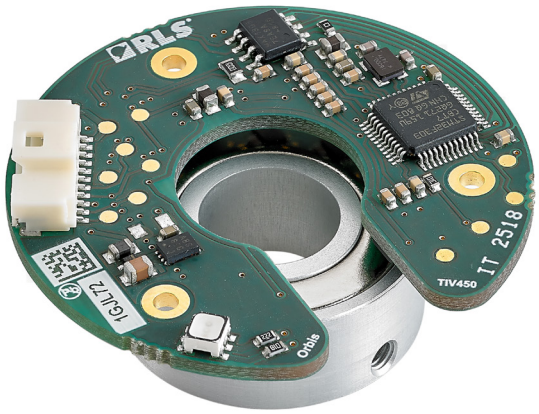
Application note
BRD09_01
Issue 1, 1st June 2020

Using advanced functions with Orbis™ over SPI



NOTE: Image does not represent all product variants.

Brief description

Programming the Orbis encoder via the SPI interface makes it possible to:

- Set position offset (zero position),
- Set multiturn counter value (if available),
- Perform the self-calibration function,
- Store the current Orbis configuration parameters in a non-volatile memory,
- Reset Orbis configuration parameters to the factory settings.

Programming is done by sending separate bytes to the encoder.

Each byte represents the first MOSI byte in the SPI frame. Each byte must be in a separate data frame (an NCS signal cycle only transmits one byte via MOSI). The following describes the command sequences for each command.

Each of the programming options must be started with the special unlocking sequence. This prevents unintentional changes to the encoder configuration parameters. The sequence of each programming option is described below.

NOTE: Delay between each byte sent during programming must be at least 1 ms.

Orbis™ application note

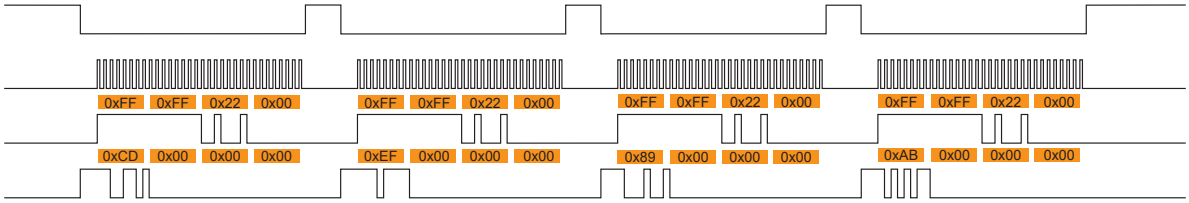
Unlocking sequence

To unlock the encoder and enable programming, four exact bytes must be sent in specified order. They are listed in table below:

Byte in sequence	Programming command byte	
	HEX	ASCII
B1	0xCD	/
B2	0xEF	/
B3	0x89	/
B4	0xAB	/

Whenever the sequence is interrupted by a wrong byte, the unlock stage is reset and must be started from the beginning.

Image below shows the unlocking sequence captured by the logic analyzer.



Only one byte (command) must be sent to the encoder in one SPI transaction. In the example above, 3 additional bytes are transmitted, since the application also reads encoder position data at the same time as programming the encoder. The SPI transaction can be terminated at any time with NCS signal high. If the encoder has been successfully unlocked, the fifth byte in a complete sequence must be one of the valid programming command bytes described below. If the fifth byte is not one of them, the encoder is “locked” again.

Programming command bytes

The programming command bytes vary depending on the desired programming function. The valid programming bytes are listed in the following table. Some of them perform the required task immediately (i.e. resetting the factory settings), while others require additional data bytes (i.e. setting position offset). Specific tasks are described in the following sections. When programming is complete, the programming option returns to the ‘locked’ phase.

Programming feature	Programming command byte		Additional data bytes required
	HEX	ASCII	
Setting zero offset	0x5A	'Z'	4
Multiturn counter setting	0x4D	'M'	4
Configuration parameters save	0x63	'c'	0
Reset to factory defaults	0x72	'r'	0
Triggering self-calibration	0x41	'A'	0

Setting zero offset

First transfer zero position offset into RAM. Second, send the command to store this value into non-volatile memory.
Pos_offset parameter has max value of 16383.

