
L648x devices: high power stepper motor controllers

Enrico Poli**Introduction**

The L6480 and L6482 devices are motor controllers providing a flexible solution for the high power stepper motor applications. The devices can be controlled by a host microcontroller through a fast SPI interface and are able to execute a complete set of motion commands.

This document describes how the devices can be configured and gives some suggestions on the operation and application design.

The current control algorithm of the devices - voltage mode driving for the L6480 and the advanced current control for the L6482 device - will be not investigated in this document.

Please refer to the respective application notes AN4144 "Voltage mode control operation and parameter optimization" and AN4158 "Peak current control with automatic decay adjustment and predictive current control: basics and setup" for more details about them.

Contents

1	The L6480 and L6482 communication interface	3
1.1	Communication protocol	3
1.2	Daisy chain	3
1.3	5 V and 3.3 V communication interface	5
2	Motion engine control	6
2.1	Speed tracking commands	7
2.2	Positioning commands	8
	Change the target position of the on-the-fly command	9
2.3	Stop commands	10
2.4	Initializing position using GoUntil and ReleaseSW commands	10
3	Supply management	12
4	Gate driving circuitry	14
4.1	Turn-on sequence	16
4.2	Turn-off sequence	16
5	Protections	18
5.1	Overtemperature protection	18
5.2	Overcurrent protection	18
5.3	Gate drivers undervoltage	19
5.4	V_S undervoltage	20
6	Stall detection (L6480 only)	21
7	Main clock source	22
8	Layout suggestions	23
9	Revision history	24

1 The L6480 and L6482 communication interface

The device (always slave) can be driven by an MCU (always master) sending commands through an 8-bit SPI interface. The 8-bit shift register of the device is kept enabled while the \overline{CS} input is forced low. During this time, at every raising edge of the serial clock (CK), the SDI input is stored into the shift register. At CK falling edges the SDO output is updated according to the last bit of the shift register.

When the \overline{CS} input is raised, the device catches the shift register content and interprets its value as a command or an argument of the previously received command.

All the bytes are sent through the SPI data lines starting from the most significant bit.

1.1 Communication protocol

The communication protocol is based on single byte commands that can be followed by a command argument up to 3 byte long which must be transmitted starting from the most significant byte.

Part of the information needed to execute the target operation could be embedded into the command byte, for example the target register address in the GetParam and SetParam commands, and the argument provides extra data as well as the target position of the GoTo command.

By default the response byte of the device is h00 (hexadecimal format). Some commands, for example those used to read the value of a register, generate a response from the device up to 3 byte long which is transmitted during the following transmission cycles starting from the most significant byte.

1.2 Daisy chain

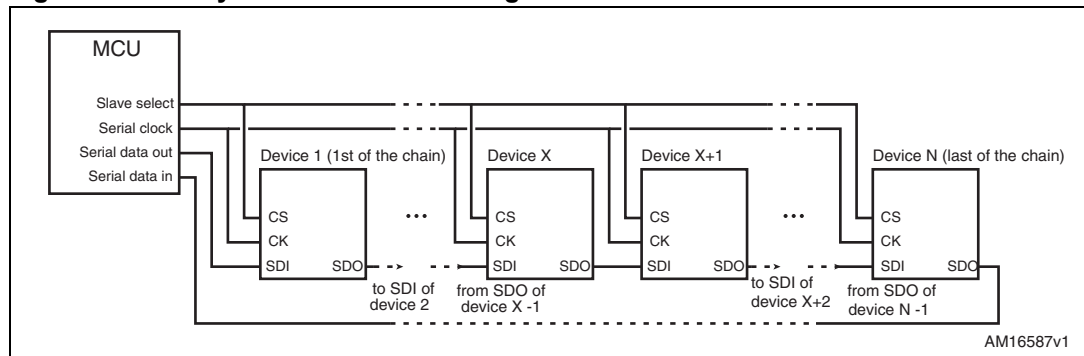
The device is compatible with the daisy chain architecture allowing the MCU to drive multiple devices with a single SPI interface.

The daisy chain architecture is obtained as follows:

- Master serial clock line is connected to the CK input of each slave device.
- Master slave select line is connected to the \overline{CS} input of each slave device.
- Master serial data output (MOSI) is connected to the SDI input of the first slave of the chain.
- The SDO output of each slave device is connected to the SDI input of the next one, last slave excluded.
- Master serial data input (MISO) is connected to the SDO output of the last slave of the chain.

The connection diagram of the configuration is shown in [Figure 1](#).

Figure 1. Daisy chain connection diagram



In this configuration, the chain of slaves acts as a single slave with an SPI device of N byte. Each communication cycle, for example when the master needs to transmit/receive a byte from/to a slave, the master must fill all the shift registers of the slaves before raising the CS line.

The devices are addressed according to the position of the byte in the communication cycle: the first byte transmitted by the master is received by the last device of the chain; the second one is received by the last-but-one slave and so on down to the last transmitted byte which is received by the first slave of the chain. The response bytes from the device chain are addressed to the same way: the first byte received by the master has been transmitted by the last device of the chain; the second one has been transmitted by the last-but-one slave and so on down to the last received byte which has been transmitted by the first slave of the chain.

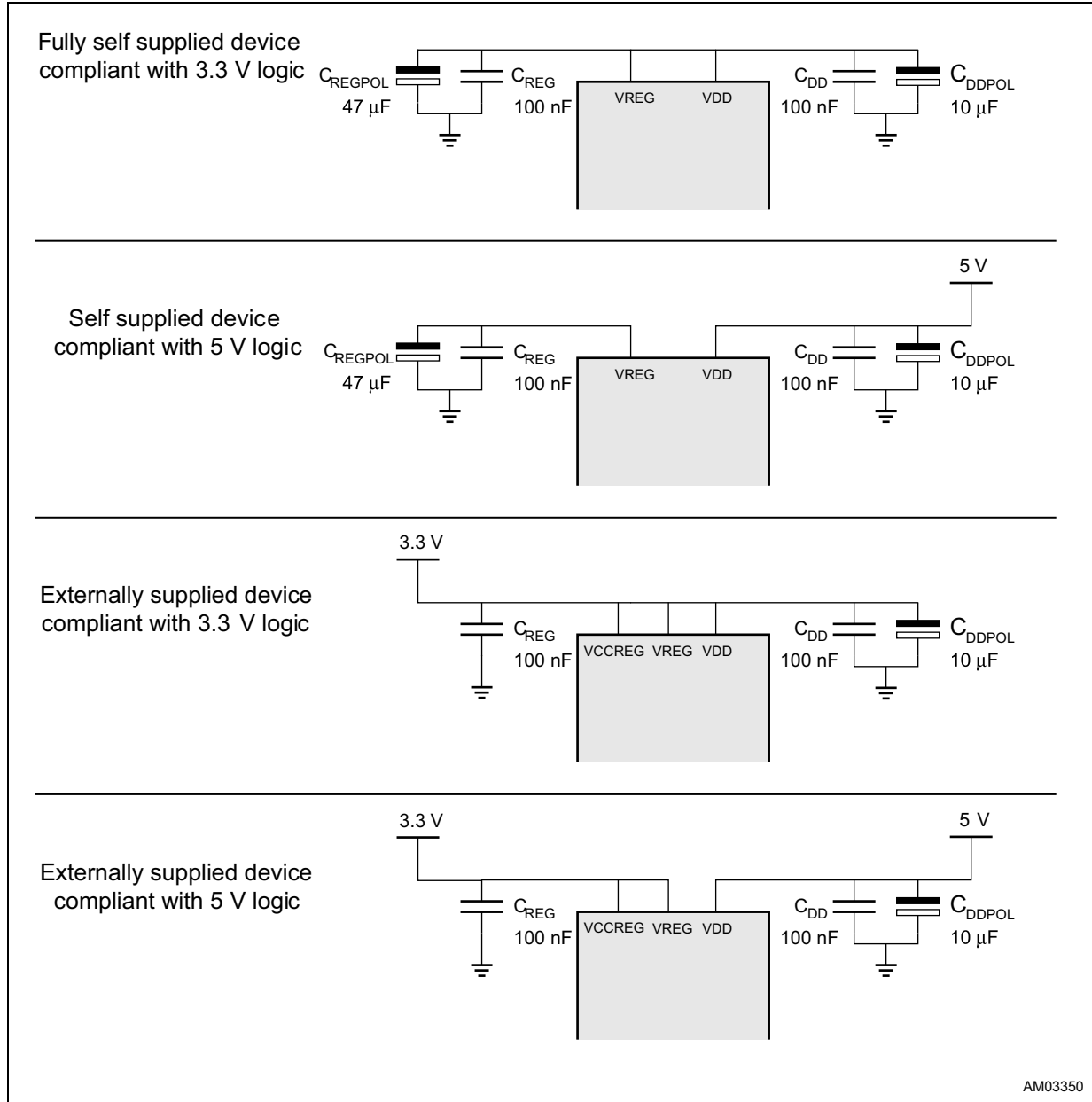
In theory, the number of slaves that an MCU can drive using the daisy chain configuration is unlimited; in practice the maximum number of devices connected to the same SPI depends on the clock skew.

