





DATE: 10/Oct./2006

## Communication Protocol Specification For SCIP2.0 Standard

	All Page Modified				2008/03/21	Kawata	PR5446
 X 8	Web Link, LED blinking after changed to SCIP2.0, MDMS-Command explanation, Bit rate as a standard.			2,4,6,9	2007/08/03	Kawata	PR5325
 X 6	Published paper link, URG_Configurer, string character explanation, encode example			2,3,4,	2006/12/08	Kawata	PR5205
 X 3	Explanation on F/W update mode			3, 12	2006/11/01	Kawata	PR5181
SYMBOL	AMENDED REASON			PAGES	DATE	AMENDED	DWG.NO
APPROVED	CHECKED	DRAFTED	DESIGNED	TITLE	Communication Protocol Specification For SCIP2.0 Standard		
Mori	Maejima	Kawata	Kawata	DRAWING NO.			
					<b>C-42-03320B</b>		1/25

## Table of Contents

1. Introduction
  2. Switching to SCIP2.0
  3. Changing the Default Settings
  4. Interface and Settings
  5. Measurement Direction and Points
  6. Data Encoding and Decoding
    - 2-Character Encoding
    - 3-Character Encoding
    - 4-Character Encoding
  7. Communication Format
  8. Sensor Commands
    - MDMS-Command
    - GDGS-Command
    - BM-Command
    - QT-Command
    - RS-Command
    - TM-Command
    - SS-Command
    - CR-Command
    - HS-Command
    - DB-Command
    - VV-Command
    - PP-Command
    - II Command
  9. Response to Invalid Commands
- Annex A

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	2/25
-------	--	----------------	--------------------	------

## 1. Introduction

SCIP2.0 standard is developed by sensor interface research group<sup>\*1</sup> with the aim of providing flexible and efficient sensor interfacing for robotic applications. This document describes communication system in SCIP2.0 and applicable to all sensors<sup>\*2</sup> compatible with this standard.

\*1 Intelligent Robot Laboratory, University of Tsukuba (<http://www.roboken.esys.tsukuba.ac.jp>)

\*2 For details see, [H. Kawata, W. Santosh, T. Mori, A. Ohya and S. Yuta, "Development of ultra-small lightweight optical range sensor system", IEEE/RSJ International Conference on Intelligent Robots and Systems \(IROS2005\), pp.3277-3282 2005-8 Edmonton](#)

## 2. Switching to SCIP2.0 (For URG-04LX)

If the sensor is compatible to SCIP2.0 but currently running on SCIP1.1 it is necessary to switch the sensor mode to use SCIP2.0 system. Such cases are, when the dispatched sensor is set to run in SCIP1.1 by default or after updating sensor with SCIP2.0 compatible firmware. Switching is not necessary in sensor models that run in SCIP2.0 mode by default.

Use the following SCIP2.0 Switching Command to change the sensor mode. Sensor will accept SCIP2.0 commands only after successfully switching the mode. Sensor will return to SCIP1.1 on restart thus it is necessary to transmit this command whenever user wants to use SCIP2.0 system.

### [SCIP2.0 Switching Command]

(HOST → SENSOR)

S	C	I	P	2	.	0	LF
---	---	---	---	---	---	---	----

(SENSOR → HOST)

S	C	I	P	2	.	0	LF	Status	LF	LF
---	---	---	---	---	---	---	----	--------	----	----

- Status: 0 --- Changed to SCIP2.0 successfully.
- LF : Line Feed

### NOTE:

Communication system in SCIP1.1 and SCIP2.0 are not inter-compatible. User should exclusively use single system to avoid errors.

## 3. Changing the Default Settings

Sensor parameter will return to default setting values whenever the sensor is restarted. Some of these parameters can be changed from the initially set values to newly defined values so that the sensor will retain them during restart. To change the settings download and use "URG\_Configurer.exe" application from the company's website (<http://www.hokuyo-aut.jp/login/index.html>). Table 1 shows the sensor compatibility for setting change.

Table 1: Compatibility to Change the Default Parameters

	URG-04LX	UBG-04LX-F01	UHG-08LX	UTM-30LX
Change from SCIP1.1 to SCIP2.0 (Non Reversible)	YES	NO	NO	NO
RS232C Default Communication Speed Change	YES	YES	NO	NO
Default Motor Speed Change	YES	YES	YES	NO

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	3/25
-------	--	----------------	--------------------	------



## 4. Interface and Setting

Sensor is equipped with RS232C and/or USB for interfacing with an external device, the Host. Communication can be done via any one of these interfacing channels. In sensors having both interfaces if host and sensor is connected with both channels, USB connection will have the priority. It is also possible to switch between USB and RS232C connection by plugging and unplugging the USB cable on the sensor side even when the sensor is operating.

