



1) Scope

This is detailed description of the commands supported by the SL900A device. The SL900A device complies to the EPC Gen2 standard. For the specification of EPC Gen2 commands, please refer to the EPC Gen2 specification version 1.2.

The Cool-log™ command set is defined in compliance with the EPC Gen2 custom command specification.

Table of contents:

- 1) Scope..... 1
- 2) Supported Commands..... 2
- 3) Supported EPC Gen2 commands..... 3
- 4) Cool-Log Custom Commands..... 5
- 5) Cool-log™ command set (custom commands)..... 7
 - a) Set Password..... 8
 - b) Set Log Mode..... 8
 - c) Set Log Limits..... 9
 - d) Get Measurement Setup..... 9
 - e) Set SFE Parameters..... 10
 - f) Set Calibration Data..... 10
 - g) End Log..... 11
 - h) Start Log..... 11
 - i) Get Log State..... 11
 - j) Get Calibration Data..... 12
 - k) Get Battery Level..... 12
 - l) Set Shelf Life..... 13
 - m) Initialize..... 13
 - n) Get Sensor Value..... 14
 - o) Open Area..... 14
 - p) Access FIFO..... 15

2) Supported Commands

Some commands can be password protected by 3 different passwords: System password (S), Application password (A) or Measurement password (M).

#	COMMAND	COMMAND CODE	SECURITY LEVEL	DEFINITION
01	QueryRep	0b00	/	EPC Gen2 anticollision round command
02	ACK	0b01	/	EPC Gen2 anticollision round command
03	Query	0b1000	/	EPC Gen2 anticollision round command
04	QueryAdjust	0b1001	/	EPC Gen2 anticollision round command
05	Select	0b1010	/	EPC Gen2 anticollision round command
06	NAK	0xC0	/	EPC Gen2 anticollision round command
07	Req_RN	0xC1	/	Request for a new 16-bit random number
08	Read	0xC2	A or M	Reads the selected block in the specified memory bank
09	Write	0xC3	A or M	Writes the selected block in the specified memory bank
10	Kill	0xC4	/	Kills the transponder – no RFID access is possible after this command (SPI remains active)
11	Lock	0xC5	/	Locks the selected memory banks
12	Access	0xC6	/	Puts the transponder to the secured state
13	BlockWrite	0xC7	A or M	Writes the selected block in the specified memory bank
14	BlockErase	0xC8	A or M	Erases the selected block in the specified memory bank
The cool-Log commands are defined as EPC custom commands. All custom commands have a 16-bit command code, where the 1 st command code is defined as 0xE0, the second command code is in the table below.				
15	Set Password	0xA0	S, M or A	Sets the passwords to EEPROM
16	Set Log Mode	0xA1	S	Sets logging mode
17	Set Log Limits	0xA2	S	Sets the measurement limits for limits logging mode
18	Get measurement setup	0xA3	S	Reads 4 system blocks - Start time, Log limits, Log mode, and Delay time + application area size
19	Set SFE parameters	0xA4	S	Sets parameter for the External sensor front end
20	Set Calibration Data	0xA5	S	Sets the calibration data for the temperature sensor and timer
21	End Log	0xA6	S	Stops the log procedure and returns the chip to Standby mode
22	Start Log	0xA7	S	Starts the timer and the selected log procedure
23	Get Log State	0xA8	S	Gets the log state of the chip
24	Get calibration data	0xA9	S	Reads the internal and external calibration data
25	Get Battery level	0xAA	/	Measures the battery voltage
26	Set Shelf Life	0xAB	/	Set the shelf life parameters
27	Initialize	0xAC	S	Initializes the chip and sets the application area size and the logging delay
28	Get Sensor Value	0xAD	/	Measures the specified sensor – temperature, ext. sensor1 or ext. sensor 2
29	Open Area	0xAE	/	Opens access to the specified EEPROM area
30	Access FIFO	0xAF	/	Reads or writes the 8-byte FIFO register (for fast SPI to RFID data transfer)

Table 1: EPC Gen2 and cool-Log™ Command Overview

3) Supported EPC Gen2 commands

QueryREP - #01

The `QUERY_REP` command instructs tags to decrement their slot counter and is specified for one out of 4 sessions. If the slot counter becomes 0 after decrementing, the tag will backscatter its RN16.

