 61

chanhvinh.com



Table 48. ResetDevice command structure . . . 61
Table 49. SoftStop command structure . . . 62
Table 50. HardStop command structure . . . 62
Table 51. SoftHiZ command structure . . . 62
Table 52. HardHiZ command structure . . . 63
Table 53. GetStatus command structure . . . 63
Table 54. HTSSOP28 package mechanical data . . . 66
Table 55. POWERSO36 package mechanical data . . . 68
Table 56. Document revision history . . . 69

chanhvinh.com



# List of figures

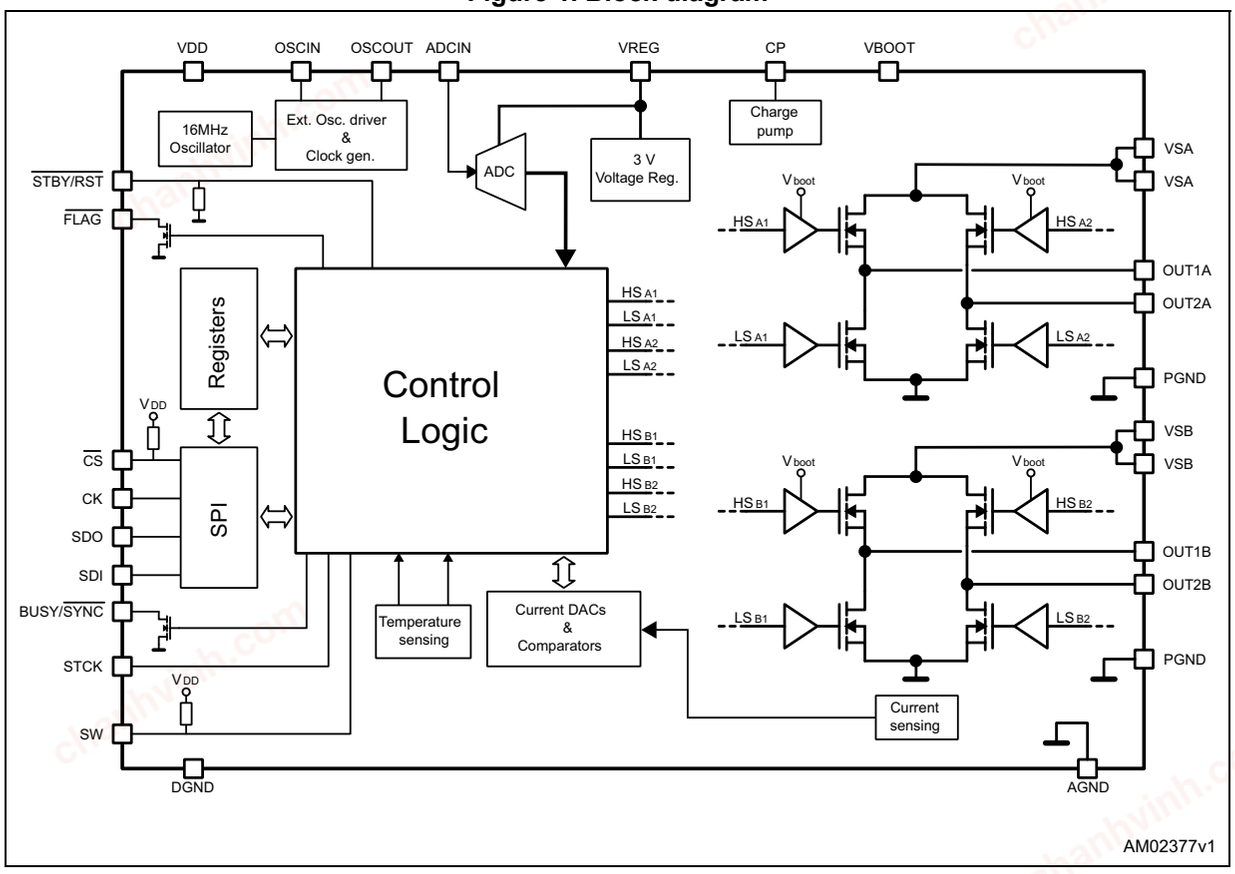
Figure 1. Block diagram . . . 8
Figure 2. HTSSOP28 pin connection (top view) . . . 16
Figure 3. POWERSO36 pin connection (top view) . . . 16
Figure 4. Bipolar stepper motor control application using the L6472 . . . 19
Figure 5. Charge pump circuitry . . . 21
Figure 6. Normal mode and microstepping (16 microsteps) . . . 22
Figure 7. Automatic full-step switching . . . 22
Figure 8. Constant speed command examples . . . 24
Figure 9. Positioning command examples . . . 25
Figure 10. Motion command examples . . . 25
Figure 11. OSCIN and OSCOUT pin configurations . . . 28
Figure 12. External switch connection . . . 30
Figure 13. Internal 3 V linear regulator . . . 31
Figure 14. Predictive current control . . . 33
Figure 15. Non-predictive current control . . . 34
Figure 16. Adaptive decay - fast decay tuning . . . 35
Figure 17. Adaptive decay switch from normal to slow + fast decay mode and vice-versa . . . 36
Figure 18. Fast decay tuning during the falling steps . . . 37
Figure 19. SPI timings diagram . . . 38
Figure 20. Daisy chain configuration . . . 39
Figure 21. Command with 3-byte argument . . . 55
Figure 22. Command with 3-byte response . . . 55
Figure 23. Command response aborted . . . 55
Figure 24. HTSSOP28 package outline . . . 65
Figure 25. POWERSO36 package outline . . . 67

chanhvinh.com



# 1 Block diagram

**Figure 1. Block diagram**



chanhvinh.com



# 2 Electrical data

## 2.1 Absolute maximum ratings

**Table 2. Absolute maximum ratings**

| Symbol | Parameter | Test condition | Value | Unit |
|---|---|---|---|---|
| $V_{DD}$ | Logic interface supply voltage | | 5.5 | V |
| $V_S$ | Motor supply voltage | $V_{SA} = V_{SB} = V_S$ | 48 | V |
| $V_{GND, diff}$ | Differential voltage between AGND, PGND and DGND | | ±0.3 | V |
| $V_{boot}$ | Bootstrap peak voltage | | 55 | V |
| $V_{REG}$ | Internal voltage regulator output pin and logic supply voltage | | 3.6 | V |
| $V_{ADCIN}$ | Integrated ADC input voltage range (ADCIN pin) | | -0.3 to +3.6 | V |
| $V_{OSC}$ | OSCIN and OSCOUT pin voltage range | | -0.3 to +3.6 | V |
| $V_{out_diff}$ | Differential voltage between $V_{SA}$, $OUT1_A$, $OUT2_A$, PGND and $V_{SB}$, $OUT1_B$, $OUT2_B$, PGND pins | $V_{SA} = V_{SB} = V_S$ | 48 | V |
| $V_{LOGIC}$ | Logic inputs voltage range | | -0.3 to +5.5 | V |
| $I_{out}$(1) | R.m.s. output current | | 3 | A |
| $I_{out_peak}$(1) | Pulsed output current | $T_{PULSE}$ < 1 ms | 7 | A |
| $T_{OP}$ | Operating junction temperature | | -40 to 150 | °C |
| $T_s$ | Storage temperature range | | -55 to 150 | °C |
| $P_{tot}$ | Total power dissipation ($T_A$ = 25 °C) | (2) | 5 | W |

1. Maximum output current limit is related to metal connection and bonding characteristics. Actual limit must satisfy maximum thermal dissipation constraints.
2. HTSSOP28 mounted on the EVAL6472H.

## 2.2 Recommended operating conditions

**Table 3. Recommended operating conditions**

| Symbol | Parameter | Test condition | Value | | | Unit |
|---|---|---|---|---|---|---|
| $V_{DD}$ | Logic interface supply voltage | 3.3 V logic outputs | | 3.3 | | V |
| | | 5 V logic outputs | | 5 | | |
| $V_S$ | Motor supply voltage | $V_{SA} = V_{SB} = V_S$ | 8 | | 45 | V |
| $V_{out_diff}$ | Differential voltage between $V_{SA}$, $OUT1_A$, $OUT2_A$, PGND and $V_{SB}$, $OUT1_B$, $OUT2_B$, PGND pins | $V_{SA} = V_{SB} = V_S$ | | | 45 | V |
| $V_{REG,in}$ | Logic supply voltage | $V_{REG}$ voltage imposed by external source | 3.2 | 3.3 | | V |
| $V_{ADC}$ | Integrated ADC input voltage (ADCIN pin) | | 0 | | $V_{REG}$ | V |

chanhvinh.com



## 2.3 Thermal data

**Table 4. Thermal data**

| Symbol | Parameter | Package | Typ. | Unit |
|---|---|---|---|---|
| $R_{thJA}$ | Thermal resistance junction ambient | HTSSOP28(1) | 22 | °C/W |
| | | POWERSO36(2) | 12 | |

1. HTSSOP28 mounted on the EVAL6472H Rev 1.0 board: four-layer FR4 PCB with a dissipating copper surface of about 40 $cm^2$ on each layer and 15 via holes below the IC.
2. POWERSO36 mounted on the EVAL6472PD Rev 1.0 board: four-layer FR4 PCB with a dissipating copper surface of about 40 $cm^2$ on each layer and 22 via holes below the IC.

chanhvinh.com



# 3 Electrical characteristics

$V_{SA}$ = $V_{SB}$ = 36 V; $V_{DD}$ = 3.3 V; internal 3 V regulator; $T_J$ = 25 °C, unless otherwise specified.

**Table 5. Electrical characteristics**

| Symbol | Parameter | Test condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| **General** | | | | | | |
| $V_{SthOn}$ | $V_S$ UVLO turn-on threshold | | 7.5 | 8.2 | 8.9 | V |
| $V_{SthOff}$ | $V_S$ UVLO turn-off threshold | | 6.6 | 7.2 | 7.8 | V |
| $V_{SthHyst}$ | $V_S$ UVLO threshold hysteresis | | 0.7 | 1 | 1.3 | V |
| $I_q$ | Quiescent motor supply current | Internal oscillator selected; $V_{REG}$ = 3.3 V ext; CP floating | | 0.5 | 0.65 | mA |
| $T_{j(WRN)}$ | Thermal warning temperature | | | 130 | | °C |
| $T_{j(SD)}$ | Thermal shutdown temperature | | | 160 | | °C |
| **Charge pump** | | | | | | |
| $V_{pump}$ | Voltage swing for charge pump oscillator | | | 10 | | V |
| $f_{pump,min}$ | Minimum charge pump oscillator frequency(1) | | | 660 | | kHz |
| $f_{pump,max}$ | Maximum charge pump oscillator frequency(1) | | | 800 | | kHz |
| $I_{boot}$ | Average boot current | $f_{sw,A}$ = $f_{sw,B}$ = 15.6 kHz POW_SR = ‘10’ | | 1.1 | 1.4 | mA |
| **Output DMOS transistor** | | | | | | |
| $R_{DS(on)}$ | High-side switch on-resistance | $T_j$ = 25 °C, $I_{out}$ = 3 A | | 0.37 | | Ω |
| | | $T_j$ = 125 °C,(2) $I_{out}$ = 3 A | | 0.51 | | |
| | Low-side switch on-resistance | $T_j$ = 25 °C, $I_{out}$ = 3 A | | 0.18 | | |
| | | $T_j$ = 125 °C,(2) $I_{out}$ = 3 A | | 0.23 | | |
| $I_{DSS}$ | Leakage current | OUT = $V_S$ | | | 3.1 | mA |
| | | OUT = GND | -0.3 | | | |
| $t_r$ | Rise time(3) | POW_SR = '00', $I_{out}$ = +1 A | | 100 | | ns |
| | | POW_SR = '00', $I_{out}$ = -1 A | | 80 | | |
| | | POW_SR = ‘11’, $I_{out}$ = ±1 A | | 100 | | |
| | | POW_SR = ‘10’, $I_{out}$ = ±1 A | | 200 | | |
| | | POW_SR = ‘01’, $I_{out}$ = ±1 A | | 300 | | |

chanhvinh.com



Table 5. Electrical characteristics (continued)

| Symbol | Parameter | Test condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $t_f$ | Fall time (3) | POW_SR = '00'; $I_{out}$ = +1 A | | 90 | | ns |
| | | POW_SR = '00'; $I_{out}$ = -1 A | | 110 | | |
| | | POW_SR = '11', $I_{out}$ = ±1 A | | 110 | | |
| | | POW_SR = '10', $I_{out}$ = ±1 A | | 260 | | |
| | | POW_SR = '01', $I_{load}$ = ±1 A | | 375 | | |
| $SR_{out_r}$ | Output rising slew rate | POW_SR = '00', $I_{out}$ = +1 A | | 285 | | V/µs |
| | | POW_SR = '00', $I_{out}$ = -1 A | | 360 | | |
| | | POW_SR = '11', $I_{out}$ = ±1 A | | 285 | | |
| | | POW_SR = '10', $I_{out}$ = ±1 A | | 150 | | |
| | | POW_SR = '01', $I_{out}$ = ±1 A | | 95 | | |
| $SR_{out_f}$ | Output falling slew rate | POW_SR = '00', $I_{out}$ = +1 A | | 320 | | V/µs |
| | | POW_SR = '00', $I_{out}$ = -1 A | | 260 | | |
| | | POW_SR = '11', $I_{out}$ = ±1 A | | 260 | | |
| | | POW_SR = '10', $I_{out}$ = ±1 A | | 110 | | |
| | | POW_SR = '01', $I_{out}$ = ±1 A | | 75 | | |
| **Deadtime and blanking** | | | | | | |
| $t_{DT}$ | Deadtime(1) | POW_SR = '00' | | 250 | | ns |
| | | POW_SR = '11', $f_{OSC}$ = 16 MHz | | 375 | | |
| | | POW_SR = '10', $f_{OSC}$ = 16 MHz | | 625 | | |
| | | POW_SR = '01', $f_{OSC}$ = 16 MHz | | 875 | | |
| $t_{blank}$ | Blanking time(1) | POW_SR = '00' | | 250 | | ns |
| | | POW_SR = '11', $f_{OSC}$ = 16 MHz | | 375 | | |
| | | POW_SR = '10', $f_{OSC}$ = 16 MHz | | 625 | | |
| | | POW_SR = '01', $f_{OSC}$ = 16 MHz | | 875 | | |
| **Source-drain diodes** | | | | | | |
| $V_{SD,HS}$ | High-side diode forward ON voltage | $I_{out}$ = 1 A | | 1 | 1.1 | V |
| $V_{SD,LS}$ | Low-side diode forward ON voltage | $I_{out}$ = 1 A | | 1 | 1.1 | V |
| $t_{rrHS}$ | High-side diode reverse recovery time | $I_{out}$ = 1 A | | 30 | | ns |
| $t_{rrLS}$ | Low-side diode reverse recovery time | $I_{out}$ = 1 A | | 100 | | ns |
| **Logic inputs and outputs** | | | | | | |
| $V_{IL}$ | Low logic level input voltage | | | | 0.8 | V |
| $V_{IH}$ | High logic level input voltage | | 2 | | | V |
| $I_{IH}$ | High logic level input current(4) | $V_{IN}$ = 5 V | | | 1 | µA |

chanhvinh.com



Table 5. Electrical characteristics (continued)

