APPLICATIONS

The NCV70628 is ideally suited for small positioning applications. Target markets include: automotive (headlamp alignment, HVAC, idle control, cruise control), industrial equipment (lighting, fluid control, labeling, process control, XYZ tables, robots...) and building automation (HVAC, surveillance, satellite dish, renewable energy systems). Suitable applications typically have multiple axes or require mechatronics solutions with the driver chip mounted directly on the motor.

Table 1. ORDERING INFORMATION

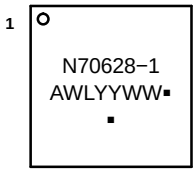
Part No.	Peak Current	End Market/Version	Package*	Shipping†
NCV70628MW001R2G	800/1100 mA (Note 1)	Automotive	QFN32 with step−cut wettable flank (Pb−Free)	5000 / Tape & Reel

*For additional information on our Pb-Free strategy and soldering details, please download the **onsemi** Soldering and Mounting Techniques Reference Manual, SOLDERRM/D.

†For information on tape and reel specifications, including part orientation and tape sizes, please refer to our Tape and Reel Packaging Specifications Brochure, BRD8011/D.

1. The device boost current. This applies for operation under the thermal warning level only.

MARKING DIAGRAM



QFN32

A = Assembly Location
WL = Wafer Lot
YY = Year
WW = Work Week
G or ▪ = Pb−Free Package

(Note: Microdot may be in either location)

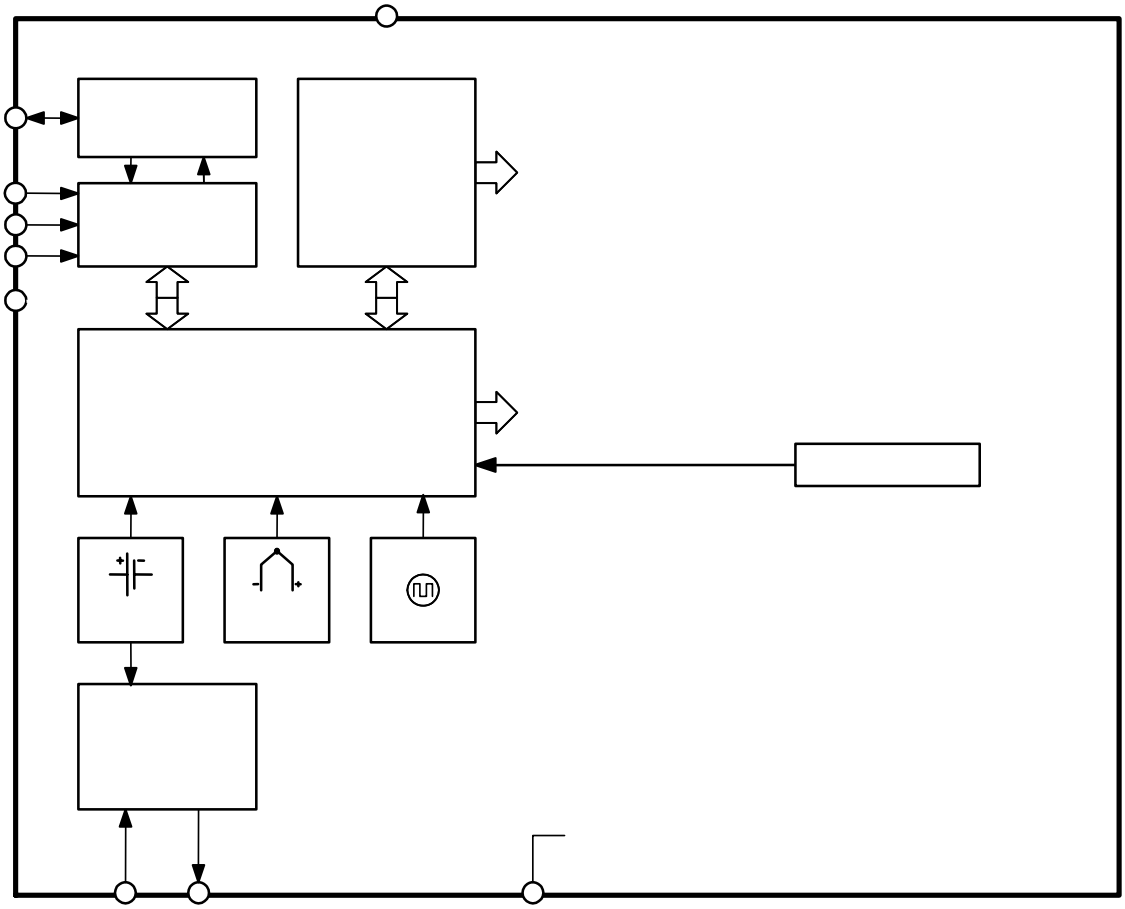


Figure 1. Block Diagram

Table 4. PIN DESCRIPTIONS − QFN PACKAGE

Pin No.	Pin Name	Pin Description	
1, 2	MXP	Positive end of phase X coil	
3, 4, 21, 22	VBB	Battery voltage supply	
5, 7, 20	NC	Not used	
6	SWI	Switch input	
8	HW0	Bit 0 of LIN−ADD	To be tied to GND or V_{DD}
9	HW1	Bit 1 of LIN−ADD	
10	VDD	Internal supply (needs external decoupling capacitor)	
11	GND	Ground	
12	TST1	Test pin (to be tied to ground in normal operation)	
13	LIN	LIN−bus connection	
14, 15	GNDL	Ground	
16	HW2	Bit 2 LIN−ADD	
17	TST2	Test pin (to be tied to ground in normal operation)	
18	TST3	Test pin (to be tied to ground in normal operation)	
19	TST4	Test pin (to be tied to ground in normal operation)	
23, 24	MYN	Negative end of phase Y coil	
25, 26, 31, 32	GNDPW	Ground	
27, 28	MYP	Positive end of phase Y coil	
29, 30	MXN	Negative end of phase X coil	

PACKAGE THERMAL RESISTANCE

The NCV70628 is available in thermally optimized QFN32 package. For the optimizations, the package has an exposed thermal pad which has to be soldered to the PCB ground plane. The ground plane needs thermal vias to conduct the heat to the bottom layer. Figure 3 gives examples for good power distribution solutions.

The thermal resistances are presented in Table 5: Thermal resistance.

Table 5. THERMAL RESISTANCE

Characteristics	Package	Symbol	Min	Typ	Max	Unit
Thermal Resistance, Junction−to−Exposed Pad (Note 9)	QFN32	$R_{\theta JP}$	−	14	−	K/W

9. Also includes typical solder thickness under the Exposed Pad (EP).

Figure 3. Example of QFN32 PCB Ground Plane Layout. Preferred layout at top and bottom connected with through−hole filled vias

DC PARAMETERS

The DC parameters are guaranteed over junction temperature from ĭ40 to 145 C and V_{BB} in the operating range from 5.5 to 29 V, unless otherwise specified. Convention: currents flowing into the circuit are defined as positive.

