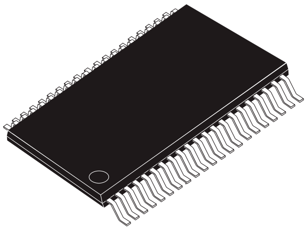
L6480

Microstepping motor controller with motion engine and SPI

Datasheet - production data



HTSSOP38

Features

- Operating voltage: 7.5 V - 85 V
- Dual full bridge gate driver for N-channel MOSFETs
- Fully programmable gate driving
- Embedded Miller clamp function
- Programmable speed profile
- Up to 1/128 microstepping
- Sensorless stall detection
- Integrated voltage regulators
- SPI interface
- Low quiescent standby currents
- Programmable non dissipative overcurrent protection
- Overtemperature protection

Applications

- Bipolar stepper motor

Description

The L6480 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 full bridge gate driver for N-channel MOSFET power stages with embedded non dissipative overcurrent protection. Thanks to a unique voltage mode driving mode which compensates for BEMF, bus voltage and motor winding variations, the microstepping of a true 1/128-step resolution is achieved. 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 protection trip point, deadtime, PWM frequency, etc.) are sent through a standard 5-Mbit/s SPI. A very rich set of protections (thermal, low bus voltage, overcurrent and motor stall) makes the L6480 device “bullet proof”, as required by the most demanding motor control applications.

Table 1. Device summary

Order codes	Package	Packaging
L6480H	HTSSOP38	Tube
L6480HTR	HTSSOP38	Tape and reel

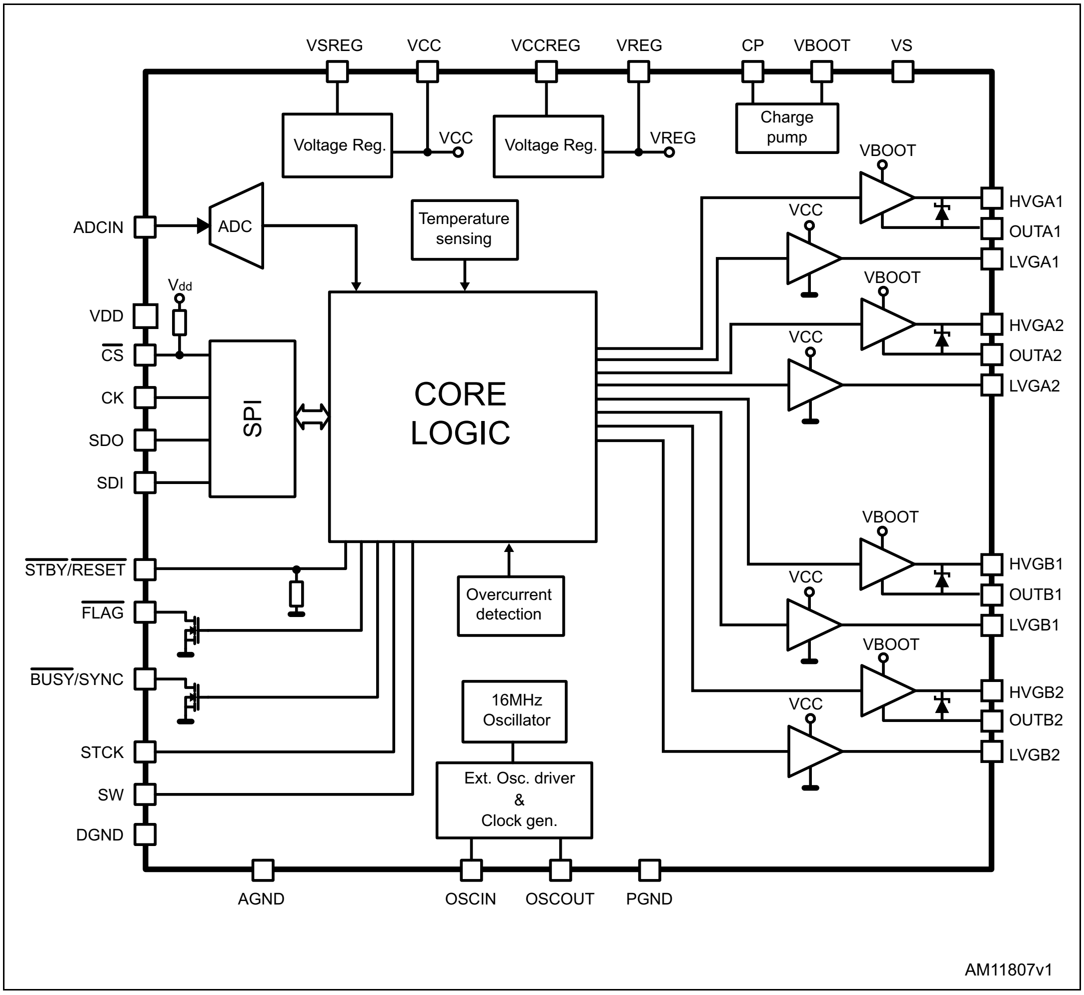
Contents

List of tables

List of figures

1 Block diagram

Figure 1. Block diagram



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_{REG}	Logic supply voltage		3.6	
V_S	Motor supply voltage		95	V
V_{CC}	Low-side gate driver supply voltage		18	V
V_{BOOT}	Boot voltage		100	V
ΔV_{BOOT}	High-side gate driver supply voltage (V_{BOOT} - V_S)		0 to 20	V
V_{SREG}	Internal V_{CC} regulator supply voltage		95	V
V_{CCREG}	Internal V_{REG} regulator supply voltage		18	V
V_{OUT1A} V_{OUT2A} V_{OUT1B} V_{OUT2B}	Full bridge output voltage	DC	-5 to V_{BOOT}	V
		AC	-15 to V_{BOOT}	
SR_{out}	Full bridge outputs slew rate (10% - 90%)		10	V/ns
V_{HVG1A} V_{HVG2A} V_{HVG1B} V_{HVG2B}	High-side output driver voltage		V_{OUT} to V_{BOOT}	V
ΔV_{HVG1A} ΔV_{HVG2A} ΔV_{HVG1B} ΔV_{HVG2B}	High-side output driver to respective bridge output voltage(V_{HVG} - V_{OUT})		15	V
V_{LVG1A} V_{LVG2A} V_{LVG1B} V_{LVG2B}	Low-side output driver voltage		V_{CC} + 0.3	V
$I_{GATE-CLAMP}$	High-side gate voltage clamp current capability		100	mA
V_{ADCIN}	Integrated ADC input voltage range (ADCIN pin)		-0.3 to 3.6	V
V_{out_diff}	Differential voltage between VBOOT, VS, OUT1A, OUT2A, PGND and VBOOT, VS, OUT1B, OUT2B, PGND pins		100	V
V_{in}	Logic inputs voltage range		-0.3 to 5.5	V
T_{OP} T_s	Storage and operating junction		-40 to 150	°C
P_{tot}	Total power dissipation (T_{amb} = 25 °C)	[1]	4	W

[1] HTSSOP38 mounted on a four-layer FR4 PCB with a dissipating copper surface of about 30 cm^2.

2.2 Recommended operating conditions

Table 3. Recommended operating conditions

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

2.3 Thermal data

Table 4. Thermal data

Symbol	Parameter	Package	Typ.	Unit
R_{thj-a}	Thermal resistance junction to ambient	HTSSOP38(1)	31	°C/W

1. HTSSOP38 mounted on a four-layer FR4 PCB with a dissipating copper surface of about 30 cm^2.

3 Electrical characteristics

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

Table 5. Electrical characteristics

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
General						
V_{CCthOn}	V_{CC} UVLO turn-on threshold	UVLO_VAL set high(1)	9.9	10.4	10.9	V
		UVLO_VAL set low(1)	6.5	6.9	7.3	V
$V_{CCthOff}$	V_{CC} UVLO turn-off threshold	UVLO_VAL set high(1)	9.5	10	10.5	V
		UVLO_VAL set low(1)	5.9	6.3	6.7	V
$\Delta V_{BOOTthOn}$	V_{BOOT} - V_S UVLO turn-on threshold	UVLO_VAL set high(1)	8.6	9.2	9.8	V
		UVLO_VAL set low(1)	5.7	6	6.3	V
$\Delta V_{BOOTthOff}$	V_{BOOT} - V_S UVLO turn-off threshold	UVLO_VAL set high(1)	8.2	8.8	9.5	V
		UVLO_VAL set low(1)	5.3	5.5	5.8	V
$V_{REGthOn}$	V_{REG} turn-on threshold	(1)	2.8	3	3.18	V
$V_{REGthOff}$	V_{REG} turn-off threshold	(1)	2.2	2.4	2.5	V
I_{VREGqu}	Undervoltage V_{REG} quiescent supply current	$V_{CCREG} = V_{REG} < 2.2$ V(1)		40		µA
I_{VREGq}	Quiescent V_{REG} supply current	$V_{CCREG} = V_{REG} = 3.3$ V, internal oscillator selected(1)		3.8		mA
I_{VSREGq}	Quiescent V_{SREG} supply current	$V_{CCREG} = V_{CC}$ = 15V		6.5		mA
Thermal protection						
$T_{j(WRN)Set}$	Thermal warning temperature			135		°C
$T_{j(WRN)Rec}$	Thermal warning recovery temperature			125		°C
$T_{j(OFF)Set}$	Thermal bridge shutdown temperature			155		°C
$T_{j(OFF)Rec}$	Thermal bridge shutdown recovery temperature			145		°C
$T_{j(SD)Set}$	Thermal device shutdown temperature			170		°C
$T_{j(SD)Rec}$	Thermal device shutdown recovery temperature			130		°C
Charge pump						
V_{pump}	Voltage swing for charge pump oscillator			V_{CC}		V
$f_{pump,min}$	Minimum charge pump oscillator frequency(2)			660		kHz
$f_{pump,max}$	Maximum charge pump oscillator frequency(2)			800		kHz
R_{pumpHS}	Charge pump high-side $R_{DS(on)}$ resistance			10		Ω

Table 5. Electrical characteristics (continued)

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
R_{pumpLS}	Charge pump low-side $R_{DS(ON)}$ resistance			10		Ω
I_{boot}	Average boot current			2.6		mA
Gate driver outputs						
$I_{GATE,Sink}$	Programmable high-side and low-side gate sink current	V_S = 38 V V_{HVGX} - V_{OUTX} > 3 V V_{LVGX} > 3 V	2.4	4	5.6	mA
			5.4	8	10.6	
			11.3	16	20.7	
			17.3	24	30.7	
			23.2	32	40.8	
			50.2	64	77.8	
			81	96	113	
$I_{GATE,Source}$	Programmable high-side and low-side gate source current	V_S = 38 V V_{BOOTX} - V_{HVGX} > 3.5 V V_{CC}-V_{LVGX} > 3.5 V	2.8	4	5.2	mA
			5.8	8	10.2	
			12	16	20	
			18	24	30	
			24	32	40	
			51	64	77	
			82	96	112	
I_{OB}	High-side and low-side turn-off overboost gate current		85	103	117	mA
$R_{CLAMP(LS)}$	Low-side gate driver Miller clamp resistance			6.5	10	Ω
$R_{CLAMP(HS)}$	High-side gate driver Miller clamp resistance			3	10	Ω
$V_{GATE-CLAMP}$	High-side gate voltage clamp	$I_{GATE-CLAMP=10\ mA}$		16.7		v
t_{cc}	Programmable constant gate current time(2)	TCC = ’00000’		125		ns
		TCC = 11111		3750		
t_{OB}	Programmable. Turn-off overboost; gate current time(2)	TBOOST = ’001’, internal oscillator		62.5		ns
		TBOOST = ’111’		1000		
I_{DSS}	Leakage current	OUT = V_S			0.1	mA
		OUT = GND	-0.1			mA
t_r	Rise time (10% - 90%)	I_{GATE} = 96 mA V_{CC} = 15 V C_{GATE} = 15 nF		2.5		µs
t_f	Fall time (90%-10%)	I_{GATE} = 96 mA V_{CC} = 15 V C_{GATE} = 15 nF		2.5		µs

Table 5. Electrical characteristics (continued)

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
SRgate	Gate driver output slew rate	I_{GATE}= 96 mA V_{CC} = 15 V C_{GATE} = 15 nF		6		V/µs
Deadtime and blanking						
t_{DT}	Programmable deadtime[2]	TDT = '00000'		125		ns
		TDT = ’11111’		4000		
t_{blank}	Programmable blanking time[2]	TBLANK = '000'		125		ns
		TBLANK = ’111’		1000		
Logic						
V_{IL}	Low level logic input voltage				0.8	V
V_{IH}	High level logic input voltage		2			V
I_{IH}	High level logic input current	V_{IN} = 5 V, VDDIO = 5 V			1	µA
I_{IL}	Low level logic input current	V_{IN} = 0 V, VDDIO = 5 V	-1			µA
V_{OL}	Low level logic output voltage(3)	V_{DD} = 3.3 V, I_{OL} = 4 mA			0.3	V
		V_{DD} = 5 V, I_{OL} = 4 mA			0.3	
V_{OH}	High level logic output voltage	V_{DD} = 3.3 V, I_{OH} = 4 mA	2.4			V
		V_{DD} = 5 V, I_{OH} = 4 mA	4.7			
R_{PUCS}	CS pull-up resistor			430		kΩ
R_{PDRST}	STBY/RESET pull-down resistor			450		
R_{PUSW}	SW pull-up resistor			80		
$t_{high,STCK}$	Step-clock input high time		300			ns
$t_{low,STCK}$	Step-clock input low time		300			ns
Internal oscillator and external oscillator driver						
$f_{osc,int}$	Internal oscillator frequency	T_j = 25 °C	-5%	16	+5%	MHz
$f_{osc,ext}$	Programmable external oscillator frequency		8		32	MHz
$V_{OSCOUTH}$	OSCOUT clock source high level voltage	Internal oscillator	2.4			V
$V_{OSCOUTL}$	OSCOUT clock source low level voltage	Internal oscillator			0.3	V
$t_{rOSCOUT}$ $t_{fOSCOUT}$	OSCOUT clock source rise and fall time	Internal oscillator			10	ns
t_{high}	OSCOUT clock source high time	Internal oscillator		31.25		ns
t_{extosc}	Internal to external oscillator switching delay			3		ms
t_{intosc}	External to internal oscillator switching delay				100	µs