The number of slaves limits the communication speed also because every time a byte has to be transmitted to a device, the whole N slave chain has to be filled transmitting N - 1 extra bytes.

1.3 5 V and 3.3 V communication interface

The device can be configured to operate both with 3.3 V or 5 V standard logic as shown in [Figure 2](#).

Figure 2. Logic interface supply scenarios



2 Motion engine control

The L648x devices integrate a motion engine providing a full set of commands. The motion engine generates the step sequence according to the programmed speed profile and the requested command.

The speed profile represents the operation boundaries, defined by the acceleration, deceleration, maximum and minimum speed, which should be respected to ensure the proper functioning of the application.

The devices allow all the parameters to be set independently:

- Acceleration and deceleration values range from 14.55 up to 59590 steps/s². The device can also be set to use an infinite acceleration and deceleration value; in this case, both the acceleration and deceleration phases are totally skipped.
- Maximum speed value ranges from 15.25 steps/s up to 15610 step/s.
- Minimum speed value ranges from 0 up to 976 steps/s.

The acceleration, deceleration and minimum speed parameters can be modified when the motor is stopped only. The maximum speed can be also changed when the motor is running, but the new value is only considered at next command execution.

The commands supported by the motion engine are listed in [Table 1](#).

Table 1. Command list

Name	Length (bytes)	Description	Notes
Move	4 (including 3 of arguments)	Performing the target number of microsteps as per requested direction.	Can be executed when the motor is stopped only.
GoTo	4 (including 3 of arguments)	Reaching the absolute target position (ABS_POS register) using the shortest path.	Not accepted while another command is under execution.
GoTo_DIR	4 (including 3 of arguments)	Reaching the absolute target position (ABS_POS register) running as per requested direction.	Not accepted while another command is under execution.
GoHome	1	Reaching the home position (all zeroes) using the shortest path.	Not accepted while another command is under execution.
GoMark	1	Reaching the position stored into the MARK register using the shortest path.	Not accepted while another command is under execution.
Run	4 (including 3 of arguments)	Reaching the target speed in the requested direction.	Always accepted and immediately executed (if present, the previous command is aborted)
StepClock	1	Switching the device in step-clock mode imposing the direction.	Can be executed when the motor is stopped only.
GoUntil	4 (including 3 of arguments)	Reaching the speed target in the requested direction and stopping when SW input is forced low (falling edge).	Always accepted and immediately executed (if present, the previous command is aborted)
ReleaseSW	1	Running the motor at low speed in the requested direction and stopping when SW input is forced high (rising edge).	Always accepted and immediately executed (if present, the previous command is aborted)

Table 1. Command list (continued)

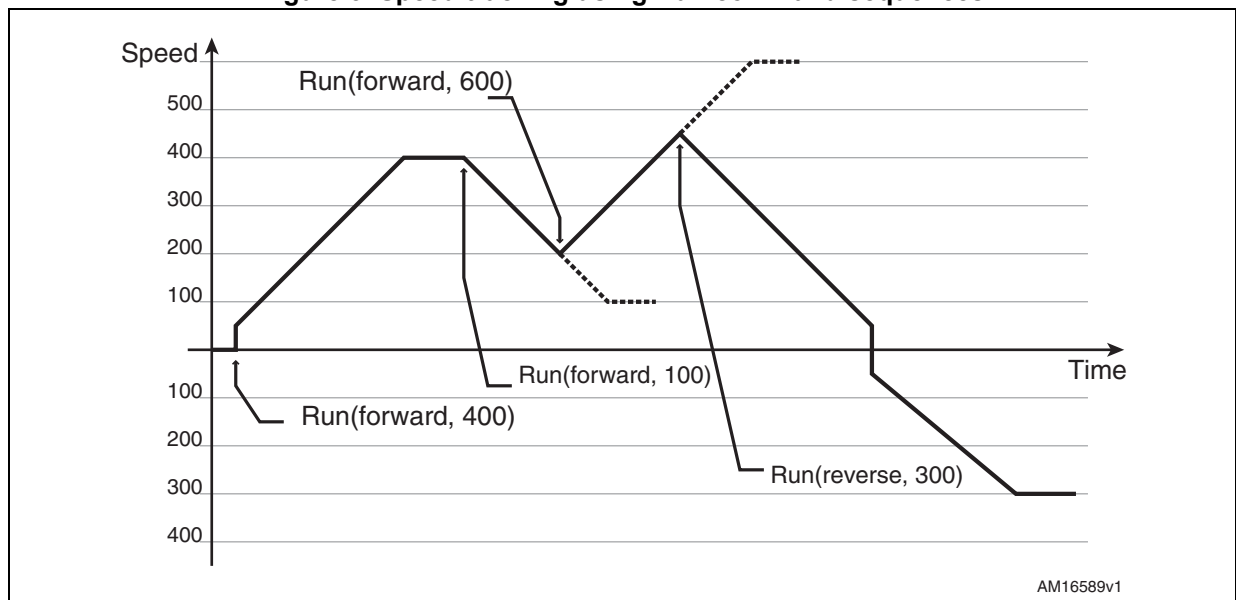
Name	Length (bytes)	Description	Notes
SoftStop	1	Stopping the motor in accordance to the programmed speed profile.	Always accepted and immediately executed (if present, the previous command is aborted)
HardStop	1	Stopping the motor immediately (infinite deceleration).	Always accepted and immediately executed (if present, the previous command is aborted)
SoftHiZ	1	Stopping the motor in accordance to the programmed speed profile and then disabling the power bridges.	Always accepted and immediately executed (if present, the previous command is aborted)
HardHiZ	1	Disabling the power bridges immediately.	Always accepted and immediately executed (if present, the previous command is aborted)

2.1 Speed tracking commands

During the speed tracking, the device dynamically changes the motor speed according to the application requirements. The Run command can be used to achieve this result.

The Run command sets the speed target and direction which the motor has to reach. Both speed target and direction can be changed in real time through a new Run command (see [Figure 3](#)).