Table 6. DC PARAMETERS

Symbol	Pin(s)

Table 6. DC PARAMETERS

Symbol	Pin(s)	Parameter	Test Conditions	Min	Typ	Max	Unit
THERMAL WARNING & SHUTDOWN							
T_{tw}		Thermal warning (Notes 13 and 14)		150	157	165	C
T_{tsd}		Thermal shutdown (Note 15)			T_{tw} + 10		C
Tlow		Low temperature warning (Note 15)			T_{tw} − 157		C

Table 6. DC PARAMETERS

Symbol	Pin(s)	Parameter	Test Conditions	Min	Typ	Max	Unit
SWITCH INPUT AND HARDWIRE ADDRESS INPUT							
$Rt_{\_OFF}$	SWI HW2						

AC PARAMETERS

The AC parameters are guaranteed over junction temperature from ĭ40 to 145 C and V_{BB} in the operating range from 5.5 to 29 V, unless otherwise specified. The LIN transmitter and receiver physical layer parameters are compliant to LIN rev. 2.x.

Table 7. AC PARAMETERS

Symbol	Pin(s)	Parameter	Test Conditions	Min	Typ	Max	Unit
POWERUP							
T_{pu}		Power−up time	Guaranteed by design			10	ms
INTERNAL OSCILLATOR							
f_{osc}		Frequency of internal oscillator	V_{BB} = 14 V	3.6	4.0	4.4	MHz
LIN TRANSMITTER CHARACTERISTICS ACCORDING TO LIN V2.x							
D1	LIN	Duty cycle 1 = $t_{Bus\_rec(min)}$ / (2 x t_{Bit}); See Figure 4	THRec(max) = 0.744 x V_{BB} THDom(max) = 0.581 x V_{BB}; V_{BB} = 7.0 V...18 V; t_{Bit} = 50 μs	0.396			
D2		Duty cycle 2 = $t_{Bus\_rec(max)}$ / (2 x t_{Bit}); See Figure 4	THRec(min) = 0.422 x V_{BB} THDom(min) = 0.284 x V_{BB}; V_{BB} = 7.6 V...18 V; t_{Bit} = 50 μs			0.581	
D3		Duty cycle 3 = $t_{Bus\_rec(min)}$ / (2 x t_{Bit})	THRec(max) = 0.778 x V_{BB} THDom(max) = 0.616 x V_{BB}; V_{BB} = 7.0 V...18 V; t_{Bit} = 96 μs	0.417			
D4		Duty cycle 4 = $t_{Bus\_rec(max)}$ / (2 x t_{Bit})	THRec(min) = 0.389 x V_{BB} THDom(min) = 0.251 x V_{BB}; V_{BB} = 7.6 V...18 V; t_{Bit} = 96 μs			0.590	
LIN RECEIVER CHARACTERISTICS ACCORDING TO LIN V2.x							
trx_pdr	LIN	Propagation delay bus dominant to RxD = low	V_{BB} = 7.0 V & 18 V; See Figure 4			6	μs
trx_pdf		Propagation delay bus recessive to RxD = high	V_{BB} = 7.0 V & 18 V; See Figure 4			6	μs
trx_sym		Symmetry of receiver propagation delay	trx_pdr − trx_pdf	−2		+2	μs
SWITCH INPUT AND HARDWIRE ADDRESS INPUT							
T_{sw}	SWI HW2	Scan pulse period (Note 22)	V_{BB} = 14 V		1024		μs
T_{sw_on}		Scan pulse duration (Note 22)	V_{BB} = 14 V		128		μs

22. Derived from the internal oscillator
23. See `SetMotorParam` and PWM Regulator

Table 7. AC PARAMETERS

Symbol	Pin(s)	Parameter	Test Conditions	Min	Typ	Max	Unit

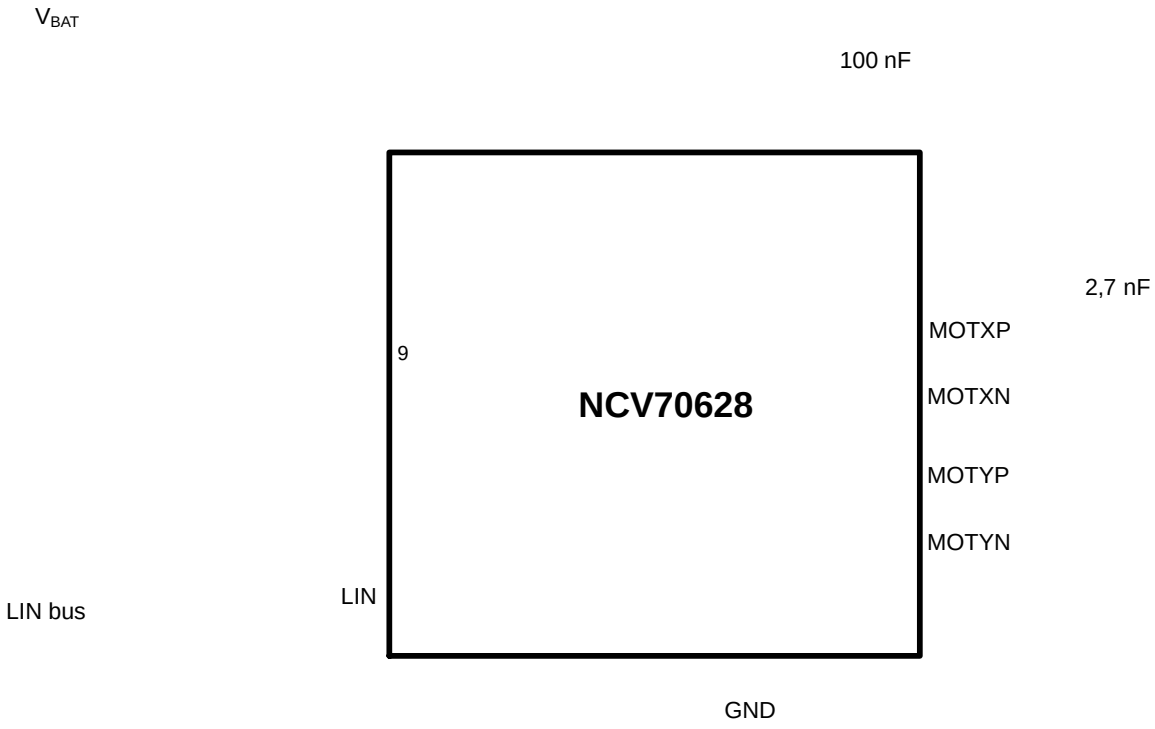


Figure 5. Typical Application

POSITIONING PARAMETERS

Stepping Modes

One of four possible stepping modes can be programmed:

Half ĭ stepping

1/4 micro ĭ stepping

1/8 micro ĭ stepping

1/16 micro ĭ stepping

Maximum Velocity

For each stepping mode, the maximum velocity Vmax can be programmed to 16 possible values given in the table below.

The accuracy of Vmax is derived from the internal oscillator. Under special circumstances it is possible to change the Vmax parameter while a motion is ongoing. All 16 entries for the Vmax parameter are divided into four groups. When changing Vmax during a motion the application must take care that the new Vmax parameter stays within the same group.