Table 5. Electrical characteristics (continued)

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
SPI						
$f_{CK,MAX}$	Maximum SPI clock frequency(4)		5			MHz
t_{rCK} t_{fCK}	SPI clock rise and fall time(4)				1	µs
t_{hCK} t_{lCK}	SPI clock high and low time(4)		90			ns
t_{setCS}	Chip select setup time(4)		30			ns
t_{holCS}	Chip select hold time (4)		30			ns
t_{disCS}	Deselect time(4)		625			ns
t_{setSDI}	Data input setup time(4)		20			ns
t_{holSDI}	Data input hold time(4)		30			ns
t_{enSDO}	Data output enable time(4)				95	ns
t_{disSDO}	Data output disable time(4)				95	ns
t_{vSDO}	Data output valid time (4)				35	ns
t_{holSDO}	Data output hold time(4)		0			ns
PWM modulators						
f_{PWM}	Programmable PWM frequency(2)	f_{osc} = 32 MHz F_PWM_INT = ’11X’ F_PWM_DEC = ’000’		5.6		kHz
		f_{osc} = 32 MHz F_PWM_INT = ’000’ F_PWM_DEC = ’111’		125		
N_{PWM}	PWM resolution			8		bit
Overcurrent protection						
V_{OCD}	Programmable overcurrent detection voltage V_{DS} threshold	OCD_TH = ‘11111’	800	1000	1100	mV
		OCD_TH = ‘00000’	27	31	35	mV
		OCD_TH = ‘01001’	270	312.5	344	mV
		OCD_TH = ‘10011’	500	625	688	mV
$t_{OCD,Comp}$	OCD comparator delay			100	200	ns
$t_{OCD,Flag}$	OCD to flag signal delay time			230	530	ns
$t_{OCD,SD}$	OCD to shutdown delay time	OCD_TH = '11111' OCD event to 90% of gate voltage		400	630	ns
Stall detection						
V_{STALL}	Programmable stall detection V_{DS} voltage threshold	STALL_TH = '11111'		1000		mV
		STALL_TH = '00000'		31		

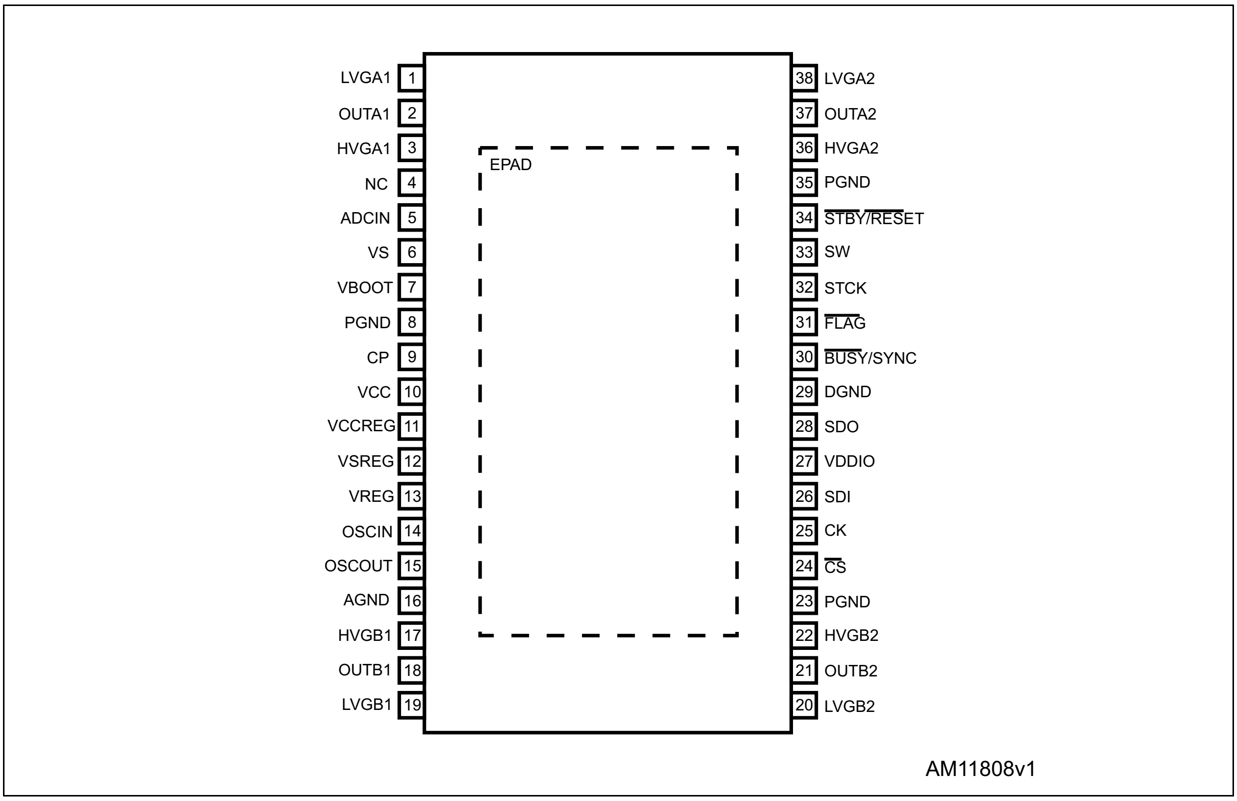
Table 5. Electrical characteristics (continued)

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
Standby						
I_{STBY}	Standby mode supply current (VSREG pin)	V_{CC} = V_{CCREG} = 7.5 V V_{SREG} = 48 V		42		μA
		VCC = VCCREG = 7.5 V V_{SREG} = 18 V		37.5		
$I_{STBY,vreg}$	Standby mode supply current (VREG pin)			6		μA
$t_{STBY,min}$	Minimum standby time			0.5		ms
$t_{logicwu}$	Logic power-on and wake-up time			500		μs
t_{cpwu}	Charge pump power-on and wake-up time	Power bridges disabled, C_p = 10 nF, C_{boot} = 220 nF, V_{CC} = 15 V		1		ms
Internal voltage regulators						
V_{CCOUT}	Internal V_{CC} voltage regulator output voltage	Low (default), I_{CC}= 10 mA	7.3	7.5		V
		High, I_{CC} = 10 mA	14	15		
$V_{SREG, drop}$	V_{SREG} to V_{CC} dropout voltage	I_{CC} = 50 mA			3	V
P_{CC}	Internal V_{CC} voltage regulator power dissipation				2.5	W
V_{REGOUT}	Internal V_{REG} voltage regulator output voltage	I_{REG} = 10 mA	3.135	3.3		V
$V_{CCREG, drop}$	V_{CCREG} to V_{REG} dropout voltage	I_{REG} = 50 mA			3	V
I_{REGOUT}	Internal V_{REG} voltage regulator output current	VREG pin shorted to ground.		125		mA
$I_{REGOUT,STBY}$	Internal V_{REG} voltage regulator output standby current	VREG pin shorted to ground.		55		mA
P_{REG}	Internal V_{REG} voltage regulator power dissipation				0.5	W
Integrated analog to digital converter						
N_{ADC}	Analog to digital converter resolution			5		bit
$V_{ADC,ref}$	Analog to digital converter reference voltage			3.3		V
f_S	Analog to digital converter sampling frequency	(2)		f_{PWM}		kHz
$V_{ADC,UVLO}$	ADCIN UVLO threshold		1.05	1.16	1.35	V

1. Guaranteed in the temperature range -25 to 125 °C.
2. The value accuracy is dependent on oscillator frequency accuracy (*Section 6.8 on page 27*).
3. $\overline{\text{FLAG}}$ and $\overline{\text{BUSY}}$ open drain outputs included.
4. See *Figure 19 on page 41*.

4 Pin connection

Figure 2. Pin connection (top view)



Pin list

Table 6. Pin description

No.	Name	Type	Function
11	VCCREG	Power supply	Internal V_{REG} voltage regulator supply voltage
13	VREG	Power supply	Logic supply voltage
27	VDD	Power supply	Logic interface supply voltage
12	VSREG	Power supply	Internal V_{CC} voltage regulator supply voltage
10	VCC	Power supply	Gate driver supply voltage
14	OSCIN	Analog input	Oscillator pin1. To connect an external oscillator or clock source.
15	OSCOUT	Analog output	Oscillator pin2. To connect an external oscillator. When the internal oscillator is used, this pin can supply a 2/4/8/16 MHz clock.
9	CP	Output	Charge pump oscillator output
7	VBOOT	Power supply	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

Table 6. Pin description (continued)

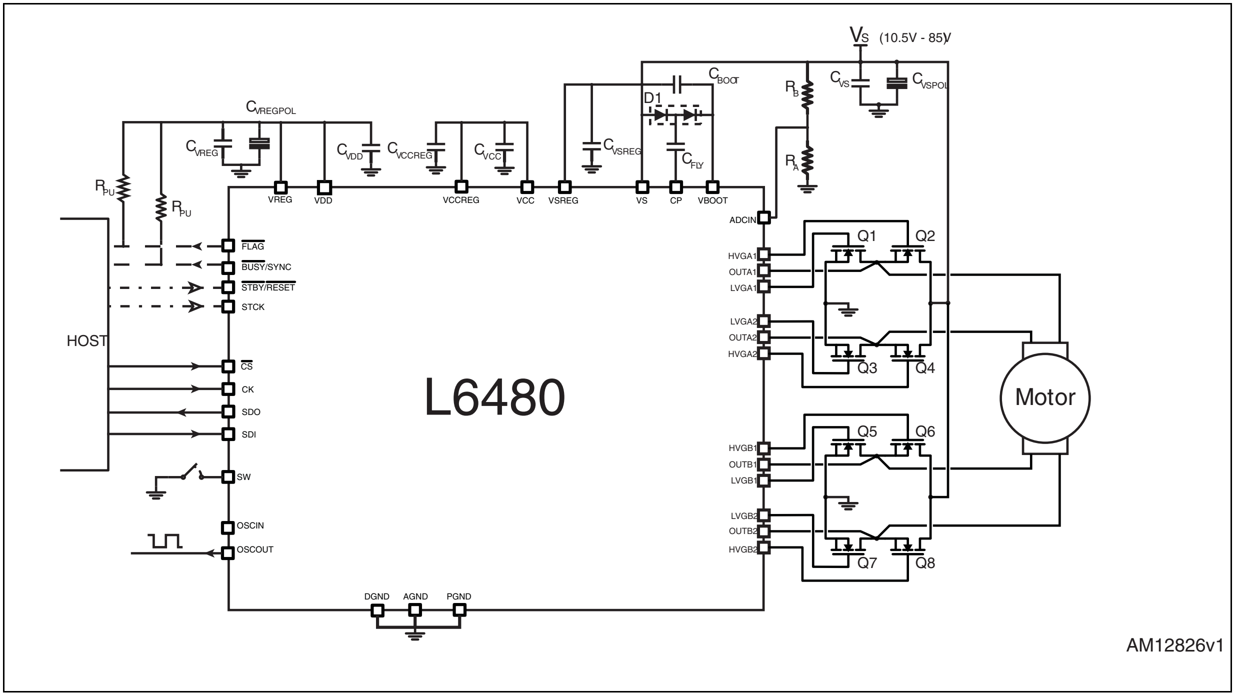
No.	Name	Type	Function
6	VS	Power supply	Motor voltage
3	HVGA1	Power output	High-side half-bridge A1 gate driver output.
36	HVGA2	Power output	High-side half-bridge A2 gate driver output
17	HVGB1	Power output	High-side half-bridge B1 gate driver output
22	HVGB2	Power output	High-side half-bridge B2 gate driver output
1	LVGA1	Power output	Low-side half-bridge A1 gate driver output
38	LVGA2	Power output	Low-side half-bridge A2 gate driver output
19	LVGB1	Power output	Low-side half-bridge B1 gate driver output
20	LVGB2	Power output	Low-side half-bridge B2 gate driver output
8, 23, 35	PGND	Ground	Power ground pins. They must be connected to other ground pins.
2	OUTA1	Power input	Full bridge A output 1
37	OUTA2	Power input	Full bridge A output 2
18	OUTB1	Power input	Full bridge B output 1
21	OUTB2	Power input	Full bridge B output 2
16	AGND	Ground	Analog ground. It must be connected to other ground pins.
33	SW	Logical input	External switch input pin
29	DGND	Ground	Digital ground. It must be connected to other ground pins
28	SDO	Logical output	Data output pin for serial interface
26	SDI	Logical input	Data input pin for serial interface
25	CK	Logical input	Serial interface clock
24	CS	Logical input	Chip select input pin for serial interface
30	$\overline{\text{BUSY}}$/SYNC	Open drain output	By default, the $\overline{\text{BUSY}}$ /SYNC pin is forced low when the device is performing a command. The pin can be programmed in order to generate a synchronization signal.
31	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).
34	STBY RESET	Logical input	Standby and reset pin. LOW logic level puts the device in Standby mode and reset logic. If not used, should be connected to V_{REG}.
32	STCK	Logical input	Step-clock input
EPAD	Exposed pad	Ground	Exposed pad. It must be connected to other ground pins.