USB has Communication Device Class (CDC) standard with its settings similar to RS232C. Programs written for RS232C can also be used for USB.

When using USB as an interfacing device it is necessary to install USB driver on the host. Driver for Windows operating system can be downloaded from the company website.

<http://www.hokuyo-aut.jp/login/index.html>

For Macintosh operating system standard built in driver can be used (Mac OS X Leopard or later versions only). See URG programming guide for details.

For Linux systems CDC-ACM can be used. For details contact your Linux distributor.

See URG-Programming guide<sup>\*3</sup> for details.

*\*3 Scheduled to be released around April 2008.*

Settings details for RS232C and USB are given below.

### RS232C

Bit Rate :19.2, 38.4<sup>\*4</sup>, 57.6, 115.2, 250, 500, 750 Kbps  
Parity :None  
Data Bit :8  
Stop Bit :1  
Flow Control :None

### USB

Version :2.0  
Speed :Full Speed (9Mbps<sup>\*5</sup>)  
Class :Communication Device Class

*\*4 Not compatible with URG-04LX*

*\*5 Compatible with CDC-ACM Ver0.25 or later versions in Linux operating system.*

### Note on USB Connection:

When using USB connection, port should be opened only after the OS assigns the number to the device (enumeration). Access to the device from the application should be done only when the host-device configuration is complete and host recognizes the device.

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	4/25
-------	--	----------------	--------------------	------

## 5. Measurement Direction and Data Points

This section gives some basic information on sensor's measurement parameters. These parameters are important when reading the measurement data from sensor.

Figure 1 shows sensor's measurement details. The scanner rotates in an anti-clockwise direction when viewed from top. Detection Range (E) is maximum angle the sensor scans for measurement. Angular Resolution is defined as the 360degree divided by the Slit Division (F).

Measurement points are called Steps. Step 0 is the first measurement point. Step A is the initial measurement point in the detection range. Step B is the sensor front step. Step C is the end point of the detection range. Step D is the last measurement point. Table 1 shows the measurement parameters of some sensor models.

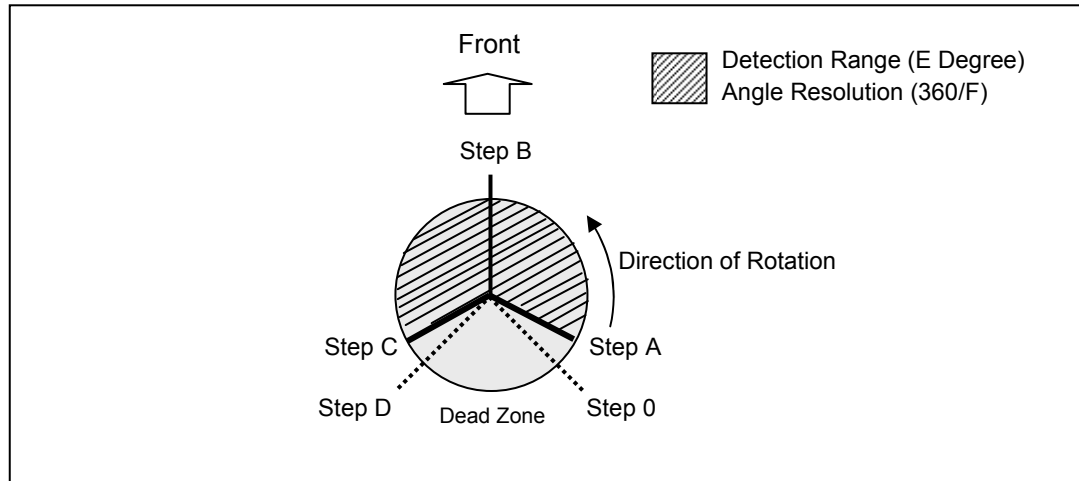


Figure 1: Measurement Parameters

Table 2: Measurement Parameters of Sensor Models

		URG-04LX	UBG-05LX-F01	UHG-08LX	UTM-30LX	UTM-30LX (Sample)
Step 0	First Measurement Point	0	0	0	0	0
Step A	Initial measurement Step of Detection Range	44	44	0	0	0
Step B	Sensor Front Step	384	384	384	540	562 <sup>*6</sup>
Step C	End point of Detection Range	725	725	768	1080	1100
Step D	Last Measurement Point	768	768	768	1080	1120
E	Detection Range	239.77	239.77	270.35	270.25	282.00
F	Slit Division	1024	1024	1024	1440	1440

<sup>\*6</sup> PP-Command shows 560 as a center step in sample version instead of 562.

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	5/25
-------	--	----------------	--------------------	------

## 6. Data Encoding and Decoding

Sensor's data are encoded to reduce the transmission time between host and sensor. These data should be decoded at the host side before processing them. There are three types of encoding technique applied in the sensor depending upon the data size.