Table 9. MAXIMUM VELOCITY SELECTION TABLE

Vmax Index				Stepping Mode			
Hex	Dec	Vmax (full step/s)	Group	Half−stepping (half−step/s)	1/4th Micro−stepping (micro−step/s)	1/8th Micro−stepping (micro−step/s)	1/16th Micro−stepping (micro−step/s)
0	0	99	A	197	395	790	1579
1	1	136	B	273	546	1091	2182
2	2	167		334	668	1335	2670
3	3	197		395	790	1579	3159
4	4	213		425	851	1701	3403
5	5	228		456	912	1823	3647
6	6	243		486	973	1945	3891
7	7	273	C	546	1091	2182	4364
8	8	303		607	1213	2426	4852
9	9	334		668	1335	2670	5341
A	10	364		729	1457	2914	5829
B	11	395		790	1579	3159	6317
C	12	456		912	1823	3647	7294
D	13	546	D	1091	2182	4364	8728
E	14	729		1457	2914	5829	11658
F	15	973		1945	3891	7782	15564

Minimum Velocity

Once the maximum velocity is chosen, 16 possible values can be programmed for the minimum velocity Vmin. The table below provides the obtainable values in full−step/s. The accuracy of Vmin is derived from the internal oscillator. It is not recommended to change the Vmin while a motion is ongoing.

Table 10. OBTAINABLE VALUES IN FULL−STEP/s FOR THE MINIMUM VELOCITY

Vmin Index		Vmax Factor	Vmax (Full−step/s)															
			A	B						C							D	
Hex	Dec		99	136	167	197	213	228	243	273	303	334	364	395	456	546	729	973
0	0	1	99	136	167	197	213	228	243	273	303	334	364	395	456	546	729	973
1	1	1/32	3	4	5	6	6	7	7	8	8	10	10	11	13	15	19	27
2	2	2/32	6	8	10	11	12	13	14	15	17	19	21	23	27	31	42	57
3	3	3/32	9	12	15	18	19	21	22	25	27	31	32	36	42	50	65	88
4	4	4/32	12	16	20	24	26	28	30	32	36	40	44	48	55	65	88	118
5	5	5/32	15	21	26	31	32	35	37	42	46	51	55	61	71	84	111	149
6	6	6/32	18	25	31	36	39	42	45	50	55	61	67	72	84	99	134	179
7	7	7/32	21	30	36	43	46	50	52	59	65	72	78	86	99			

Acceleration and Deceleration

Sixteen possible values can be programmed for Acc (acceleration and deceleration between Vmin and Vmax). The table below provides the obtainable values in full ǐ step/s^2. One observes restrictions for some combinations of acceleration index and maximum speed. It is not recommended to change the Acc value while a motion is ongoing.

Position Ranges

A position is coded by using the binary two's complement format. According to the positioning commands used and to the chosen stepping mode, the position range will be as shown in the following table.

Table 13. POSITION RANGE

Command	Stepping Mode	Position Range	Full Range Excursion	Number of Bits in micro stepping
SetPosition	Half−stepping	−4096 to +4095	8192 half−steps	13
	1/4th micro−stepping	−8192 to +8191	16384 micro−steps	14
	1/8th micro−stepping	−16384 to +16383	32768 micro−steps	15
	1/16th micro−stepping	−32768 to +32767	65536 micro−steps	16

When using the command SetPosition, although coded on 16 bits, the position word is shifted to the left by a certain number of bits, according to the stepping mode.

Secure Position

A secure position can be programmed. It is mapped to the positioned full range but coded in 11 bits, thus having a lower resolution than normal positions, as shown in the following table. See also command GotoSecurePosition and LIN lost behavior.

Table 14. SECURE POSITION

Stepping Mode	Secure Position Resolution
Half−stepping	4 half−steps
1/4th micro−stepping	8 micro−steps (1/4th)
1/8th micro−stepping	16 micro−steps (1/8th)
1/16th micro−stepping	32 micro−steps (1/16th)

Important

NOTES: For the FailSafe functionality and SetDualPosition command, the secure position is disabled in case the programmed value has the code "10000000000" (0x400 or most negative position). For the GotoSecurePosition command there is no disabling possible. By receiving this command the secure positioning is always executed, even when the secure position has the value 0x400.

The resolution of the secure position is limited to 9 bit at start−up. The OTP register is copied in RAM as illustrated below. The RAM bits SecPos1 and SecPos0 are set to 0.

SecPos10

STRUCTURAL DESCRIPTION

Refer to the Block Diagram in Figure 1.

Stepper Motordriver

The Motordriver receives the control signals from the control logic. The main features are:

- Two H−bridges, designed to drive a stepper motor with two separated coils. Each coil (X and Y) is driven by one H−bridge, and the driver controls the currents flowing through the coils. The rotational position of the rotor, in unloaded condition, is defined by the ratio of current flowing in X and Y. The torque of the stepper motor when unloaded is controlled by the magnitude of the currents in X and Y.
- The control block for the H−bridges, including the PWM control, the synchronous rectification and the internal current sensing circuitry.
- Two pre−scale 4−bit DAC's to set the maximum magnitude of the current through X and Y.
- Two DAC's to set the correct current ratio through X and Y.
- A boost function that increases the current during cold conditions.

Battery voltage monitoring is also performed by this block, which provides the required information to the control logic part. The same applies for detection and reporting of an electrical problem that could occur on the coils.

Control Logic (Position Controller and Main Control)

The control logic block stores the information provided by the LIN interface (in a RAM or an OTP memory) and digitally controls the positioning of the stepper motor in terms of speed and acceleration, by feeding the right signals to the motor driver state machine.

It will take into account the successive positioning commands to properly initiate or stop the stepper motor in order to reach the set point in a minimum time.

It also receives feedback from the motor driver part in order to manage possible problems and decide on internal actions and reporting to the LIN interface.

Motion Detection

Motion detection is based on the back−emf generated internally in the running motor. When the motor is blocked, e.g. when it hits the end position, the velocity, and as a result also the generated back−emf, is disturbed. The NCV70628 senses the back−emf and compares the value with an independent threshold level. If the back−emf becomes lower than the threshold, the running motor is stopped.

LIN Interface

The LIN interface implements the physical layer and the MAC and LLC layers according to the OSI reference model. It provides and gets information to and from the control logic block, in order to drive the stepper motor, to configure the way this motor must be driven, or to get information such as actual position or diagnosis (temperature, battery voltage, electrical status...) and pass it to the LIN master node.

Miscellaneous

The NCV70628 also contains the following:

- An internal oscillator, needed for the LIN protocol handler as well as the control logic and the PWM control of the motor driver.
- An internal trimmed voltage source for precise referencing.
- A protection block featuring a thermal shutdown and a power−on−reset circuit.
- A 3.3 V regulator (from the battery supply) to supply the internal logic circuitry.

FUNCTIONS DESCRIPTION

This chapter describes the following functional blocks in more detail:

- Position controller
- Main control and register, OTP memory + ROM
- Motor driver

The Motion detection and LIN controller are discussed in separate chapters.

POSITION CONTROLLER

Positioning and Motion Control

A positioning command will produce a motion as illustrated in Figure 6. A motion starts with an acceleration phase from minimum velocity (Vmin) to maximum velocity (Vmax) and ends with a symmetrical deceleration. This is defined by the control logic according to the position required by the application and the parameters programmed by the application during the configuration phase. The current in the coils is also programmable.