5 Typical applications

Table 7. Typical application values

Name	Value
C_{VSPOL}	220 µF
C_{VS}	220 nF
C_{BOOT}	470 nF
C_{FLY}	47 nF
C_{VSREG}	100 nF
C_{VCC}	470 nF
C_{VCCREG}	100 nF
C_{VREG}	100 nF
$C_{VREGPOL}$	22 µF
C_{VDD}	100 nF
D1	Charge pump diodes
Q1, Q2, Q3, Q4, Q5, Q6, Q7, Q8	STD25N10F7
R_{PU}	39 kΩ
R_A	1.8 kΩ (V_S = 85 V)
R_B	91 kΩ (V_S = 85 V)

Figure 3. Typical application schematic



6 Functional description

6.1 Device power-up

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

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

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

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

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

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

6.2 Logic I/O

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

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

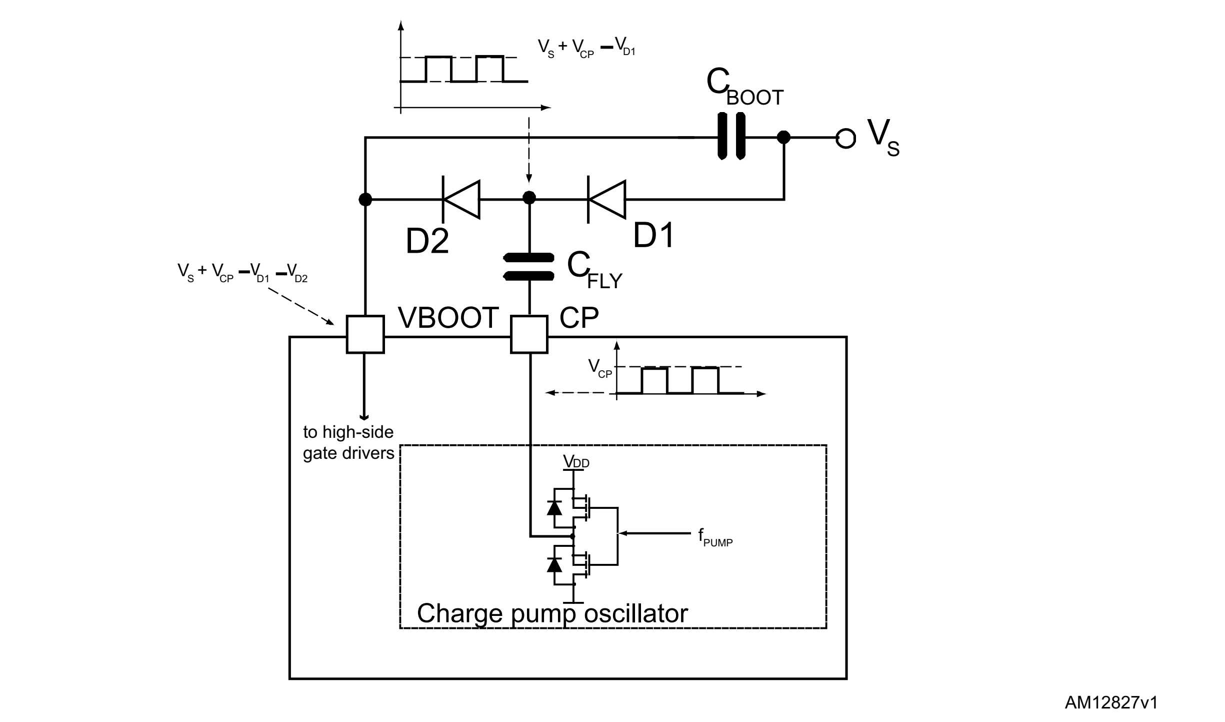
Pins $\overline{\text{FLAG}}$ and $\overline{\text{BUSY}}$/SYNC are open drain outputs.

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

6.3 Charge pump

To ensure the correct driving of the high-side gate drivers, a voltage higher than the motor power supply voltage needs to be applied to the VBOOT pin. The high-side gate driver supply voltage V_{BOOT} is obtained through an oscillator and a few external components realizing a charge pump (see *Figure 4*).

Figure 4. Charge pump circuitry

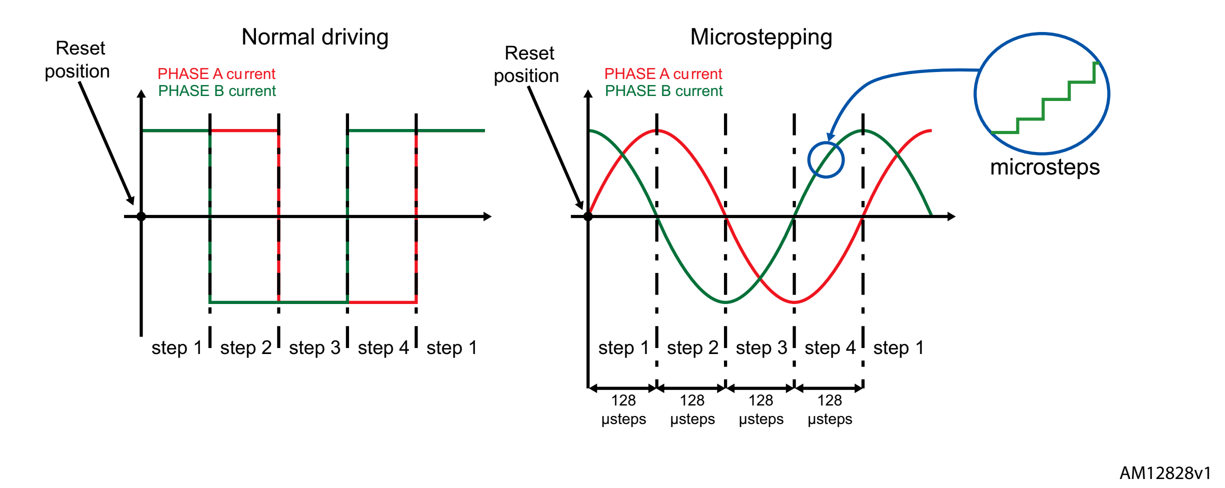


6.4 Microstepping

The driver is able to divide the single step into up to 128 microsteps. Stepping mode can be programmed by the STEP_SEL parameter in the STEP_MODE register (*Table 20 on page 50*).

Step mode can be only changed when bridges are disabled. Every time the 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 (*Section 6.5*) becomes meaningless.

Figure 5. Normal mode and microstepping (128 microsteps)



Automatic Full-step and Boost modes

When motor speed is greater than a programmable full-step speed threshold, the L6480 switches automatically to Full-step mode; the driving mode returns to microstepping when motor speed decreases below the full-step speed threshold.

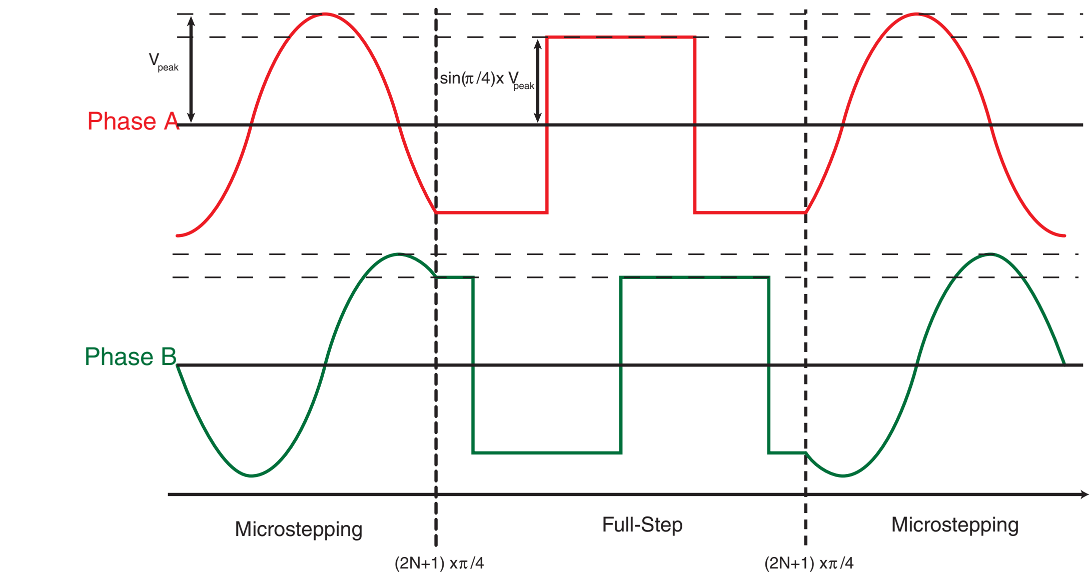
The switching between the microstepping and Full-step mode and vice-versa is always performed at an electrical position multiple of π/4 (*Figure 6* and *Figure 7*).

Full-step speed threshold is set through the related parameter in the FS_SPD register (*Section 9.1.9 on page 47*).

When the BOOST_MODE bit of the FS_SPD register is low (default), the amplitude of the voltage squarewave in Full-step mode is equal to the peak of the voltage sine wave multiplied by sine(π/4) (*Figure 6*). This avoids the current drop between the two driving modes.

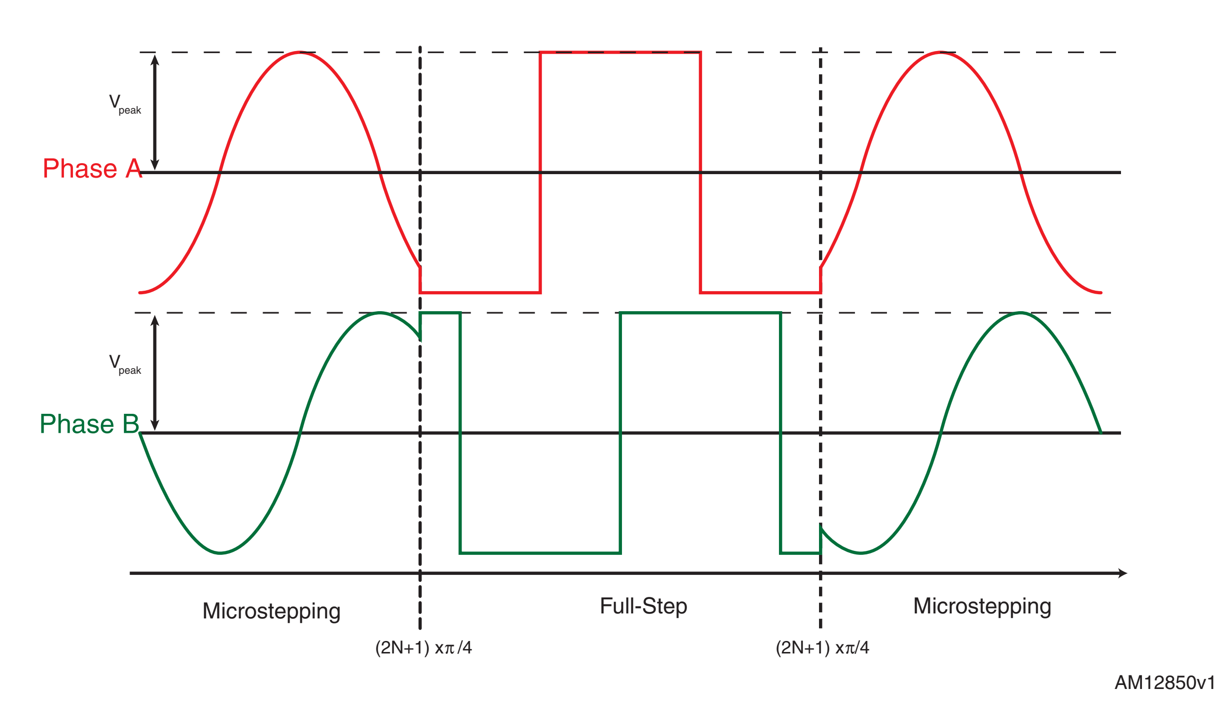
When the BOOST_MODE bit of the FS_SPD register is high, the amplitude of the voltage squarewave in Full-step mode is equal to the peak of the voltage sine wave (*Figure 7*). That improves the output current increasing the maximum motor torque.

Figure 6. Automatic Full-step switching in Normal mode



AM12829v1

Figure 7. Automatic Full-step switching in Boost mode



6.5 Absolute position counter

An internal 22-bit register (ABS_POS) records all the motor motions according to the selected step mode; the stored value unit is equal to the selected step mode (full, half, quarter, etc.). The position range is from -2^{21} to $+2^{21}-1$ steps (see *Section 9.1.1 on page 44*).

6.6 Programmable speed profiles

The user can easily program a customized speed profile defining independently acceleration, deceleration, maximum and minimum speed values by ACC, DEC, MAX_SPEED and MIN_SPEED registers respectively (see *Section 9.1.5 on page 45*, *9.1.6 on page 46*, *9.1.7 on page 46* and *9.1.8 on page 46*).

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 the selected step mode. Acceleration and deceleration parameters range from 2^{-40} to $(2^{12}-2) \bullet 2^{-40}$ step/tick^2 (equivalent to 14.55 to 59590 step/s^2).

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

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

6.7 Motor control commands

The L6480 can accept different types of commands:

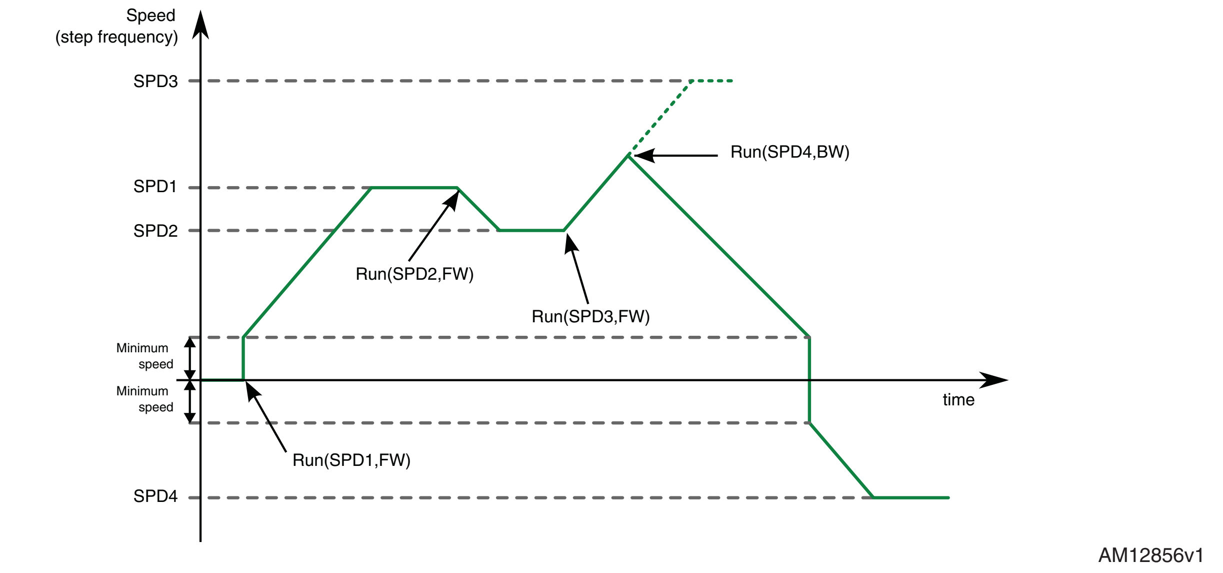
- constant speed commands (Run, GoUntil, ReleaseSW)
- absolute positioning commands (GoTo, GoTo_DIR, GoHome, GoMark)
- motion commands (Move)
- stop commands (SoftStop, HardStop, SoftHiz, HardHiz).