- 5.1 Two-Character Encoding
- 5.2 Three-Character Encoding
- 5.3 Four-Character Encoding

### 6.1 Two-Character Encoding

This encoding technique is applied to express data having maximum length of 12 bits. Encoding is done by separating data into upper and lower 6 bits and then 30H is added to convert them into ASCII characters. Figure 2 and 3 shows the Two-Character encoding and decoding example.

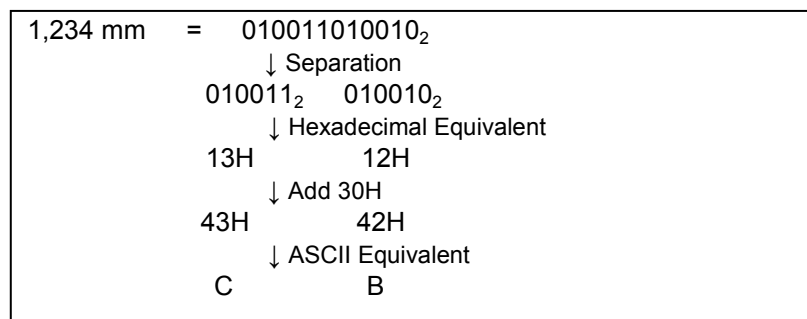


Figure 2: 2-Character Encoding Example

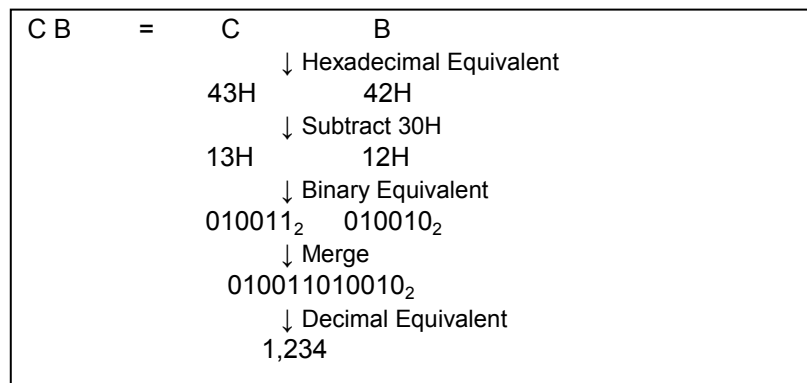


Figure 3: 2-Character Decoding Example

## 6.2. Three-Character Encoding

This encoding technique is applied to express data having maximum length of 18 bits. Encoding is done by separating data into upper, middle and lower 6 bits and then 30H is added to convert them into ASCII characters. Figure 4 and 5 shows the Three-Character encoding and decoding example.

5,432 mm	=	1010100111000 <sub>2</sub>
		↓ Separation
		000001 <sub>2</sub> 010100 <sub>2</sub> 111000 <sub>2</sub>
		↓ Hexadecimal Equivalent
		1H 14H 38H
		↓ Add 30H
		31H 44H 68H
		↓ ASCII Equivalent
		1 D h

**Figure 4: 3-Character Encoding Example**

1 D h	=	1	D	h
			↓ Hexadecimal Equivalent	
		31H	44H	68H
			↓ Subtract 30H	
		1H	14H	38H
			↓ Binary Equivalent	
		000001 <sub>2</sub>	010100 <sub>2</sub>	111000 <sub>2</sub>
			↓ Merge	
		000001010100111000 <sub>2</sub>		
		↓ Decimal Equivalent		
		5,432		

**Figure 5: 3-Character Decoding Example**

### 6.3. Four-Character Encoding

This encoding technique is applied to express data having maximum length of 24 bits. Encoding is done by separating data into four parts of 6 bits each and then 30H is added to convert them into ASCII characters. Figure 6 and 7 shows the Four-Character encoding and decoding example.

16,000,000 msec = 111101000010010000000000 <sub>2</sub>			
↓ Separation			
111101 <sub>2</sub>	000010 <sub>2</sub>	010000 <sub>2</sub>	000000 <sub>2</sub>
↓ Hexadecimal Equivalent			
3DH	2H	10H	0H
↓ Add 30H			
6DH	32H	40H	30H
↓ ASCII Equivalent			
m	2	@	0