ACK - #02

When a tag receives the `ACK` command in the Reply state, it will transition to the Acknowledged state and backscatter the EPC. The EPC can be truncated if this has been requested by the reader in the `SELECT` command. The `ACK` command can also be processed in the Open or Secured states, but in this case no state transition will occur.

Query - #03

The `QUERY` command initiates and specifies an inventory round. It sets the TX and RX data rates. It also defines the number of slots used for the inventory round. When the tag receives the `QUERY` command it will calculate a random RN16 if it has a matching Sel and Target. The tag will backscatter the RN16 value in case the slot counter is loaded with 0.

QueryAdjust - #04

The `QUERY_ADJUST` command increments or decrements the Q number (number of slots) for the current inventory round.

Select - #05

The `SELECT` command selects a tag population that will participate in the inventory round, based on user-defined criteria. The tag can receive any number of successive `SELECT` commands.

NAK - #06

When a tag receives the `NAK` command it will transition to the Arbitrate state, unless it is in the Kill or Ready states. The tag will not send any reply to the `NAK` command.

Req_RN - #07

The `REQ_RN` command will instruct the tag to backscatter a new RN16. When a tag in the Acknowledged state receives a correct `REQ_RN` command it will transition to the Open or Secured state. When the tag is in the Open or Secured state it will backscatter a new RN16 and no state transition will occur.

Read - #08

The Read command instructs the tag to read and backscatter a part or all of the Reserved, EPC, TID or User memory.

Write - #09

The `WRITE` command allows the interrogator to write a word (16 bits) in the tags Reserved, EPC, TID or User memory. Prior to sending the Write command the interrogator has to send the `REQ_RN` command in order to receive a new RN16 that will be used for cover-coding the data by

EXOR-ing it with the RN16. In case the data writing has been successful, the tag will backscatter the response within 20ms after receiving the command.

Kill - #10

The `KILL` command is used to permanently disable a tag. When the tag receives the correct multi-step Kill procedure, it will transition to the Killed state and will not send any response thereafter.

Lock - #11

The `LOCK` command instructs the tag to lock the specified block of the EEPROM memory. The Kill and Access passwords can be Read/Write locked, while the EPC, TID and User block can only be Write locked. The command will only be executed in the Secured state.

Access - #12

The `ACCESS` command with a correct password and correct multi-step procedure instructs the tag to transition from the Open to the Secured state. When the tag has successfully received the multi-step access procedure it will backscatter its handle.

BlockWrite - #13

The `BLOCK_WRITE` command writes a *single word of data (16 bits)* to the specified memory address. It provides faster data writing than the `WRITE` command as it does not need a new RN16 for every word of data that has to be written. In case the data writing has been successful, the tag will backscatter the response within 20ms after receiving the command.

BlockErase - #14

The `BLOCK_ERASE` command erases a single word in the specified memory bank. In case the erase has been successful, the tag will backscatter the response within 20ms after receiving the command.

4) Cool-Log Custom Commands

Set Password - #15

The `SET PASSWORD` command sets the password for the specified memory area. This is the System area, Application area and Measurement area. The System area is in the Reserved memory bank. The Application and Measurement areas are in the User memory bank. In case the command has executed successfully, the tag will backscatter the response within 20ms after receiving the command.

Set Log Mode - #16

The `SET LOG MODE` command sets various parameters for the logging procedure. In case the command has executed successfully, the tag will backscatter the response within 20ms after receiving the command.

Set Log Limits - #17

The `SET LOG LIMITS` command write the 4 limits that are going to be used for logging measurement data. The limits are: Extreme upper limit, Upper limit, Lower limit and Extreme lower limit. In case the command has executed successfully, the tag will backscatter the response within 20ms after receiving the command.

Get Measurement Setup - #18

The `GET MEASUREMENT SETUP` command reads 4 system blocks - Start time, Log limits, Log mode and Delay time.

Set SFE Parameters - #19

The `SET SFE PARAMETERS` command sets the parameters for the External sensor front end.

Set Calibration Data - #20

The `SET CALIBRATION DATA` command sets the calibration values for the internal temperature sensor.

WARNING – the factory preset calibration data can be overwritten. It is advised to read the calibration data, change only the required bits and write back with the `SET CALIBRATION DATA` command.

End Log - #21

The `END LOG` command stops the logging procedure and returns the chip to passive mode. It also stops the timer.