| Symbol | Parameter | Test condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $I_{IL}$ | Low logic level input current(5) | $V_{IN}$ = 0 V | -1 | | | µA |
| $V_{OL}$ | Low logic level output voltage (6) | $V_{DD}$ = 3.3 V, $I_{OL}$ = 4 mA | | | 0.3 | V |
| | | $V_{DD}$ = 5 V, $I_{OL}$ = 4 mA | | | 0.3 | |
| $V_{OH}$ | High logic level output voltage | $V_{DD}$ = 3.3 V, $I_{OH}$ = 4 mA | 2.4 | | | V |
| | | $V_{DD}$ = 5 V, $I_{OH}$ = 4 mA | 4.7 | | | |
| $R_{PU}$ $R_{PD}$ | CS pull-up and STBY pull-down resistors | $\overline{CS}$ = GND; $\overline{STBY/RST}$ = 5 V | 335 | 430 | 565 | kΩ |
| $I_{logic}$ | Internal logic supply current | 3.3 V $V_{REG}$ externally supplied, internal oscillator | | 3.7 | 4.3 | mA |
| $I_{logic,STBY}$ | Standby mode internal logic supply current | 3.3 V $V_{REG}$ externally supplied | | 2 | 2.5 | µA |
| $f_{STCK}$ | Step-clock input frequency | | | | 2 | MHz |
| **Internal oscillator and external oscillator driver** | | | | | | |
| $f_{osc,i}$ | Internal oscillator frequency | $T_j$ = 25 °C, $V_{REG}$ = 3.3 V | -3% | 16 | +3% | MHz |
| $f_{osc,e}$ | Programmable external oscillator frequency | | 8 | | 32 | MHz |
| $V_{OSCOUTH}$ | OSCOUT clock source high level voltage | Internal oscillator 3.3 V $V_{REG}$ externally supplied; $I_{OSCOUT}$ = 4 mA | 2.4 | | | V |
| $V_{OSCOUTL}$ | OSCOUT clock source low level voltage | Internal oscillator 3.3 V $V_{REG}$ externally supplied; $I_{OSCOUT}$ = 4 mA | | | 0.3 | V |
| $t_{rOSCOUT}$ $t_{fOSCOUT}$ | OSCOUT clock source rise and fall time | Internal oscillator | | | 20 | ns |
| $t_{extosc}$ | Internal to external oscillator switching delay | | | 3 | | ms |
| $t_{intosc}$ | External to internal oscillator switching delay | | | 1.5 | | µs |
| **SPI** | | | | | | |
| $f_{CK,MAX}$ | Maximum SPI clock frequency(7) | | 5 | | | MHz |
| $t_{rCK}$ $t_{fCK}$ | SPI clock rise and fall time(7) | $C_L$ = 30 pF | | | 25 | ns |
| $t_{hCK}$ $t_{lCK}$ | SPI clock high and low time(7) | | 75 | | | ns |
| $t_{setCS}$ | Chip select setup time(7) | | 350 | | | ns |
| $t_{holCS}$ | Chip select hold time(7) | | 10 | | | ns |
| $t_{disCS}$ | De-select time(7) | | 800 | | | ns |
| $t_{setSDI}$ | Data input setup time(7) | | 25 | | | ns |
| $t_{holSDI}$ | Data input hold time(7) | | 20 | | | ns |
| $t_{enSDO}$ | Data output enable time(7) | | | | 38 | ns |

chanhvinh.com



Table 5. Electrical characteristics (continued)

| Symbol | Parameter | Test condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $t_{disSDO}$ | Data output disable time(7) | | | | 47 | ns |
| $t_{vSDO}$ | Data output valid time(7) | | | | 57 | ns |
| $t_{holSDO}$ | Data output hold time(7) | | 37 | | | ns |
| **Switch input (SW)** | | | | | | |
| $R_{PUSW}$ | SW input pull-up resistance | SW = GND | 60 | 85 | 110 | kΩ |
| **Current control** | | | | | | |
| $I_{STEP,max}$ | Max. programmable reference current | | | 4 | | A |
| $I_{STEP,min}$ | Min. programmable reference current | | | 31 | | mA |
| **Overcurrent protection** | | | | | | |
| $I_{OCD,MAX}$ | Maximum programmable overcurrent detection threshold | OCD_TH = ‘1111’ | | 6 | | A |
| $I_{OCD,MIN}$ | Minimum programmable overcurrent detection threshold | OCD_TH = ‘0000’ | | 0.375 | | A |
| $I_{OCD,RES}$ | Programmable overcurrent detection threshold resolution | | | 0.375 | | A |
| $t_{OCD,Flag}$ | OCD to flag signal delay time | $dI_{out}/d_t$ = 350 A/µs | | 650 | 1000 | ns |
| $t_{OCD,SD}$ | OCD to shutdown delay time | $dI_{out}/d_t$ = 350 A/µs POW_SR = '10' | | 600 | | µs |
| **Standby** | | | | | | |
| $I_{qSTBY}$ | Quiescent motor supply current in standby conditions | $V_S$ = 8 V | | 26 | 34 | µA |
| | | $V_S$ = 36 V | | 30 | 36 | |
| $t_{STBY,min}$ | Minimum standby time | | | 10 | | µs |
| $t_{logicwu}$ | Logic power-on and wake-up time | | | 38 | 45 | µs |
| $t_{cpwu}$ | Charge pump power-on and wake-up time | Power bridges disabled, $C_p$ = 10 nF, $C_{boot}$ = 220 nF | | 650 | | µs |
| **Internal voltage regulator** | | | | | | |
| $V_{REG}$ | Voltage regulator output voltage | | 2.9 | 3 | 3.2 | V |
| $I_{REG}$ | Voltage regulator output current | | | | 40 | mA |
| $V_{REG, drop}$ | Voltage regulator output voltage drop | $I_{REG}$ = 40 mA | | 50 | | mV |
| $I_{REG,STBY}$ | Voltage regulator standby output current | | | | 10 | mA |

chanhvinh.com



**Table 5. Electrical characteristics (continued)**

| Symbol | Parameter | Test condition | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| **Integrated analog-to-digital converter** | | | | | | |
| $N_{ADC}$ | Analog-to-digital converter resolution | | | 5 | | bit |
| $V_{ADC,ref}$ | Analog-to-digital converter reference voltage | | | $V_{REG}$ | | V |
| $f_S$ | Analog-to-digital converter sampling frequency | | | $f_{OSC}$/ 512 | | kHz |

1. Accuracy depends on oscillator frequency accuracy.
2. Tested at 25 °C in a restricted range and guaranteed by characterization.
3. Rise and fall time depends on motor supply voltage value. Refer to $SR_{out}$ values in order to evaluate the actual rise and fall time.
4. Not valid for the $\overline{\text{STBY/RST}}$ pin which has an internal pull-down resistor.
5. Not valid for the SW and CS pins which have an internal pull-up resistor.
6. $\overline{\text{FLAG}}$, $\overline{\text{BUSY}}$ and SYNC open drain outputs included.
7. See *Figure 19: SPI timings diagram on page 38* for details.

chanhvinh.com



# 4 Pin connection

**Figure 2. HTSSOP28 pin connection (top view)**

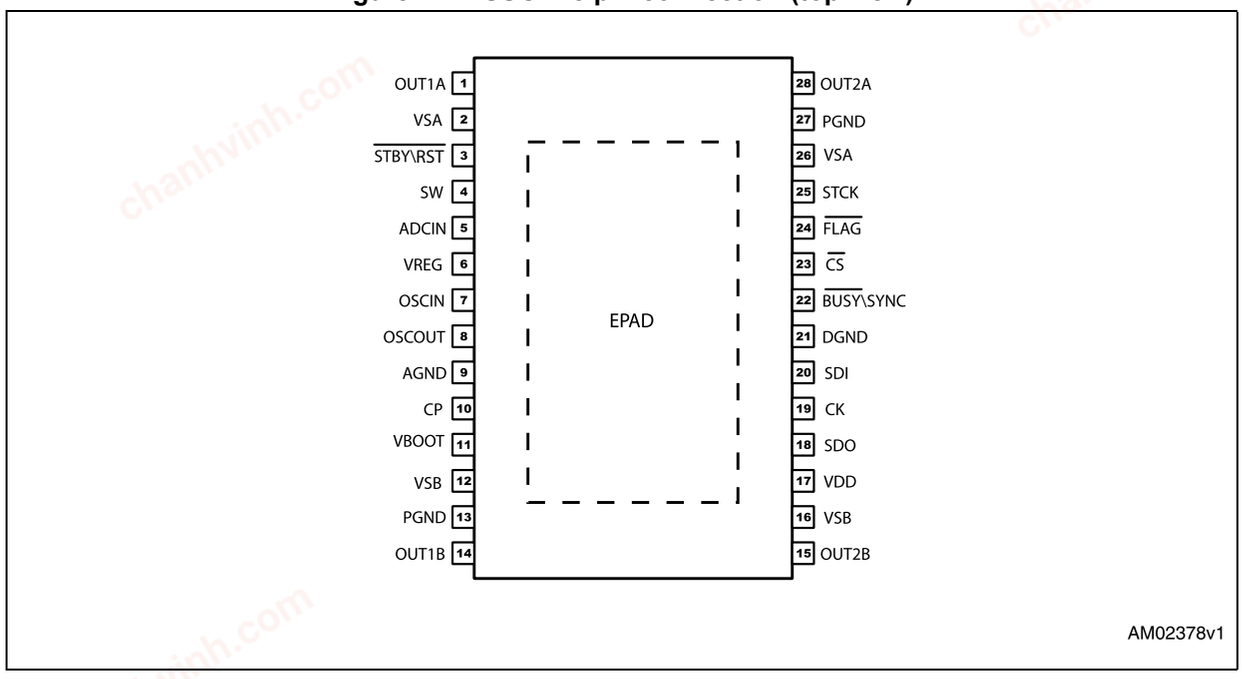
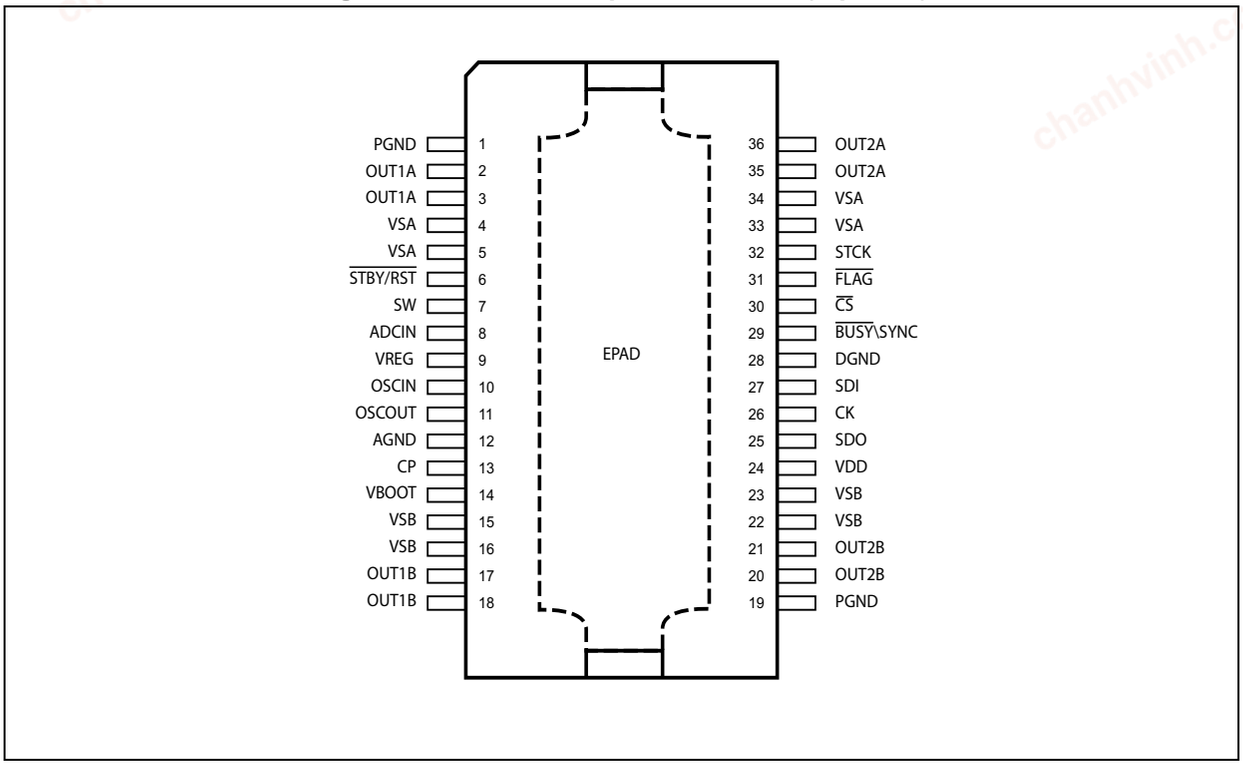



**Figure 3. POWERSO36 pin connection (top view)**



chanhvinh.com



# Pin list

**Table 6. Pin description**

| No. | Name | Type | Function |
|---|---|---|---|
| 17 | VDD | Power | Logic output supply voltage (pull-up reference) |
| 6 | VREG | Power | Internal 3 V voltage regulator output and 3.3 V external logic supply |
| 7 | OSCIN | Analog input | Oscillator pin 1. To connect an external oscillator or clock source. If this pin is unused, it should be left floating. |
| 8 | OSCOUT | Analog output | Oscillator pin 2. To connect an external oscillator. When the internal oscillator is used this pin can supply 2/4/8/16 MHz. If this pin is unused, it should be left floating. |
| 10 | CP | Output | Charge pump oscillator output |
| 11 | Vboot | Supply voltage | Bootstrap voltage needed for driving the high-side power DMOS of both bridges (A and B). |
| 5 | ADCIN | Analog input | Internal analog-to-digital converter input |
| 2 | VSA | Power supply | Full bridge A power supply pin. It must be connected to VSB. |
| 26 | | | |
| 12 | VSB | Power supply | Full bridge B power supply pin. It must be connected to VSA. |
| 16 | | | |
| 27 | PGND | Ground | Power ground pin |
| 13 | | | |
| 1 | OUT1A | Power output | Full bridge A output 1 |
| 28 | OUT2A | Power output | Full bridge A output 2 |
| 14 | OUT1B | Power output | Full bridge B output 1 |
| 15 | OUT2B | Power output | Full bridge B output 2 |
| 9 | AGND | Ground | Analog ground. |
| 4 | SW | Logical input | External switch input pin. If not used the pin should be connected to VDD. |
| 21 | DGND | Ground | Digital ground |
| 22 | $\overline{\text{BUSY}}$\SYNC | Open drain output | By default, this BUSY pin is forced low when the device is performing a command. Otherwise the pin can be configured to generate a synchronization signal. |
| 18 | SDO | Logic output | Data output pin for serial interface |
| 20 | SDI | Logic input | Data input pin for serial interface |
| 19 | CK | Logic input | Serial interface clock |
| 23 | $\overline{\text{CS}}$ | Logic input | Chip select input pin for serial interface |
| 24 | $\overline{\text{FLAG}}$ | 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). |

chanhvinh.com



Table 6. Pin description (continued)

| No. | Name | Type | Function |
|---|---|---|---|
| 3 | $\overline{\text{STBY\RST}}$ | Logic input | Standby and reset pin. LOW logic level resets the logic and puts the device into standby mode. If not used, it should be connected to VDD |
| 25 | STCK | Logic input | Step-clock input |
| EPAD | Exposed pad | Ground | Internally connected to PGND, AGND and DGND pins |