Figure 3. Speed tracking using Run command sequences



2.2 Positioning commands

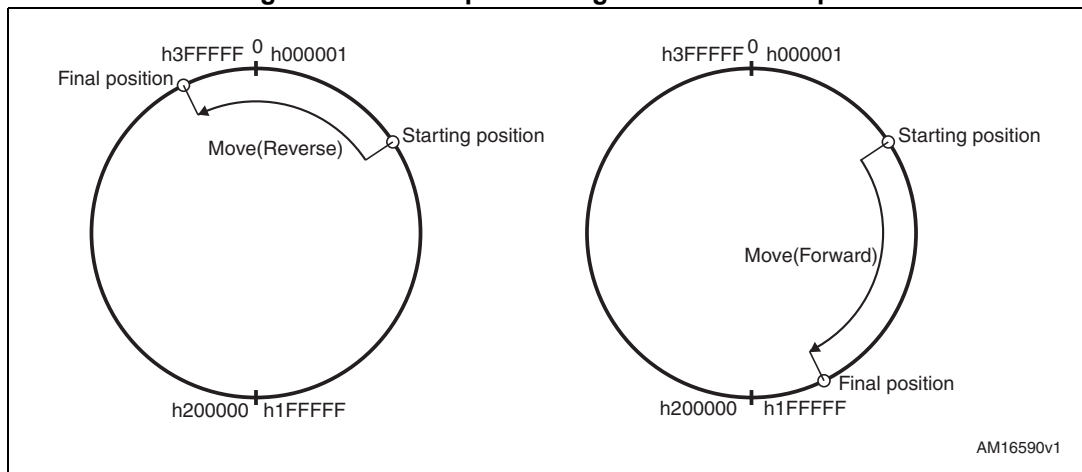
The motion engine, integrated into the devices, allows the position of the motor in a target position based on integrated ABS_POS register.

The ABS_POS register traces all the motion performed by the motor adding a unit to each microstep completed in forward direction and subtracting a unit when the microstep is performed in reverse direction.

The target positioning can be directly imposed indicating the ABS_POS register value (absolute position) or the distance between the current position and the target one (relative position).

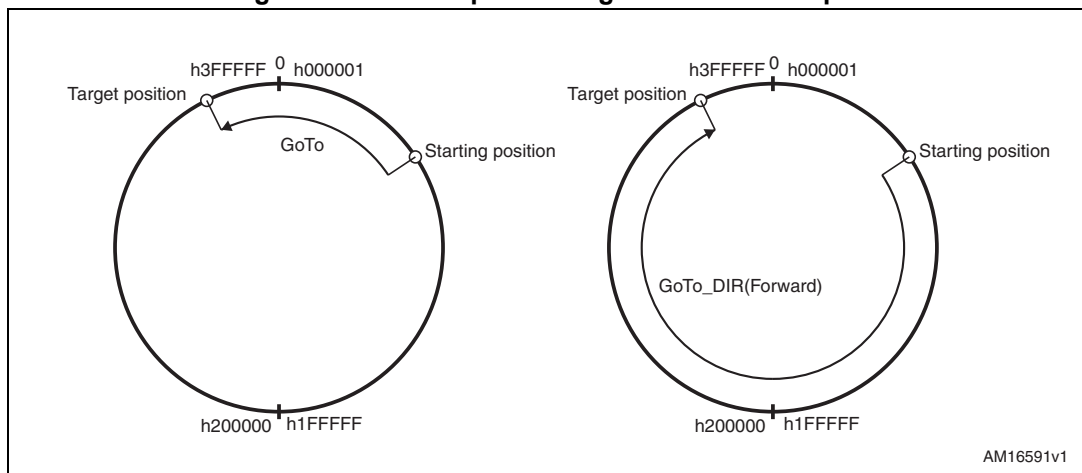
The relative positioning command is Move. The motion engine executes this command only when the motor is stopped in order to avoid unexpected behaviors (e.g.: the target of number of steps is not enough to allow the speed profile compliance).

Figure 4. Relative positioning command example



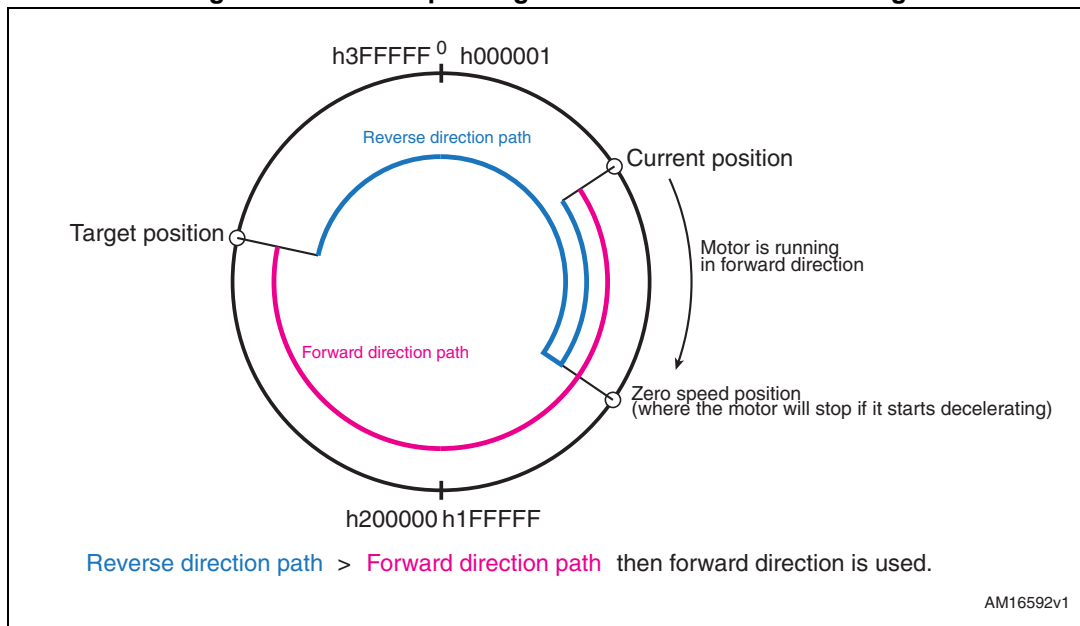
The absolute positioning commands are GoTo and GoTo_DIR. The former moves the motor to the position target choosing the rotation direction according to a minimum path algorithm (i.e.: the lower number of microsteps is executed) whereas the latter imposes the direction directly.

Figure 5. Absolute positioning command example



If a GoTo command is requested when the motor is running, the minimum path algorithm also considers the steps required to reverse the direction (see [Figure 6](#)).

Figure 6. Minimum path algorithm when motor is running



The GoTo and GoTo_DIR can only be executed when no other commands are under execution.

Change the target position of the on-the-fly command

If a Run command is sent to the device during the execution of a GoTo command, it aborts the previous command. This effect can be used to change the target position of the motion engine on-the-fly.

Following the suggested sequence of operations:

1. Read the current motor speed (SPEED register) and direction (DIR bit in the STATUS register).
2. Send a Run command setting the target speed and direction equal to the values obtained at point 1.
3. Wait for the execution command monitoring the BUSY pin or the BUSY flag in the STATUS register.
4. Send the new positioning command.

This operation could introduce a small error in the generation of the speed profile. If the target position is changed a high number of times (tracking position) the error increases and anomalous behaviors could occur.

2.3 Stop commands

The motor can be stopped through the stop commands. These commands can be sent at any time and they are executed immediately.

The SoftStop command stops the motor fitting the deceleration value of the speed profile whereas HardStop command stops the motor immediately (infinite deceleration).

The SoftHiZ and HardHiZ commands operate similarly, but the power bridges are disabled as soon as the zero speed is reached (the high impedance status is forced).