Start Log - #22

The `START LOG` command starts the logging procedure and sets the Start time in UTC format. In logging state the chips automatically performs the measurements and data logging in the specified time intervals. Supported is also a delayed start, which means that the chip will start with the logging procedure with a specified delay after it receives the `START LOG` command. This command also starts the Interrupt mode of operation where the measurements and data-logging are driven from external events.

Get Log State - #23

The `GET LOG STATE` command gets the log state of following parameters: measurement status and out of limits counter. This gives the ability to quickly check the state of the package without the need to read the whole temperature data log.

Get Calibration Data - #24

The `GET CALIBRATION DATA` command reads the calibration data for the internal and external sensors.

Get Battery Level - #25

The `GET BATTERY LEVEL` command measures and reads the voltage level of the battery.

Set Shelf Life - #26

The `SET SHELF LIFE` command writes the shelf life algorithm parameters and enables the dynamic shelf life calculation.

Initialize - #27

The `INITIALIZE` command sets the size of the application data area and sets the delay time. The command clears the measurement status and limits counter blocks.

Get Sensor Value - #28

The `GET SENSOR VALUE` command measures and backscatters the value of the specified sensor – internal, external 1 or external 2.

Open Area - #29

The `OPEN AREA` command opens the specified area of the memory (System, Application, and Measurement). The password is stored in a RAM location and compared with the password in EEPROM. When the tag leaves the RF field this RAM location is cleared.

Access FIFO - #30

The `ACCESS FIFO` command can read or write the 8-byte FIFO. The FIFO can also be accessed from the SPI so this command can be used for fast data transfer between a microcontroller connected to the SPI and an RFID reader.

5) Cool-log™ command set (custom commands)

Upon receiving a valid command the tag always transmits a reply. If the command can not be executed, the tag replies with the following error message:

Reply structure (error):

SOF	Header	Error code	Handle	CRC	EOF
Pilot tone + preamble	1 bit [1]	8 bits	16 bits	16 bits	Dummy bit [1]

The error codes are defined as:

Error code	Error name	Error description	Condition
00000000	Other error	For error s that are not covered by the other specified error codes	
00000011	Memory overrun	The specified memory location does not exist or the EPC length field is not supported by the tag	The EBV address is outside the physical address of the EEPROM or outside the specified memory bank.
00000100	Memory locked	The specified memory location is locked and/or permalocked and can not be read or written.	The lock bit for the specified memory bank or password is set.
00001011	Insufficient power	The tag has insufficient power to perform the memory write operation.	This error code can only be set in fully passive mode when the supply voltage is to low.
10100000	Incorrect password	The password is incorrect – tag is not open.	The IDS custom password protection is active.
10100010	Battery measurement error	The battery measurement can not be started.	The tag is fully passive and there is no battery attached.
10100011	Command not allowed	Command is not allowed in active state.	Custom commands that can modify logging and calibration parameters are not allowed when the tag is in active state (RTC running).
10100110	EEPROM busy error	The memory can not be accessed as the measurement unit or SPI is accessing the EEPROM.	This error is reported when the EEPROM is used by the SPI or measurement unit.

a) Set Password

The **SET PASSWORD** command writes a 32-bit password to the EEPROM. The password protection for the specified area is automatically enabled if the password is any other value except 0.

Command structure:

SOF	Custom	Command code	Password level	Password	Handle	CRC
Frame-sync	0xE0	0xA0	8 bits	32 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The "Password level" bits are:

Password level bits:

Bits b7 – b2 are X.

b1 b0	Passw. level
0 0	Not allowed
0 1	System
1 0	Application
1 1	Measurement

When the System area is open for writing, the Set password can change the passwords for all 3 password levels. When the System area is write-protected, the Set password command can not change the System password, but it can change the Application password, if the Application area is open, and the Measurement password when the Measurement area is open.

b) Set Log Mode

The **SET LOG MODE** command sets the logging form, storage rule, enables sensors that are used in the logging process and sets the logging interval (in 1 second steps).

Command structure:

SOF	Custom	Command code	Log mode	Handle	CRC
Frame-sync	0xE0	0xA1	24 bits	16 bits	16 bits

In case the operation is successful, the following reply will be sent:

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The "Log mode" field is composed as:

Bit number	23 ... 21	20	19	18	17	16	15 ... 1	0
Function	Logging form [2:0]	Storage rule	Ext.1 sensor enable	Ext.2 sensor enable	Temp. sensor enable	Battery check enable	Log interval [14:0]	RFU

c) Set Log Limits

The SET LOG LIMITS command writes the 4 limits that are used in the logging process. All 4 limits are 10 bits long.