chanhvinh.com

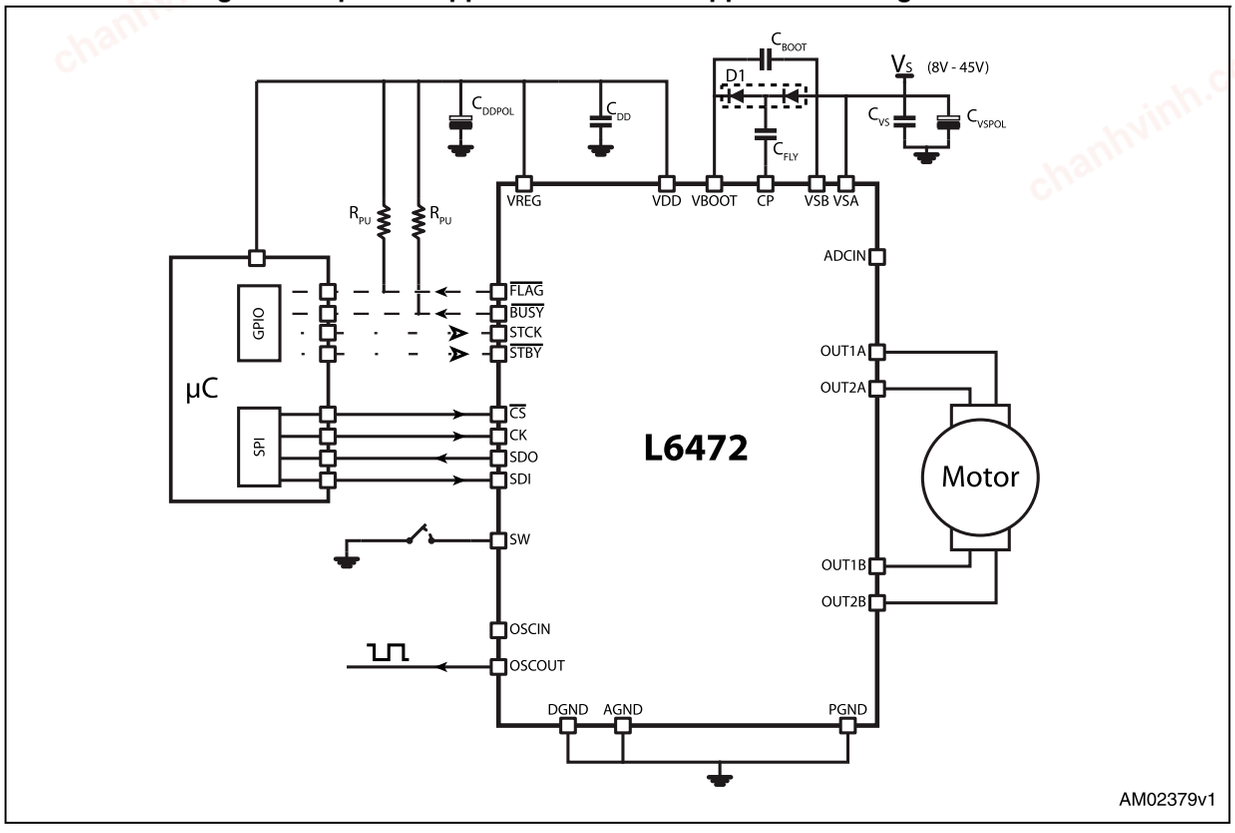


# 5 Typical applications

**Table 7. Typical application values**

| Name | Value |
|---|---|
| $C_{VS}$ | 220 nF |
| $C_{VSPOL}$ | 100 µF |
| $C_{REG}$ | 100 nF |
| $C_{REGPOL}$ | 47 µF |
| $C_{DD}$ | 100 nF |
| $C_{DDPOL}$ | 10 µF |
| D1 | Charge pump diodes |
| $C_{BOOT}$ | 220 nF |
| $C_{FLY}$ | 10 nF |
| $R_{PU}$ | 39 kΩ |
| $R_{SW}$ | 100 Ω |
| $C_{SW}$ | 10 nF |

**Figure 4. Bipolar stepper motor control application using the L6472**



chanhvinh.com



# 6 Functional description

## 6.1 Device power-up

At the end of power-up, the device state is the following:

- Registers 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 Z)
- UVLO bit in the STATUS register is forced low (fail condition)
- FLAG output is forced low.

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

- $V_S$ is greater than $V_{SthOn}$
- $V_{REG}$ is greater than $V_{REGth}$ = 2.8 V (typ.)
- Internal oscillator is operative.

Any motion command causes the device to exit from High Z state (HardStop and SoftStop included).

## 6.2 Logic I/O

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

Pin SDO is a TTL/CMOS compatible logic output. The VDD pin voltage sets the logic output pin voltage range; when it is connected to VREG or a 3.3 V external supply voltage, the output is 3.3 V compatible. When VDD is connected to a 5 V supply voltage, SDO is 5 V compatible.

VDD is not internally connected to $V_{REG}$, an external connection is always needed.

A 10 µF capacitor should be connected to the VDD pin in order to obtain a proper operation.

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

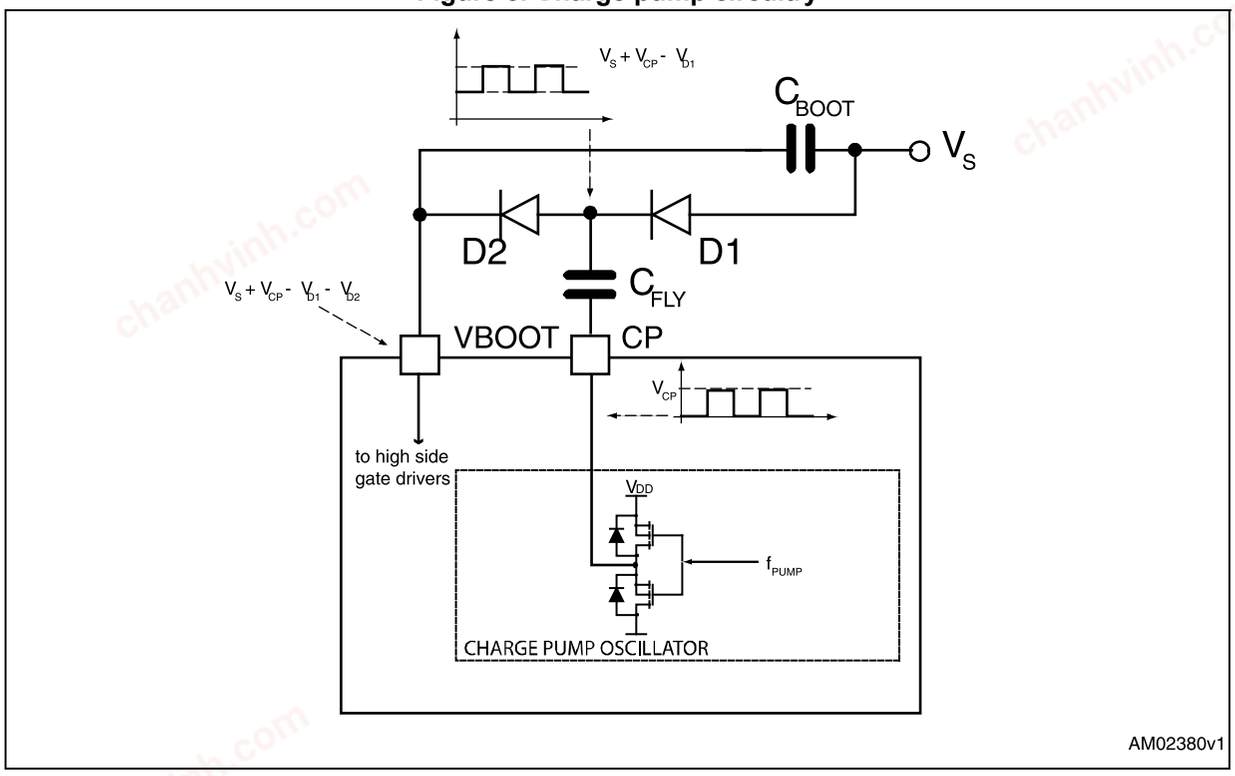
## 6.3 Charge pump

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 Vboot is obtained through an oscillator and a few external components realizing a charge pump (see *Figure 5*).

chanhvinh.com



Figure 5. Charge pump circuitry




## 6.4 Microstepping

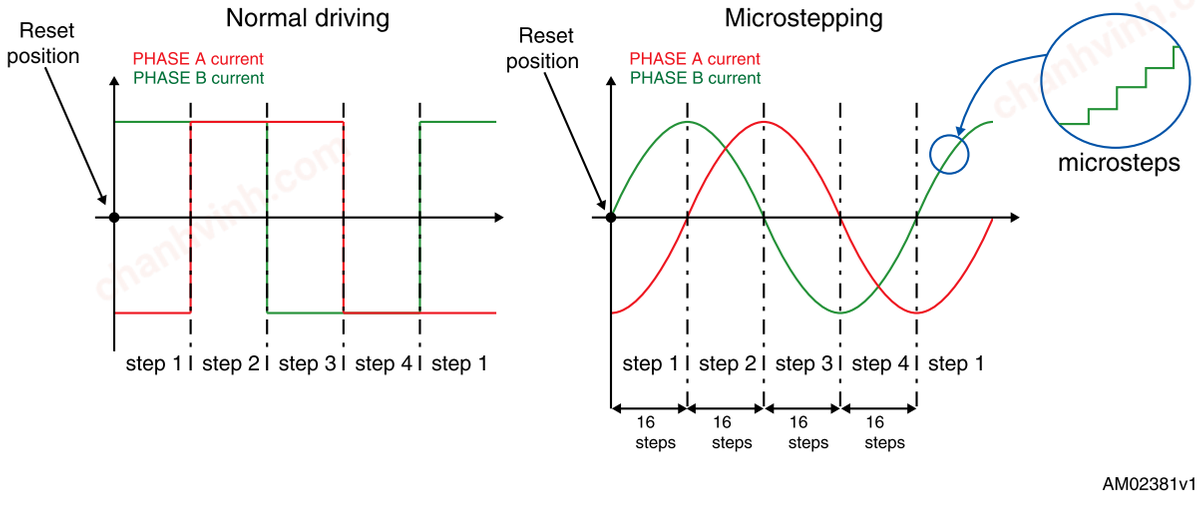
The driver is able to divide the single step into up to 16 microsteps. Step mode can be programmed by the STEP_SEL parameter in the STEP_MODE register (see *Table 20 on page 47*).

Step mode can only be changed when bridges are disabled. Every time step mode is changed, the electrical position (i.e. the point of microstepping sine wave that is generated) is reset to zero, and the absolute position counter value (see *Section 6.5*) becomes meaningless.

chanhvinh.com



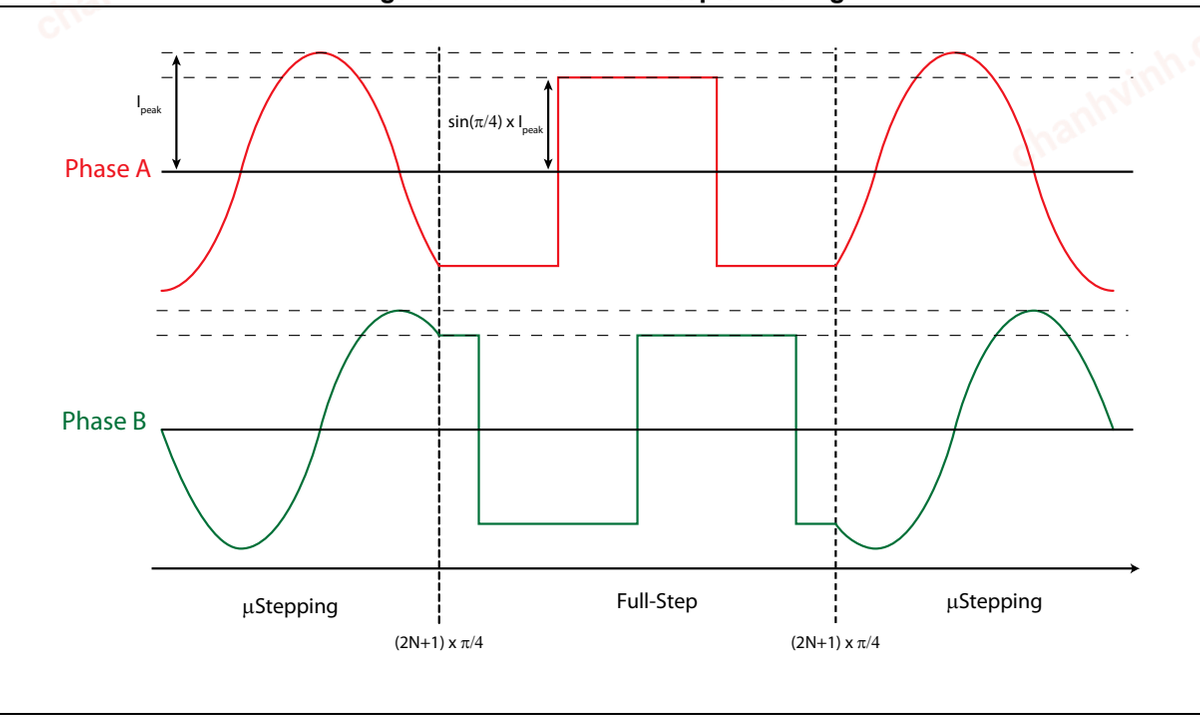
**Figure 6. Normal mode and microstepping (16 microsteps)**




## Automatic full-step mode

When motor speed is greater than a programmable full-step speed threshold, the L6472 switches automatically to full-step mode (see *Figure 7*); the driving mode returns to microstepping when motor speed decreases below the full-step speed threshold. The full-step speed threshold is set through the FS_SPD register (see *Section 9.1.9 on page 44*).

**Figure 7. Automatic full-step switching**



chanhvinh.com



## 6.5 Absolute position counter

An internal 22-bit register (ABS_POS) keeps track of the motor motion 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 $+2^{21}-1$ (μ) steps (see *Section 9.1.1 on page 41*).

## 6.6 Programmable speed profiles

The user can easily program a customized speed profile, independently defining acceleration, deceleration, maximum and minimum speed values through the ACC, DEC, MAX_SPEED and MIN_SPEED registers respectively (see *Section 9.1.5 on page 42*, *9.1.6 on page 42*, *9.1.7 on page 43* and *9.1.8 on page 43*).

When a command is sent to the device, the integrated logic generates the microstep frequency profile that performs a motor motion compliant to 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 selected step mode.

Acceleration and deceleration parameters range from $2^{-40}$ to $(2^{12}-2) \cdot 2^{-40}$ step/tick2 (equivalent to 14.55 to 59590 step/s2).

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).

## 6.7 Motor control commands

The L6472 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 9.2 on page 54*.

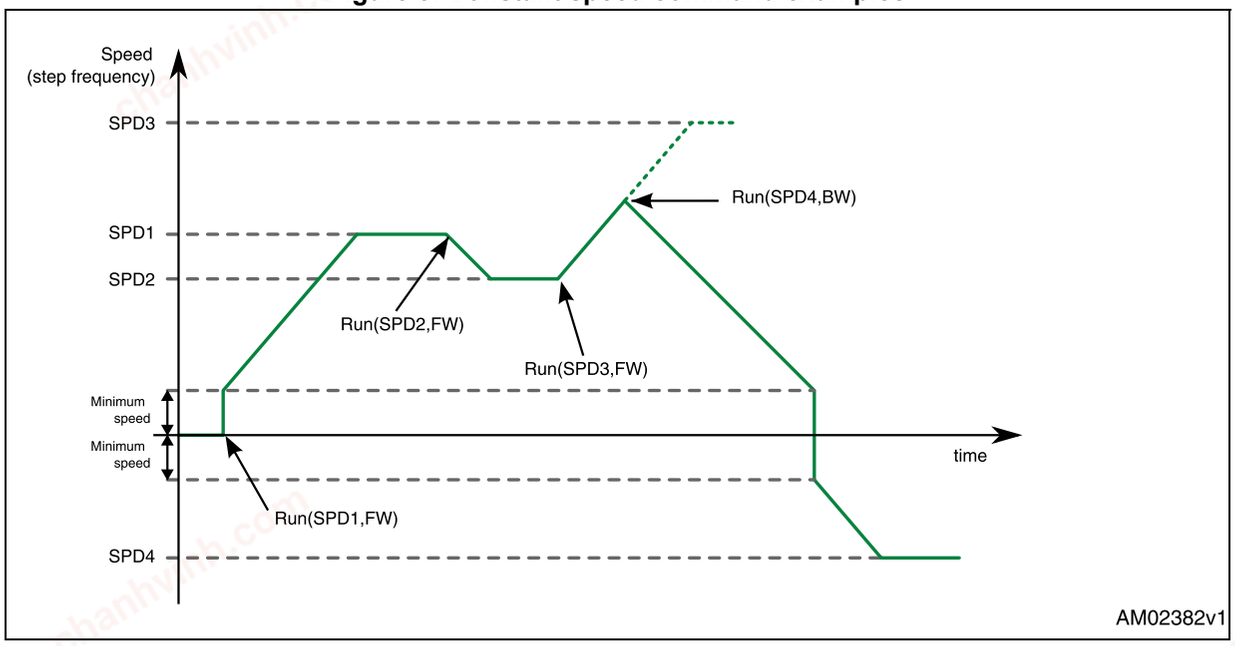
chanhvinh.com



### 6.7.1 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 8. Constant speed command examples**




### 6.7.2 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 by performing the minimum path (minimum physical distance) or forcing a direction (see *Figure 9*).