Figure 6: 3-Character Encoding Example

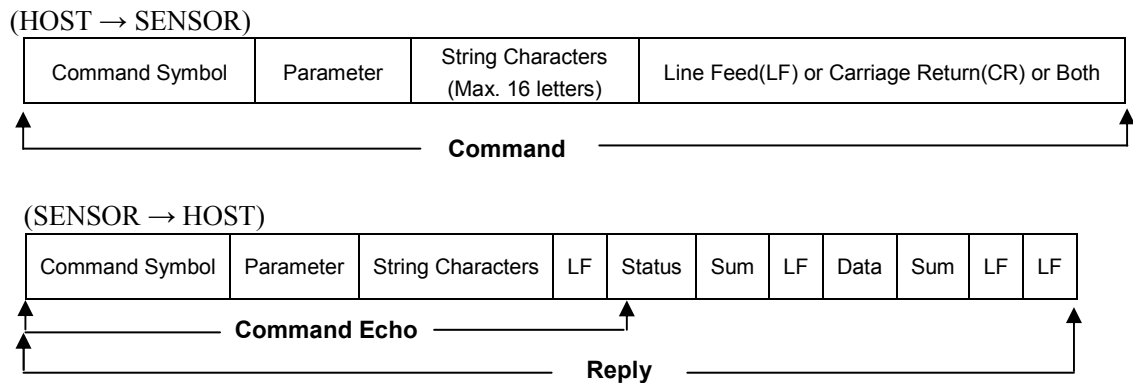
m2@0	=	m	2	@	0
↓ Hexadecimal Equivalent					
6DH		32H		40H	30H
↓ Subtract 30H					
3DH		2H		10H	0H
↓ Binary Equivalent					
111101 <sub>2</sub>		000010 <sub>2</sub>		010000 <sub>2</sub>	000000 <sub>2</sub>
↓ Merge					
111101000010010000000000 <sub>2</sub>					
↓ Decimal Equivalent					
16,000,000					

Figure 7: 3-Character Decoding Example



## 7. Communication Format

Sensor and host exchange data using set of predefined commands. These commands have specific format known as communication format. Figure 8 and 9 shows the basic communication format between host and sensor in SCIP2.0. Communication is initiated from host by sending a **Command** that consists of Command-Symbol, Parameter, String Characters followed by line feed or carriage return or both. When sensor receive commands it sends **Reply** with the **Command Echo**, status, sum, line feed, data related to the command, sum and two continuous line feed as a termination code.



- **Command Symbol** is 2 bytes code at the beginning of every command. Each command has specific symbols for verification.
- **Parameter** is information that is required to change sensor settings or to request the additional data.
- **String Characters** are optional information in the command. They are used to verify the reply when same command is repeated more than once, such as by sending different String Characters in each command and checking the command echo. Maximum length of the String Characters is 16-letters made with combination of any English letters, numbers, blank space and symbols '.', '\_', '+', '@'. They must be separated by semicolon- ';' -at the beginning to separate from parameter.
- **Line Feed (LF) or Carriage Return (CR)** is terminating code. Command can have LF or CR or both as termination code but reply will always have two continuous LF as its termination code.
- **Status** is 2 bytes data in the reply that informs normal processing if command is authenticated or errors if undefined, invalid or incomplete command is received by sensor. Status other than 00 and 99 are error codes
- **Sum** is 1 byte data used in authentication. It is calculated by adding data between two linefeeds, taking the lower 6 byte of this sum and finally adding 30H to this sum.

**Example:**

$$[\text{LF}] \text{Hokuyo} [\text{LF}] = 48\text{H} + 6\text{fH} + 6\text{bH} + 75\text{H} + 79\text{H} + 6\text{fH} = 27\text{fH} = 1001 \text{ } 111111_2$$

$$\text{Sum} = 111111_2 = 3\text{fH} + 30\text{H} = 6\text{fH} = \text{o}$$

- **Data** is main information related to the command. It is separated by LF and sum after every 64 bytes if exceeds 64 bytes.
- **IMPORTANT:** \$(24H) is a reserved letter for special mode. Do not use it in the commands.

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	9/25
-------	--	----------------	--------------------	------

## 8. Sensor Commands

Sensor commands are predefined codes in communication format (see section 6). Host and sensor exchange data using these commands. There are 13-types of predefined sensor commands in SCIP2.0.

### NOTE:

- Host can send multiple commands at one time to sensor and sensor replies to each command progressively. However sensor will not accept multiple commands of same type sent at once. In such case sensor will reply only once and sends error code thereafter.
- Comparing the command echo and sum can identify errors in communication.
- All the characters in the **Command** and **Reply** are ASCII code.

### 8.1 [MDMS-Command]

This is a sensor data acquisition command. Whenever sensor receives this command it transmits the eco with status '00' followed by the reply having measurement data that was taken after the command was received. Laser switches on automatically before the measurement and switched off after completing the number of scans defined in the command.

(HOST→ SENSOR)

M (4dH)	D (44H) or S (53H)	Starting Step (4bytes)	End Step (4 bytes)	Cluster Count (2bytes)	Scan Interval (1 byte)
Number of Scans (2 bytes)	String Characters (max 16-letters)	LF (1 byte)			

- Command symbol can be either MD or MS. MD is for three character encoded data and MS is for two character encoded data (see section 5).
- Starting Step and End Step:  
Starting step and End Step can be any points between 0 and maximum step (see section 4). End Step should be always greater than Starting step.