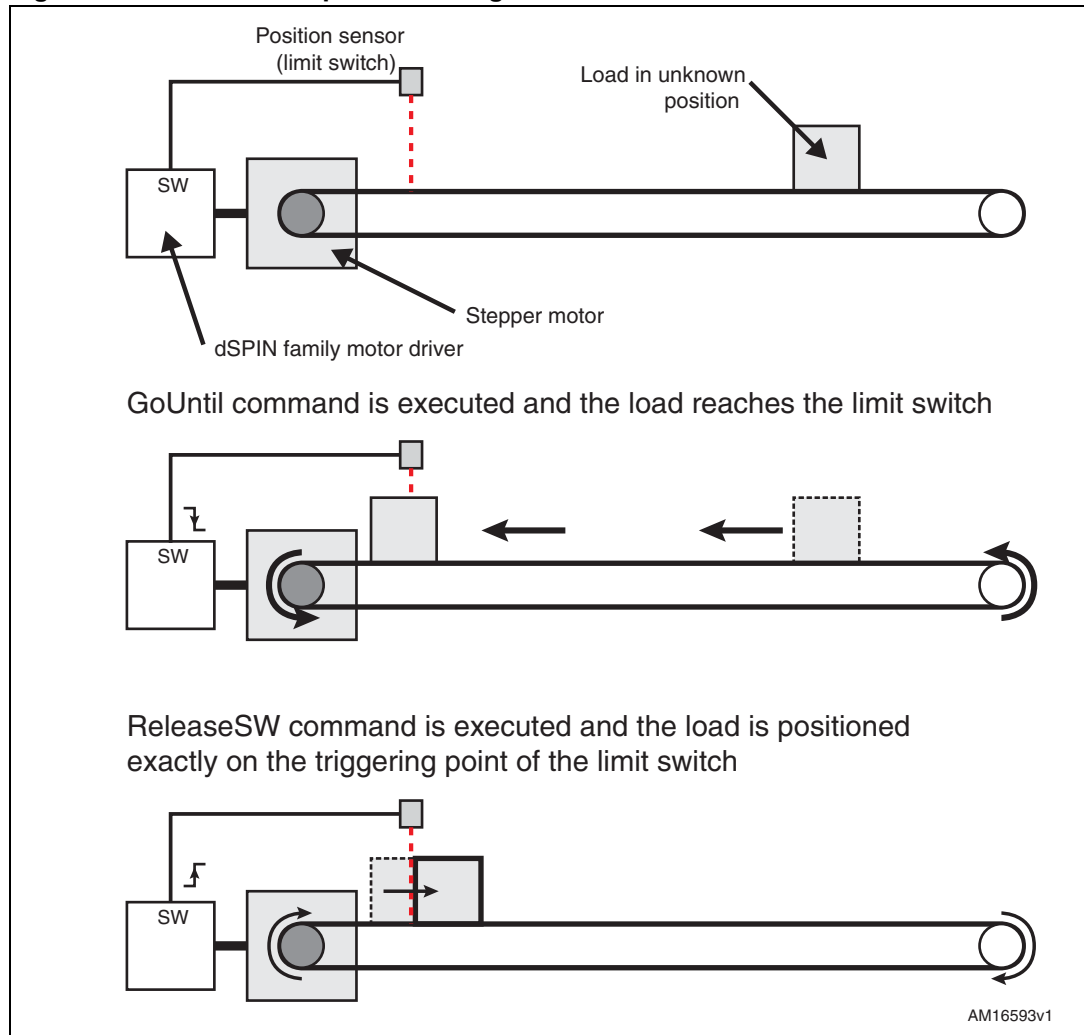
2.4 Initializing position using GoUntil and ReleaseSW commands

The Information stored into the ABS_POS register can be initialized according to an external position sensor. In this way a relation between ABS_POS value and mechanical position of the motor is established.

The position initialization sequence, as shown in [Figure 7](#), is the following:

1. In power-up status, the load position is unknown.
2. Using GoUntil command the load is moved to the limit switch at high speed.
3. When the load reaches the limit switch the SW input of the device is forced low. The motor decelerates and then stops.
4. Considering the high speed used to approach the load to the limit switch, a significant error in the positioning could happen.
5. Using ReleaseSW command the load is moved away from the limit switch at low speed.
6. As soon as the threshold position of the limit switch is crossed by the load the SW input of the device is forced high. The motor stops immediately.
7. The load is positioned in correspondence to the threshold position of the limit switch with high precision.

Figure 7. Initialization position using GoUntil and ReleaseSW commands



3 Supply management

The L648x devices integrate two linear voltage regulators which can be used for the generation of the supply voltages of the gate drivers and the control logic. This allows the device to operate both using a single supply source (the motor supply) and using separated supply sources for gate driving and logic parts.

The supply pins of the device are listed in [Table 2](#).

Table 2. Device supply pins

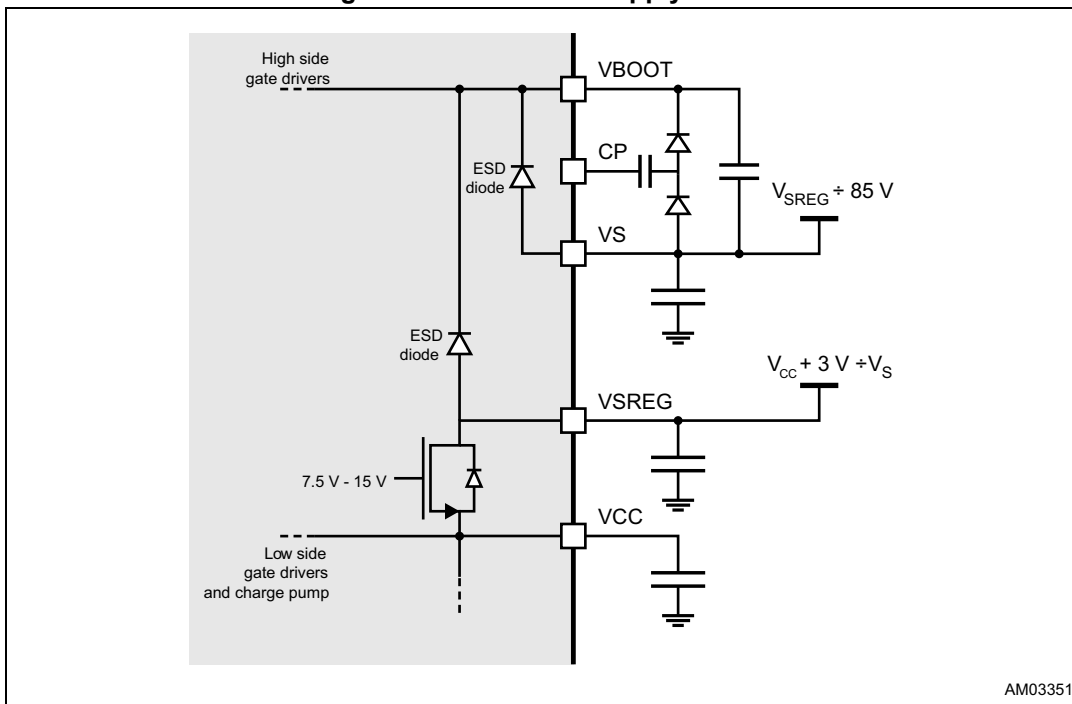
Pin	Voltage range (operative)	Description
VS	From V_{SREG} to 85 V	Motor supply voltage. It is used by the device only as a reference voltage for internal circuitry.
VBOOT	From V_S to $V_S + V_{CC}$	High-side gate drivers supply voltage. It is generated through a charge pump circuit. The charge pump is supplied by the VCC pin.
VSREG	From $V_{CC} + 3\text{ V}$ to V_S or shorted to VCC (VCC directly supplied)	Input voltage of the VCC linear voltage regulator.
VCC	From 7.5 V to 15 V	Output voltage of the VCC linear voltage regulator and low-side gate drivers supply voltage. It is also the supply voltage of the charge pump circuit generating the V_{BOOT} voltage.
VCCREG	From 6.3 V to V_{CC} or shorted to VREG (VREG directly supplied)	Input voltage of the VREG linear voltage regulator.
VREG	3.3 V	Output voltage of the VREG linear voltage regulator and logic supply voltage.
VDD	3.3 V or 5 V	Digital outputs supply voltage.

The supply circuit for the gate drivers is shown in [Figure 8](#). When the V_{CC} voltage is generated through the integrated linear regulator the VSREG pin must be connected to a supply voltage ranging from $V_{CC} + 3\text{ V}$ (3 V is the dropout voltage of the linear regulator) and the motor supply voltage V_S . V_{SREG} must be always lower than V_{BOOT} otherwise the ESD protection diode is turned on and the device can be damaged.

The device can be protected from this event adding an external low drop diode between the VSREG and VS pins, charge pump diodes should be low drop too.