For detailed command descriptions refer to *Section 9.2 on page 62*.

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



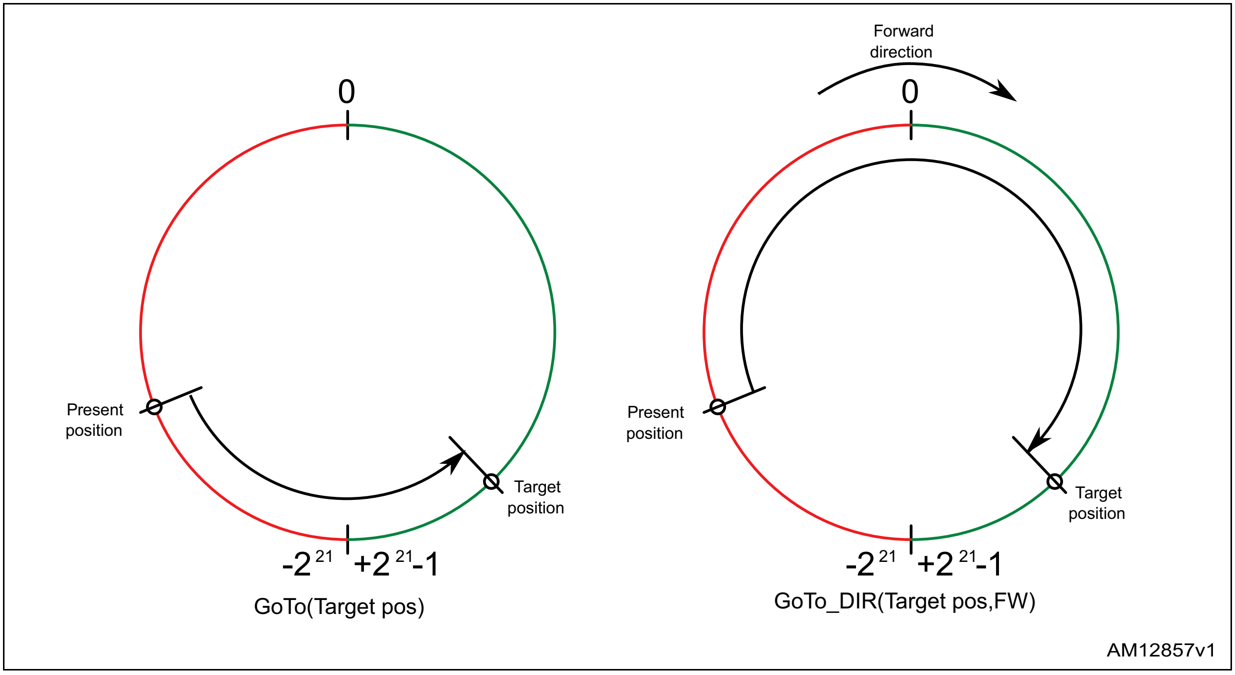
6.7.2 Positioning commands

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

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

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

Figure 9. Positioning command examples



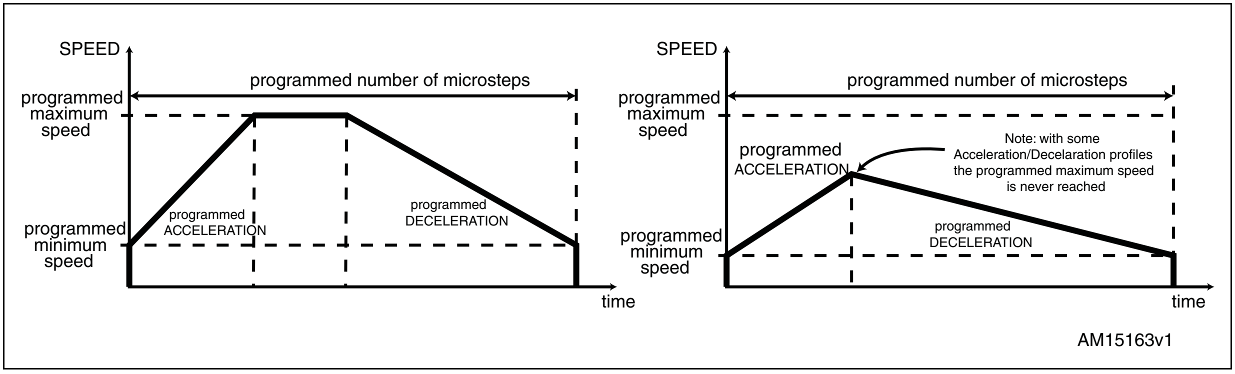
6.7.3 Motion commands

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*).

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 MIN_SPEED value is reached and then stops the motor keeping the rotor position (a holding torque is applied).

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

The SoftHiZ command causes the motor to decelerate with a programmed deceleration value until the MIN_SPEED value is reached and then forces the bridges into 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 one microstep in the programmed direction and absolute position is consequently updated.

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

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 commands47

can be used in combination with external switch input (see *Section 6.14 on page 31*) to easily initialize the motor position.

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

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

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

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

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

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

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 EXT_CLK and OSC_SEL parameters in the CONFIG register (see *Table 32 on page 55*).

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

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

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 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 (C_L) 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

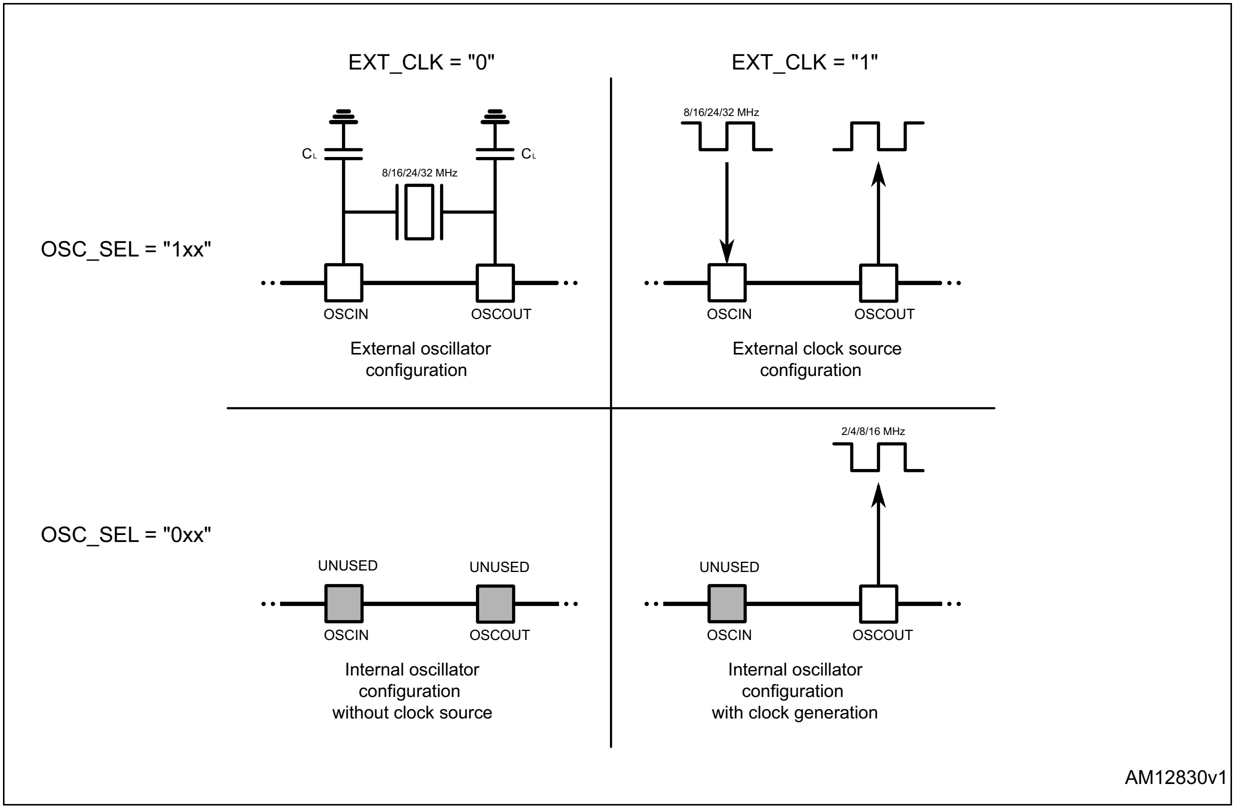
Crystal/resonator freq. [1]	C_L [2]
8 MHz	25 pF (ESR_{max} = 80 Ω)
16 MHz	18 pF (ESR_{max} = 50 Ω)
24 MHz	15 pF (ESR_{max} = 40 Ω)
32 MHz	10 pF (ESR_{max} = 40 Ω)

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

If a direct clock source is used, it must be connected to the OSCIN pin and the OSCOUT pin supplies the inverted OSCIN signal (see *Figure 11*).

The L6480 device integrates a clock detection system that resets the device in case of the failure of the external clock source (direct or crystal/resonator). The monitoring of the clock source is disabled by default, it can be enabled setting high the WD_EN bit in the GATECFG1 register (*Section 9.1.21 on page 52*). When the external clock source is selected, the device continues to work with the integrated oscillator for t_{extosc} milliseconds and then the clock management system switches to the OSCIN input.

Figure 11. OSCIN and OSCOUT pin configuration


EXT_CLK = "0"
EXT_CLK = "1"
OSC_SEL = "1xx"
OSC_SEL = "0xx"
CL
CL
8/16/24/32 MHz
8/16/24/32 MHz
OSCIN
OSCOUT
External oscillator configuration
External clock source configuration
2/4/8/16 MHz
UNUSED
UNUSED
UNUSED
Internal oscillator configuration without clock source
Internal oscillator configuration with clock generation
AM12830v1

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

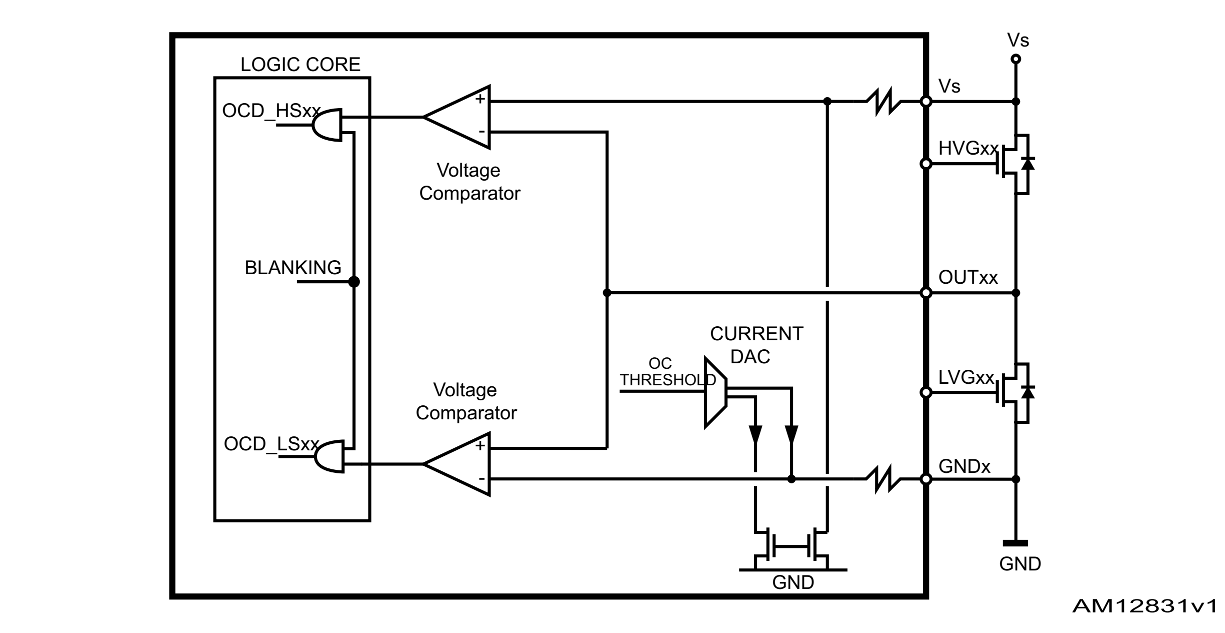
When OSCOUT is UNUSED it should be left floating.

6.9 Overcurrent detection

The L6480 measures the load current of each half-bridge sensing the V_{DS} voltage of all the Power MOSFETs (*Figure 12*). When any of the V_{DS} voltages rise over the programmed threshold, the OCD flag in the STATUS register is forced low until the event expires and a GetStatus command is sent to the device (*Section 9.1.24 on page 59* and *Section 9.2.20 on page 71*). The overcurrent event expires when all the Power MOSFET V_{DS} voltages fall below the programmed threshold.