#### Example:

To obtain data from step 10 to 750.

Starting point : 0010 (30H,30H, 31H, 30H)

End Point : 0750 (30H, 37H, 35H, 30H)

- Cluster Count:  
Cluster Count is the number of adjacent steps that can be merged into single data and has a range 0 to 99. When cluster count is more than 1, step having minimum measurement value (excluding error) in the cluster will be the output data.

**Example:** If Cluster Count is 3 and measurement values of 3 adjacent steps in this cluster are 3059, 3055 and 3062, the received data from the sensor will be 3055.

- Scan Interval and  
Skipping the number of scans when obtaining multiple scan data can be set in Scan Interval. The value should be in decimal.
- Number of Scans:  
User can request number of scan data by supplying the count in Number of Scan. If Number of Scan is set to 00 the data is supplied indefinitely unless canceled using [QT-Command] or [RS-Command]. The value should be in decimal.

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	10/25
-------	--	----------------	--------------------	-------



(SENSOR → HOST)

When Status is not '99' or '00'

M	D or S	Starting Step	End Step	Cluster Count	Scan Interval
Number of Scans		LF	String Characters	LF	
Status	Sum	LF	LF		

(SENSOR → HOST)

M	D or S	Starting Step	End Step	Cluster Count	Scan Interval
Number of Scans		LF	String Characters	LF	
0	0	P	LF	LF	

Transmitted only once even for commands with multiple numbers of scan after receiving MDMS-Command during normal operation.

When Data is less than 64 bytes

M	D or S	Starting Step	End Step	Cluster Count	Scan Interval
Remaining Scans		LF	String Characters	LF	
9	9	b	LF	Time Stamp (4byte)	Sum LF
Data	Sum	LF	LF		

When data is more than 64 bytes and terminates without remaining bytes

M	D or S	Starting Step	End Step	Cluster Count	Scan Interval
Remaining Scans		LF	String Characters	LF	
9	9	b	LF	Time Stamp (4byte)	Sum LF
Data Block 1 (64 byte)			Sum	LF	
-----			Sum	LF	
Data Block N (64 byte)			Sum	LF	LF

When data is more than 64 bytes and terminates with remaining bytes

M	D or S	Starting Step	End Step	Cluster Count	Scan Interval
Remaining Scans		LF	String Characters	LF	
9	9	b	LF	Time Stamp (4byte)	Sum LF
Data Block 1 (64 byte)			Sum	LF	
-----			Sum	LF	
Data Block N-1 (64 byte)			Sum	LF	
Data Block N (n byte)			Sum	LF	LF

- When the host requests multiple measurement data, reply from the sensor will be the number of remaining scan (Remaining Scans) in the echo instead of Number of Scan.
- Status:
  - 00 --- Command received without any Error
  - 01 --- Starting Step has non-numeric value.
  - 02 --- End Step has non-numeric value.
  - 03 --- Cluster Count has non-numeric value.
  - 04 --- End Step is out of range.
  - 05 --- End Step is smaller than Starting Step.
  - 06 --- Scan Interval has non-numeric value.
  - 07 --- Number of Scan has non-numeric value.
  - 21~49 --- Processing stopped to verify the error.
  - 50~97 --- Hardware trouble (such as laser, motor malfunctions etc.)
  - 98 --- Resumption of process after confirming normal laser operation.
- Data: If measurement data exceeds the maximum range the output will be the maximum value (ex. 5600)

**Note:**

During continuous measurement if diagnostics suspects some malfunction, data processing will be temporarily stopped to confirm the error. Confirmation may take up to 10 sec, during this period the status will be 21~49. If no malfunction is detected sensor resumes the measurement after transmitting the status 98. Otherwise it will transmit the corresponding error status (see Annex A for details).

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	11/25
-------	--	----------------	--------------------	-------

- Time Stamp:

Sensor has 24bit internal timer with 1msec resolution. Time stamp is a timer value at 0<sup>th</sup> step of every scan (Figure 1). Received Time Stamp is Four-Character encoded data (see section 5).

**Note:** Measurement data may contain error codes under certain circumstances. Table 3 and 4 shows the list of error codes and error details.

Table 3: ERROR CODES (URG-04LX, UBG-04LX-F01, UHG-08LX)

Error Code	Error Details
0	Detected object is possibly at 22m
1	Reflected light has low intensity
2	Reflected light has low intensity
6	Others
7	Distance data on the preceding and succeeding steps have errors
8	Intensity difference of two waves
9	The same step had error in the last two scan
10~17	Others
18	Error reading due to strong reflective object
19	Non-Measurable Step

Table 4: ERROR CODES (UTM-30LX)

Error Code	Error Details
1	No object in the range
2	Object is too near (Internal Error)
3	Measurement Error (May be due to Interference)
4	Object out of range ( At the near end)
5	Other Errors

## 8.2 [GDGS-Command]

Whenever sensor receives this command it supplies the latest measurement data to the host. If the laser is switched off, it should be switched on by sending BM-Command (see section 7.2) before the measurement. Laser should be switched off if necessary by sending QT-Command (see section 7.3) after measurement is complete.