Command structure:

SOF	Custom	Command code	Log limits	Handle	CRC
Frame-sync	0xE0	0xA2	40 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The "Log Limits" field is composed as:

Bit number	39 ... 30	29 ... 20	19 ... 10	9 ... 0
Function	Extreme limit lower	Lower limit	Upper limit	Extreme limit upper

d) Get Measurement Setup

The GET MEASUREMENT SETUP command will read the current system setup of the chip.

Command structure:

SOF	Custom	Command code	Handle	CRC
Frame-sync	0xE0	0xA3	16 bits	16 bits

Successful Reply structure:

SOF	Header	Start Time	Log limits	Log mode	Log interval	Delay time	Application data	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	32 bits	40 bits	8 bits	16 bits	16 bits	16 bits	16 bits	16 bits	Dummy bit [1]

The "Log Limits" field is composed as:

Bit number	39 ... 30	29 ... 20	19 ... 10	9 ... 0
Function	Extreme limit lower	Lower limit	Upper limit	Extreme limit upper

The "Log mode" field is composed as:

Bit number	7 ... 5	4	3	2	1	0
Function	Logging form [2:0]	Storage rule	Ext.1 sensor enable	Ext.2 sensor enable	Temp. sensor enable	Battery check enable

The "Log interval" field is composed as:

Bit number	15 ... 1	0
Function	Log interval [14:0]	RFU

The “Delay time” field is composed as:

Bit number	15 ... 4	3 ... 2	1	0
Function	Delay time [11:0]	RFU [1:0]	Delay mode [0 – timer, 1 – external switch]	IRQ+timer enable

The “Application data” field is composed as:

Bit number	15 ... 7	6 ... 3	2 ... 0
Function	Number of words for application data [8:0]	RFU [3:0]	Broken word pointer [2:0]

e) Set SFE Parameters

The **SET SFE PARAMETERS** command writes the Sensor Front End parameters to the memory. Those parameters include the range preset values for the external sensor inputs, external sensor types and the also the sensor that will be used for limits comparison.

Command structure:

SOF	Custom	Command code	SFE parameters	Handle	CRC
Frame-sync	0xE0	0xA4	16 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The “SFE parameters” field is composed as:

Bit number	15 ... 11	10 ... 6	5 ... 4	3	2	1 ... 0
Function	Rang [4:0]	Seti [4:0]	EXT1 [1:0]	EXT2	Autorange disable	Verify sensor ID [1:0]

f) Set Calibration Data

The **SET CALIBRATION DATA** write to the calibration block in the EEPROM memory. The calibration data is preset during manufacturing, but can also be changed in the application if needed. The **SET CALIBRATION DATA** will write only to the EEPROM, but it will not update the calibration values in the calibration registers. The calibration registers are automatically updated with each **START LOG** command.

Command structure:

SOF	Custom	Command code	Calibration data	Handle	CRC
Frame-sync	0xE0	0xA5	56 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The “Calibration data” field is composed of 7 bytes (refer to device specification for detailed description).

g) End Log

The **END LOG** command stops the logging procedure and turns off the real time clock. It also clears the Active flag that is store in the “System status” field in the EEPROM.

Command structure:

SOF	Custom	Command code	Handle	CRC
Frame-sync	0xE0	0xA6	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

h) Start Log

The **START LOG** command starts the logging process. It refreshes the data in the calibration registers, enables the RTC, writes the Start time and sets the Active bit in the “System status” field in the EEPROM.

Command structure:

SOF	Custom	Command code	Start time	Handle	CRC
Frame-sync	0xE0	0xA7	32 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The “Start time” field is composed as:

Bit number	31 ... 26	25 ... 22	21 ... 17	16 ... 12	11 ... 6	5 ... 0
Function	Year [5:0]	Month [3:0]	Day [4:0]	Hour [4:0]	Minute [5:0]	Second [5:0]

i) Get Log State

The **GET LOG STATE** command reads the status of the logging process. The command can be used to quickly determine the current state of the product, together with the Shelf life and the Limit counter.