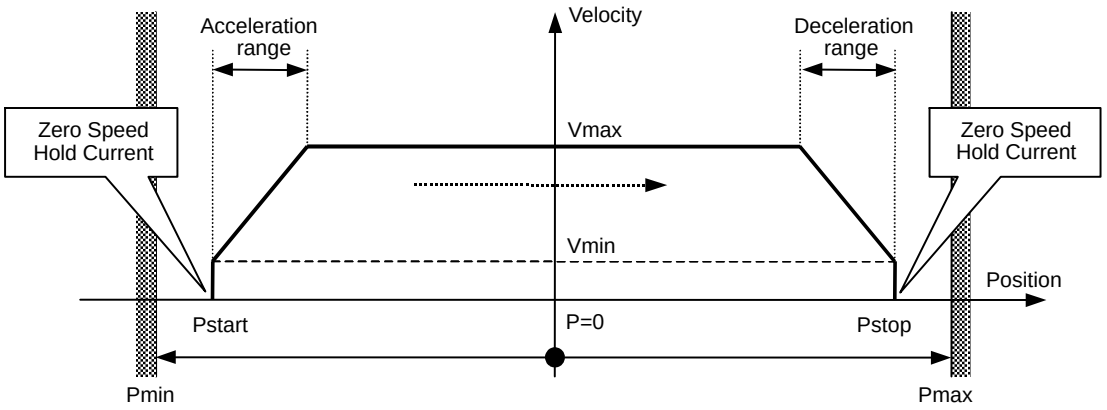


Figure 6. Position and Motion Control

Table 15. POSITION RELATED PARAMETERS

Parameter	Reference
Pmax – Pmin	See Positioning
Zero Speed Hold Current	See Ihold
Maximum Current	See Irun
Acceleration and Deceleration	See Acceleration and Deceleration
Vmin	See Minimum Velocity
Vmax	See Maximum Velocity
Stabilization Time	See Stabilization Time

Different positioning examples are shown in the next table.

Table 16. POSITIONING EXAMPLES

Short motion.	

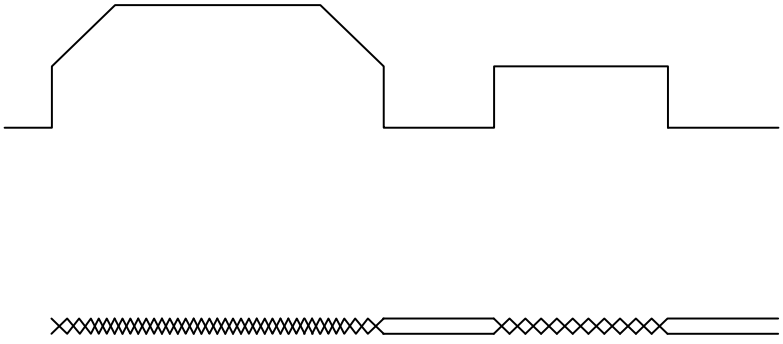


Table 17. STATE DIAGRAM OF THE HW2 COMPARATOR

Previous State	DriveLS	DriveHS	HW2_Cmp	New State	Condition	Drawing

R2VBAT

A resistor is connected between VBAT and HW2. Every 1024 μs S_{BOT}


T warning level

SetPosition2motors, SetPosParam and GotoSecurePosition, even if the <UV2> flag and <Steploss> flags are NOT cleared.

If however the V_{BB} voltage remains below UV2 level voltage level for more than 15 seconds, the device will enter <Shutdown> state and the target position is overwritten by Actual Position. This state can be exited only if V_{BB} is > UV1 voltage level and an incoming command GetStatus or GetFullStatus is received.

Important Notes:

1. In the case of Autarkic positioning, care needs to be taken because accumulated steploss can cause a significant deviation between physical and stored actual position.
2. The SetDualPosition command will only be executed after clearing the <UV2> and <Steploss> flags.
3. RAM reset occurs when Vdd < VddReset (digital PowerŏOnŏReset level).
4. The Autarkic function remains active as long as V_{DD} > VddReset.
5. LIN timeout is not being monitored in Autarkic mode in <ShutUnder> state to avoid LIN lost procedure activation in Autarkic mode or right after Vdd > UV1 (LIN communication is disabled when Vdd < UV2).

OTP REGISTER

OTP Memory Structure

The table below shows how the parameters to be stored in the OTP memory are located.

Table 19. OTP MEMORY STRUCTURE

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	TSD2	TSD1	TSD0	IREF4	IREF3	IREF2	IREF1	IREF0
0x01	SecPosA	ADM	BG4	BG3	BG2	BG1	BG0	TSD3
0x02	AbsThr3	AbsThr2	AbsThr1	AbsThr0	PA3	PA2	PA1	PA0
0x03	Irun3	Irun2	Irun1	Irun0	Ihold3	Ihold2	Ihold1	Ihold0
0x04	Vmax3	Vmax2	Vmax1	Vmax0	Vmin3	Vmin2	Vmin1	Vmin0
0x05	SecPos10	SecPos9	SecPos8	Shaft	Acc3	Acc2	Acc1	Acc0
0x06	SecPos7	SecPos6	SecPos5	SecPos4	SecPos3	SecPos2	Failsafe	SleepEn
0x07	UV3debT	UV3Thr2	UV3Thr1	UV3Thr0	StepMode1	StepMode0	LOCKBT	LOCKBG
0x08	SecPos10A	SecPos9A	SecPos8A	OSC4	OSC3	OSC2	OSC1	OSC0
0x09	SecPos7A	SecPos6A	SecPos5A					

UV3Thr[2:0] Under voltage threshold voltage for UV3 and UV1.

Index	UV3Thr			UV3 Level	UV1 Level
0	0	0	0	5.90	6.62
1	0	0	1	6.30	7.07
2	0	1	0	6.70	7.52
3	0	1	1	7.10	7.97
4	1	0	0	7.50	8.41
5	1	0	1	7.90	8.86
6	1	1	0	8.30	9.31
7	1	1	1	8.70	9.76

Irun[3:0] Current amplitude value to be fed to each coil of the stepper motor. The table below provides the 16 possible values for <IRUN>.