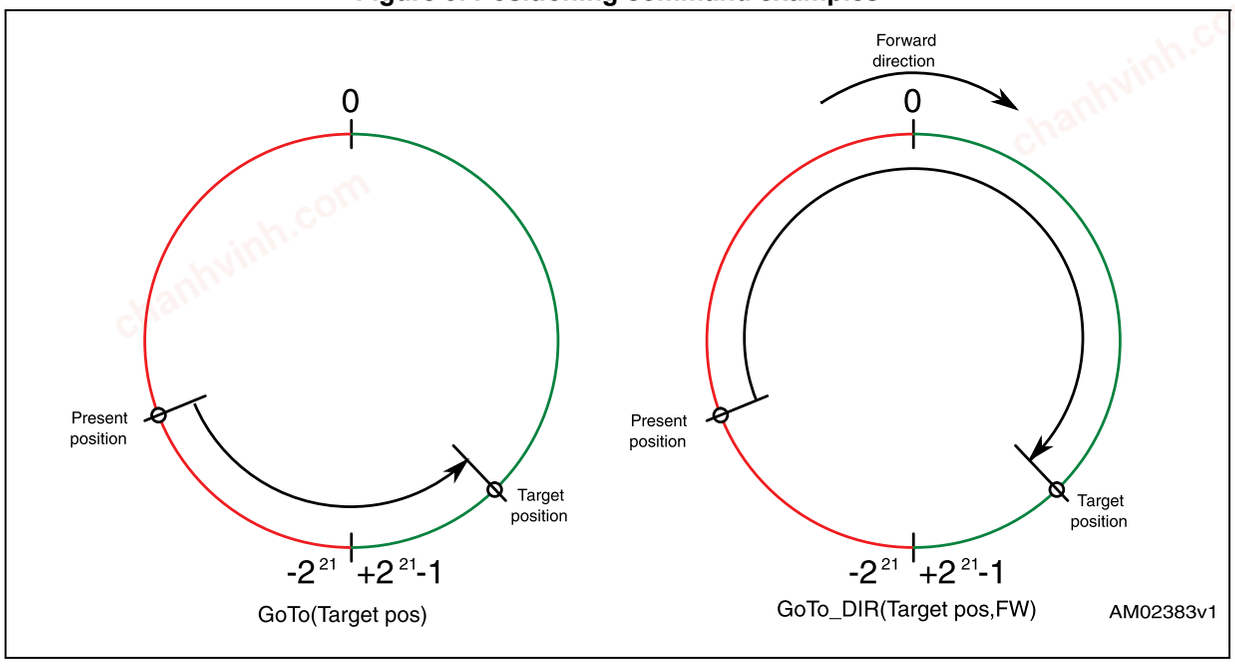
The performed motor motion is compliant to programmed speed profile boundaries (acceleration, deceleration, minimum and maximum speed).

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

chanhvinh.com



**Figure 9. Positioning command examples**



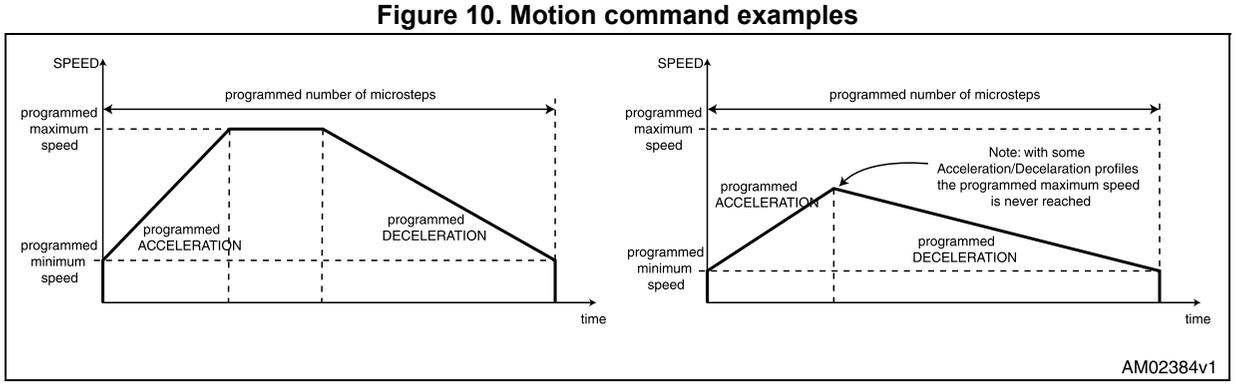

## 6.7.3 Motion commands

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 10*).

The 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 10. Motion command examples**




## 6.7.4 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 maintaining the rotor position (a holding torque is applied).

chanhvinh.com



The HardStop command stops the motor instantly, ignoring deceleration constraints and maintaining 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 high-impedance state (no holding torque is present).

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

### 6.7.5 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 by one microstep in the programmed direction and the 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 the motor status is considered stopped whatever the STCK signal frequency (the MOT_STATUS parameter in the STATUS register equal to g00h).

### 6.7.6 GoUntil and ReleaseSW commands

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

The GoUntil and ReleaseSW commands can be used in combination with external switch input (see *Section 6.13 on page 30*) 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 e0f, the motor does not decelerate but it immediately stops (as a HardStop command).

The ReleaseSW command makes the motor run at the 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.

chanhvinh.com



## 6.8 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 25 on page 50*).

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

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

### 6.8.1 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 the OSC_SEL value); otherwise it is unused (see *Figure 11*).

### 6.8.2 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 11*). The crystal/resonator and load capacitors (CL) must be placed as close as possible to the pins. Refer to *Table 8* for the choice of the load capacitor value according to the external oscillator frequency.

**Table 8. CL values according to external oscillator frequency**

| Crystal/resonator freq. (1) | CL (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) |

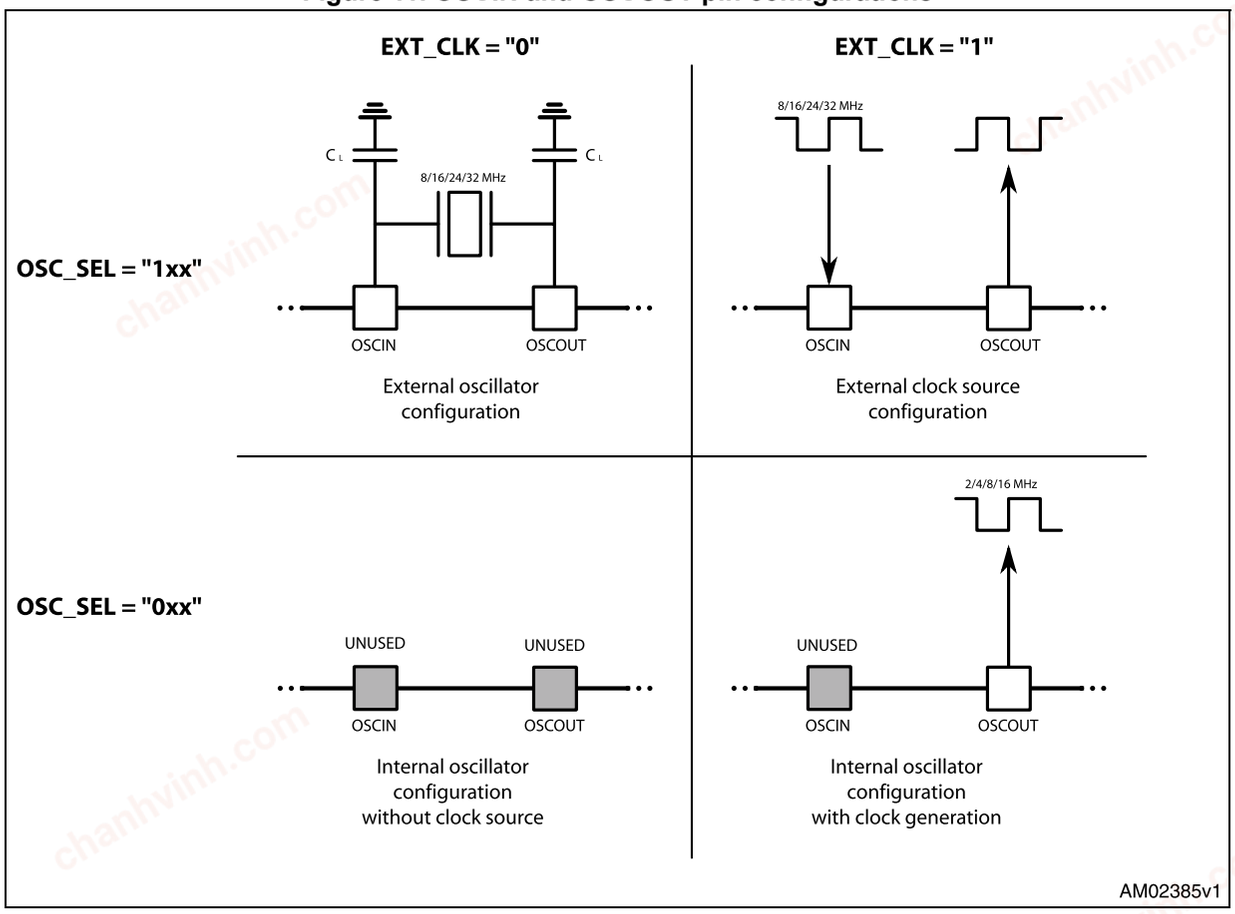
1. First harmonic resonance frequency.
2. Lower ESR value allows the driving of greater load capacitors.

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 11*).

chanhvinh.com



Figure 11. OSCIN and OSCOUT pin configurations




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

*When OSCOUT is UNUSED it should be left floating.*

## 6.9 Overcurrent detection

When the current in any of the Power MOSFETs exceeds a programmed overcurrent threshold, the STATUS register OCD flag is forced low until the overcurrent event expires and a GetStatus command is sent to the IC (see *Section 9.1.19 on page 52* and *9.2.20 on page 63*). The overcurrent event expires when all the Power MOSFET currents fall below the programmed overcurrent threshold.

The overcurrent threshold can be programmed through the OCD_TH register in one of 16 available values ranging from 375 mA to 6 A with steps of 375 mA (see *Table 18 on page 47*).

It is possible to set if an overcurrent event causes or not the MOSFET turn-off (bridges in high-impedance status) acting on the OC_SD bit in the CONFIG register (see *Section 9.1.18 on page 49*). The OCD flag in the STATUS register is raised anyway (see *Table 26 on page 50*).

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

chanhvinh.com



**Warning:** **The overcurrent shutdown is a critical protection feature. It is not recommended to disable it.**

## 6.10 Undervoltage lockout (UVLO)

The L6472 provides motor supply UVLO protection. When the motor supply voltage falls below the $V_{SthOff}$ threshold voltage, the STATUS register UVLO flag is forced low. When a GetStatus command is sent to the IC, and the undervoltage condition expires, the UVLO flag is released (see *Section 9.1.19 on page 52* and *9.2.20 on page 63*). The undervoltage condition expires when the motor supply voltage goes over the $V_{SthOn}$ threshold voltage. When the device is in the undervoltage condition, no motion command can be performed. The UVLO flag is forced low by logic reset (power-up included) even if no UVLO condition is present.

## 6.11 Thermal warning and thermal shutdown

An internal sensor allows the L6472 to detect when the device internal temperature exceeds a thermal warning or an overtemperature threshold.

When the thermal warning threshold ($T_{j(WRN)}$) is reached, the TH_WRN bit in the STATUS register is forced low (see *Section 9.1.19*) until the temperature decreases below $T_{j(WRN)}$ and a GetStatus command is sent to the IC (see *Section 9.1.19* and *9.2.20*).

When the thermal shutdown threshold ($T_{j(OFF)}$) is reached, the device goes into the thermal shutdown condition: the TH_SD bit in the STATUS register is forced low, the power bridges are disabled bridges in high-impedance state and the HiZ bit in the STATUS register is raised (see *Section 9.1.19*).

The thermal shutdown condition only expires when the temperature goes below the thermal warning threshold ($T_{j(WRN)}$).

On exiting the thermal shutdown condition, the bridges are still disabled (HiZ flag high); whichever motion command makes the device exit from High Z state (HardStop and SoftStop included).

## 6.12 Reset and standby

The device can be reset and put into standby mode through a dedicated pin. When the $\overline{STBY}$\$\overline{RST}$ pin is driven low, the bridges are left open (High Z state), the internal charge pump is stopped, the SPI interface and control logic are disabled, and the internal 3 V voltage regulator maximum output current is reduced to IREG,STBY; as a result, the L6472 heavily reduces the power consumption. At the same time the register values are reset to default and all protection functions are disabled. $\overline{STBY}$\$\overline{RST}$ input must be forced low at least for $t_{STBY,min}$ in order to ensure the complete switch to standby mode.

On exiting standby mode, as well as for IC power-up, a delay of up to $t_{logicwu}$ must be given before applying a new command to allow proper oscillator and logic startup and a delay of up to $t_{cpwu}$ must be given to allow the charge pump startup.

chanhvinh.com



On exiting standby mode the bridges are disabled (HiZ flag high) and whichever motion command causes the device to exit High Z state (HardStop and SoftStop included).

**Warning: It is not recommended to reset the device when outputs are active. The device should be switched to high-impedance state before being reset.**

## 6.13 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 12*).

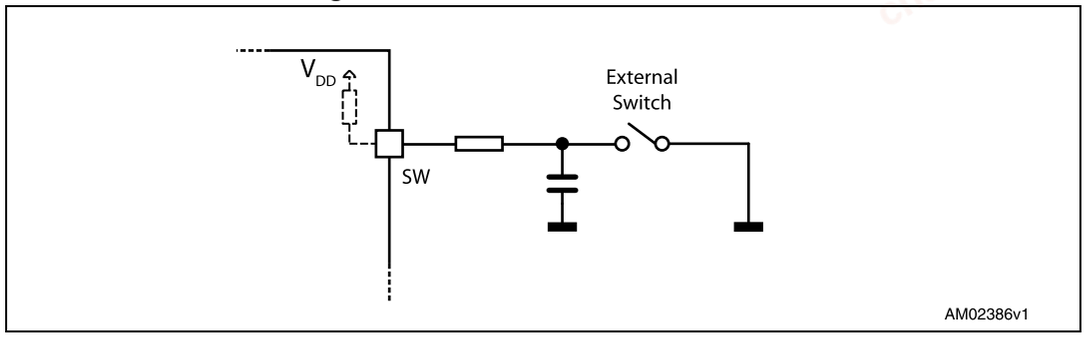
The SW_F bit of the STATUS register indicates if the switch is open (‘0’) or closed (‘1’) (see *Section 9.1.19 on page 52*); the bit value is refreshed at every system clock cycle (125 ns). The SW_EVN flag of the STATUS register is raised when a switch turn-on event (SW input falling edge) is detected (see *Section 9.1.19*). A GetStatus command releases the SW_EVN flag (see *Section 9.2.20 on page 63*).

By default a switch turn-on event causes a HardStop interrupt (SW_MODE bit of the CONFIG register set to ‘0’). Otherwise (SW_MODE bit of the 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 26 on page 50*).

The switch input can be used by the GoUntil and ReleaseSW commands as described in *Section 9.2.10 on page 59* and *9.2.11 on page 60*.

If the SW input is not used, it should be connected to VDD.

**Figure 12. External switch connection**




## 6.14 Programmable DMOS slew rate, deadtime and blanking time

Using the POW_SR parameter in the CONFIG register, it is possible to set the commutation speed of the power bridge output (see *Table 28 on page 51*).

chanhvinh.com



## 6.15 Integrated analog-to-digital converter

The L6472 integrates an $N_{ADC}$ bit ramp-compare analog-to-digital converter with a reference voltage equal to VREG. The analog-to-digital converter input is available through the ADCIN pin and the conversion result is available in the ADC_OUT register (see *Section 9.1.13 on page 46*). The sampling frequency is equal to the clock frequency divided by 512.