The overcurrent threshold can be programmed by the OCD_TH register in one of 32 available values ranging from 31.25 mV to 1 V with steps of 31.25 mV (*Table 18 on page 50 Section 9.1.17 on page 50*).

Figure 12. Overcurrent detection-principle scheme



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

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

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

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

6.10 Undervoltage lockout (UVLO)

The L6480 provides a programmable gate driver supply voltage UVLO protection. When one of the supply voltages of the gate driver (V_{CC} for the low sides and V_{BOOT} - V_S for the high sides) falls below the respective turn-off threshold, an undervoltage event occurs. In this case, all MOSFETs are immediately turned off and the UVLO flag in the STATUS register is forced low.

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

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

Table 9. UVLO thresholds

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

6.11 VS undervoltage lockout (UVLO_ADC)

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

The ADCIN undervoltage event doesn’t turn off the MOSFETs of the power bridges.

The motor supply voltage undervoltage detection can be performed by means of this feature, connecting the ADCIN pin to VS through a voltage divider as described in *Section 7.5 on page 39*.

6.12 Thermal warning and thermal shutdown

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

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

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

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

The thermal condition of the device is shown by TH_STATUS bits in the STATUS register (*Table 10*).

Table 10. Thermal protection summarizing table

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

6.13 Reset and standby

The device can be reset and put into Standby mode through the $\overline{\text{STBY}}/\overline{\text{RESET}}$ pin. When it is forced low, all the MOSFETs are turned off (High Z state), the charge pump is stopped, the SPI interface and control logic are disabled and the internal V_{REG} voltage regulator maximum output current is limited; as a result, the L6480 heavily reduces the power consumption. At the same time the register values are reset to their default and all the protection functions are disabled. The $\overline{\text{STBY}}/\overline{\text{RESET}}$ 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 must be given before applying a new command to allow proper oscillator and charge pump startup. Actual delay could vary according to the values of the charge pump external components.

On exiting Standby mode all the MOSFETs are off and the HiZ flag is high.

The registers can be reset to the default values without putting the device into Standby mode through the ResetDevice command (*Section 9.2.14 on page 69*).

6.14 External switch (SW pin)

The SW input is internally pulled up to V_{DD} and detects if the pin is open or connected to ground (see *Figure 13*).

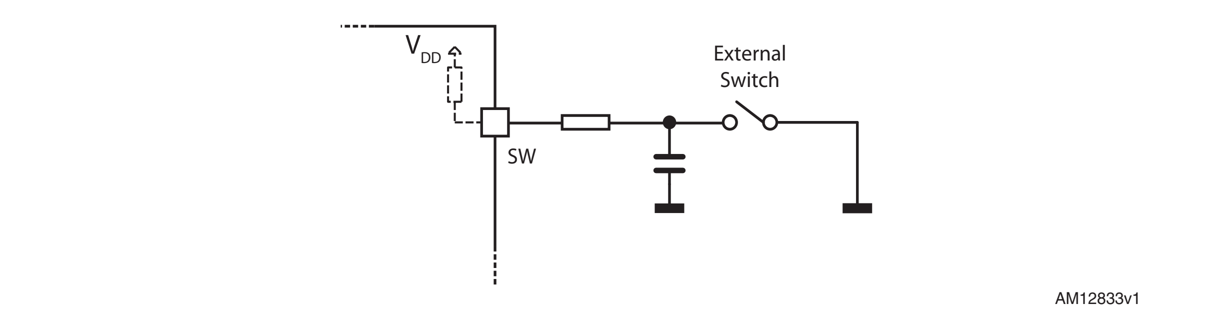
The SW_F bit of the STATUS register indicates if the switch is open (‘0’) or closed (‘1’) (*Section 9.1.24 on page 59*); the bit value is refreshed at every system clock cycle (125 ns). The SW_EVN flag of the STATUS register is raised when a switch turn-on event (SW input falling edge) is detected (*Section 9.1.24*). A GetStatus command releases the SW_EVN flag (*Section 9.2.20 on page 71*).

By default, a switch turn-on event causes a HardStop interrupt (SW_MODE bit of CONFIG register set to ‘0’). Otherwise (SW_MODE bit of CONFIG register set to ‘1’), switch input events do not cause interrupts and the switch status information is at the user’s disposal (*Table 32 on page 55* and *Section 9.1.24 on page 59*).

The switch input can be used by GoUntil and ReleaseSW commands as described in *Section 9.2.10 on page 67* and *Section 9.2.11 on page 68*.

If the SW input is not used, it should be connected to V_{DD}.

Figure 13. External switch connection



6.15 Programmable gate drivers

The L6480 device integrates eight programmable gate drivers that allow the fitting of a wide range of applications.

The following parameters can be adjusted:

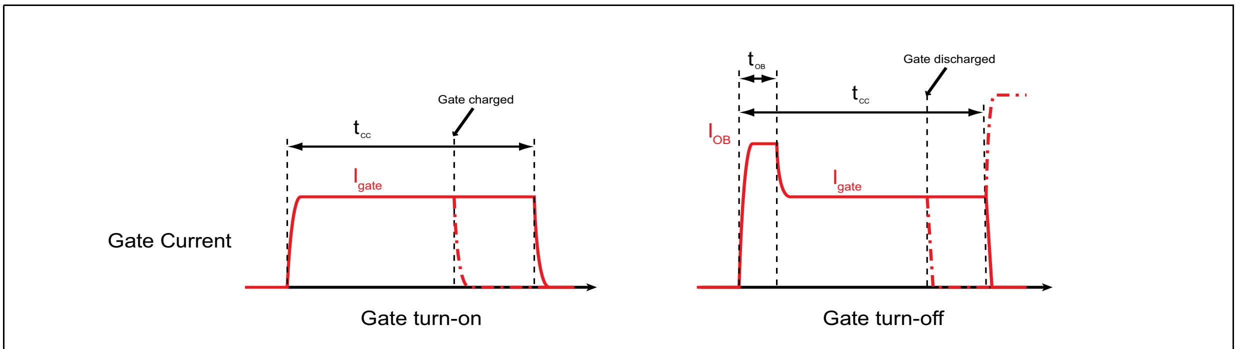
- gate sink/source current (I_{GATE})
- controlled current time (t_{CC})
- turn-off overboost time (t_{OB}).

During turn-on, the gate driver charges the gate forcing an I_{GATE} current for all the controlled current time period. At the end of the controlled current phase the gate of the external MOSFET should be completely charged, otherwise the gate driving circuitry continues to charge it using a holding current.

This current is equal to I_{GATE} for the low-side gate drivers and 1 mA for the high-side ones.

During turn-off, the gate driver discharges the gate sinking an I_{GATE} current for all the controlled current time period. At the beginning of turn-off an overboost phase can be added: in this case the gate driver sinks an I_{OB} current for the programmed t_{OB} period in order to rapidly reach the plateau region. At the end of the controlled current time the gate of the external MOSFET should be completely charged, otherwise the gate driving circuitry discharges it using the integrated Miller clamp.

Figure 14. Gate driving currents



The gate current can be set to one of the following values: 4, 8, 16, 24, 32, 64 and 96 mA through the IGATE parameter in the GATECFG1 register (see *Section 9.1.21 on page 52*).

Controlled current time can be programmed within range from 125 ns to 3.75 µs with a resolution of 125 ns (TCC parameter in GATECFG1 register) (see *Section 9.1.21*).

Turn-off overboost time can be set to one of the following values: 0, 62.5, 125, 250 ns (TBOOST parameter in GATECFG1 register). The 62.5 ns value is only available when clock frequency is 16 MHz or 32 MHz; when clock frequency is 8 MHz it is changed to 125 ns and when a 24-MHz clock is used it is changed to 83.3 ns. (see *Section 9.1.21*).

6.16 Deadtime and blanking time

During the bridge commutation, a deadtime is added in order to avoid cross conductions. The deadtime can be programmed within a range from 125 ns to 4 µs with a resolution of 125 ns (TDT parameter in the GATECFG2 register) (see *Section 9.1.22 on page 53*).

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

The duration of blanking time is programmable through the TBLANK parameter in the GATECFG2 register at one of the following values: 125, 250, 375, 500, 625, 750, 875, 1000 ns (see *Section 9.1.22*).

6.17 Integrated analog to digital converter

The L6480 integrates an N_{ADC} bit ramp-compare analog to digital converter with a reference voltage equal to V_{REG}. The analog to digital converter input is available through the ADCIN pin and the conversion result is available in the ADC_OUT register (*Section 9.1.16 on page 49*).

Sampling frequency is equal to the programmed PWM frequency.

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

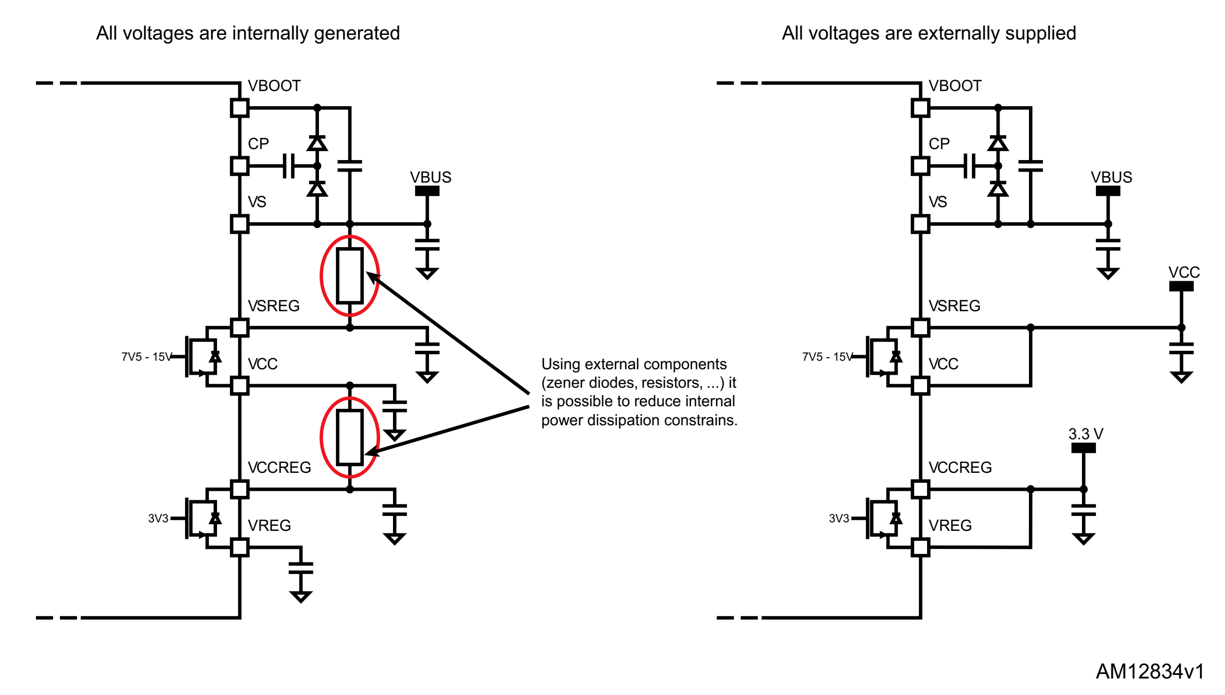
6.18 Supply management and internal voltage regulators

The L6480 integrates two linear voltage regulators: the first one can be used to obtain gate driver supply starting from a higher voltage (e.g. the motor supply one). Its output voltage can be set to 7.5 V or 15 V according to the VCCVAL bit value (CONFIG register). The second linear voltage regulator can be used to obtain the 3.3 V logic supply voltage.

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

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

Figure 15. Device supply pin management


All voltages are internally generated
All voltages are externally supplied
VBOOT
CP
VS
VBUS
VSREG
VCC
7V5 - 15V
VCCREG
VREG
3V3
3.3 V
Using external components (zener diodes, resistors, ...) it is possible to reduce internal power dissipation constrains.
AM12834v1

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

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

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

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

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

6.19 BUSY/SYNC pin

This pin is an open drain output which can be used as busy flag or synchronization signal according to the SYNC_EN bit value (STEP_MODE register) (see *Section 9.1.19 on page 50*).

6.20 FLAG pin

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

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

It is possible to mask one or more alarm conditions by programming the ALARM_EN register (see *Section 9.1.20 on page 52* and *Table 23 on page 52*). 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 case of daisy chain configuration, FLAG pins of different ICs can be or-wired to save host controller GPIOs.

7 Phase current control

The L6480 controls the phase current applying a sinusoidal voltage to motor windings. Phase current amplitude is not directly controlled but depends on phase voltage amplitude, load torque, motor electrical characteristics and rotation speed. Sine wave amplitude is proportional to the motor supply voltage multiplied by a coefficient (K_{VAL}). K_{VAL} ranges from 0 to 100% and the sine wave amplitude can be obtained through the following formula:

Equation 1

$$V_{OUT} = V_S \cdot K_{VAL}$$

Different K_{VAL} values can be programmed for acceleration, deceleration and constant speed phases and when the motor is stopped (HOLD phase) through KVAL_ACC, KVAL_DEC, KVAL_RUN and KVAL_HOLD registers (*Section 9.1.10 on page 47*). KVAL value is calculated according to the following formula:

Equation 2

$$K_{VAL} = [(K_{VAL\_X} + \mathrm{BEMF\_COMP}) \times \mathrm{VSCOMP} \times \mathrm{K\_THERM}] \times \mathrm{microstep}$$

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

The L6480 offers various methods to guarantee a stable current value, allowing the compensation of:

- low speed distortion (*Section 7.3*)
- back electromotive force (*Section 7.4*)
- motor supply voltage variation (*Section 7.5*)
- windings resistance variation (*Section 7.6 on page 40*).

7.1 PWM sine wave generators

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

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

Equation 3

$$f_{PWM} = \frac{f_{OSC}}{512 \cdot N} \cdot m$$

'*N*' is the integer division factor and '*m*' is the multiplication factor. 'N' and 'm' values can be programmed by F_PWM_INT and F_PWM_DEC parameters in the CONFIG register (see *Table 38 on page 57* and *Table 39 on page 57*, *Section 9.1.23 on page 54*).