Vmax[3:0] Maximum velocity

Index	Vmax

Table 21. RAM REGISTERS

Register	Mnemonic	Lengt-h (bit)	Related Commands	Comment	Reset State
PWM Jitter	PWMJEn	1	GetFullStatus SetStallParam	'1' means jitter is added	'0'
100% duty cycle Stall Enable	DC100StEn	1	GetFullStatus SetStallParam	'1' means stall detection is enabled in case PWM regulator runs at δ = 100%	'0'
PWM frequency	PWMFreq	1	GetFullStatus SetMotorParam	'0' means ~ 22 KHz, '1' means ~ 44 KHz	'0'
Boost function	I_BOOST_ ENB	1	GetFullStatus SetMotorParam	'0' means boost function is enabled. See also Motor current boost function	'1'
Stabilization time	Tstab	3	GetFullStatus SetMotorParam	See also Motor stopping phase and Motion detection	'011'

Table 22. FLAGS TABLE

Flag	Mnemonic	Length (bit)	Related Commands	Comment	Reset State
LIN Timeout Error	TimE	1	GetFullStatus	Timeout error occurred	'0'
LIN Data Error	DataE	1	Internal use	'1' when one of the three errors occurred: checksum error, stop bit error or frame length error	'0'
LIN Header Error	HeadE	1	GetFullStatus	PID Parity error or PID stop bit error occurred	'0'
LIN Bit Error	BitE	1	Internal use	'1' when received bit value is different from the one being transmitted	'0'
LIN Response Error	LIN_E	1	GetActualPos GetFullStatus	LIN response error occurred (Checksum error, Stop bit error, Frame length error or difference in bit sent and bit monitored on LIN bus)	'0'
Electrical defect	ElDef	1	GetActualPos GetStatus GetFullStatus	<OVC1> or <OVC2> or 'open−load on coil X' or 'open−load on coil Y Resets only after Get(Full)Status	'0'
External switch status	ESW	1	GetActualPos GetStatus GetFullStatus	'0' = open '1' = close	'0'
Electrical flag	HS	1	Internal use	<UV2> or <ElDef> or <VDDreset>	'0'
Motion status	Motion	3	GetActualPos GetFullStatus	"000" = Stop, last movement was inner (CCW) motion "100" = Stop, last movement was outer (CW) motion "001" = inner (CCW) motion acceleration "010" = inner (CCW) motion deceleration "011" = inner (CCW) motion max. speed "101" = outer (CW) motion acceleration "110" = outer (CW) motion deceleration "111" = outer (CW) motion max. speed	"000"
Over current in coil X	OVC1	1	GetFullStatus	'1' = over current; reset only after GetFullStatus	'0'
Over current in coil Y	OVC2	1	GetFullStatus	'1' = over current; reset only after GetFullStatus	'0'
Secure position enabled	SecEn	1	Internal use	'0' if <SecPos> = "100 0000 0000" '1' otherwise	n.a.
Circuit going to Sleep mode	Sleep	1	Internal use	'1' = Sleep mode reset by LIN command	'0'
Step loss	StepLoss				

Table 22. FLAGS TABLE

Flag	Mnemonic	Length (bit)	Related Commands	Comment	Reset State
Stall	Stall	1	GetActualPos GetFullStatus GetStatus		‘0’
Motor stop	Stop	1	Internal use		‘0’
Temperature info	Tinfo	2	GetActualPos GetStatus GetFullStatus	“00” = normal temperature range “01” = low temperature warning “10” = high temperature warning “11” = motor shutdown	

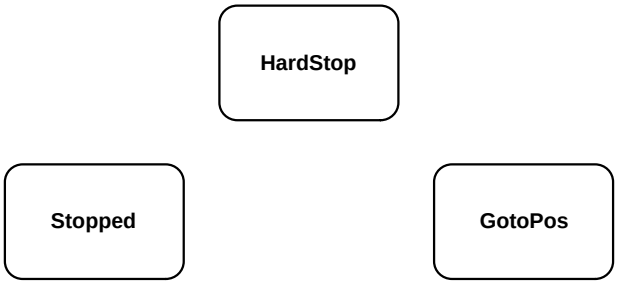
Priority Encoder

The table below describes the simplified state management performed by the main control block.

With the Following Color Code:

Command Ignored	Transition to Another State	Master is responsible for proper update (see Note 37)

32. Leaving <Sleep> state is equivalent to power−on−reset.
33. After power−on−reset, the <Standby> state is entered.
34. A Dual Position sequence runs with a separate set of RAM registers. The parameters that are not specified in a Dual Position command are loaded with the values stored in RAM at the moment the Dual Position sequence starts. <AccShape> is forced to '1' during second motion. <AccShape> at '0' will be taken into account after the Dual Position sequence. A GetFullStatus command will return the default parameters for <Vmax> and <Vmin> stored in RAM.
35. The <Sleep> flag is set to '1' when LIN timeout or GotoSleep command occurs. It is reset by the next LIN command (<Sleep> is cancelled if not activated yet).
36. Shutdown state can be left only when <TSD> and <HS> flags are reset.
37. Flags can be reset only after the master could read them via a GetStatus or GetFullStatus command, and provided the physical


HardStop
Stopped
GotoPos

PWM Jitter

To lower the power spectrum for the fundamental and higher harmonics of the PWM

This can be illustrated in the following sequence given as an application example. The master can check whether there is a problem or not and decide which application strategy to adopt.

Table 27. EXAMPLE OF POSSIBLE SEQUENCE USED TO DETECT AND DETERMINE CAUSE OF MOTOR SHUTDOWN

Tj Ttsd or V_{BB} UV2 (>15s) or <ElDef> = '1'	SetPosition frame	GetFullStatus or GetStatus frame

<StepLoss> and <Stall> flags are set. These flags can only be reset by sending a GetFullStatus command.

If Stall appears during DualPosition then the first phase is cancelled (via internal hardstop) and after timeout Tstab (see AC table) the second phase at Vmin starts.

When the <Stall> flag is set, the position controller will generate an internal HardStop. As a consequence also the <Steploss> flag will be set. The position in the internal counter will be copied to the <ActPos> register. All flags can be read out with the GetStatus or GetFullStatus command.

Important Remark
(limited to motion detection flags / parameters):

Using GetFullStatus will read **AND** clear the following flags: <Steploss>, <Stall> and <AbsStall>. New positioning is possible and the <ActPos> register will be further updated.

Using GetStatus will read **AND** clear **ONLY** the <Steploss> flag. The <Stall> and <AbsStall>

LIN CONTROLLER

General Description

The LIN (local interconnect network) is a serial communications protocol that efficiently supports the control of mechatronics nodes in distributed automotive applications. The physical interface implemented in the NCV70628 is compliant to the LIN rev. 2.2a specification. It features a slave node, thus allowing for:

- single−master / multiple−slave communication
- self synchronization without quartz or ceramics resonator in the slave nodes
- guaranteed latency times for signal transmission
- single−signal−wire communication
- automatic transmission speed detection (between 1 and 19.2 kbit/s)
- configuration flexibility
- data and protected identifier checksum (classic checksum for LIN diagnostic and configuration frames, enhanced checksum for other frames) security and error detection
- detection of defective nodes in the network

It includes the analog physical layer and the digital protocol handler.

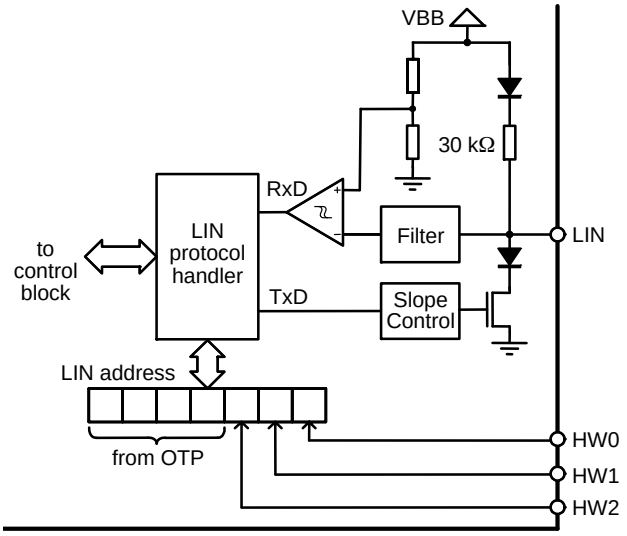


Figure 20. LIN Interface

The analog circuitry implements a low side driver with a pull−up resistor as a transmitter, and a resistive divider with a comparator as a receiver. The specification of the line driver/receiver follows the ISO 9141 standard with some enhancements regarding the EMI behavior.