The ADC_OUT value can be used for the torque regulation or can remain at the user's disposal.

## 6.16 Internal voltage regulator

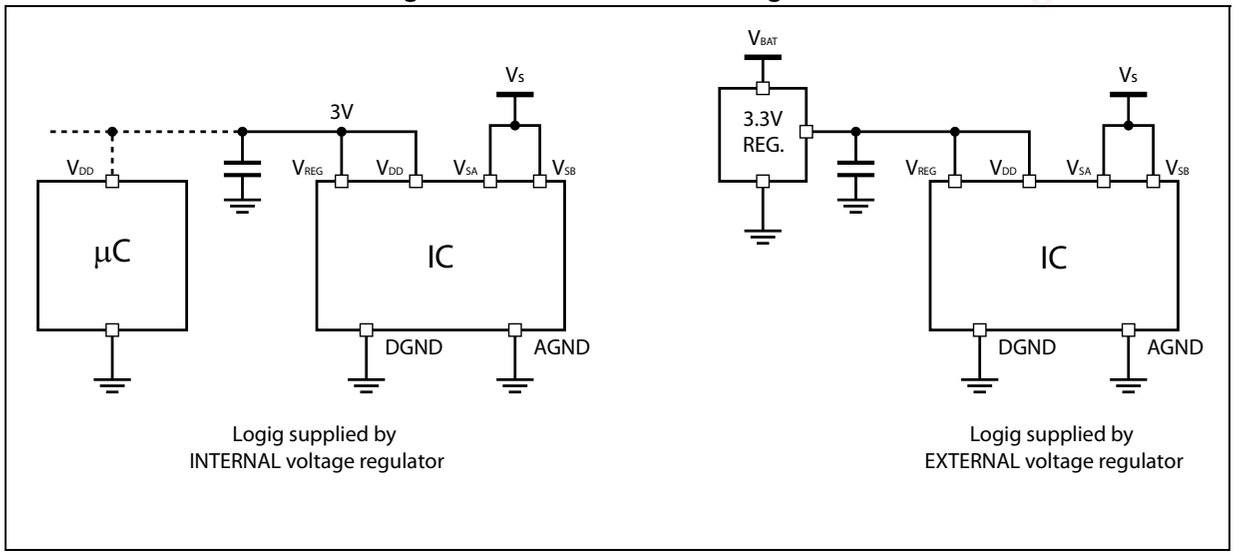
The L6472 device integrates a voltage regulator which generates a 3 V voltage starting from motor power supply (VSA and VSB). In order to make the voltage regulator stable, at least 22 µF should be connected between the VREG pin and ground (the suggested value is 47 µF).

The internal voltage regulator can be used to supply the VDD pin in order to make the device digital output range 3.3 V compatible (*Figure 13*). A digital output range 5 V compatible can be obtained connecting the VDD pin to an external 5 V voltage source. In both cases, a 10 µF capacitance should be connected to the VDD pin in order to obtain a correct operation.

The internal voltage regulator is able to supply a current up to $I_{REG,MAX}$, internal logic consumption included ($I_{logic}$). When the device is in standby mode the maximum current that can be supplied is $I_{REG, STBY}$, internal consumption included ($I_{logic, STBY}$).

If an external 3.3 V regulated voltage is available, it can be applied to the VREG pin in order to supply all the internal logic and avoid power dissipation of the internal 3 V voltage regulator (*Figure 13*). The external voltage regulator should never sink current from the VREG pin.

**Figure 13. Internal 3 V linear regulator**



chanhvinh.com



## 6.17 BUSY\SYNC pin

This pin is an open drain output which can be used as the busy flag or synchronization signal according to the SYNC_EN bit value (STEP_MODE register).

### 6.17.1 BUSY operation mode

The pin works as busy signal when the SYNC_EN bit is set low (default condition). In this mode the output is forced low while a constant speed, absolute positioning or motion command is under execution. The $\overline{\text{BUSY}}$ pin is released when the command has been executed (target speed or target position reached). The STATUS register includes a BUSY flag that is the BUSY pin mirror (see *Section 9.1.19 on page 52*).

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

### 6.17.2 SYNC operation mode

The pin works as a synchronization signal when the SYNC_EN bit is set high. In this mode a step-clock signal is provided on the output according to a SYNC_SEL and STEP_SEL parameter combination (see *Section 9.1.16 on page 47*).

## 6.18 FLAG pin

By default an internal open drain transistor pulls the FLAG pin to ground when at least one of the following conditions occur:

- Power-up or standby/reset exit
- Overcurrent detection
- Thermal warning
- Thermal shutdown
- UVLO
- Switch turn-on event
- Wrong command
- Non-performable command.

It is possible to mask one or more alarm conditions by programming the ALARM_EN register (see *Table 23 on page 49*). If the corresponding bit of the ALARM_EN register is low, the alarm condition is masked and it does not cause a FLAG pin transition; all other actions imposed by alarm conditions are performed anyway. In the case of daisy chain configuration, the FLAG pins of different ICs can be OR-wired to save host controller GPIOs.

chanhvinh.com



# 7 Phase current control

The L6472 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 7.1*. Furthermore, the L6472 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 9.1.11 on page 45*, *9.1.12 on page 45*, *9.1.13 on page 46* and *9.1.18 on page 49* 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 7.4 on page 37*). The output current amplitude can also be regulated by the ADCIN voltage value (see *Section 6.15*).

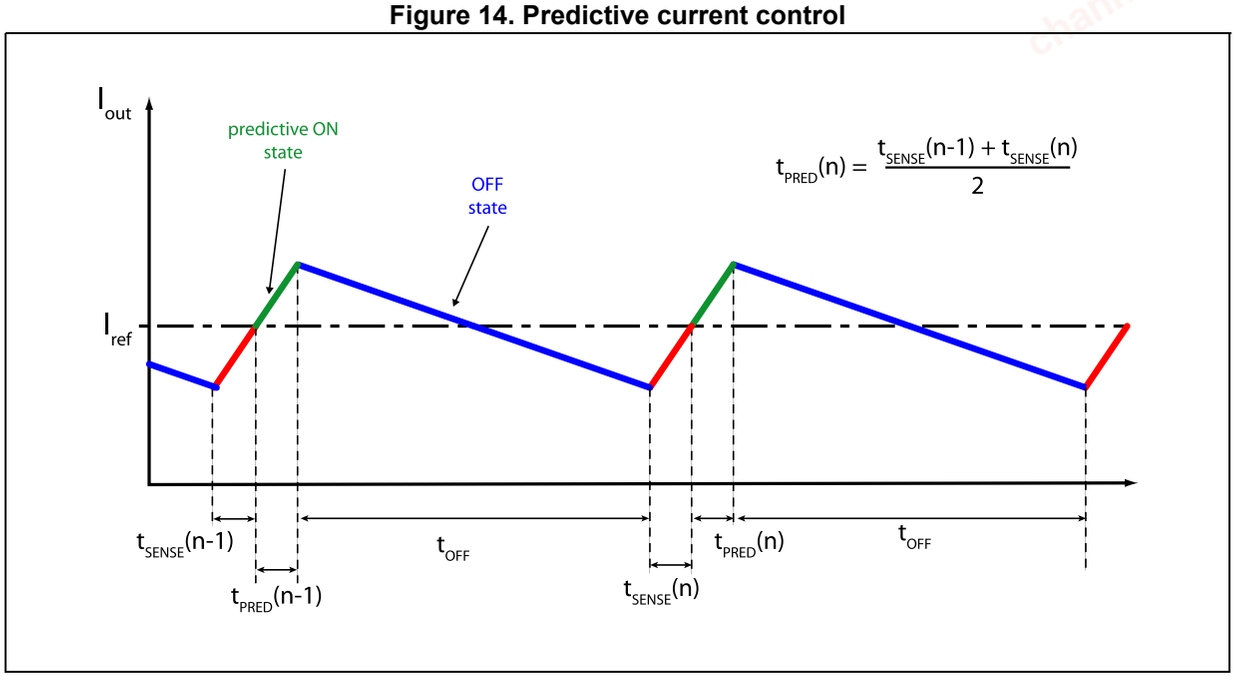
Each bridge is driven by an independent control system that shares the control parameters only with other bridges.

## 7.1 Predictive current control

Unlike a classical peak current control system, that causes 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 14*.

**Figure 14. Predictive current control**



chanhvinh.com



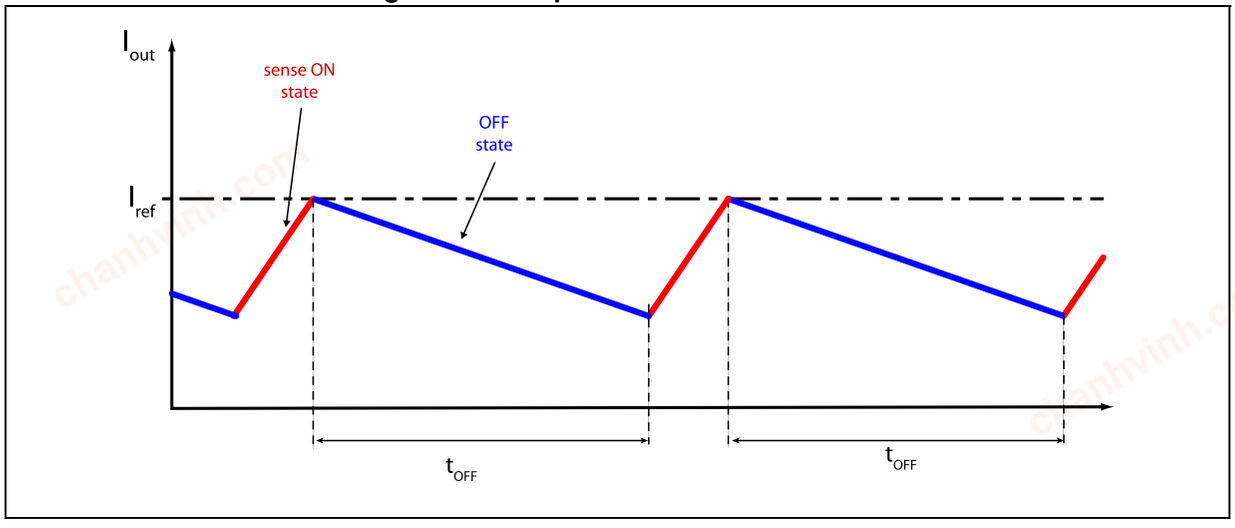
At the end of the predictive ON state the power stage is set in the OFF state for a fixed time, as in a constant $t_{OFF}$ current control. During the OFF state both slow and fast decay can be performed; the better decay combination is automatically selected by the L6472, as described in *Section 7.2*.

As shown in *Figure 14*, 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 the PRED_EN bit in the CONFIG register low (default condition). In this case, after the sense phase ($t_{SENSE}$) the power stage is set in the OFF state, as shown in *Figure 15*.

**Figure 15. Non-predictive current control**




## 7.2 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 by turning on both the low-side MOSFETs of the power stage and the current recirculates in the lower half of the bridge (slow decay).

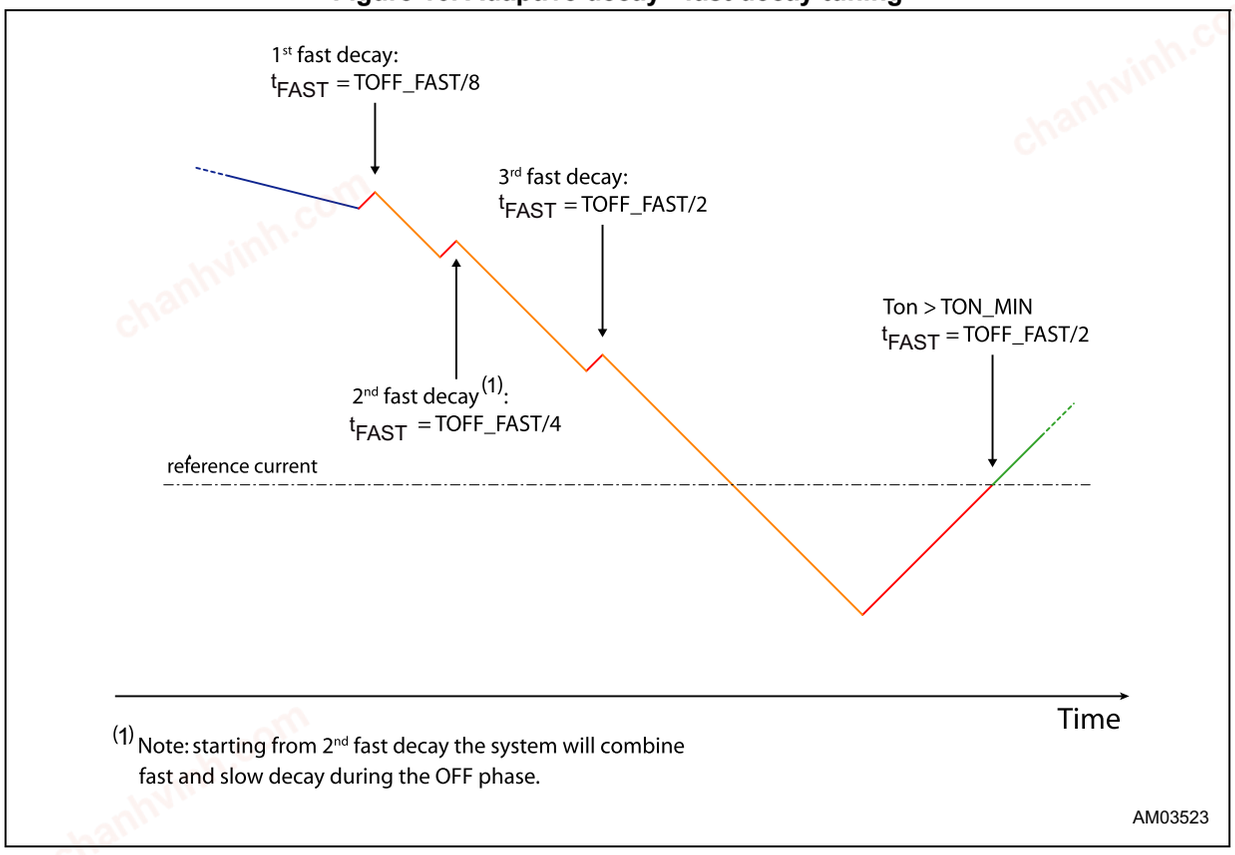
If, during a PWM cycle, the target current threshold is reached in a time shorter than the TON_MIN value, a fast decay of TOFF_FAST/8 (T_FAST register) is immediately performed turning on the opposite MOS of both half-bridges and the current recirculates back to the supply bus.