Byte in sequence	Programming command byte		Purpose
	HEX	ASCII	
B1	0xCD	/	Unlock sequence
B2	0xEF	/	
B3	0x89	/	
B4	0xAB	/	
B5	0x5A	'Z'	Setting zero offset command
B6	0x00	/	Pos_offset
B7	0x00	/	
B8	HH (High byte)		
B9	LL (Low byte)		

Programming example: Encoder position offset setting to 5144 (0x1418) counts

Bytes listed in the table should be sent to the encoder in exact order with at least 1 ms of delay between bytes.

Byte in sequence	Programming command byte		Purpose
	HEX	ASCII	
B1	0xCD	/	Unlock sequence
B2	0xEF	/	
B3	0x89	/	
B4	0xAB	/	
B5	0x5A	'Z'	Setting zero offset command
B6	0x00	/	Encoder position shifted for 5144 counts
B7	0x00	/	
B8	0x14	/	
B9	0x18	/	

To store offset position in a non-volatile memory, encoder's configuration parameters must be saved in accordance to "Saving configuration parameters in a non-volatile memory".

Multiturn counter

Multiturn counter value can be between 0 and 65535.
Saving to non-volatile memory is performed on power-off.

Byte in sequence	Programming command byte		Purpose
	HEX	ASCII	
B1	0xCD	/	Unlock sequence
B2	0xEF	/	
B3	0x89	/	
B4	0xAB	/	
B5	0x4D	'M'	Multiturn counter setting command
B6	0x00	/	New multiturn counter value
B7	0x00	/	
B8	HH (High byte)		
B9	LL (Low byte)		

Saving to non-volatile memory

Bytes listed in table below should be sent to the encoder in exact order.
After this sequence is completed, encoder is non-responsive for a few milliseconds.

Byte in sequence	Programming command byte		Purpose
	HEX	ASCII	
B1	0xCD	/	Unlock sequence
B2	0xEF	/	
B3	0x89	/	
B4	0xAB	/	
B5	0x63	'c'	Save configuration parameters command

Reset to factory defaults

Command will reset zero position offset to 0 and self-calibration parameters to factory defaults.
Bytes listed in table below should be sent to the encoder in exact order.
After this sequence is completed, encoder is non-responsive for a few milliseconds.

Byte in sequence	Programming command byte		Purpose
	HEX	ASCII	
B1	0xCD	/	Unlock sequence
B2	0xEF	/	
B3	0x89	/	
B4	0xAB	/	
B5	0x72	'r'	Reset to factory defaults command

Triggering self-calibration

The self-calibration function eliminates eccentricity-caused error, which is a dominant part of the encoder accuracy and is caused by the eccentric mounting of the ring. This function removes the one sine wave per revolution error.
If multiturn counter is being used in the encoder and if rotational speed is higher than ±300 RPM, it may have incorrect value after self-calibration. Multiturn error flag will be set in such case.

Requirements:

- Free mechanical rotation for full turn (360°)
- Good signal throughout the calibration angle
- Maximum available time is 10 seconds
- Direction is not important
- Maximum speed during self-calibration is 600 RPM
- Self-calibration must be started when no error is present (green LED)
- LED must be visible for checking self-calibration status

Byte in sequence	Programming command byte		Purpose
	HEX	ASCII	
B1	0xCD	/	Unlock sequence
B2	0xEF	/	
B3	0x89	/	
B4	0xAB	/	
B5	0x41	'A'	Triggering self-calibration command

Saving to non-volatile memory is automatic if calibration was sucessfull and LED was fast flashing green.
LED must be visible for checking self-calibration status otherwise self-calibration status can not be checked.
Rotational speed and direction during self-calibration could be uneven. The only requirement is that the shaft makes at least one complete turn during 10 second period after the command was sent.
Currently there is no command available to read the status, if self-calibration was successfully completed or not. Please observe the LED. Fast flashing green means the procedure has finished successfully.

Head office

RLS merilna tehnika d.o.o.
Poslovna cona Žeje pri Komendi
Pod vrbami 2
SI-1218 Komenda
Slovenia

T +386 1 5272100
F +386 1 5272129
E sales@rls.si
www.rls.si

Document issues

Issue	Date	Page	Corrections made
1	1. 6. 2020	-	New document