Slave Operational Range for Proper Self Synchronization

The LIN interface will synchronize properly in the following conditions:

Vbat 8 V

LIN communication is disabled when Vbat < UV2

Ground shift between master node and slave node < 1 V

It is highly recommended to use the same type of reverse battery voltage protection diode for the Master and the Slave nodes.

FUNCTIONAL DESCRIPTION

Analog Part

The transmitter is a low−side driver with a pull−up resistor and slope control. The receiver mainly consists of a comparator with a threshold equal to $V_{BB}/2$. Figure 4 shows the characteristics of the transmitted and received signal. See AC Parameters for timing values.

Bit Sample Timing

The LIN uses a clock whose frequency is $16*(1/t_{bit})$. The byte field is synchronized at the falling edge of the start bit. The byte field synchronization has an accuracy of t_{BFS}.

After the byte field synchronization the data bit itself is sampled within the window between the earliest bit sample time and the latest bit sample time. The majority of the samples define the bit level.

Table 31. BIT SAMPLE TIMING

Parameter	Comment	Test Condition	Min	Typ	Max	Unit
t_{BFS}	Value of accuracy of the byte field detection	Guaranteed by a digital test			2/16	

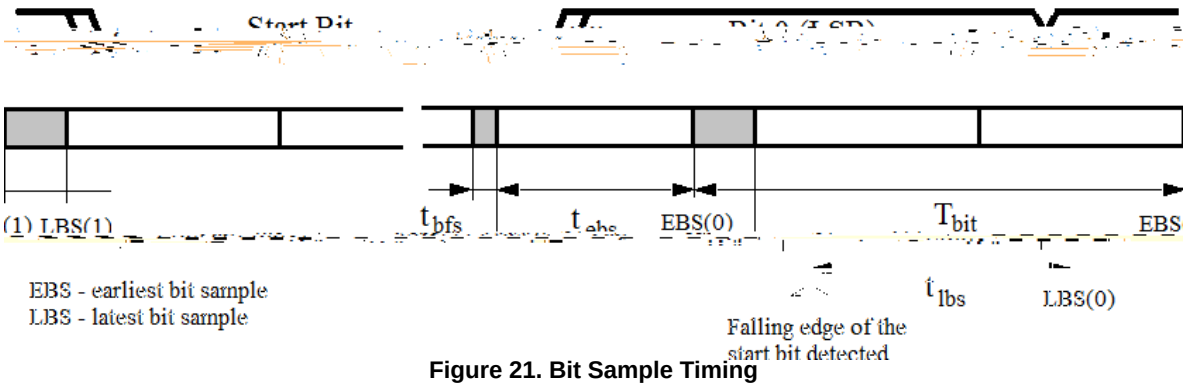


Figure 21. Bit Sample Timing

Protocol Handler

This block implements:

Bit synchronization

Bit timing

The MAC layer

The LLC layer

The supervisor

LIN Error Register

The LIN interface implements a register containing an error status of the LIN communication. This register is as follows:

Table 32. LIN ERROR REGISTER

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Not used	Not used	Not used	Response error flag (LIN_E)	**Time** out error flag	**Data** error Flag	**Header** error Flag	**Bit** error Flag

With:

Response error flag (LIN_E): OR function of Data error flag, Bit error flag

Data error flag: Checksum error, data stop bit error or frame length error

Header error flag: PID parity error or PID stop bit error or sync field stop bit error

Time out error flag: Detected dominant pulse duration is greater than maximum allowed header duration (invalid break + sync detected)

Bit error flag: Difference in bit sent and bit monitored on the LIN bus

Time out error flag has only informational purpose. Flags Response error, Data error and Bit error are reset by GetFullStatus and GetActualPos frames. Flags Header error and Time out error are reset by GetFullStatus frame.

Physical Address of the Circuit

The circuit must be provided with a physical address in order to discriminate this circuit from other ones on the LIN bus. This address is coded on 7 bits, yielding the theoretical possibility of 125 different circuits on the same bus due to LIN NAD field restriction (only addresses from 1 to 125 are allowed). If node address 127 is supplied, all slave nodes shall react to the frame (broadcast function). However the maximum number of nodes in a LIN network is also limited by the physical properties of the bus line. It is recommended to limit the number of nodes in a LIN network to not exceed 16. Otherwise the reduced network impedance may prohibit a fault free communication under worst case conditions. Every additional node lowers the network impedance by approximately 3%.

Node address is supplied in 8ĭbit NAD field within the LIN writing or preparing frames. Lower 7 bits of the NAD field represent the node address, the MSB bit of NAD field shall be always zero since LIN user diagnostic frames are not used on NCV70628. See Table 33 for detailed NAD field description:

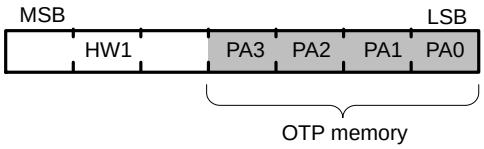
Table 33. LIN NAD FIELD

NAD	Description
0	Reserved for go to sleep command
1 – 125	Node address
126	Functional NAD (not used on NCV70628)
127	Broadcast
128 – 255	Free usage (user diagnostics, not used on NCV70628)

The node address is a combination of 4 OTP memory bits and 3 hardwired address bits (pins HW[2:0]). Depending on the Addressing Mode (<ADM> bit in OTP) the bits of the address are combined as illustrated below. OTP bit PA0 is always inverted. Due to restriction in LIN specification rev. 2.2 such combination of hardwired bits and OTP memory bits that would result into node address being 0, 126 or 127 shall be avoided.

Node address is assigned when first break and sync field is detected on LIN bus after powerĭonĭreset. In case of HW2 float state, node address assignment is performed after

HW2 state changes to low or high and next break and sync field is detected on LIN bus. LIN communication is disabled until node address is assigned. Once node address is assigned, it cannot be changed (e.g. via HW[2:0] reconfiguration) until power−on−reset occurs.



reading frame and will also contain a command indicating which kind of information is awaited from the slave (see definition of LIN standard diagnostic and configuration frames supported by NCV70628).

Dynamic Assignment of Identifiers

The protected identifier field in the LIN datagram denotes the content of the message. Six identifier bits PID[5:0] and two parity bits PID[7:6] are used to represent the content. Identifiers 0x3C and 0x3D are reserved for LIN diagnostic and configuration frames. Identifiers 0x3E and 0x3F are reserved for future LIN extensions. Slave nodes need to be very flexible to adapt itself to a given LIN network in order to avoid conflicts with slave nodes from different

LIN LOST BEHAVIOR

Introduction

When the LIN communication is inactive (stable recessive or dominant value) for duration of 4.46 s NCV70628 sets an internal flag called “LIN lost”. The functional behavior depends on the state of OTP bits <SleepEn> and <FailSafe>, and if this loss in LIN communication occurred at (or before) power on reset or in normal powered operation.