After this time, the bridge returns to the 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 7.1*. The maximum fast decay duration is set by the TOFF_FAST value.

chanhvinh.com



**Figure 16. Adaptive decay - fast decay tuning**



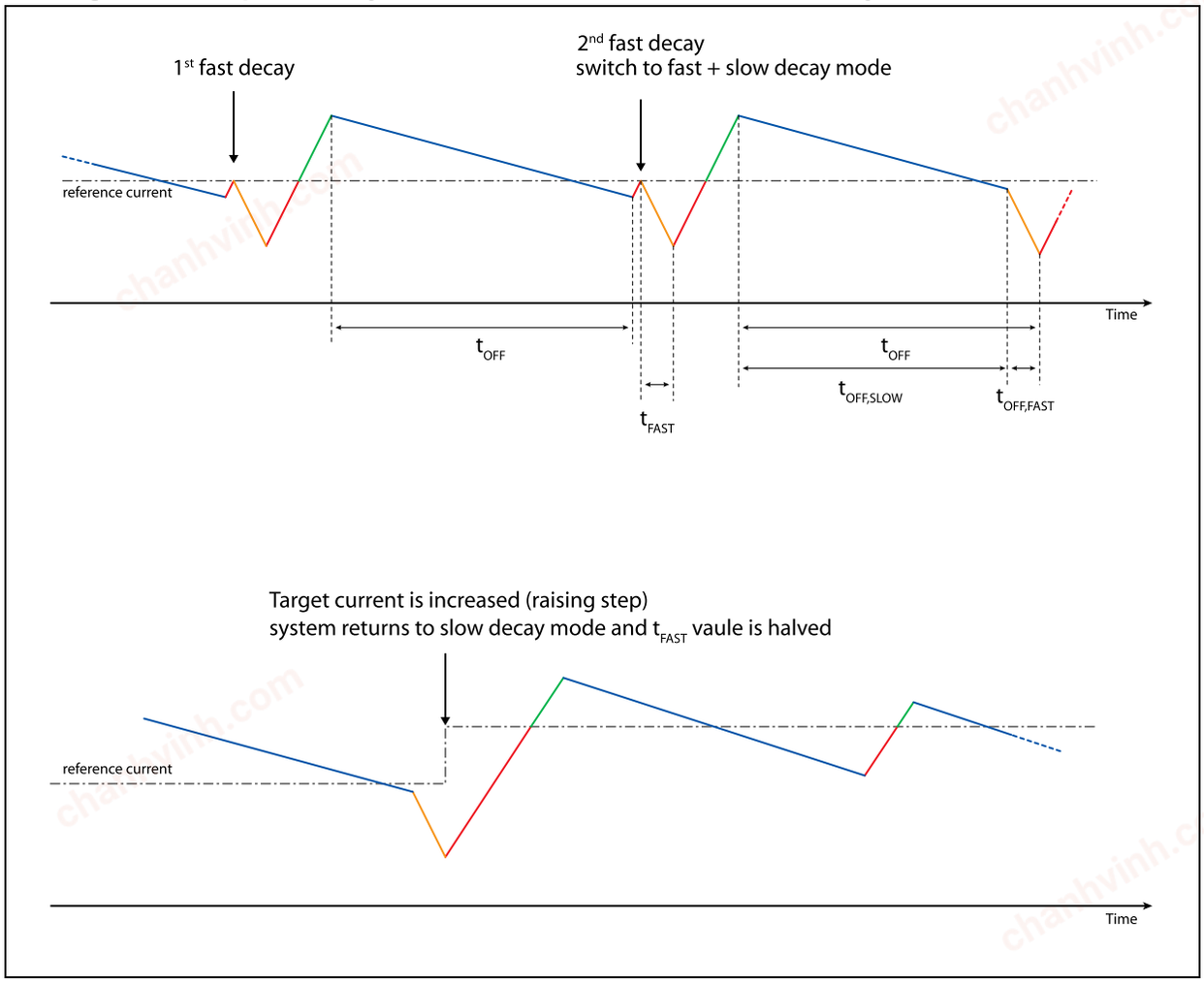

When two or more fast decays are performed with the present target current, the control system adds a fast decay at the end of every OFF time, keeping the OFF state duration constant ($t_{OFF}$ is split into $t_{OFF,SLOW}$ and $t_{OFF,FAST}$). When the current threshold is increased by a microstep change (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.

chanhvinh.com



**Figure 17. Adaptive decay switch from normal to slow + fast decay mode and vice-versa**




## 7.3 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. Anyway, exceeding the fast duration may cause a strong ripple on the step change. The L6472 device automatically adjusts these fast decays reducing the current ripple.

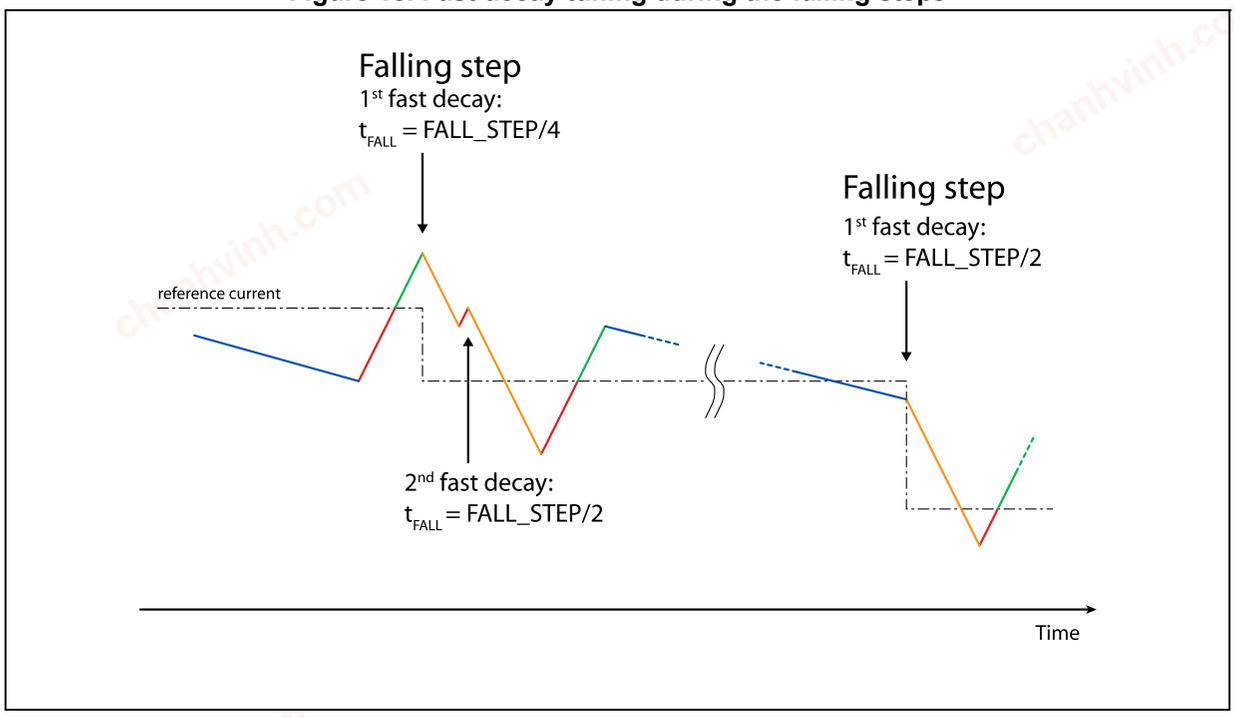
At reset, the fast decay value ($t_{FALL}$) is set to FALL_STEP/4 (T_FAST register). The $t_{FALL}$ value is doubled every time, within the same falling step, an extra fast decay is necessary to obtain an ON time greater than TON_MIN. The maximum $t_{FALL}$ value is equal to FALL_STEP.

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.

chanhvinh.com



**Figure 18. Fast decay tuning during the falling steps**




## 7.4 Torque regulation (output current amplitude regulation)

The output current amplitude can be regulated in two ways: writing the 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 *Section 9.1.14 on page 46*). 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 *Section 9.1.10 on page 44*).

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.

chanhvinh.com



# 8 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 L6472 (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.

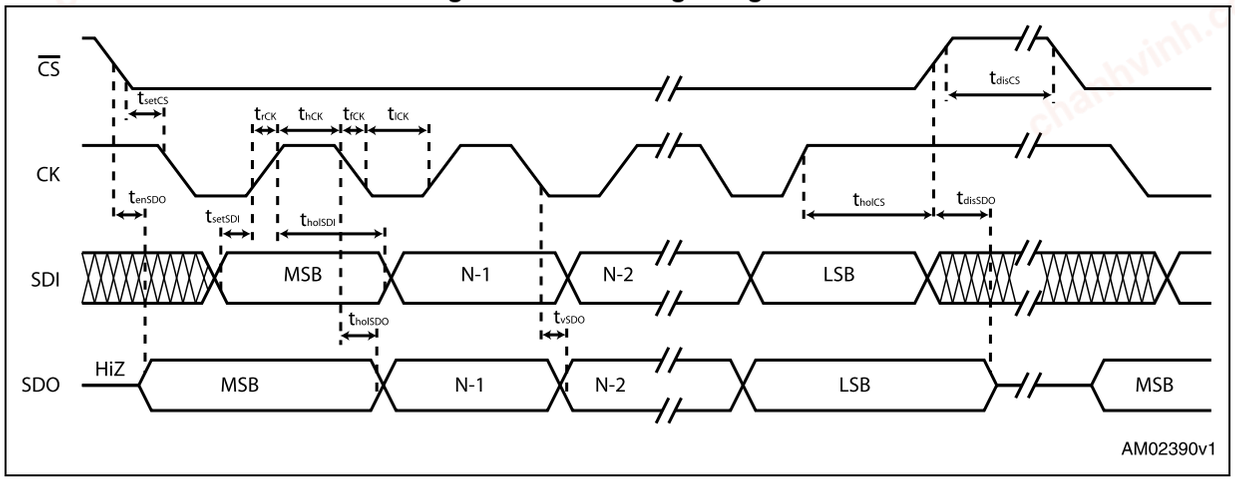
All output data bytes are shifted out of the device through the SDO output, most significant bit first. The SDO is latched on the falling edges of the CK. When a return value from the device is not available, an all zero byte is sent.

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 19* (see *Section 3 on page 11* for the respective electrical characteristics for values).

Multiple devices can be connected in a daisy chain configuration, as shown in *Figure 20*.

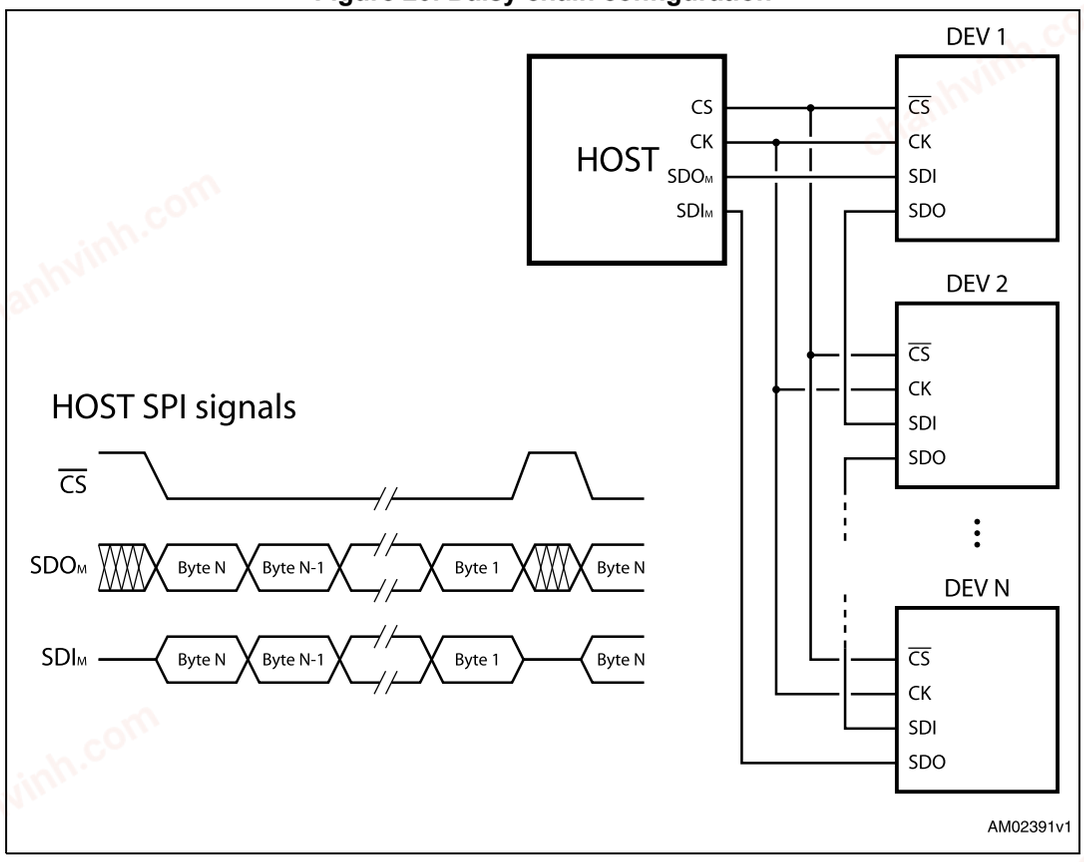
**Figure 19. SPI timings diagram**



chanhvinh.com



**Figure 20. Daisy chain configuration**



chanhvinh.com



# 9 Programming manual

## 9.1 Register and flag description

*Table 9* shows a map of the user registers available (detailed description in respective paragraphs from *Section 9.1.1 on page 41* to *Section 9.1.19 on page 52*):

**Table 9. Register map**

| Address [Hex] | Register name | Register function | Len. [bit] | Reset [Hex] | Reset value | Remarks(1) |
|---|---|---|---|---|---|---|
| h01 | ABS_POS | Current position | 22 | 000000 | 0 | R, WS |
| h02 | EL_POS | Electrical position | 9 | 000 | 0 | R, WS |
| h03 | MARK | Mark position | 22 | 000000 | 0 | 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$^2$ (2008 step/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 | 13 | 000 | 0 step/tick (0 step/s) | R, WS |
| h15 | FS_SPD | Full-step speed | 10 | 027 | 150.7e-6 step/tick (602.7 step/s) | R, WR |
| h09 | TVAL_HOLD | Holding current | 7 | 29 | 1.3125 A | R, WR |
| h0A | TVAL_RUN | Constant speed current | 7 | 29 | 1.3125 A | R, WR |
| h0B | TVAL_ACC | Acceleration starting current | 7 | 29 | 1.3125 A | R, WR |
| h0C | TVAL_DEC | Deceleration starting current | 7 | 29 | 1.3125 A | R, WR |
| h0D | RESERVED | Reserved address | 16 | | | |
| h0E | T_FAST | Fast decay/fall step time | 8 | 19 | 1µs / 5 µs | R, WH |
| h0F | TON_MIN | Minimum ON time | 7 | 29 | 20.5 µs | R, WH |
| h10 | TOFF_MIN | Minimum OFF time | 7 | 29 | 20.5 µs | R, WH |
| h11 | RESERVED | Reserved address | 8 | | | |
| h12 | ADC_OUT | ADC output | 5 | XX(2) | | R |
| h13 | OCD_TH | OCD threshold | 4 | 8 | 3.38 A | R, WR |
| h14 | RESERVED | Reserved address | 8 | | | |
| h16 | STEP_MODE | Step mode | 8 | 7(3) | 16 microsteps, no synch. | R, WH |
| h17 | ALARM_EN | Alarms enable | 8 | FF | All alarms enabled | R, WS |

chanhvinh.com



Table 9. Register map (continued)

| Address [Hex] | Register name | Register function | Len. [bit] | Reset [Hex] | Reset value | Remarks(1) |
|---|---|---|---|---|---|---|
| h18 | CONFIG | IC configuration | 16 | 2E88 | Internal oscillator, 2 MHz OSCOUT clock, supply voltage compensation disabled, overcurrent shutdown enabled, slew rate = 290 V/µs TSW = 40 µs | R, WH |
| h19 | STATUS | Status | 16 | XXXX(2) | High-impedance state, UVLO/reset flag set. | R |
| h1A | RESERVED | Reserved address | | | | |
| h1B | RESERVED | Reserved address | | | | |

1. R: Readable, WH: writable only when outputs are in high-impedance, WS: writable only when motor is stopped, WR: always writable.
2. According to startup conditions.
3. The bit 3 of the register must be set to one.

### 9.1.1 ABS_POS

The ABS_POS register contains the current motor absolute position in agreement to 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 NOTPERF_CMD flag to rise (see *Section 9.1.19 on page 52*).

### 9.1.2 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 10. 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 (see *Section 9.1.16 on page 47*); otherwise the resulting microstep sequence is incorrect.

Any attempt to write the register when the motor is running causes the command to be ignored and the NOTPERF_CMD flag to rise (see *Section 9.1.19 on page 52*).

chanhvinh.com



### 9.1.3 MARK

The MARK register contains an absolute position called MARK, in accordance with 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 $+2^{21}-1$.

### 9.1.4 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:

**Equation 1**

$$[step/s] = \frac{SPEED \cdot 2^{-28}}{tick}$$

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 NOTPERF_CMD flag to rise (see *Section 9.1.19 on page 52*).