(HOST → SENSOR)

G (47H)	D (44H) or S (53H)	Starting Step (4bytes)	End Step (4 bytes)	Cluster Count (2bytes)	String Characters	LF
---------	--------------------	---------------------------	-----------------------	---------------------------	-------------------	----

- Parameters are same as explained in MDMS-Command (see section 7.1).

(SENSOR → HOST)

- When status is not 00

G	D or S	Starting Step	End Step	Cluster Count	String Characters	LF
Status	Sum	LF	LF			

- When data is less than 64 bytes

G	D or S	Starting Step	End Step	Cluster Count	String Characters	LF
0	0	P	LF	Time Stamp	Sum	LF
Data	Sum	LF	LF			

- When data is more than 64 bytes and terminates without remaining bytes

G	D or S	Starting Step	End Step	Cluster Count	String Characters	LF
0	0	P	LF	Time Stamp	Sum	LF
Data Block 1 (64 bytes)	Sum	LF				
-----	Sum	LF				
Data Block N (64 bytes)	Sum	LF	LF			

- When data is more than 64 bytes and terminates with n-remaining bytes

G	D or S	Starting Step	End Step	Cluster Count	String Characters	LF
0	0	P	LF	Time Stamp	Sum	LF
Data Block 1 (64 bytes)	Sum	LF				
-----	Sum	LF				
Data Block N-1 (64 bytes)	Sum	LF				
Data Block N (n bytes)	Sum	LF	LF			

- Status:

- 01 --- Starting Step has non-numeric value.
- 02 --- End Step has non-numeric value
- 03 --- Cluster Count has non-numeric value.
- 04 --- End Step is out of range
- 05 --- End Step is smaller than Starting Step.
- 10 --- Laser is off.
- 50~98 --- Hardware trouble (such as laser, motor malfunction etc.)

- Time Stamp:

Same as in MDMS-Command (see section 7.1)

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	13/25
-------	--	----------------	--------------------	-------



### 8.3 [BM-Command]

This command will illuminate the sensor's laser enabling the measurement. When sensor is switched on in SCIP2.0 mode or switched to SCIP2.0 by command the laser is initially in off state by default. In this state sensor can not perform the measurement. Laser state can be verified by green LED on the sensor. Laser is off if the LED blinks rapidly and it is ON when LED glows continuously.

(HOST → SENSOR)

B (42H)	M (4dH)	String Characters	LF
---------	---------	-------------------	----

(SENSOR → HOST)

B	M	String Characters	LF	Status	Sum	LF	LF
---	---	-------------------	----	--------	-----	----	----

● Status:

- 00 --- Command received without any Error
- 01 --- Unable to control due to laser malfunction.
- 02 --- Laser is already on.

● Note:

Laser is initially switched off and sensor's measurement state is disabled by default in SCIP2.0 system.

### 8.4 [QT-Command]

This command will switch off the laser disabling sensor's measurement state.

(HOST → SENSOR)

Q (51H)	T (54H)	String Characters	LF
---------	---------	-------------------	----

(SENSOR → HOST)

Q	T	String Characters	LF	0	0	P	LF	LF
---	---	-------------------	----	---	---	---	----	----

### 8.5 [RS-Command]

This command will reset all the settings that were changed after sensor was switched on. This turns Laser off, sets motor speed and bit rate back to default as well as reset sensor's internal timer.

(HOST → SENSOR)

R (52H)	S (53H)	String Characters	LF
---------	---------	-------------------	----

(SENSOR → HOST)

R	S	String Characters	LF	0	0	P	LF	LF
---	---	-------------------	----	---	---	---	----	----

## 8.6 [TM-Command]

This command is used to adjust (match) the host and sensor time. Sensor should be switched to adjust mode before requesting its time and mode should be switched off after the adjustment. When the sensor is in adjustment mode laser is switched off and it will not accept any other commands unless the mode is terminated.

Sending multiple TM Command with different string lengths and comparing the time can estimate average data transmission time between sensor and host.

(HOST → SENSOR)

T (54H)	M (4dH)	Control Code (1 byte)	String Characters	LF
---------	---------	--------------------------	-------------------	----

- Control Code:
  - 0 --- Adjust mode on
  - 1 --- Time request
  - 2 --- Adjust mode off

(SENSOR → HOST)

1. When the Control Code is 0 or 2 or status is not 00.

T	M	Control Code	String Characters	LF	Status	Sum	LF	LF
---	---	--------------	-------------------	----	--------	-----	----	----

- Status:
  - 00 --- Command received without any Error
  - 01 --- Invalid Control Code.
  - 02 --- Adjust mode on command is received when sensor's adjust mode is already on
  - 03 --- Adjust mode off command is received when sensor's adjust mode is already off
  - 04 --- Adjust mode is off when requested for time.