Sleep Enable

The OTP bit <SleepEn> enables or disables the entering to low−power sleep mode in case of LIN time−out. By default the entering to sleep mode is disabled.

Table 37. SLEEP ENABLE SELECTION

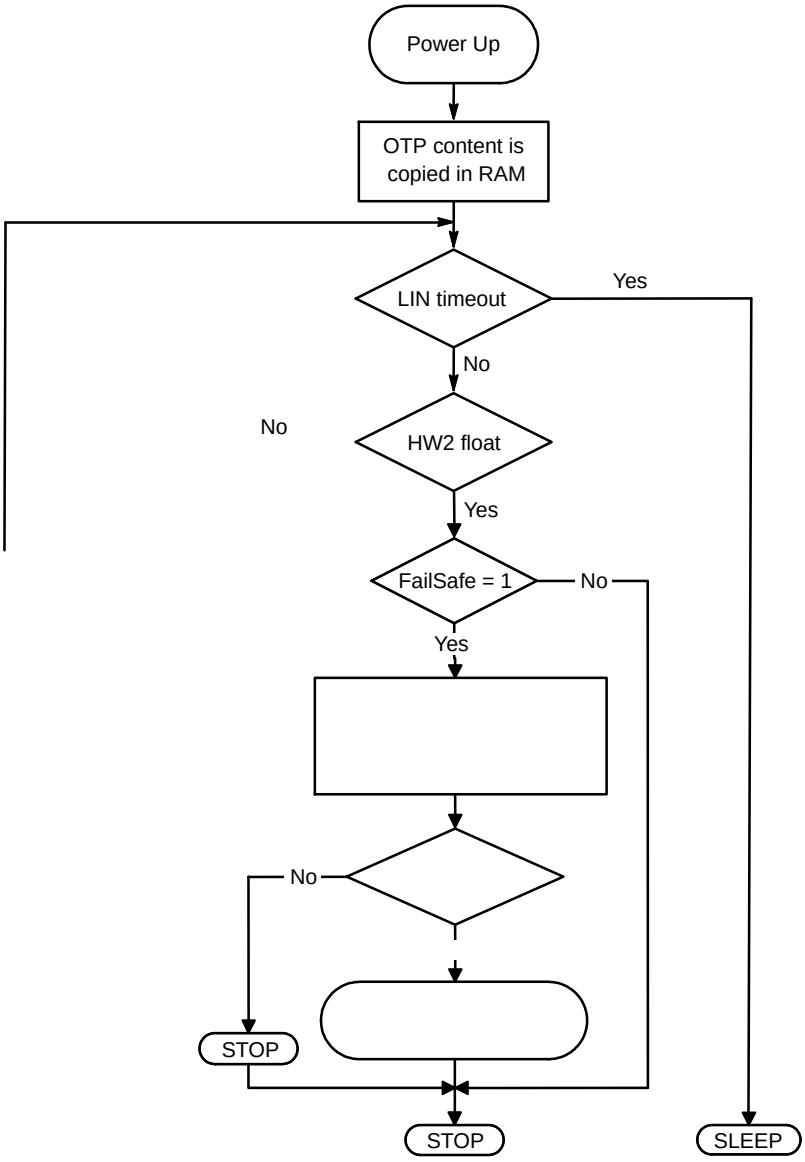
<SleepEn>	Behavior
0	Entering low−power sleep mode is disabled except from <Standby> and <Shutdown>
1	Entering low−power sleep mode enabled

Fail Safe Motion

The OTP bit <FailSafe> enables or disables an automatic motion to a predefined secure position. See also Autonomous Motion.

Table 38. FAIL SAFE ENABLE SELECTION

<FailSafe>	Behavior
0	


Power Up
OTP content is copied in RAM
LIN timeout
Yes
No
No
HW2 float
Yes
FailSafe = 1
No
Yes
No
STOP
STOP
SLEEP

LIN APPLICATION COMMANDS

Introduction

The LIN Master will have to use commands to manage

Table 43. STRUCTURE OF PID(2) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
[redacted]									
1	CMD	Command							
2	Checksum	Enhanced Checksum							

Where:

Command: Can be 0x04 for `GotoSecurePosition`, 0x05 for `HardStop`, 0x06 for `ResetPosition` or 0x0F for `SoftStop`.

APPLICATION COMMANDS

Read by identifier ID 0

Read by identifier is a standard LIN frame used for slave node identification. Response to this command varies based on supplied Identifier field (ID, Byte 4). NCV70628 implements only response to ID 0 which is the minimum required by LIN specification.

Read by identifier ID 0 provides the ability to read slave node Supplier ID, Function ID and Variant while knowing slave node address. This can be achieved by supplying node address in NAD field and using a wildcard value for Supplier ID and Function ID bytes. Slave node with specified node

Table 45. READ BY IDENTIFIER ID 0 READING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	0x7D							
1	NAD	NAD[7:0]							
2									

Table 47. GetActualPos PID(6) READING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(6)							
1	Data 1	ActPos[7:0]							
2	Data 2	ActPos[15:8]							
3	Data 3	ESW	StepLoss	ElDef	UV	TSD	TW	Tinfo[1:0]	
4	Data 4	Motion[2:0]			Stall	LIN_E	UV2	UV3	VddReset
5	Checksum	Enhanced Checksum							

GetFullStatus

This command is provided to the circuit by the LIN master to get a complete status of the circuit and the stepper motor. Refer to RAM Registers and Flags Table to see the meaning of the parameters sent to the LIN master.

Note: First response to `GetFullStatus` command will attempt to reset flags <TW>, <TSD>, <UV2>, <UV3>, <UV>, <ElDef>, <StepLoss>, <OVC1>, <OVC2>, <VddReset>, <LIN_E>, <DataE>, <BitE>, <HeadE> and <TimE>. Second response to `GetFullStatus` command will attempt to reset flags <Stall> and <AbsStall>.

The master sends PID(7) reading frame. `GetFullStatus` corresponds to 2 successive LIN inter frame responses to PID(7) reading frame. First response frame contains FrmSeq value equal to 0. Second response frame contains FrmSeq value equal to 1.

Note: It is not mandatory for the LIN master to initiate the second inter frame response if the data in the second response frame is not needed by the application.

Note: It is recommended to poll AbsStall bit (instead of Stall bit) in case LIN Master reads GetFullStatus first response followed by GetFullStatus second response.

Table 48. GetFullStatus PID(7) READING FRAME – FIRST RESPONSE

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(7)							
1	Data 1	FrmSeq	NAD[6:0]						
2	Data 2	Irun[3:0]				Ihold[3:0]			
3	Data 3	Vmax[3:0]				Vmin[3:0]			
4	Data 4	AbsThr[3:0]				Acc[3:0]			
5	Data 5	Tstab[2:0]			StepMode[1:0]		UV3Thr[2:0]		
6	Data 6	AccShape	Shaft	I_BOOST_ENB	PWMFreq	TimeE	HeadE	OVC1	OVC2
7	Data 7	Motion[2:0]			Stall	LIN_E	UV2	UV3	VddReset
8	Data 8	ESW	StepLoss	EIDef	UV	TSD	TW	Tinfo[1:0]	
9	Checksum	Enhanced Checksum							

Where:

FrmSeq: Value is 0 to identify first response frame.