Available PWM frequencies are listed in *Section 9.1.23* from *Table 40 on page 58* to *Table 43 on page 59*.

7.2 Sensorless stall detection

The L6480 is able to detect a motor stall caused by an excessive load torque. When the motor is driven using the voltage mode approach, a stall condition corresponds to an unexpected increase of the phase current. Imposing a current threshold slightly above the operative current, it is possible to detect the stall condition without speed or position sensors.

The L6480 measures the load current of each phase sensing the V_{DS} voltage of the low-side Power MOSFETs. When any of the V_{DS} voltages rise over the programmed threshold, the STEP_LOSS_X flag in the STATUS register of the respective bridge (STEP_LOSS_A or STEP_LOSS_B) is forced low. The failure flag is kept low until the V_{DS} voltages fall below the programmed threshold and a GetStatus command is sent to the device (*Section 9.1.24 on page 59* and *Section 9.2.20 on page 71*).

The stall detection threshold can be programmed in one of 32 available values ranging from 31.25 mV to 1 V with steps of 31.25 mV (see *Section 9.1.18 on page 50*).

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

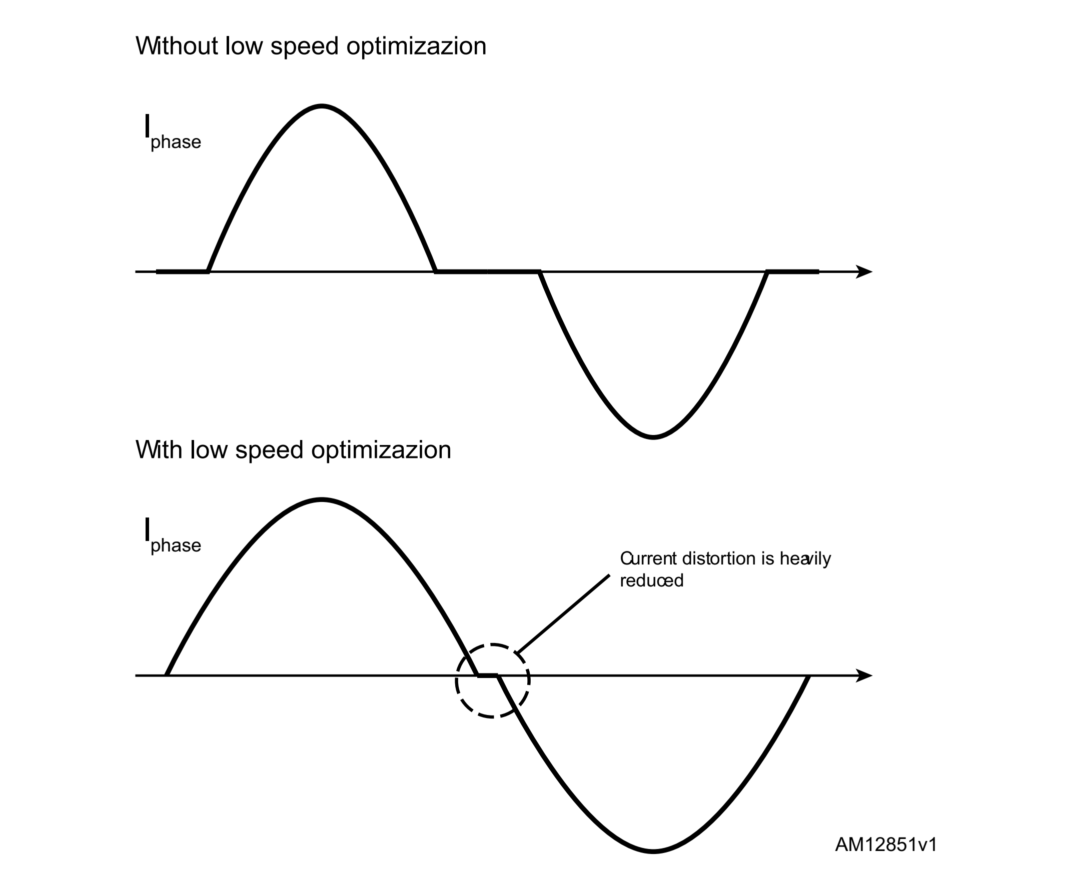
- The respective half-bridge is in high impedance state (both MOSFETs forced off)
- The respective half-bridge is commutating
- The respective half-bridge is commutated and the programmed blanking time has not yet elapsed
- The respective low-side gate is turned off.

7.3 Low speed optimization

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

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

Figure 16. Current distortion and compensation



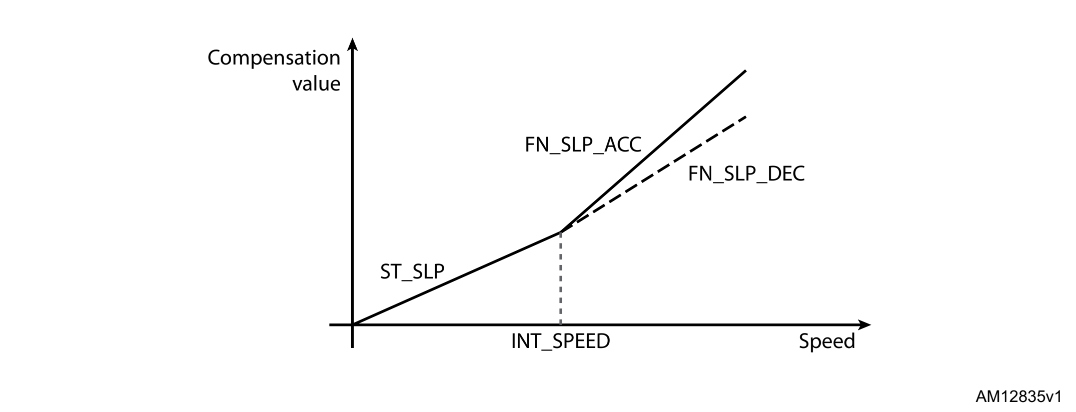
The optimization can be enabled setting high the LSPD_OPT bit in the MIN_SPEED register (*Section 9.1.8 on page 46*) and is active in a speed range from zero to MIN_SPEED. When low speed optimization is enabled, speed profile minimum speed is forced to zero.

7.4 BEMF compensation

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

The compensation curve is approximated by a stacked line with a starting slope (ST_SLP) when speed is lower than a programmable threshold speed (INT_SPEED) and a fine slope (FN_SLP_ACC and FN_SLP_DEC) when speed is greater than the threshold speed (see sections *9.1.11 on page 48*, *9.1.12 on page 48*, *9.1.13 on page 48* and *9.1.14 on page 49*).

Figure 17. BEMF compensation curve



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

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

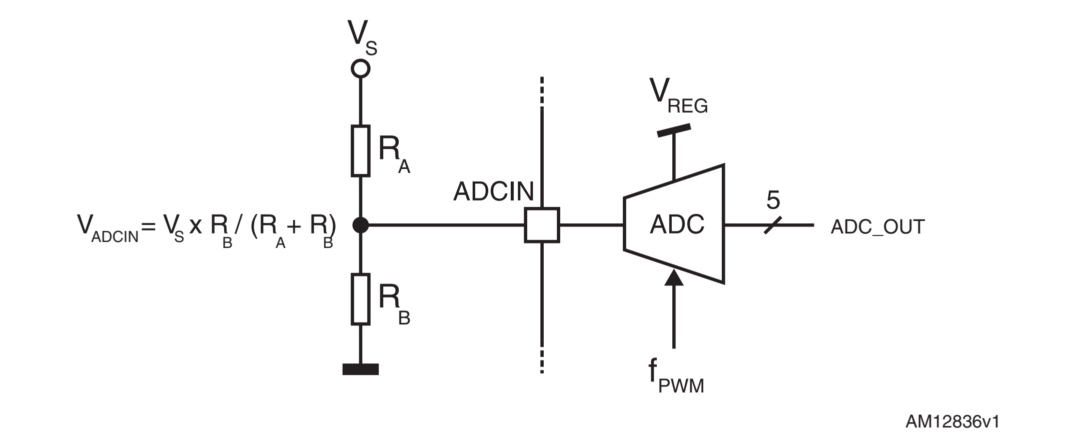
7.5 Motor supply voltage compensation

The sine wave amplitude generated by the PWM modulators is directly proportional to the motor supply voltage (V_S). When the motor supply voltage is different from its nominal value, the motor phases are driven with an incorrect voltage. The L6480 can compensate motor supply voltage variations in order to avoid this effect.

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

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

Figure 18. Motor supply voltage compensation circuit



Motor supply voltage compensation can be enabled setting high the EN_VSCOMP bit of the CONFIG register (see*Table 37 on page 56*, *Section 9.1.23 on page 54*). If the EN_VSCOMP bit is low, the compensation is disabled and the internal analog to digital converter is at the user’s disposal; the sampling rate is always equal to PWM frequency.

7.6 Winding resistance thermal drift compensation

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

The L6480 integrates a register (K_THERM) which can be used to compensate phase resistance increment due to temperature rising.

The value in the K_THERM register (*Section 9.1.15 on page 49*) multiplies duty cycle value allowing the higher phase resistance value to be faced.

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

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 L6480 (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 12* for values).

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

Figure 19. SPI timings diagram

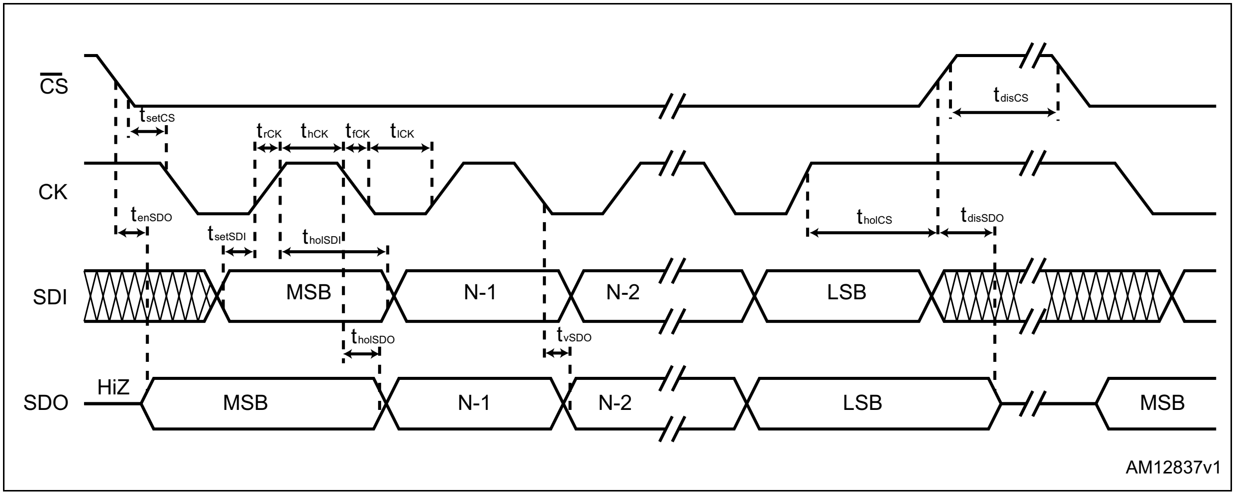
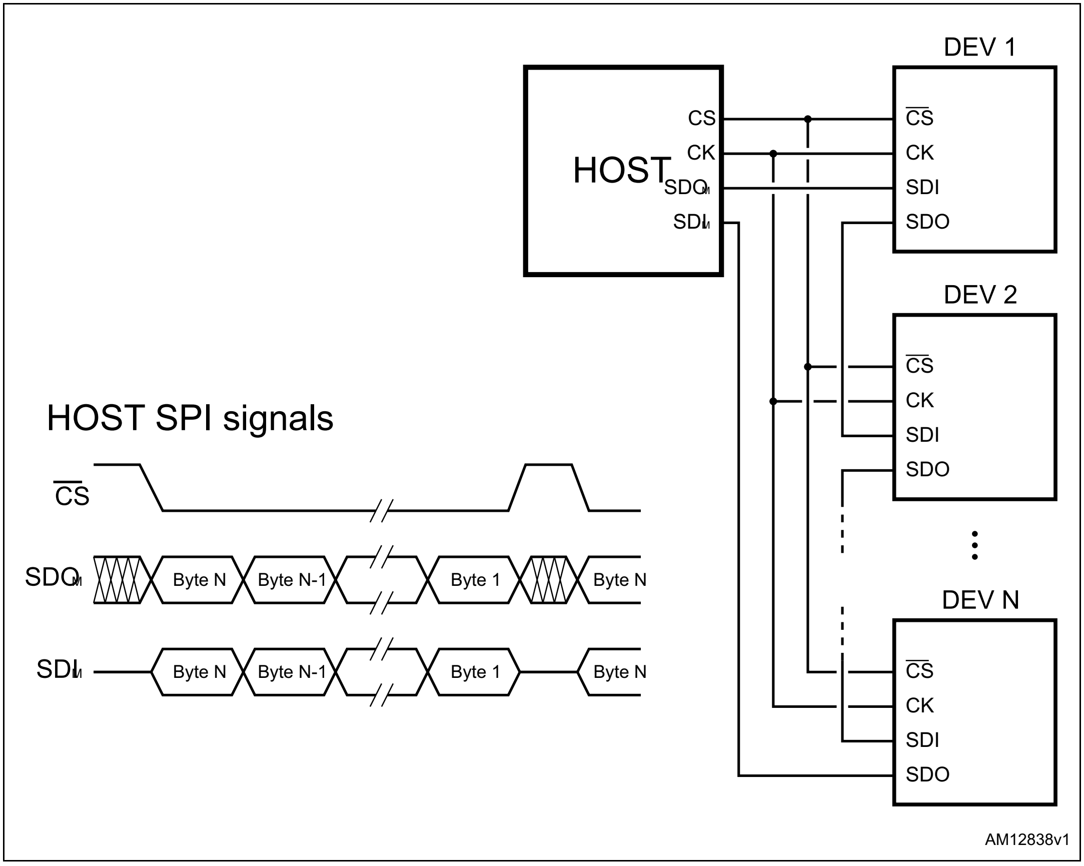