2. When the Control Code is 1

T	M	1	String Characters	LF	0	0	P	LF	Time (4 byte)	Sum	LF	LF
---	---	---	-------------------	----	---	---	---	----	------------------	-----	----	----

- Time:
  - Sensor's internal timer reading.

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	15/25
-------	--	----------------	--------------------	-------

## 8.7 [SS-Command]

This command will change the communication bit rate of the sensor when connected with RS232C.

(HOST → SENSOR)

S (53H)	S (53H)	Bit Rate (6 byte)	String Characters	LF
---------	---------	----------------------	-------------------	----

● Bit Rate:

019200 --- 19.2 Kbps  
038400 --- 38.4 Kbps (Some sensor models may not be compatible to this speed)  
057600 --- 57.6 Kbps.  
115200 --- 115.2 Kbps.  
250000 --- 250.0 Kbps  
500000 --- 500.0 Kbps  
750000 --- 750.0 Kbps.

(SENSOR → HOST)

S	S	Bit Rate	String Characters	LF	Status	Sum	LF	LF
---	---	----------	-------------------	----	--------	-----	----	----

● Status:

00 --- Command received without any Error  
01 --- Bit Rate has non-numeric value.  
02 --- Invalid Bit Rate  
03 --- Sensor is already running at the defined bit rate.  
04 --- Not compatible with the sensor model.

● Note:

1. See sensor specification for default bit rate during shipment.
2. Bit rate change is implemented only after sensor returns the status 00.
3. When sensor is connected with USB, bit rate change will not have any effect on the communication speed, but the command will be accepted and effective when connection is changed to RS232C.

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	16/25
-------	--	----------------	--------------------	-------



## 8.8 [CR-Command]

This command is used to adjust the sensor's motor speed.

When multiple sensors are used in the same environment their motors can be made to run at different speed to avoid light interference using this command.

**Note: UTM-30LX is not compatible to this command.**

(HOST → SENSOR)

C (43H)	R (52H)	Speed Parameter (2 byte)	String Characters	LF
---------	---------	-----------------------------	-------------------	----

- Speed Parameter:
  - 00 --- Default Speed.
  - 01 ~ 10 --- Changes speed to 10 different levels.
  - 99 --- Reset to initial speed.

(SENSOR → HOST)

C	R	Speed Ratio	String Characters	LF	Status	Sum	LF	LF
---	---	-------------	-------------------	----	--------	-----	----	----

- Status:
  - 00 --- Command received without any Error
  - 01 --- Invalid Speed Ratio
  - 02 --- Speed Ratio is out of range.
  - 03 --- Motor is already running on the defined speed
  - 04 --- Incompatible with the current sensor model



## 8.9 [HS-Command]

This command will switch between high sensitivity and normal sensitivity modes. Sensor's detection ability will increase about 20% in the high sensitivity mode. However there may be chances of measurement errors due to strong reflective objects near 22m.

**Note: This command is not compatible in all versions of UTM-30LX and URG-04LX with firmware versions earlier than 3.2.00.**

(HOST → SENSOR)

H (48H)	S (53H)	Parameter (1 byte)	String Characters	LF
---------	---------	-----------------------	-------------------	----

● Parameter:

- 0 --- Normal mode
- 1 --- High sensitivity mode

(SENSOR → HOST)

S	S	Bit Rate	String Characters	LF	Status	Sum	LF	LF
---	---	----------	-------------------	----	--------	-----	----	----

● Status:

- 00 --- Normal Operation
- 01 --- Parameter Error
- 02 --- Already running on the set mode
- 03 --- Incompatible with current sensor model