Command structure:

SOF	Custom	Command code	Handle	CRC
Frame-sync	0xE0	0xA8	16 bits	16 bits

Successful Reply structure:

SOF	Header	Limit counter	System status	SL-block 0&1	Current shelf life	Status flags	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	32 bits	32 bits	64 bits	24 bits	8 bits	16 bits	16 bits	Dummy bit [1]
				OPTIONAL - only when Shelf Life flag is set in the EEPROM					

The "Limit counter" field is composed as:

Bit number	31 ... 24	23 ... 16	15 ... 8	7 ... 0
Function	Extreme lower [7:0]	Lower [7:0]	Upper [7:0]	Extreme upper [7:0]

The "System status" field is composed as:

Bit number	31 ... 22	21 ... 16	15 ... 1	0
Function	Measurement address pointer [9:0]	Number of memory replacements [5:0]	Number of measurements [14:0]	Active

The "Status flags" field is composed as:

Bit number	7	6	5	4	3	2	1	0
Function	Active (logging process)	Measurement area full	Measurement overwritten	AD error	Low battery	Shelf life low error	Shelf life high error	Shelf Life expired

j) Get Calibration Data

The GET CALIBRATION DATA command reads the calibration data field and the SFE parameters field.

Command structure:

SOF	Custom	Command code	Handle	CRC
Frame-sync	0xE0	0xA9	16 bits	16 bits

Successful Reply structure:

SOF	Header	Calibration data and SFE parameters	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	72 bits	16 bits	16 bits	Dummy bit [1]

Please refer to the SL900A device specifications for a detailed content of the Calibration data field and the SFE parameters .

k) Get Battery Level

The GET BATTERY LEVEL command starts the AD conversion on the battery voltage and returns the voltage level with the battery type (1.5V or 3V).

Command structure:

SOF	Custom	Command code	Battery retrigger	Handle	CRC
Frame-sync	0xE0	0xAA	8 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	A/D error	Battery type	Zeros	Battery level	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	1 bit - error [1]	1 bit - [0 = 1.5V, 1 = 3V]	4 bits [0000]	10 bits	16 bits	16 bits	Dummy bit [1]

The application can also request the battery type re-check if the Battery retrigger field has the value "0000001", otherwise the Battery retrigger field needs to have the value "0000000".

l) Set Shelf Life

The **SET SHELF LIFE** command programs parameters for the dynamic shelf life algorithm.

Command structure:

SOF	Custom	Command code	SL Block 0	SL Block 1	Handle	CRC
Frame-sync	0xE0	0xAB	32 bits	32 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The "SL Block 0" field is composed as:

Bit number	31 ... 24	23 ... 16	15 ... 8	7 ... 0
Function	Tmax [7:0]	Tmin [7:0]	Tstd [7:0]	Ea [7:0]

The "SL Block 1" field is composed as:

Bit number	31 ... 16	15 ... 6	5 ... 4	3	2	1 ... 0
Function	SLinit [15:0]	Tinit [9:0]	Shelf life sensor ID [1:0]	Enable negative shelf life	Shelf life algorithm enable	RFU [1:0]

m) Initialize

The **INITIALIZE** command clears the System status field, the Limit counters and sets the Delay time field and the Application data field. The Initialize command is needed before the **START LOG** command as it will clear the pointers and counters. If the application needs to run the logging process from the previous point on, the Initialize command can be left out.

Command structure:

SOF	Custom	Command code	Delay time	Application data	Handle	CRC
Frame-sync	0xE0	0xAC	16 bits	16 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The "Delay time" field is composed as:

Bit number	15 ... 4	3 ... 2	1	0
Function	Delay time [11:0]	RFU [1:0]	Delay mode [0 – timer, 1 – external switch]	IRQ+timer enable

The "Application data" field is composed as:

Bit number	15 ... 7	6 ... 3	2 ... 0
Function	Number of words for application data [8:0]	RFU [3:0]	Broken word pointer [2:0]

n) Get Sensor Value

The `GET SENSOR VALUE` command starts the AD conversion on the specified sensor and returns the value.