In order to supply directly the gate driving circuitry the external supply voltage must be connected to both the VCC and VSREG pins; this way the internal linear regulator is disabled.

Figure 8. Gate drivers supply circuit



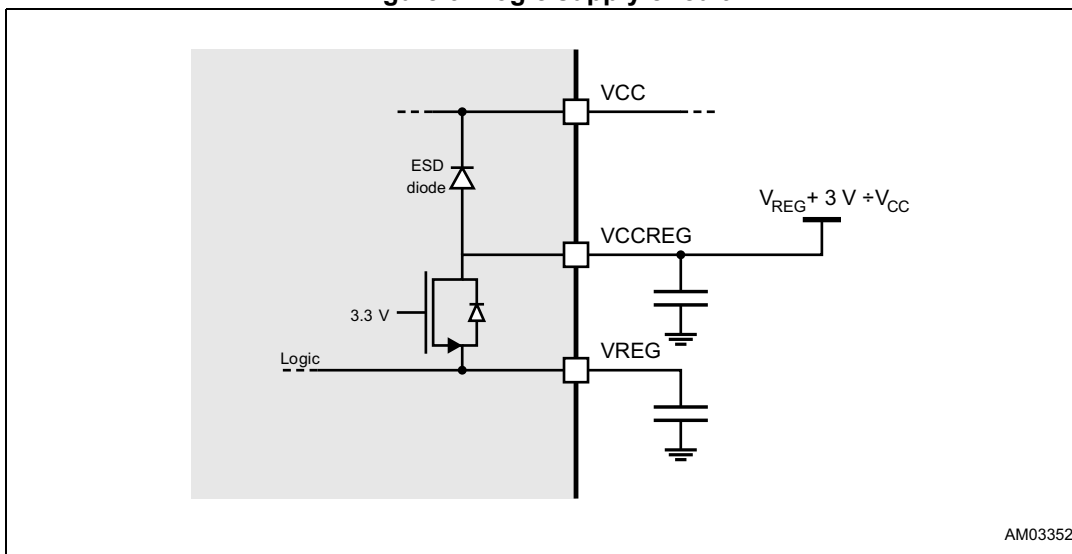
AM03351

The supply circuit for the device logic is shown in [Figure 9](#). When the V_{REG} voltage is generated through the integrated linear regulator the V_{CCREG} pin must be connected to a supply voltage ranging from $V_{REG} + 3\text{ V} = 6.3\text{ V}$ (3 V is the dropout voltage of the linear regulator) and V_{CC} . V_{CCREG} must be always lower than V_{CC} otherwise the ESD protection diode is turned on and the device can be damaged.

The device can be protected from this event adding an external low drop diode between the V_{CCREG} and V_{CC} pins.

In order to supply directly the logic the external supply voltage must be connected to both the V_{REG} and V_{CCREG} pins; this way the internal linear regulator is disabled.

Figure 9. Logic supply circuit



AM03352

4 Gate driving circuitry

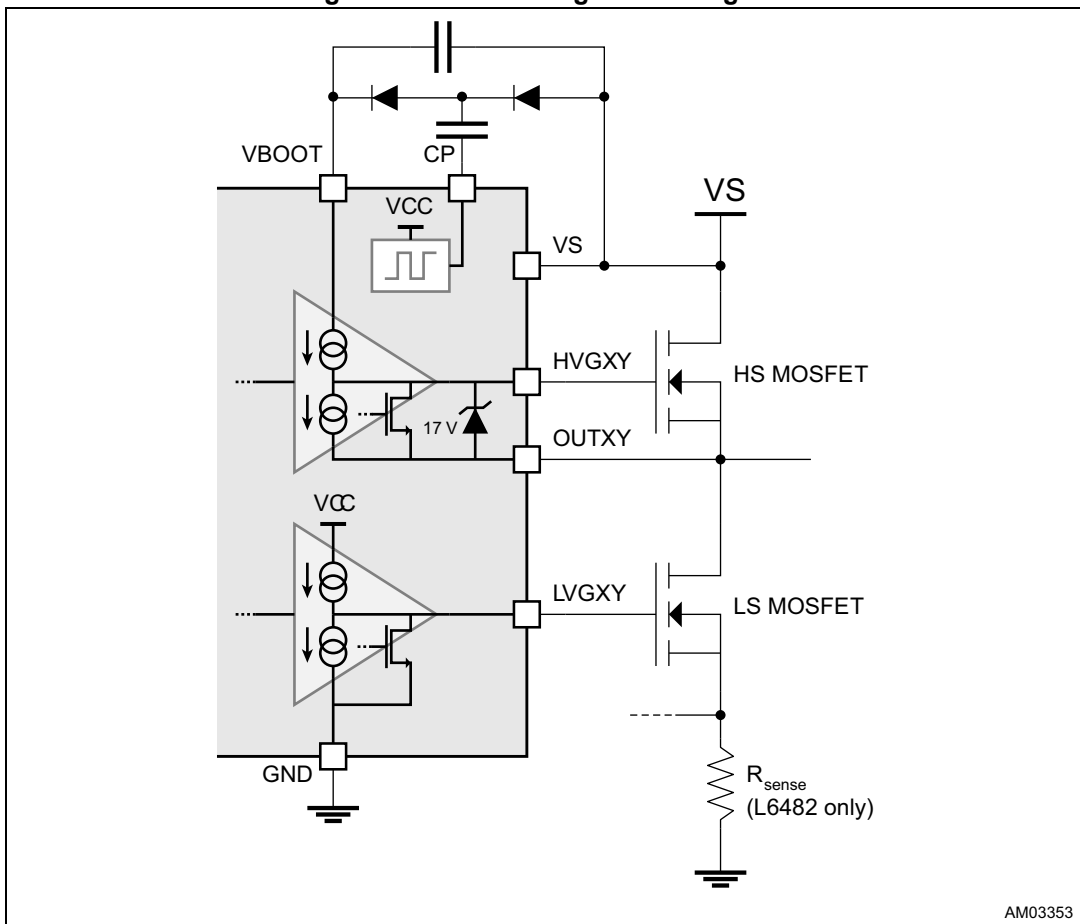
The L648x devices integrate a programmable gate driving circuitry which allows to drive a wide range of external N-channel MOSFETs.

The parameters which can be set are listed in [Table 3](#).

Table 3. Gate driving circuit parameters

Parameter	Description
Gate source/sink current	The current used to charge/discharge the gate of the MOSFETs.
Controlled current time (charging time)	The charging/discharging time of the gate of the MOSFETs.
Turn-off current boost time	The part of the discharging time during which the gate current is forced to the maximum value.
Deadtime	The time between the turn-off of a MOSFET and the turn-on of the opposite one.
Blanking time	The time after the MOSFETs commutation where the sensing circuitry is disabled in order to avoid spurious triggering.