Figure 20. Daisy chain configuration



9 Programming manual

9.1 Register and flag description

The following section shows the user registers available (detailed description in respective paragraphs from *Section 9.1.1 on page 44* to *Section 9.1.24 on page 59*):

Table 11. 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	12	000	0 step/tick (0 step/s)	R, WS
h15	FS_SPD	Full-step speed	11	027	150.7e-6 step/tick (602.7 step/s)	R, WR
h09	KVAL_HOLD	Holding K_{VAL}	8	29	$0.16 \cdot V_S$	R, WR
h0A	KVAL_RUN	Constant speed K_{VAL}	8	29	$0.16 \cdot V_S$	R, WR
h0B	KVAL_ACC	Acceleration starting K_{VAL}	8	29	$0.16 \cdot V_S$	R, WR
h0C	KVAL_DEC	Deceleration starting K_{VAL}	8	29	$0.16 \cdot V_S$	R, WR
h0D	INT_SPEED	Intersect speed	14	0408	15.4e-6 step/tick (61.5 step/s)	R, WH
h0E	ST_SLP	Start slope	8	19	250.038% s/step	R, WH
h0F	FN_SLP_ACC	Acceleration final slope	8	29	0.063% s/step 25	R, WH
h10	FN_SLP_DEC	Deceleration final slope	8	29	0.063% s/step 25	R, WH
h11	K_THERM	Thermal compensation factor	4	0	1.0	R, WR
h12	ADC_OUT	ADC output	5	XX[2]	0	R
h13	OCD_TH	OCD threshold	5	8	281.25 mV	R, WR
h14	STALL_TH	STALL threshold	5	10	531.25 mV	R, WR
h16	STEP_MODE	Step mode	4	7	BUSY/SYNC output used as BUSY, 128 µsteps	R, WH

Table 11. Register map (continued)

Address [Hex]	Register name	Register function	Len. [bit]	Reset [Hex]	Reset value	Remarks[1]
h17	ALARM_EN	Alarms enables	8	FF	All alarms enabled	R, WS
h18	GATECFG1	Gate driver configuration	11	0	I_{gate} = 4 mA, t_{CC} = 125 ns, no boost	R, WH
h19	GATECFG2	Gate driver configuration	8	0	t_{BLANK} = 125 ns, t_{DT} = 125 ns	R, WH
h1A	CONFIG	IC configuration	16	2C88	Internal 16 MHz oscillator (OSCOUT at 2 MHz), SW event causes HardStop, motor supply voltage compensation disabled, overcurrent shutdown, V_{CC} = 7.5 V, UVLO threshold low, f_{PWM} = f_{OSC} / 1024	R, WH
h1B	STATUS	Status	16	XXXX[2]	High impedance state, motor stopped, reverse direction, all fault flags released UVLO/Reset flag set	R

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.

9.1.1 ABS_POS

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

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

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

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 12. EL_POS register

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

When the EL_POS register is written by the user the new electrical position is instantly imposed. When the EL_POS register is written, its value must be masked in order to match with the step mode selected in the STEP_MODE register in order to avoid a wrong microstep value generation (*Section 9.1.19 on page 50*); 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 CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

9.1.3 MARK

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

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 4

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

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 CMD_ERROR flag to rise (*Section 9.1.24*).

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 the ACC value in step/s^2 the following formula can be used:

Equation 5

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

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 CMD_ERROR flag to rise (*Section 9.1.24*).

9.1.6 DEC

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

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

Equation 6

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

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

The available range is from 14.55to 59590 step/s2 with a resolution of 14.55 step/s2.

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

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 7

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

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 13. MIN_SPEED register

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

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

Equation 8

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

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.

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

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

9.1.9 FS_SPD

The FS_SPD register contains the following parameters:

Table 14. FS_SPD register

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

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

Equation 9

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

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

The available range is from 7.63 to 15625 step/s with a resolution of 15.25 step/s.

The BOOST_MODE bit sets the amplitude of the voltage squarewave during the full-step operation (see *Section on page 22*).

9.1.10 KVAL_HOLD, KVAL_RUN, KVAL_ACC and KVAL_DEC

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

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

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

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

The available range is from 0 to 0.996 x V_S with a resolution of 0.004 x V_S, as shown in *Table 15*.

Table 15. Voltage amplitude regulation registers

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

9.1.11 INT_SPEED

The INT_SPEED register contains the speed value at which the BEMF compensation curve changes slope (*Section 7.4 on page 38* for details). Its value is expressed in step/tick and to convert it in [step/s] the following formula can be used:

Equation 10

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

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

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

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

9.1.12 ST_SLP

The ST_SLP register contains the BEMF compensation curve slope that is used when the speed is lower than the intersect speed (*Section 7.4*). Its value is expressed in s/step and the available range is from 0 to 0.004 with a resolution of 0.000015.

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

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24*).

9.1.13 FN_SLP_ACC

The FN_SLP_ACC register contains the BEMF compensation curve slope that is used when the speed is greater than the intersect speed during acceleration (*Section 7.4* for details). Its value is expressed in s/step and the available range is from 0 to 0.004 with a resolution of 0.000015.

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

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24*).

9.1.14 FN_SLP_DEC

The FN_SLP_DEC register contains the BEMF compensation curve slope that is used when the speed is greater than the intersect speed during deceleration (*Section 7.4 on page 38* for details). Its value is expressed in s/step and the available range is from 0 to 0.004 with a resolution of 0.000015.

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

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

9.1.15 K_THERM

The K_THERM register contains the value used by the winding resistance thermal drift compensation system (*Section 7.6 on page 40*).

The available range is from 1 to 1.46875 with a resolution of 0.03125, as shown in *Table 16*.

Table 16. Winding resistance thermal drift compensation coefficient

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

9.1.16 ADC_OUT

The ADC_OUT register contains the result of the analog to digital conversion of the ADCIN pin voltage; the result is available even if the supply voltage compensation is disabled.

Any attempt to write to the register causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24*).

Table 17. ADC_OUT value and motor supply voltage compensation feature

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

9.1.17 OCD_TH

The OCD_TH register contains the overcurrent threshold value (*Section 6.9 on page 29* for details). The available range is from 31.25 mV to 1 V, steps of 31.25 mV, as shown in *Table 18*.

Table 18. Overcurrent detection threshold

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

9.1.18 STALL_TH

The STALL_TH register contains the stall detection threshold value. The available range is from 31.25 mV to 1 V with a resolution of 31.25 mV.

Table 19. Stall detection threshold

STALL_th [4…0]					Stall detection threshold
0	0	0	0	0	31.25 mV
0	0	0	0	1	62.5 mV
…	…	…	…	…	…
1	1	1	1	0	968.75 mV
1	1	1	1	1	1 V

9.1.19 STEP_MODE

The STEP_MODE register has the following structure:

Table 20. STEP_MODE register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SYNC_EN	SYNC_SEL			0[1]	STEP_SEL		

[1] When the register is written this bit must be set to 0.

The STEP_SEL parameter selects one of eight possible stepping modes:

Table 21. 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	0	0	1/16 microstep
1	0	1	1/32 microstep
1	1	0	1/64 microstep
1	1	1	1/128 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 loses meaning and should be reset.

Any attempt to write the register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

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

The synchronization signal is obtained starting from 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	EL_POS[2]
1	1	0	EL_POS[1]
1	1	1	EL_POS[0]

9.1.20 ALARM_EN

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

Table 23. ALARM_EN register

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

9.1.21 GATECFG1

The GATECFG1 register has the following structure:

Table 24. GATECFG1 register

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

The IGATE parameter selects the sink/source current used by gate driving circuitry to charge/discharge the respective gate during commutations. Seven possible values ranging from 4 mA to 96 mA are available, as shown in *Table 25*.

Table 25. IGATE parameter

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

The TCC parameter defines the duration of constant current phase during gate turn-on and turn-off sequences (*Section 6.15 on page 32*).

Table 26. TCC parameter

TCC [4…0]					Constant current time [ns]
0	0	0	0	0	125
0	0	0	0	1	250
⇓	⇓	⇓	⇓	⇓	⇓
1	1	1	0	0	3625
1	1	1	0	1	3750
1	1	1	1	0	3750
1	1	1	1	1	3750

The TBOOST parameter defines the duration of the overboost phase during gate turn-off (*Section 6.15*).

Table 27. TBOOST parameter

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

[1] Clock frequency equal to 16 MHz or 32 MHz.

[2] Clock frequency equal to 24 MHz.

[3] Clock frequency equal to 8 MHz.

The WD_EN bit enables the clock source monitoring (*Section 6.8.2 on page 27*).

9.1.22 GATECFG2

The GATECFG2 register has the following structure:

Table 28. GATECFG2 register (voltage mode)

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

The TDT parameter defines the deadtime duration between the gate turn-off and the opposite gate turn-on sequences (*Section 6.16 on page 33*).

Table 29. TDT parameter

TDT [4...0]					Deadtime [ns]
0	0	0	0	0	125
0	0	0	0	1	250
⇓	⇓	⇓	⇓	⇓	⇓
1	1	1	1	0	3875
1	1	1	1	1	4000

The TBLANK parameter defines the duration of the blanking of the current sensing comparators (stall detection and overcurrent) after each commutation (*Section 6.16*).

Table 30. TBLANK parameter

TBLANK [2...0]			Blanking time [ns]
0	0	0	125
0	0	1	250
⇓	⇓	⇓	⇓
1	1	0	875
1	1	1	1000

9.1.23 CONFIG

The CONFIG register has the following structure:

Table 31. CONFIG register

Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8
F_PWM_INT			F_PWM_DEC			VCCVAL	UVLOVAL
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
OC_SD	RESERVED	EN_VSCOMP	SW_MODE	EXT_CLK	OSC_SEL		

The OSC_SEL and EXT_CLK bits set the system clock source:

Table 32. Oscillator management

EXT_CLK	OSC_SEL[2…0]			Clock source	OSCIN	OSCOUT
0	0	0	0	Internal oscillator: 16 MHz	Unused	Unused
0	0	0	1			
0	0	1	0			
0	0	1	1			
1	0	0	0	Internal oscillator: 16 MHz	Unused	Supplies a 2-MHz clock
1	0	0	1	Internal oscillator: 16 MHz	Unused	Supplies a 4-MHz clock
1	0	1	0	Internal oscillator: 16 MHz	Unused	Supplies an 8-MHz clock
1	0	1	1	Internal oscillator: 16 MHz	Unused	Supplies a 16-MHz clock
0	1	0	0	External crystal or resonator: 8 MHz	Crystal/resonator driving	Crystal/resonator driving
0	1	0	1	External crystal or resonator: 16 MHz	Crystal/resonator driving	Crystal/resonator driving
0	1	1	0	External crystal or resonator: 24 MHz	Crystal/resonator driving	Crystal/resonator driving
0	1	1	1	External crystal or resonator: 32 MHz	Crystal/resonator driving	Crystal/resonator driving
1	1	0	0	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 33. External switch hard stop interrupt mode

SW_MODE	Switch mode
0	HardStop interrupt
1	User disposal

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

Table 34. Overcurrent event

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

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

Table 35. Programmable V_{CC} regulator output voltage

VCCVAL	V_{CC} voltage
0	7.5 V
1	15 V

The UVLOVAL bit sets the UVLO protection thresholds.

Table 36. Programmable UVLO thresholds

UVLOVAL	V_{CCthOn}	$V_{CCthOff}$	$\Delta V_{BOOTthOn}$	$\Delta V_{BOOTthOff}$
0	6.9 V	6.3 V	6 V	5.5 V
1	10.4 V	10 V	9.2 V	8.8 V

The EN_VSCOMP bit sets if the motor supply voltage compensation is enabled or not.

Table 37. Motor supply voltage compensation enable

EN_VSCOMP	Motor supply voltage compensation
0	Disabled
1	Enabled

The F_PWM_INT bits set the integer division factor of PWM frequency generation.

Table 38. PWM frequency: integer division factor

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

The F_PWM_DEC bits set the multiplication factor of PWM frequency generation.

Table 39. PWM frequency: multiplication factor

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

From *Table 40* to *Table 43* all available PWM frequencies are listed according to oscillator frequency, F_PWM_INT and F_PWM_DEC values (the CONFIG register OSC_SEL parameter must be correctly programmed).

Table 40. Available PWM frequencies [kHz]: 8-MHz oscillator frequency

	F_PWM_DEC							
F_PWM_INT	**000**	**001**	**010**	**011**	**100**	**101**	**110**	**111**
000	9.8	11.7	13.7	15.6	19.5	23.4	27.3	31.3
001	4.9	5.9	6.8	7.8	9.8	11.7	13.7	15.6
010	3.3	3.9	4.6	5.2	6.5	7.8	9.1	10.4
011	2.4	2.9	3.4	3.9	4.9	5.9	6.8	7.8
100	2.0	2.3	2.7	3.1	3.9	4.7	5.5	6.3
101	1.6	2.0	2.3	2.6	3.3	3.9	4.6	5.2
110	1.4	1.7	2.0	2.2	2.8	3.3	3.9	4.5

Table 41. Available PWM frequencies [kHz]: 16-MHz oscillator frequency

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

Table 42. Available PWM frequencies [kHz]: 24-MHz oscillator frequency

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

Table 43. Available PWM frequencies [kHz]: 32-MHz oscillator frequency

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

Any attempt to write the CONFIG register when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

9.1.24 STATUS

The STATUS register has the following structure:

Table 44. STATUS register

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

When the HiZ flag is high it indicates that the bridges are in high impedance state. 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 events (power-up included).

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

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

The STEP_LOSS_A and STEP_LOSS_B flags are forced low when a stall condition is detected on bridge A or bridge B respectively.