Command structure:

SOF	Custom	Command code	Sensor type	Handle	CRC
Frame-sync	0xE0	0xAD	8 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	A/D error	Range/Limit	Sensor value	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	1 bit - error [1]	5 bits RANGE - for external sensors LIMIT CURRENT - for self heating compensation	10 bits	16 bits	16 bits	Dummy bit [1]

The "Sensor type" field is composed as

Bit number	7 ... 2	1 ... 0
Function	RFU [5:0] – all 0's	Sensor type: 00 – temperature sensor 01 – External sensor 1 10 – External sensor 2 11 – Battery voltage

o) Open Area

The `OPEN AREA` command opens the specified area (System, Application, and Measurement) that is protected by a password.

Command structure:

SOF	Custom	Command code	Password level	Password	Handle	CRC
Frame-sync	0xE0	0xAE	8 bits	32 bits	16 bits	16 bits

Successful Reply structure:

SOF	Header	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	16 bits	16 bits	Dummy bit [1]

The "Password level" field is composed as:

Password level bits:

Bits b7 – b2 are X.

b1 b0	Passw. level
0 0	Not allowed
0 1	System
1 0	Application
1 1	Measurement

p) Access FIFO

The `ACCESS FIFO` command can read and write data from the FIFO and can also read the FIFO status register.

Command structure:

SOF	Custom	Command code	Subcommand	Payload	Handle	CRC
Frame-sync	0xE0	0xAF	8 bits	0 ~ 8 bytes	16 bits	16 bits

Successful Reply structure:

SOF	Header	Payload	Handle	CRC	EOF
Pilot tone + preamble	1 bit [0]	0 ~ 8 bytes (data from FIFO or FIFO status register)	16 bits	16 bits	Dummy bit [1]

Possible Subcommand codes are defined as:

Subcommand bits 7 6 5	Function	Comment
1 0 0	Read data from FIFO	The bits 3-0 specify the number of bytes that will be read from FIFO
1 0 1	Write data to FIFO	The bits 3-0 specify the number of bytes that will be written to FIFO
1 1 0	Read status register	

The FIFO status register is defined as:

Bit #	Function
7	FIFO busy
6	Data ready
5	No data
4	0 – data from SPI, 1 – data from RFID
3	Number of valid bytes in FIFO register (0000 – FIFO empty, 0001 – 1 byte, 1000 – 8 bytes)
2	
1	
0	

Access FIFO command example:

Frame sync + E0 AF A5 11 22 33 44 55 +
Handle + CRC

This example command will write 5 bytes to the FIFO.

Sales & Marketing Europe

IDS Microchip AG
Wächlenstrasse 5,
CH-8832 Wollerau, Switzerland
Phone: +41 43 8446-253
Fax: +41 43 8446-250
sales@ids-microchip.com

Sales & Marketing Americas

IDS Microchip,
Toronto Office
131 Yorkminster Rd.
North York, ON M2N 2Z5, Canada
Phone: +416 227 9196
Fax: +41 43 8446-250
sales@ids-microchip.com

www.ids-microchip.com

Preliminary is product information in formative or design phase of development. Characteristic data and other specifications are design goals. IDS Microchip reserves the right to change or discontinue such products without notice.

The information furnished here by IDS is believed to be correct and accurate as of the publication date. However, IDS shall not be liable to any third party for any damages, including but not limited to personal injury, property damage, loss of profits, loss of use, interruption of business or indirect, special, incidental, or consequential damages of any kind in connection with or arising out of the furnishing, performance, or use of the technical data. No obligation or liability to any third party shall arise or flow out of IDS' rendering technical or other services.

Products sold by IDS Microchip are covered exclusively by the IDS standard warranty, patent indemnification, and other provisions appearing in IDS standard "Terms of Sale". Testing and other quality control techniques are used to the extent IDS deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed. IDS makes no warranty (express, statutory, implied and/or by description), including without limitation any warranties of merchantability and/or fitness for a particular purpose, regarding the information set forth in the Materials pertaining to IDS products, or regarding the freedom of any products described in the Materials from patent and/or other infringement.

IDS Microchip reserves the right to discontinue production and change specifications and prices, make corrections, modifications, enhancements, improvements and other changes of its products and services at any time without notice. IDS Microchip's products are intended for use in commercial applications. Applications requiring extended temperature range, unusual environmental requirements, or high reliability applications, such as military, medical life-support or life-sustaining equipment, are specifically not recommended without additional mutually agreed upon processing by IDS for such applications. IDS assumes no liability for application assistance or customer product design. Customers are responsible for their products and applications using IDS components.

cool-Log™ is a registered trademark of IDS Microchip AG, Switzerland.