Figure 10. Gate driving circuit diagram



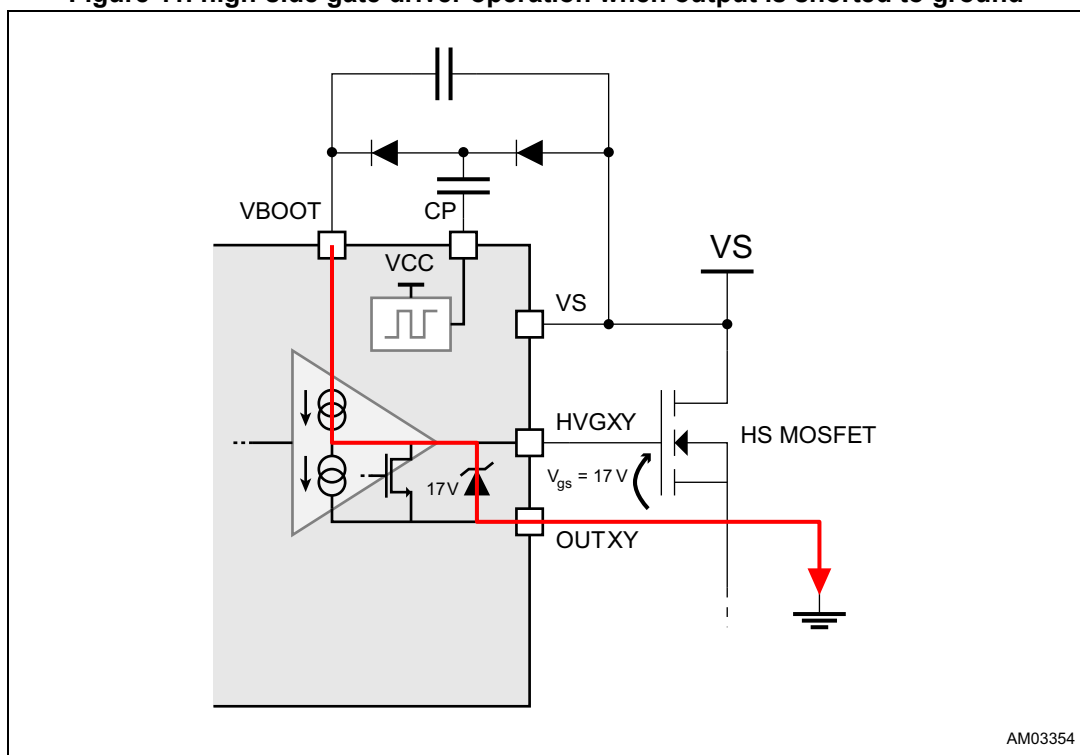
AM03353

Both the high-side and low-side gate drivers are programmable current generators which allow to charge and discharge the MOSFETs gate with a constant current. The gate drivers also integrate a Miller clamp MOSFET which avoids induced turn-on effect.

The low-side gate driver is supplied by the VCC pin which can be connected both to an external voltage source and to the integrated linear regulator which generates the supply voltage (7.5 V or 15 V according to the configuration) from the VSREG supply pin.

The high-side gate driver is supplied by charge pump circuitry. This way the on time of the high-side MOSFET is not limited as in the case of a bootstrap capacitor. A clamping diode limits the gate-source voltage of the high-side MOSFET to 17 V avoiding the breaking of the gate in case of the power stage output is short-circuited to ground. In fact in this case, if the voltage limiter was not present, the final gate-source voltage of the high-side MOSFET was equal to V_{BOOT} (see [Figure 11](#)).

Figure 11. high-side gate driver operation when output is shorted to ground

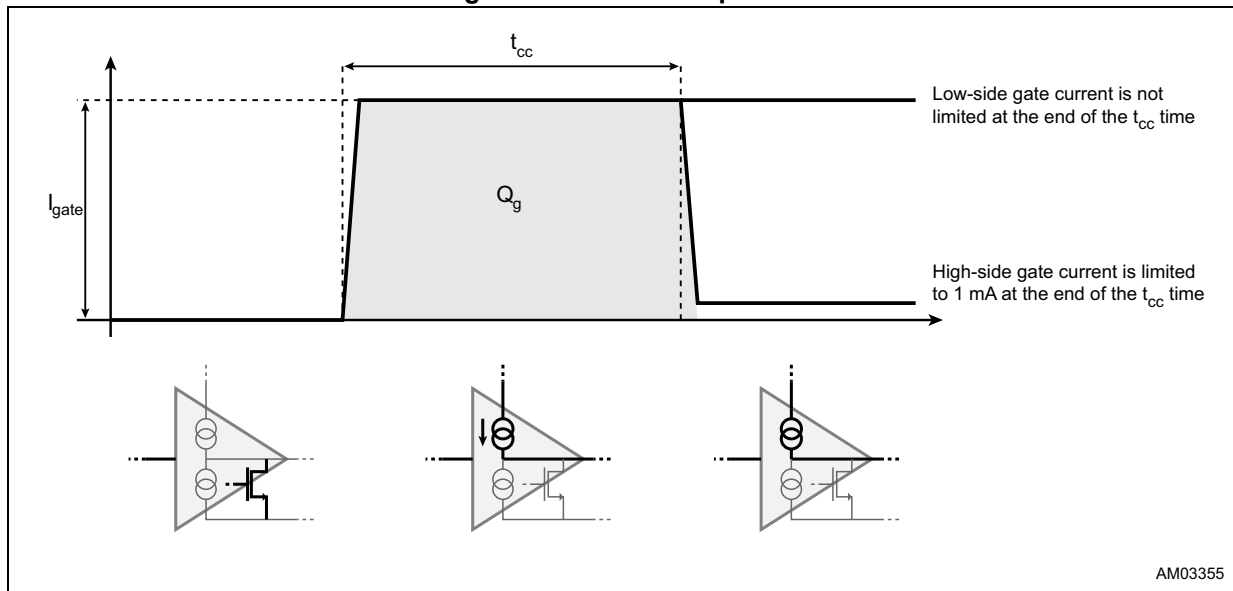


Warning: The clamping diode protects the MOSFET from the V_{gs} breakdown, but not by the high current which can be generated by the short-circuit condition.

4.1 Turn-on sequence

When one of the external MOSFET must be turned-on the gate driving circuitry disables the respective Miller clamp switch and starts forcing the programmed current into the gate. After the controlled current time expires, the high-side and the low-side drivers act differently: the low-side drivers do not limit the gate current; the high-side ones instead limit the gate current to about 1 mA.

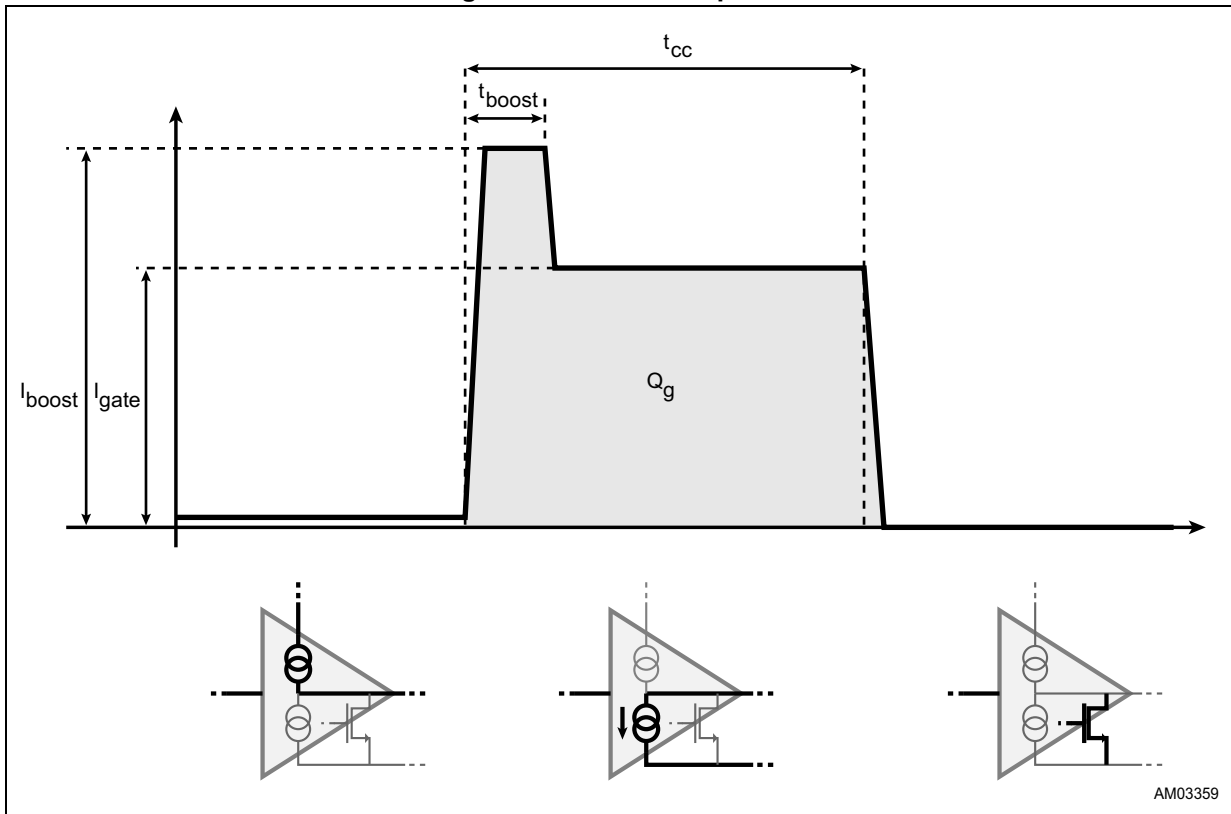
Figure 12. Turn-on sequence



4.2 Turn-off sequence

When one of the external MOSFET must be turned-off the gate driving circuitry starts sinking the programmed current from the gate. At the beginning of the controlled current time the gate current is set to the maximum for t_{boost} time. After the controlled current time expires, the Miller clamp switch is turned on.