### 9.1.5 ACC

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

In order to convert ACC value in step/s2 the following formula can be used:

**Equation 2**

$$[step/s] = \frac{ACC \cdot 2^{-40}}{tick^2}$$

where ACC 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 0xFFF value of the register is reserved and it should never be used.

Any attempt to write to the register when the motor is running causes the command to be ignored and the NOTPERF_CMD flag to rise (see *Section 9.1.19*).

### 9.1.6 DEC

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

chanhvinh.com



In order to convert the DEC value in step/s2 the following formula can be used:

**Equation 3**

$$[step/s] = \frac{DEC \cdot 2^{-40}}{tick^2}$$

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$^2$.

Any attempt to write the register when the motor is running causes the command to be ignored and the NOTPERF_CMD flag to rise (see *Section 9.1.19 on page 52*).

### 9.1.7 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:

**Equation 4**

$$[step/s] = \frac{MAXSPEED \cdot 2^{-18}}{tick}$$

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.

### 9.1.8 MIN_SPEED

The MIN_SPEED register contains the following parameters:

**Table 11. MIN_SPEED register**

| Bit12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 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:

**Equation 5**

$$[step/s] = \frac{MINSPEED \cdot 2^{-24}}{tick}$$

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

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

Any attempt to write the register when the motor is running causes the NOTPERF_CMD flag to rise.

chanhvinh.com



### 9.1.9 FS_SPD

The FS_SPD register contains the threshold speed. When the actual speed exceeds this value 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.

**Equation 6**

$$[step/s] = \frac{(FSSPD + 0.5) \cdot 2^{-18}}{tick}$$

If the FS_SPD value is set to h3FF (max.) the system always works in microstepping mode (SPEED must go beyond the threshold to switch to full-step mode). Setting FS_SPD to zero does not have the same effect as setting 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 15625 step/s with a resolution of 15.25 step/s.

### 9.1.10 TVAL_HOLD, TVAL_RUN, TVAL_ACC and TVAL_DEC

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

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

The TVAL_ACC register contains the current value that is assigned to the torque regulation DAC during acceleration.

The TVAL_DEC register contains the current value that is assigned to the torque regulation DAC during deceleration.

The available range is from 31.25 mA to 4 A with a resolution of 31.25 mA, as shown in *Table 12*.

**Table 12. Torque regulation by TVAL_HOLD, TVAL_ACC, TVAL_DEC and TVAL_RUN registers**

| TVAL_X [6 … 0] | | | | | | | Output current amplitude |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 31.25 mA |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 62.5 mA |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 3.969 A |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 A |

chanhvinh.com



### 9.1.11 T_FAST

The T_FAST register contains the maximum fast decay time (TOFF_FAST) and the maximum fall step time (FALL_STEP) used by the current control system (see *Section 7.2 on page 34* and *7.3 on page 36* for details):

**Table 13. 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 14. 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 | 30 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 NOTPERF_CMD to rise (see *Section 9.1.19 on page 52*).

### 9.1.12 TON_MIN

The TON_MIN register contains the minimum ON time value used by the current control system (see *Section 7.2 on page 34*).

The available range for both parameters is from 0.5 µs to 64 µs.

**Table 15. 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 the NOTPERF_CMD to rise (see *Section 9.1.19*).

chanhvinh.com



### 9.1.13 TOFF_MIN

The TOFF_MIN register contains the minimum OFF time value used by the current control system (see *Section 7.1 on page 33* for details).

The available range for both parameters is from 0.5 µs to 64 µs.

**Table 16. 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 NOTPERF_CMD to rise (see *Section 9.1.19 on page 52*).

### 9.1.14 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 the NOTPERF_CMD flag to rise (see *Section 9.1.19*).

**Table 17. ADC_OUT value and torque regulation feature**

| VADCIN/ VREG | ADC_OUT [4.0] | | | | | Output current amplitude |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 125 mA |
| 1/32 | 0 | 0 | 0 | 0 | 1 | 250 mA |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 30/32 | 1 | 1 | 1 | 1 | 0 | 3.875 A |
| 31/32 | 1 | 1 | 1 | 1 | 1 | 4 A |

chanhvinh.com



### 9.1.15 OCD_TH

The OCD_TH register contains the overcurrent threshold value (see *Section 6.9 on page 28* for details). The available range is from 375 mA to 6 A, steps of 375 mA, as shown in *Table 18*.

**Table 18. Overcurrent detection threshold**

| OCD_TH [3 … 0] | | | | Overcurrent detection threshold |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 375 mA |
| 0 | 0 | 0 | 1 | 750 mA |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1 | 1 | 1 | 0 | 5.625 A |
| 1 | 1 | 1 | 1 | 6 A |

### 9.1.16 STEP_MODE

The STEP_MODE register has the following structure:

**Table 19. STEP_MODE register**

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

1. When the register is written this bit should be set to 1.

*When the STEP_MODE register is written, the bit #3 is to be set to 1, otherwise anomalous behaviors could occur.*

The STEP_SEL parameter selects one of five possible stepping modes:

**Table 20. Step mode selection**

| STEP_SEL[2 … 0] | | | Step mode |
|---|---|---|---|
| 0 | 0 | 0 | Full-step |
| 0 | 0 | 1 | Half-step |
| 0 | 1 | 0 | 1/4 microstep |
| 0 | 1 | 1 | 1/8 microstep |
| 1 | X | X | 1/16 microstep |

Every time the step mode is changed, the electrical position (i.e. the point of microstepping sine wave that is generated) is reset to the first microstep.

**Warning: Every time STEP_SEL is changed the value in the ABS_POS register looses meaning and should be reset.**

chanhvinh.com



Any attempt to write the register when the motor is running causes the command to be ignored and the NOTPERF_CMD flag to rise (see *Section 9.1.19*).

When when SYNC_EN bit is set low, $\overline{\text{BUSY}}$/SYNC output is forced low during the commands execution, otherwise, when the SYNC_EN bit is set high, the $\overline{\text{BUSY}}$/SYNC output provides a clock signal according to the SYNC_SEL parameter.

**Table 21. SYNC output frequency**

| | | STEP_SEL ($f_{FS}$ is the full-step frequency) | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | **000** | **001** | **010** | **011** | **100** | **101** | **110** | **111** |
| **SYNC_SEL** | **000** | $f_{FS}/2$ | $f_{FS}/2$ | $f_{FS}/2$ | $f_{FS}/2$ | $f_{FS}/2$ | $f_{FS}/2$ | $f_{FS}/2$ | $f_{FS}/2$ |
| | **001** | NA | $f_{FS}$ | $f_{FS}$ | $f_{FS}$ | $f_{FS}$ | $f_{FS}$ | $f_{FS}$ | $f_{FS}$ |
| | **010** | NA | NA | $2 \cdot f_{FS}$ | $2 \cdot f_{FS}$ | $2 \cdot f_{FS}$ | $2 \cdot f_{FS}$ | $2 \cdot f_{FS}$ | $2 \cdot f_{FS}$ |
| | **011** | NA | NA | NA | $4 \cdot f_{FS}$ | $4 \cdot f_{FS}$ | $4 \cdot f_{FS}$ | $4 \cdot f_{FS}$ | $4 \cdot f_{FS}$ |
| | **100** | NA | NA | NA | NA | $8 \cdot f_{FS}$ | $8 \cdot f_{FS}$ | $8 \cdot f_{FS}$ | $8 \cdot f_{FS}$ |
| | **101** | NA | NA | NA | NA | NA | NA | NA | NA |
| | **110** | NA | NA | NA | NA | NA | NA | NA | NA |
| | **111** | NA | NA | NA | NA | NA | NA | NA | NA |

The synchronization signal is obtained starting from the electrical position information (EL_POS register) according to *Table 22*:

**Table 22. SYNC signal source**

| SYNC_SEL[2 … 0] | | | Source |
|---|---|---|---|
| 0 | 0 | 0 | EL_POS[7] |
| 0 | 0 | 1 | EL_POS[6] |
| 0 | 1 | 0 | EL_POS[5] |
| 0 | 1 | 1 | EL_POS[4] |
| 1 | 0 | 0 | EL_POS[3] |
| 1 | 0 | 1 | UNUSED (1) |
| 1 | 1 | 0 | UNUSED (1) |
| 1 | 1 | 1 | UNUSED (1) |

1. When this value is selected the BUSY output is forced low.

chanhvinh.com



### 9.1.17 ALARM_EN

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

**Table 23. ALARM_EN register**

| ALARM_EN bit | Alarm condition |
|---|---|
| 0 (LSB) | Overcurrent |
| 1 | Thermal shutdown |
| 2 | Thermal warning |
| 3 | Undervoltage |
| 4 | UNUSED |
| 5 | UNUSED |
| 6 | Switch turn-on event |
| 7 (MSB) | Wrong or non-performable command |

### 9.1.18 CONFIG

The CONFIG register has the following structure:

**Table 24. CONFIG register**

| Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 |
|---|---|---|---|---|---|---|---|
| PRED_EN | TSW | | | | | POW_SR | |
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| OC_SD | RESERVED | EN_TQREG | SW_MODE | EXT_CLK | OSC_SEL | | |

chanhvinh.com



The OSC_SEL and EXT_CLK bits set the system clock source:

**Table 25. 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 | Unused | Supplies an 8-MHz clock |
| 1 | 0 | 1 | | 1 | 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 | Ext clock source: 8 MHz (Crystal/resonator driver disabled) | Clock source | Supplies inverted OSCIN signal |
| 1 | 1 | 0 | 1 | Ext clock source: 16 MHz (Crystal/resonator driver disabled) | Clock source | Supplies inverted OSCIN signal |
| 1 | 1 | 1 | 0 | Ext clock source: 24 MHz (Crystal/resonator driver disabled) | Clock source | Supplies inverted OSCIN signal |
| 1 | 1 | 1 | 1 | Ext 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 26. 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:

chanhvinh.com



**Table 27. Overcurrent event**

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

The POW_SR bits set the slew rate value of the power bridge output:

**Table 28. Programmable power bridge output slew rate values**

| POW_SR [1 … 0] | | Output slew rate (1) [V/µs][1] |
|---|---|---|
| 0 | 0 | 320 |
| 0 | 1 | 75 |
| 1 | 0 | 110 |
| 1 | 1 | 270 |

1. See $S_{Rout_r}$ and $S_{Rout_f}$ parameters in *Table 5 on page 11* for details.

The TQREG bit sets if the torque regulation (see *Section 7.4 on page 37*) is performed through ADCIN voltage (external) or the TVAL_HOLD, TVAL_ACC, TVAL_DEC and TVAL_RUN registers (internal):

**Table 29. External torque regulation enable**

| TQREG | External torque regulation enable |
|---|---|
| 0 | Internal registers |
| 1 | ADC input |

The TSW parameter is used by the current control system and it sets the target switching period.

**Table 30. 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 motor is running causes the command to be ignored and the NOTPERF_CMD flag to rise (see *Section 9.1.19*).

chanhvinh.com



### 9.1.19 STATUS

**Table 31. STATUS register**

| Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 |
|---|---|---|---|---|---|---|---|
| SCK_MOD | X | X | OCD | TH_SD | TH_WRN | UVLO | WRONG_CMD |
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| NOTPERF_CMD | 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. Any motion command causes 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 event (power-up included). The TH_WRN, TH_SD, OCD flags are active low and indicate respectively thermal warning, thermal shutdown and overcurrent detection events.

The NOTPERF_CMD and WRONG_CMD flags are active high and indicate, respectively, that the command received by SPI can't 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).

The UVLO, TH_WRN, TH_SD, OCD, NOTPERF_CMD, WRONG_CMD and SW_EVN flags 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 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 SCK_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 32. STATUS register DIR bit**

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

chanhvinh.com



MOT_STATUS indicates the current motor status:

**Table 33. 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 NOTPERF_CMD to rise.

chanhvinh.com



## 9.2 Application commands

A summary of commands is given in *Table 34*.

**Table 34. 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 the PARAM register |
| GetParam (PARAM) | 001 | [PARAM] | | | | Returns the stored value in the 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 into step-clock mode and imposes DIR direction |
| Move (DIR,N_STEP) | 010 | 0 | 0 | 00 | DIR | Makes N_STEP (micro) steps in DIR direction (non-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 |
| ReleaseSW (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 (set 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 |

chanhvinh.com

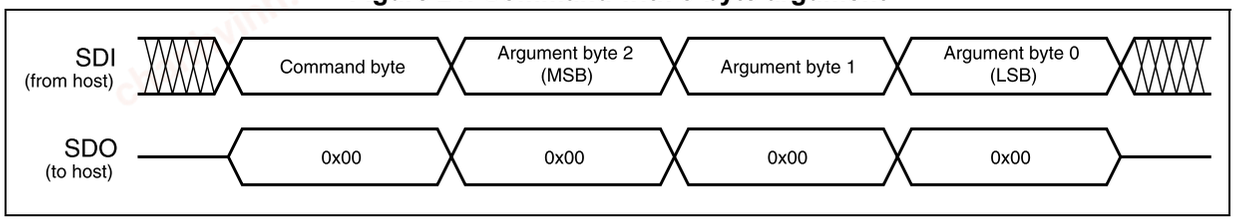


### 9.2.1 Command management

The host microcontroller can control motor motion and configure the L6472 device through a complete set of commands.

All commands are composed by a single byte. After the command byte, some argument bytes should be needed (see *Figure 21*). Argument length can vary from 1 to 3 bytes.

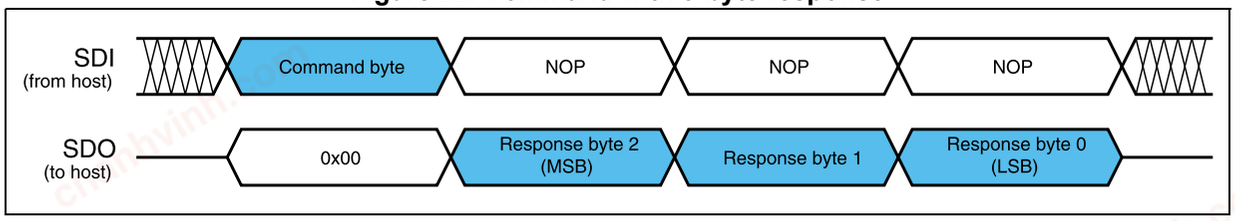
**Figure 21. Command with 3-byte argument**




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 22*).

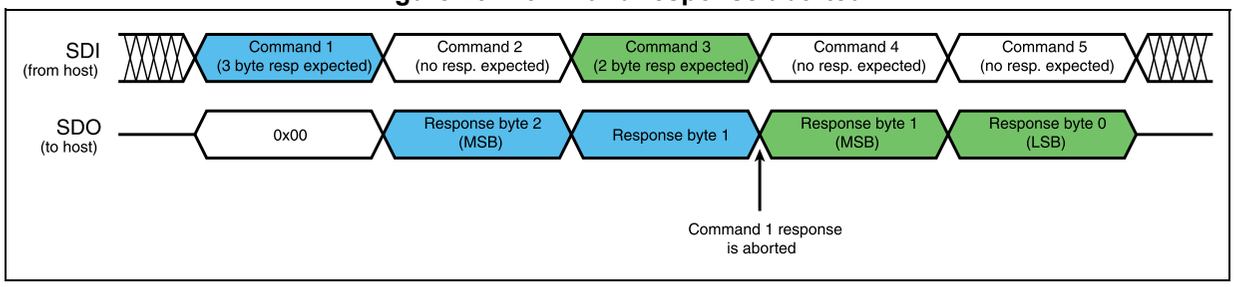
Response length can vary from 1 to 3 bytes.

**Figure 22. 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 23*).

**Figure 23. Command response aborted**




When a byte that does not correspond to a command is sent to the IC, it is ignored and the WRONG_CMD flag in the STATUS register is raised (see *Section 9.1.19*).

chanhvinh.com



### 9.2.2 Nop

**Table 35. 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.

### 9.2.3 SetParam (PARAM, VALUE)

**Table 36. 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) | | | | | | | | |
| VALUE Byte 1 (if needed) | | | | | | | | |
| VALUE Byte 0 | | | | | | | | |

The SetParam command sets the PARAM register value equal to VALUE; PARAM is the respective register address listed in *Table 9 on page 40*.