NAD[6:0]: Node address LSBs.

GetBemf

This command is provided to the circuit by the LIN master to get a result of last back EMF measurement. The Coil bit determines whether the measurement was performed on coil X (Coil = 0) or Y (Coil = 1). Refer to Flags Table to see the meaning of the parameters sent to the LIN master.

GetBemf corresponds to following LIN PID(10) frame.

Table 52. GetBemf PID(10) READING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(10)							
1	Data 1	1	BEMF_DC100	Coil	BEMF_OUT[4:0]				
2	Checksum	Enhanced Checksum							

GotoSecurePosition

This command is provided by the LIN master to one, all or a group of the stepper motors to move to the secure position <SecPos[10:0]>. It can also be internally triggered if the LIN bus communication is lost, after an initialization phase, or prior to going into sleep mode. See the priority encoder description for more details. The priority encoder table also acknowledges the cases where a GotoSecurePosition command will be ignored. This command is executed regardless of <SecEn> value.

Note: One slave node can be addressed using its node address in NAD field of PID(1) writing frame. All slave nodes can be addressed using broadcast NAD in PID(1) writing frame. A certain group of nodes can be addressed by first assigning equal PID(2) code and then issuing PID(2) writing frame.

GotoSecurePosition corresponds to the following LIN PID(1) and PID(2) writing frames.

Table 53. GotoSecurePosition PID(1) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(1)							
1	NAD	NAD[7:0] (0x7F for broadcast)							
2	CMD	0x04							
3	Checksum	Enhanced Checksum							

Table 54. GotoSecurePosition PID(2) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(2)							
1	CMD	0x04							
2									

Table 55. HardStop PID(1) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(1)							
1	NAD	NAD[7:0] (0x7F for broadcast)							
2	CMD	0x05							
3	Checksum	Enhanced Checksum							

Table 56. HardStop PID(2) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(2)							
1	CMD	0x05							
2	Checksum	Enhanced Checksum							

ResetPosition

This command is provided to the circuit by the LIN master to reset <ActPos> and <TagPos> registers to zero. This can be helpful to prepare for instance a relative positioning. The reset position command sets the internal flag “Reference done”.

Note: One slave node can be addressed using its node address in NAD field of PID(1) writing frame. All slave nodes can be addressed using broadcast NAD in PID(1) writing frame. A certain group of nodes can be addressed by first assigning equal PID(2) code and then issuing PID(2) writing frame.

ResetPosition corresponds to the following LIN PID(1) and PID(2) writing frames.

Table 57. ResetPosition PID(1) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(1)							
1	NAD	NAD[7:0] (0x7F for broadcast)							
2	CMD	0x06							
3	Checksum	Enhanced Checksum							

Table 58. ResetPosition PID(2) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(2)							
1	CMD	0x06							
2	Checksum	Enhanced Checksum							

SetDualPosition

This command is provided to the circuit by the LIN master in order to perform a positioning of the motor using two different velocities. See Dual Positioning. After Dual positioning the internal flag “Reference done” is set.

Note: This sequence cannot be interrupted by another positioning command.

SetDualPosition corresponds to the following LIN PID(0) writing frame.

Table 59. SetDualPosition PID(0) WRITING FRAME

Byte	Content

SetOtpParam

This

Table 63. SetPosition PID(3) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(3)							
1	NAD	NAD[7:0] (0x7F for broadcast)							
2	Data 1	Pos[7:0]							
3	Data 2	Pos[15:8]							
4	Checksum	Enhanced Checksum							

Table 64. SetPosition PID(4) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(4)							
1	Data 1	Pos[7:0]							
2	Data 2	Pos[15:8]							
3	Checksum	Enhanced Checksum							

SetPosition2Motors

This command is provided to the circuit by the LIN Master to drive two motors to a given absolute position. See Positioning for more details.

The priority encoder table (see Priority Encoder) describes the cases where a `SetPosition2Motors` command will be ignored.

SetPosition2Motors corresponds to the following PID(5) LIN writing frame.

Table 65. SetPosition2Motors PID(5) WRITING FRAME

Byte	Content	Structure							
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	PID	PID(5)							
1	Data 1	NAD1[7:0] (broadcast not allowed, broadcast NAD will cause the command will be ignored)							
2	Data 2	Pos1[7:0]							
3	Data 3	Pos1[15:8]							
4	Data 4	NAD2[7:0] (broadcast not allowed, broadcast NAD will cause the command will be ignored)							
5	Data 5	Pos2[7:0]							
6	Data 6	Pos2[15:8]							
7	Checksum	Enhanced Checksum							

Where:

NAD1[7:0]: Node address of the first slave. In case broadcast NAD is used, the slave node will ignore this command though slave node specified by NAD2[7:0] shall perform the command anyway.

NAD2[7:0]: Node address of the second slave. In case broadcast NAD is used, the slave node will ignore this command though slave node specified by NAD1[7:0] shall perform the command anyway.

Pos1[15:0]: Signed 16−bit target position of first motor.

Pos2[15:0]: Signed 16−bit target position of second motor.

SetPosParam

This command is provided to the circuit by the LIN Master to drive one motor to a given absolute position. It also sets some of the values for the stepper motor parameters such as minimum and maximum velocity.

SetPosParam corresponds to the following PID(0) LIN writing frame.

Table 66. SetPosParam PID(0) WRITING FRAME

Byte	Content	Structure		
		Bit 7	Bit 6	Bit 5

SoftStop

If a SoftStop command occurs during a motion of the stepper motor, it provokes an immediate deceleration to Vmin (see Minimum Velocity) followed by a stop, regardless of the position reached. Once the motor is stopped, TagPos register is overwritten with value in ActPos register to ensure keeping the stop position.

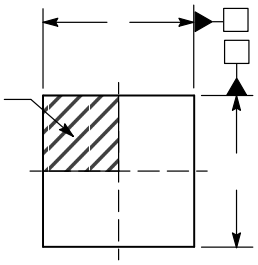
Command SoftStop occurs in the following cases:

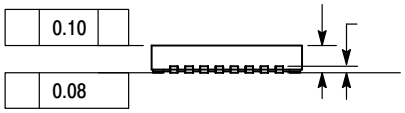
- The chip temperature rises above the thermal shutdown threshold (see DC Parameters and Temperature Management);
- The VBB drops under the UV3 level; (see DC Parameters and Battery Voltage Management);
- The LIN master requests a SoftStop. Hence SoftStop will correspond to the following LIN PID(1) and PID(2) writing frames.

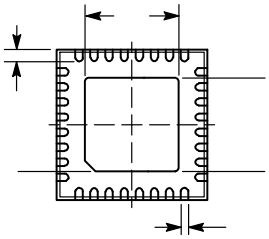
Note: One slave node can be addressed using its node

CASE 484AB
ISSUE D

DATE 07 SEP 2018







XXXXXXXX
XXXXXXXX
AWLYYWW▪
▪

XXXXX = Specific Device Code
A = Assembly Location
WL = Wafer Lot
YY = Year
WW = Work Week
▪ = Pb Free Package