Figure 13. Turn-off sequence



AM03359

5 Protections

The L648x devices provide a complete set of protections designed to prevent damaging the device in critical conditions.

The implemented protections are:

- Overtemperature (see [Section 5.1](#))
- Overcurrent (see [Section 5.2](#))
- Gate drivers undervoltage (see [Section 5.3](#))
- V_S undervoltage (see [Section 5.4](#))

5.1 Overtemperature protection

The overtemperature protection disables the power stage of the device when the temperature of the chip exceeds the safe operation conditions.

The device provides a 3-levels protection: when the temperature exceeds the warning threshold an alarm condition is set, but the device operation is still guaranteed. If the temperature further increases, the output shutdown threshold is triggered and the gate drivers are disabled. The device is kept in this condition until the junction temperature decreases below the release threshold. A second protection is triggered if the device temperature continues increasing even when the gate drivers are disabled. This protection turns off the internal voltage regulators and it is released only when the device temperature falls below the warning threshold.

5.2 Overcurrent protection

The overcurrent protection monitors the drain-source voltage drop of all the external power MOSFETs and disables the power stage when the programmed threshold is reached. No information about the specific MOSFET or bridge causing the failure is available.

As soon as the overcurrent protection is triggered the device is locked in a safe condition (all MOSFETs are turned off) and is kept in this condition until the OCD failure flag is released by a `GetStatus` command (see the Section 6.9 in the L6480 and L6482 datasheet for details).

When the device is locked in safe state no commands which enable the bridges can be executed (e.g.: `Move`, `Run`, `GoTo`, `HardStop`, etc.). The commands are simply ignored, no non-performable signaling is returned by the device.

Warning: The overcurrent protection can be disabled setting the `OC_SD` pin of the `CONFIG` register to zero. However it is not recommended to disable this protection.

The relation between the voltage threshold which is programmed into the device register and the actual load current is defined by [Equation 1](#):

Equation 1

$$I_{OCDth} = V_{OCDth} \times R_{ds(ON)}$$

Where $R_{ds(ON)}$ is the ON resistance of the external MOSFET. Considering that the on resistance value varies with temperature and gate voltage, the V_{CC} value and the operative temperature of the power stage must be considered when the threshold value is defined.

5.3 Gate drivers undervoltage

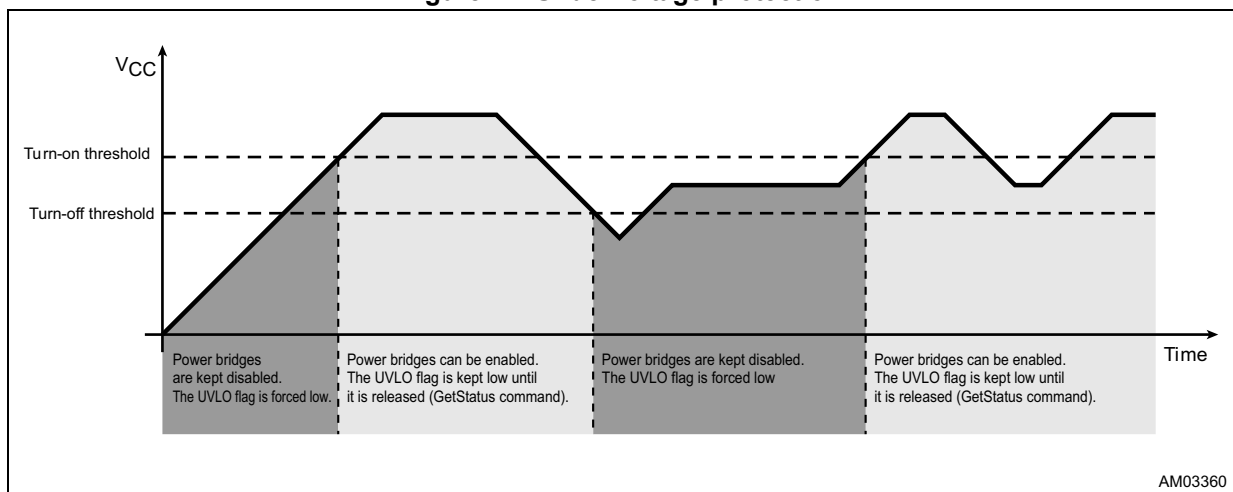
The undervoltage protection avoids the gate driving circuitry of the device to operate with a supply voltage below a programmed value. This way a minimum gate-source voltage for the external MOSFET is guaranteed.

At power-up the device is in undervoltage status: the power stage is kept disabled until the V_{CC} voltage is below the turn-on threshold. In this condition all the commands enabling the bridges (e.g.: Move, Run, GoTo, HardStop, etc.) are ignored and the UVLO failure flag is forced low.

When the turn-on threshold is reached the power stage is operative. The UVLO flag is kept low until it is released through a GetStatus command.

The device returns in undervoltage status if the supply voltage falls below the turn-off threshold.

Figure 14. Undervoltage protection



Two possible threshold values can be set through the UVLOVAL bit in the CONFIG register.

5.4 V_S undervoltage

The device also provides an undervoltage detection feature for the power stage supply (V_S) through the integrated ADC. Connecting the ADCIN pin to V_S through a voltage divider the motor supply voltage is constantly monitored.

When the ADC input voltage falls below 1.16 V (typical), the protection is triggered and the UVLO_ADC failure flag is asserted. The failure flag can be released through a GetStatus command after the ADC input voltage returns above the threshold value. The protection does not affect the device operation.

If the voltage divider is sized in order to obtain $V_{REG}/2$ voltage at the ADCIN pin when V_S is at its nominal, the V_S undervoltage threshold corresponds to a reduction of the 30% of the nominal supply voltage.

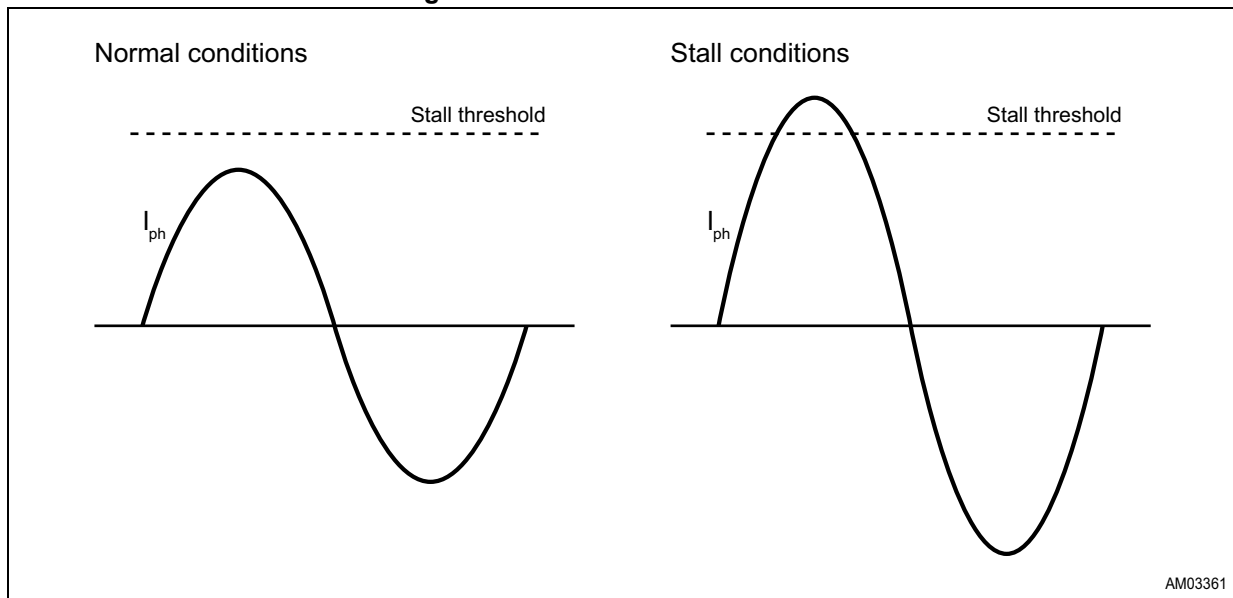
6 Stall detection (L6480 only)