The CMD_ERROR flag is active high and indicates 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).

TH_STATUS bits indicate the current device thermal status (*Section 6.12 on page 30*):

Table 45. STATUS register TH_STATUS bits

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

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

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

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

The DIR bit indicates the current motor direction:

Table 46. STATUS register DIR bit

DIR	Motor direction
1	Forward
0	Reverse

MOT_STATUS indicates the current motor status:

Table 47. STATUS register MOT_STATUS bits

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

Any attempt to write to the register causes the command to be ignored and the CMD_ERROR to rise (*Section 9.1.24*).

9.2 Application commands

The commands summary is given in *Table 48*.

Table 48. Application commands

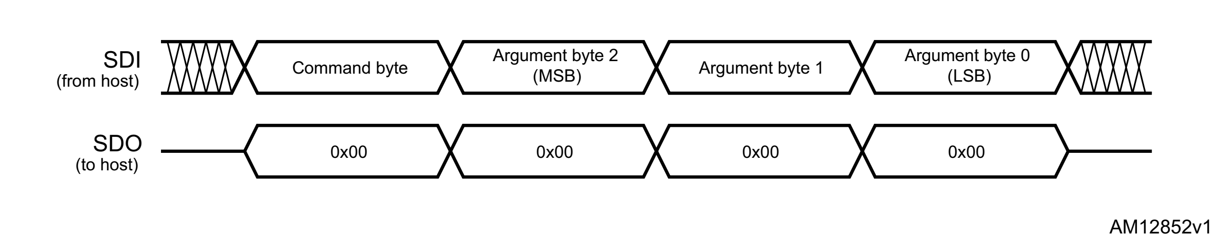
Command mnemonic	Command binary code					Action
	[7…5]	[4]	[3]	[2…1]	[0]	
NOP	000	0	0	00	0	Nothing
SetParam (PARAM, VALUE)	000	[PARAM]				Writes VALUE in PARAM register
GetParam (PARAM)	001	[PARAM]				Returns the stored value in PARAM register
Run (DIR, SPD)	010	1	0	00	DIR	Sets the target speed and the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

9.2.1 Command management

The host microcontroller can control motor motion and configure the L6480 through a complete set of commands.

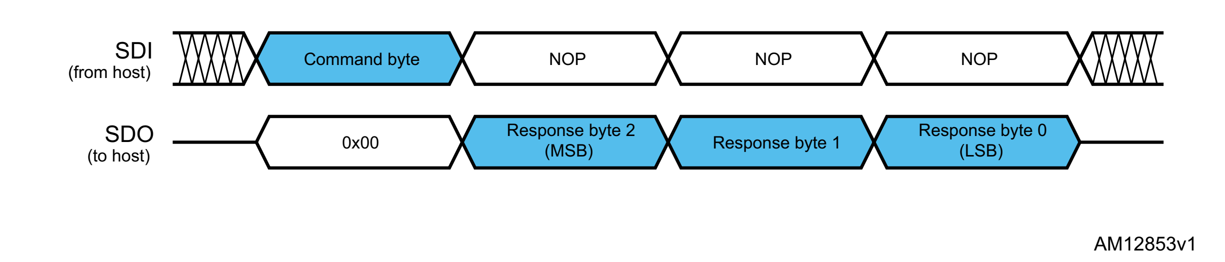
All commands are composed by a single byte. After the command byte, some bytes of arguments should be needed (see *Figure 21*). Argument length can vary from 1 to 3 bytes.

Figure 21. Command with 3-byte argument



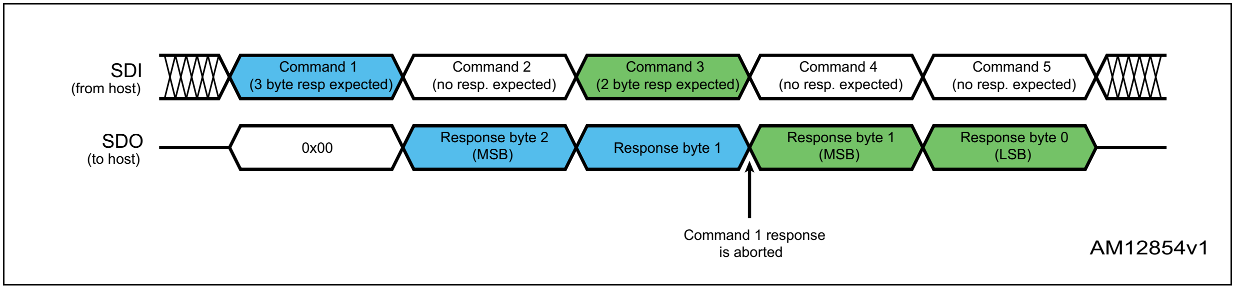
By default, the device returns an all zero response for any received byte, the only exceptions are GetParam and GetStatus commands. When one of these commands is received, the following response bytes represent the related register value (see *Figure 22*). Response length can vary from 1 to 3 bytes.

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 CMD_ERROR flag in the STATUS register is raised (see paragraph *Section 9.1.24 on page 59*).

9.2.2 Nop

Table 49. 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 50. 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 11 on page 43*.

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 11*).

Some registers cannot be written (see *Table 11*); any attempt to write one of those registers causes the command to be ignored and the CMD_ERROR flag to rise at the end of the command byte, as if an unknown command code were sent (see *Section 9.1.24 on page 59*).

Some registers can only be written in particular conditions (see *Table 11*); any attempt to write one of those registers when the conditions are not satisfied causes the command to be ignored and the CMD_ERROR flag to rise at the end of the last argument byte (see *Section 9.1.24*).

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

9.2.4 GetParam (PARAM)

Table 51. GetParam command structure

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

This command reads the current PARAM register value; PARAM is the respective register address listed in *Table 11 on page 43*.

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 11*).

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

All registers can be read anytime.

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

9.2.5 Run (DIR, SPD)

Table 52. Run command structure

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

The Run command produces a motion at SPD speed; the direction is selected by the DIR bit: '1' forward or '0' reverse. The SPD value is expressed in step/tick (format unsigned fixed point 0.28) that is the same format as the SPEED register (*Section 9.1.4 on page 45*).

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 53. 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 (*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 (*Section 6.7.5* and *9.1.24 on page 59*).

The device exits Step-clock mode when a constant speed, absolute positioning or motion command is sent through SPI. Motion direction is imposed by the respective StepClock command argument and can 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.

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

Any attempt to perform a StepClock command when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

9.2.7 Move (DIR, N_STEP)

Table 54. Move command structure

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

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

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

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

Any attempt to perform a move command when the motor is running causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24*).

9.2.8 GoTo (ABS_POS)

Table 55. 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 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.

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

Any attempt to perform a GoTo command when a previous command is under execution (BUSY low) causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

9.2.9 GoTo_DIR (DIR, ABS_POS)

Table 56. GoTo_DIR command structure

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

The GoTo_DIR command produces a motion to ABS_POS absolute position imposing a forward (DIR = '1') or a reverse (DIR = '0') rotation. The ABS_POS value is always 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 be given only when the previous motion command has been completed (BUSY flag released).

Any attempt to perform a GoTo_DIR command when a previous command is under execution (BUSY low) causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24*).

9.2.10 GoUntil (ACT, DIR, SPD)

Table 57. 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 (*Section 6.14 on page 31*), the ABS_POS register is reset (if ACT = '0') or the ABS_POS register value is copied into the MARK register (if ACT = '1'); the system then performs a SoftStop command.

The SPD value is expressed in step/tick (format unsigned fixed point 0.28) that is the same format as the SPEED register (*Section 9.1.4 on page 45*).

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

If the SW_MODE bit of the CONFIG register is set low, the external switch turn-on event causes a HardStop interrupt instead of the SoftStop one (*Section 6.14 on page 31* and *Section 9.1.23 on page 54*).

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 58. 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 59. 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 (*Section 9.2.9*).

The GoHome command keeps the BUSY flag low until the home position is reached. This command can be given only when the previous motion command has been completed. Any attempt to perform a GoHome command when a previous command is under execution (BUSY low) causes the command to be ignored and the CMD_ERROR to rise (*Section 9.1.24 on page 59*).

9.2.13 GoMark

Table 60. GoMark command structure

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

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

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

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

Any attempt to perform a GoMark command when a previous command is under execution (BUSY low) causes the command to be ignored and the CMD_ERROR flag to rise (*Section 9.1.24 on page 59*).

9.2.14 ResetPos

Table 61. ResetPos command structure

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

The ResetPos command resets the ABS_POS register to zero. The zero position is also defined as the HOME position (*Section 6.5 on page 23*).

9.2.15 ResetDevice

Table 62. 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 (*Section 6.1 on page 20*). The command can be performed only when the device is in high impedance state.

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

9.2.16 SoftStop

Table 63. 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 (*Section 9.1.6 on page 46*).

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 64. 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 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 65. 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 (*Section 9.1.6 on page 46*). 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.

9.2.19 HardHiZ

Table 66. 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 67. 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.

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.

HTSSOP38 package information

Figure 24. HTSSOP38 package outline

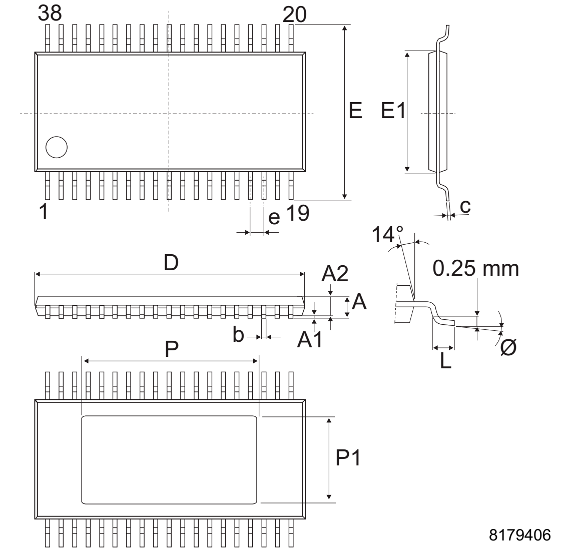
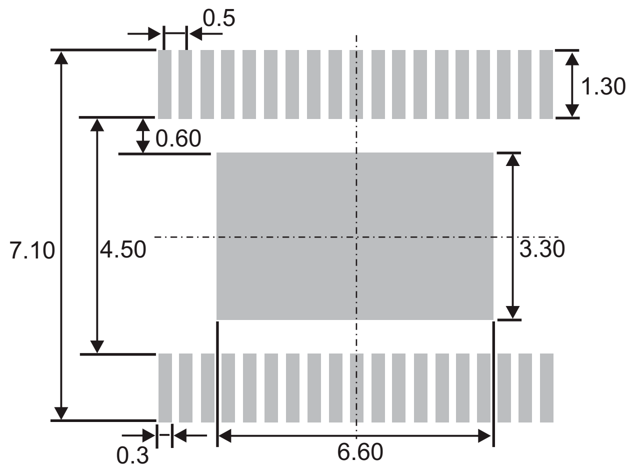


Table 68. HTSSOP38 mechanical data

Symbol	Dimensions (mm)		
	Min.	Typ.	Max.
A		-	1.1
A1	0.05	-	0.15
A2	0.85	0.9	0.95
b	0.17	-	0.27
c	0.09	-	0.20
D	9.60	9.70	9.80
E1	4.30	4.40	4.50
e	-	0.50	-
E	-	6.40	-
L	0.50	0.60	0.70
P	6.40	6.50	6.60
P1	3.10	3.20	3.30
∅	0°	-	8°

Figure 25. HTSSOP38 footprint



11 Revision history

FLAG output

Table 69. Document revision history

Date	Revision	Changes
13-Jun-2012	1	Initial release.
04-Oct-2012	2	Updated *Table 2*, *Table 5*, *Table 6*, *Table 9*, *Table 31*, *Table 36*. Update *Figure 8*, *Figure 9*, *Figure 10*. Minor text changes.
19-Dec-2012	3	Changed the title. Inserted footnote in *Table 2* and in *Table 4*. Updated *Table 18* and *Table 19*.
12-Dec-2013	4	Updated *Section 6.3*, (replaced “integrated MOSFETs” by “gate drivers”). Updated *Section 6.9* to *Section 6.13*, *Section 7.2* (replaced “gates” by “MOSFETs”). Added cross-references to *Section 9*. Updated *Section 9.1.22* (replaced “TCC parameter” by “TDT parameter”). Updated *Section 9.2.15* (Added “The command can be performed only when the device is in high impedance state”). Updated *Section 10* (reversed order of *Figure 24* and *Table 68*, updated titles). Minor modifications throughout document.
19-May-2014	5	Updated *Table 2 on page 10* [added (V_{BOOT} - V_S) to ΔV_{BOOT}]. Updated *Table 5 on page 12* (updated values of $t_{high,STCK}$, $t_{low,STCK}$, and t_{high}). Updated *Table 7 on page 19* (replaced STD25NF10 by STD25N10F7). Updated *Section 6.1 on page 20* (removed V_{CC} and V_{BOOT}, added FLAG output...). Updated *Section 6.4 on page 21* (replaced “the first microstep” by “zero”). Replaced NOTPERF_CMD and WRONG_CMD flag by CMD_ERROR flag throughout document. Updated *Section 9.1.5 on page 45* (replaced sentence: “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.”). Updated *Section 9.1.6 on page 46* (removed sentence: “When the device is working in infinite acceleration mode this value is ignored.”). Updated *Table 44 on page 59* (replaced TH_SD by TH_STATUS). Updated title of *Table 47 on page 61* (replaced MOT_STATE by MOT_STATUS). Updated cross-references throughout document.
05-Mar-2015	6	Updated main title *on page 1* (removed cSPIN™). Updated *Table 11 on page 43* (h15 - Len. [bit]: replaced 10 by 11).
23-Mar-2015	7	Updated *Table 14 on page 47* (“Bit 11” removed). Minor modifications throughout document.