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 9*).

Some registers cannot be written (see *Table 9*); any attempt to write one of these registers causes the command to be ignored and the WRONG_CMD flag to rise at the end of the command byte as if an unknown command code was sent (see *Section 9.1.18 on page 49*).

Some registers can only be written in particular conditions (see *Table 9*); any attempt to write one of these registers when the conditions are not satisfied causes the command to be ignored and the NOTPERF_CMD flag to rise at the end of last argument byte (see *Section 9.1.19 on page 52*).

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

### 9.2.4 GetParam (PARAM)

**Table 37. 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 |

chanhvinh.com



This command reads the current PARAM register value; PARAM is the respective register address listed in *Table 9 on page 40*.

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 9*).

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

All registers can be read anytime.

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

### 9.2.5 Run (DIR, SPD)

**Table 38. 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) which is the same format as the SPEED register (see *Section 9.1.4 on page 42*).

*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.

### 9.2.6 StepClock (DIR)

**Table 39. 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 6.7.5 on page 26*) 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 6.7.5* and *Section 9.1.18 on page 49*).

The device exits from step-clock mode when a constant speed, absolute positioning or motion command is sent through SPI. Motion direction is imposed by the respective

chanhvinh.com



StepClock command argument and can by changed by a new StepClock command without exiting step-clock mode.

Events that cause bridges to be forced into high-impedance state (overtemperature, overcurrent, etc.) do not cause the device to leave step-clock mode. 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 NOTPERF_CMD flag to rise (see *Section 9.1.19 on page 52*).

### 9.2.7 Move (DIR, N_STEP)

**Table 40. 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 NOTPERF_CMD flag to rise (see *Section 9.1.19*).

### 9.2.8 GoTo (ABS_POS)

**Table 41. GoTo 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 the ABS_POS absolute position through the shortest path. The ABS_POS 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.).

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

chanhvinh.com



This command can only be given 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 NOTPERF_CMD flag to rise (see *Section 9.1.19 on page 52*).

### 9.2.9 GoTo_DIR (DIR, ABS_POS)

**Table 42. 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 the ABS_POS absolute position imposing a forward (DIR = '1') or a reverse (DIR = '0') rotation. The ABS_POS 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.).

The GoTo_DIR command keeps the BUSY flag low until the target speed is reached. This command can only be given 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 NOTPERF_CMD flag to rise (see *Section 9.1.19*).

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

**Table 43. 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 6.13 on page 30*), 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) which is the same format as the SPEED register (see *Section 9.1.4 on page 42*).

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.

chanhvinh.com



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 6.13 on page 30* and *9.1.18 on page 49*).

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.

### 9.2.11 ReleaseSW (ACT, DIR)

**Table 44. 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.

### 9.2.12 GoHome

**Table 45. 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 9.2.9*).

The GoHome command keeps the BUSY flag low until the home position is reached. This command can only be given 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 the NOTPERF_CMD to rise (see *Section 9.1.19 on page 52*).

chanhvinh.com



### 9.2.13 GoMark

Table 46. 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 only be given 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 NOTPERF_CMD flag to rise (see *Section 9.1.19 on page 52*).

### 9.2.14 ResetPos

Table 47. 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 HOME position (see *Section 6.5 on page 23*).

### 9.2.15 ResetDevice

Table 48. 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 6.1 on page 20*).

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

chanhvinh.com



### 9.2.16 SoftStop

**Table 49. 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 9.1.6 on page 42*).

When the motor is in high-impedance state, a SoftStop command forces the bridges to exit from 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.

### 9.2.17 HardStop

**Table 50. 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 high-impedance state, a HardStop command forces the bridges to exit from 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.

### 9.2.18 SoftHiZ

**Table 51. 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; the deceleration value used is the one stored in the DEC register (see *Section 9.1.6*). When bridges are disabled the HiZ flag is raised.

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

chanhvinh.com



### 9.2.19 HardHiZ

**Table 52. 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 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.

### 9.2.20 GetStatus

**Table 53. 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.

chanhvinh.com



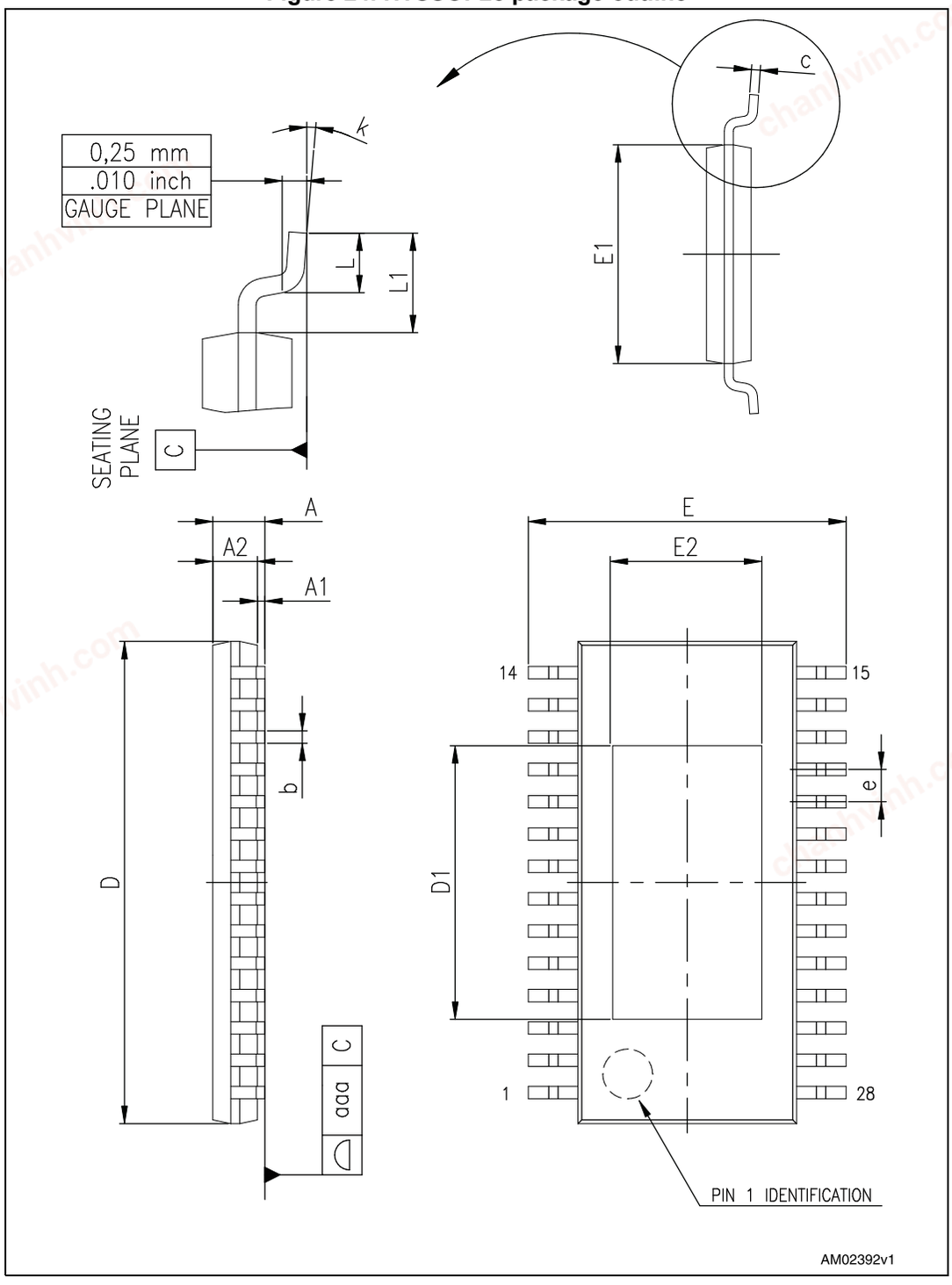
# 10 Package information

In order 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.

chanhvinh.com



**Figure 24. HTSSOP28 package outline**



chanhvinh.com



Table 54. HTSSOP28 package mechanical data

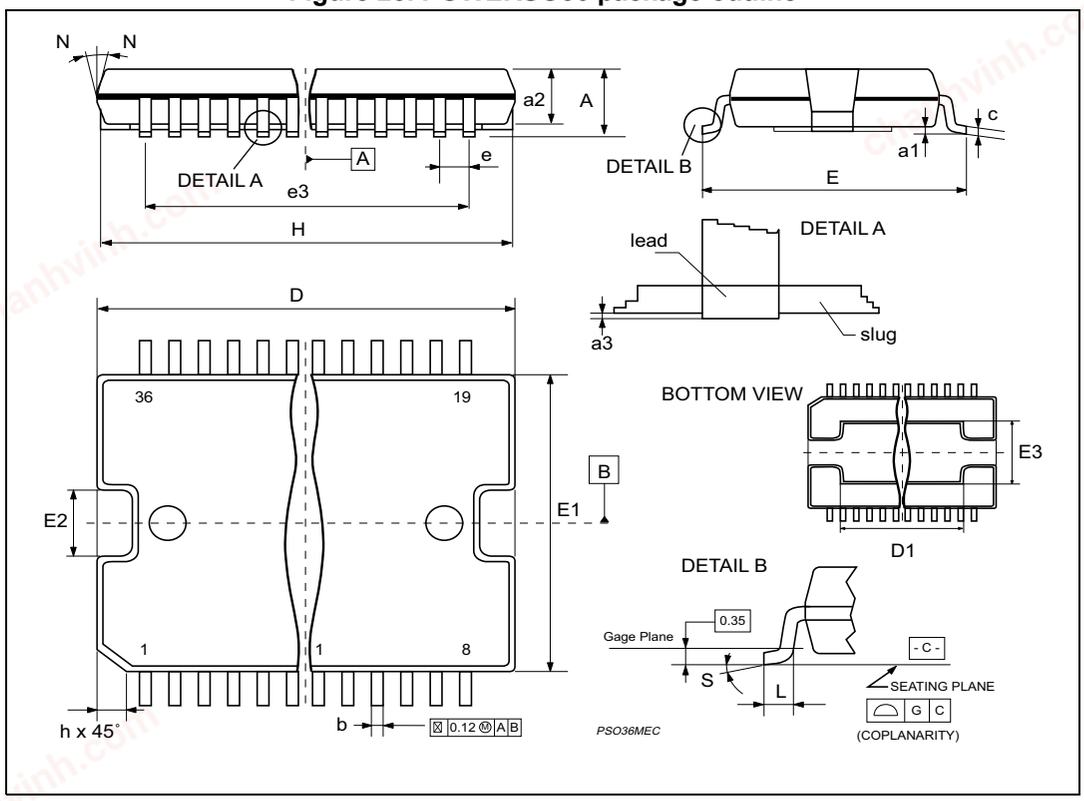
| Symbol | Dimensions (mm) | | |
|---|---|---|---|
| | Min. | Typ. | Max. |
| A | | | 1.2 |
| A1 | | | 0.15 |
| A2 | 0.8 | 1.0 | 1.05 |
| b | 0.19 | | 0.3 |
| c | 0.09 | | 0.2 |
| D(1) | 9.6 | 9.7 | 9.8 |
| D1 | | 5.5 | |
| E | 6.2 | 6.4 | 6.6 |
| E1(2) | 4.3 | 4.4 | 4.5 |
| E2 | | 2.8 | |
| e | | 0.65 | |
| L | 0.45 | 0.6 | 0.75 |
| L1 | | 1.0 | |
| K | 0° | | 8° |
| aaa | | 0.1 | |
| | | | |

1. Dimension “D” does not include mold flash, protrusions or gate burrs. Mold flash, protrusions or gate burrs do not exceed 0.15 mm per side.
2. Dimension “E1” does not include interlead flash or protrusions. Interlead flash or protrusions do not exceed 0.25 mm per side.

chanhvinh.com



**Figure 25. POWERSO36 package outline**



chanhvinh.com



Table 55. POWERSO36 package mechanical data

| Symbol | Dimensions (mm) | | |
|---|---|---|---|
| | Min. | Typ. | Max. |
| A | | | 3.60 |
| a1 | 0.10 | | 0.30 |
| a2 | | | 3.30 |
| a3 | 0 | | 0.10 |
| b | 0.22 | | 0.38 |
| c | 0.23 | | 0.32 |
| D(1) | 15.80 | | 16.00 |
| D1 | 9.40 | | 9.80 |
| E | 13.90 | | 14.50 |
| E1(1) | 10.90 | | 11.10 |
| E2 | | | 2.90 |
| E3 | 5.8 | | 6.2 |
| e | | 0.65 | |
| e3 | | 11.05 | |
| G | 0 | | 0.10 |
| H | 15.50 | | 15.90 |
| h | | | 1.10 |
| L | 0.80 | | 1.10 |
| N | | | 10° |
| S | 0° | | 8° |

1. Dimension “D/E1” does not include mold flash, protrusions or gate burrs. Mold flash, protrusions or gate burrs do not exceed 0.15 mm per side.

chanhvinh.com



# 11 Revision history

Table 56. Document revision history

| Date | Revision | Changes |
|---|---|---|
| 24-Jan-2012 | 1 | Initial release. |
| 09-Jan-2013 | 2 | Changed the title.<br>Changed $T_{OP}$ value in *Table 2*.<br>Removed $T_j$ parameter in *Table 3*.<br>Added footnote to *Table 9*.<br>Changed fast decay time in *Table 14*.<br>Changed output slew rate values in *Table 28*<br>Updated HTSSOP28 package mechanical data. |
| 16-Dec-2013 | 3 | Updated *Section 9.1.11* (updated available range for both parameters).<br>Updated *Section 10* (updated titles, reversed order of *Figure 24* and *Table 54* and *Figure 25* and *Table 55*).<br>Minor modifications throughout document. |
| 19-May-2014 | 4 | Updated *Section 6.4 on page 21* (replaced “the first microstep” by “zero”).<br>Removed Section “Infinite acceleration/deceleration mode” from page 23.<br>Updated *Section 9.1.5 on page 42* (replaced “When the ACC value is set to 0xFFF the device works in infinite acceleration mode.” by “The 0xFFF value of the register is reserved and it should never be used.”).<br>Updated *Section 9.1.6 on page 42* (removed “When the device is working in infinite acceleration mode this value is ignored.”).<br>Updated title of *Table 33 on page 53* (replaced “MOT_STATE” by “MOT_STATUS”).<br>Updated *Table 55 on page 68* (added note *1* below *Table 55*). |

chanhvinh.com



**Please Read Carefully:**

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

**UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.**

**ST PRODUCTS ARE NOT DESIGNED OR AUTHORIZED FOR USE IN: (A) SAFETY CRITICAL APPLICATIONS SUCH AS LIFE SUPPORTING, ACTIVE IMPLANTED DEVICES OR SYSTEMS WITH PRODUCT FUNCTIONAL SAFETY REQUIREMENTS; (B) AERONAUTIC APPLICATIONS; (C) AUTOMOTIVE APPLICATIONS OR ENVIRONMENTS, AND/OR (D) AEROSPACE APPLICATIONS OR ENVIRONMENTS. WHERE ST PRODUCTS ARE NOT DESIGNED FOR SUCH USE, THE PURCHASER SHALL USE PRODUCTS AT PURCHASER’S SOLE RISK, EVEN IF ST HAS BEEN INFORMED IN WRITING OF SUCH USAGE, UNLESS A PRODUCT IS EXPRESSLY DESIGNATED BY ST AS BEING INTENDED FOR “AUTOMOTIVE, AUTOMOTIVE SAFETY OR MEDICAL” INDUSTRY DOMAINS ACCORDING TO ST PRODUCT DESIGN SPECIFICATIONS. PRODUCTS FORMALLY ESCC, QML OR JAN QUALIFIED ARE DEEMED SUITABLE FOR USE IN AEROSPACE BY THE CORRESPONDING GOVERNMENTAL AGENCY.**

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.
Information in this document supersedes and replaces all information previously supplied.
The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

© 2014 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

**www.st.com**