The L6480 device also includes a sensorless stall detection system. This feature allows the device to detect the stall condition of the motor measuring the increase of the phase current caused by the sudden cancellation of the back electromotive force.

The voltage mode control applies a sinusoidal voltage to the motor phases obtaining a sinusoidal current. The amplitude of the voltage sine wave is increased with speed in order to compensate the back electromotive force effect and keep the amplitude of the phase current constant. If the motor stalls (i.e.: stops its rotation) the back electromotive force becomes null, but the voltage applied to the phase is still increased to face its effect. In this condition the phase current results higher than the nominal one and the stall condition can be easily detected. The load current is measured monitoring the drain-source voltage drop of the low-side external power MOSFETs.

The stall detection threshold must be set above the nominal peak current.

Figure 15. Stall detection threshold



The relation between the voltage threshold which is programmed into the device register and the actual load current is defined by the following formula:

Equation 2

$$I_{\text{STALLth}} = V_{\text{STALLth}} \times R_{\text{ds(ON)}}$$

Where $R_{\text{ds(ON)}}$ is the ON resistance of the external MOSFET. Considering that the on resistance value varies with temperature and gate voltage, the V_{CC} value and the operative temperature of the power stage must be considered when the threshold value is defined.

7 Main clock source

The motion engine integrated into the L648x generates the internal step clock starting from the main clock source of the system. As a consequence, the perturbations on the main clock affect the accuracy with which the device generates the speed of the motor (step rate).

The main clock also affects the following parameters:

- Gate driving timings (controlled current time, boost time, deadtime and blanking time).
- Frequency of the PWM modulators (L6480).
- On time measurement and current control timings (L6482).

The devices can operate with both external and internal clock sources. By default the clock source is the integrated oscillator operating at 16 MHz. This solution, if compared to a high precision external source, allows the maximum integration with lower precision of the generation of the speed profile (acceleration, deceleration and target speed). The effects on the current control systems (voltage mode or advanced current control) are negligible.

The device can be configured to use an external clock source or to drive a crystal/resonator. In both cases the clock frequency must be chosen between four values: 8, 16, 24 or 32 MHz.

Warning: Setting the wrong value or applying a clock frequency significantly different from the nominal value might cause failures.

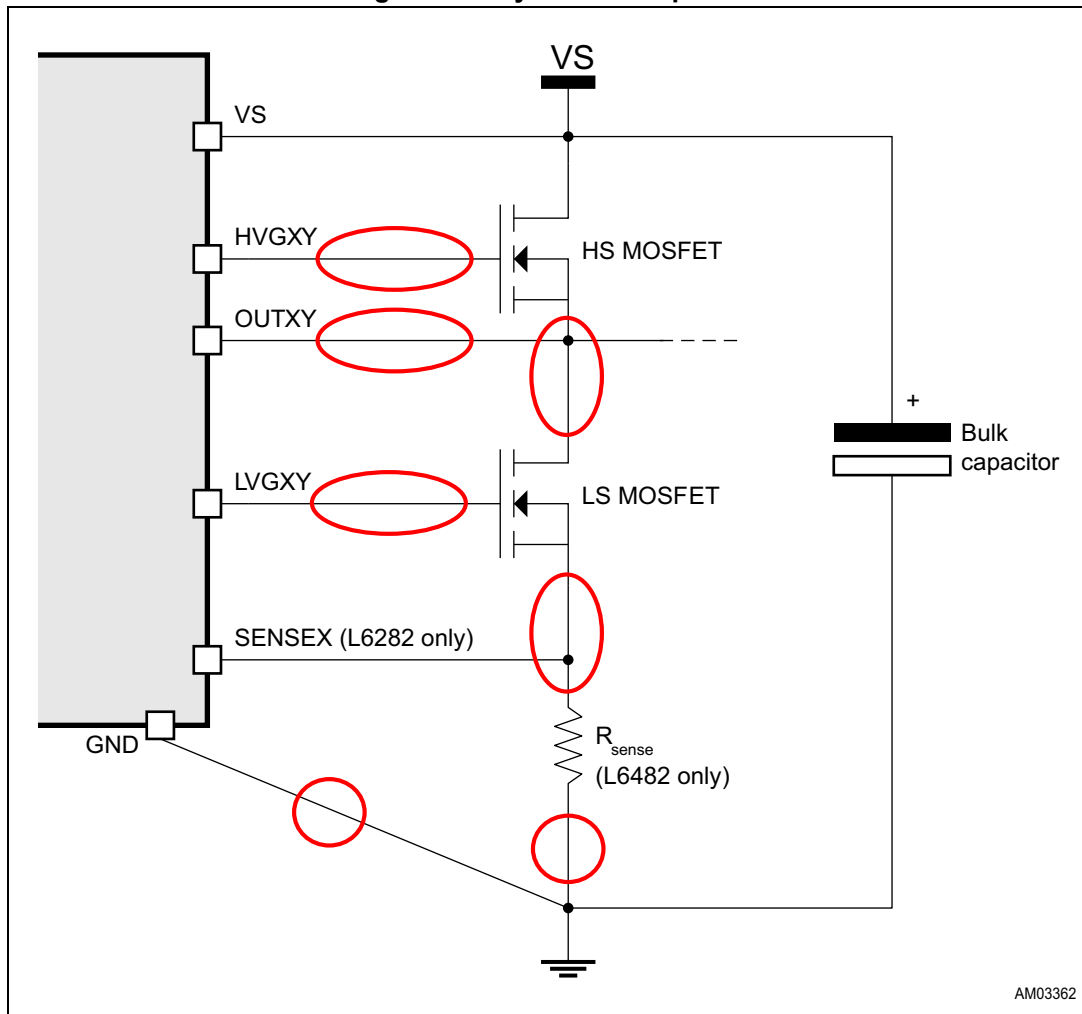
The external clock source allows obtaining higher precision of the speed profile.

8 Layout suggestions

The L648x devices require external MOSFETs for implementing the power stage which drives the motor phases. The electrical connections between the device and the external MOSFETs are critical and particular attention must be paid to reduce parasitic inductances.

Figure 16 shows the critical paths which should be designed to be as short as possible and the area of the circuits should be minimized avoiding the sensitivity of such structures to the surrounding noise. Typically, a good power system layout keeps the power MOSFETs as close as possible to the related gate driver.

Figure 16. Layout critical paths



AM03362

9 Revision history

Table 4. Document revision history

Date	Revision	Changes
11-Oct-2013	1	Initial release.
27-Mar-2015	2	Replaced main title <i>on page 1</i> by new title "L648x devices: high power stepper motor controllers". Replaced "cSPIN™ and dSPIN™ family" by "L648x" in the whole document. Minor modifications throughout document.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

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

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

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

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

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

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

© 2015 STMicroelectronics – All rights reserved