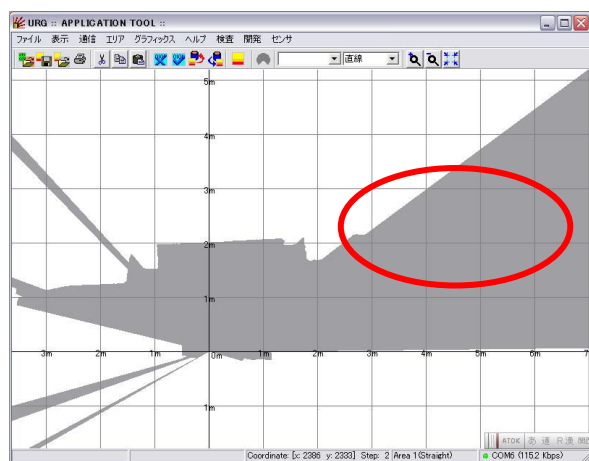


Figure 8 : Measurement in Normal Mode

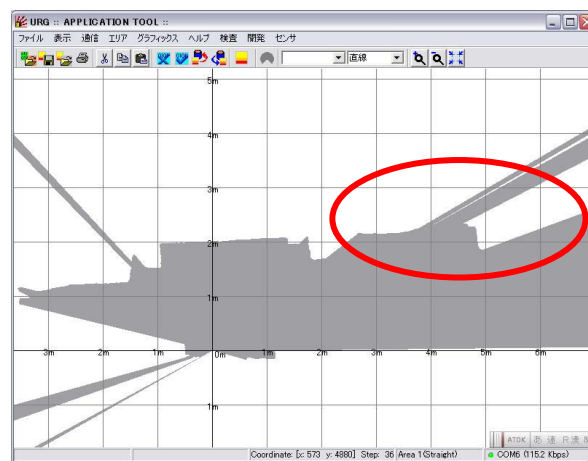


Figure 9 : Measurement in High Sensitivity Mode

## 8.10 [DB-Command]

This command will simulate the malfunction of the sensor (see Annex A). Depending upon the sent parameter sensor responds to the malfunction of MDMS, GDGS, BM and II Commands. If the parameters is '01' or '02' sensor immediately simulates the malfunction. In case the parameters is '03', '04' or '05' sensor will respond after sending the MDMS-Command having scan count more than 20 with malfunction during continuous mode.

**Note: This command is not compatible in URG-04LX with firmware versions earlier than 3.3.00.**

**Some versions of URG-04LX may not be compatible to this command.**

**Laser radiation will stop but motor keeps running during simulation.**

(HOST → SENSOR)

D (44H)	B (42H)	Parameter (2 byte)	String Characters	LF
---------	---------	-----------------------	-------------------	----

● Parameter:

- 01 --- Simulate sensor malfunction in SCIP1.1
- 02 --- Simulate sensor malfunction in SCIP2.0
- 03 --- Simulate sensor malfunction during continuous mode [Normal » Error » Normal]
- 04 --- Simulate sensor malfunction during continuous mode [Normal » Error » Malfunction]
- 05 --- Simulate sensor malfunction during continuous mode [Normal » Malfunction]
- 10 --- End simulation.

(SENSOR → HOST)

D	B	Parameter	String Characters	LF	Status	Sum	LF	LF
---	---	-----------	-------------------	----	--------	-----	----	----

● Status:

- 00 --- Normal Operation
- 01 --- Parameter Error
- 02 --- Already running on the set mode
- 03 --- Simulation mode is already ended
- 04 --- Incompatible with current sensor model
- 05 --- Sensor is in error state.



## 8.11 [VV-Command]

Sensor transmits version details such as, serial number, firmware version etc on receiving this command.

(HOST → SENSOR)

V (56H)	V	String Characters	LF
---------	---	-------------------	----

(SENSOR → HOST)

V	V	String Characters	LF	0	0	P	LF
Vendor Information		;	Sum	LF			
Product Information		;	Sum	LF			
Firmware Version		;	Sum	LF			
Protocol Version		;	Sum	LF			
Sensor Serial Number		;	Sum	LF	LF		

Example: V V [LF] 0 0 P [LF]

VEND: Hokuyo Automatic Co., Ltd;[ [LF]

PROD: SOKUIKI Sensor URG-04LX; [ [LF]

FIRM: 3.0.00, 06/10/05; m [LF]

PROT: SCIP 2.0; N [LF]

SERI: H0508486; T [LF] [LF]



## 8.12 [PP-Command]

Sensor transmits its specifications on receiving this command.

(HOST → SENSOR)

P (56H)	P	String Characters	LF
---------	---	-------------------	----

(SENSOR → HOST)

P	P	String Characters	LF	0	0	P	LF
Sensor Model				:	Sum	LF	
Minimum Measurement [mm]				:	Sum	LF	
Maximum Measurement [mm]				:	Sum	LF	
Total Number of Steps in 360° range				:	Sum	LF	
First Step of the Measurement Range				:	Sum	LF	
Last Step of the Measurement Range				:	Sum	LF	
Step number on the sensor's front axis				:	Sum	LF	
Standard motor speed [rpm]				:	Sum	LF	LF

Example: PP [LF] 0 0 P [LF]

MODL:URG-04LX(Hokuyo Automatic Co., Ltd.);b [LF]

DMIN: 20; 4 [LF]

DMAX: 5600; \_ [LF]

ARES: 1024;/ [LF]

AMIN: 44; 7 [LF]

AMAX: 725; o [LF]

AFRT: 384:6 [LF]

SCAN: 600;e [LF] [LF]



### 8.13 [II-Command]

Sensor transmits its running state on receiving this command.

(HOST → SENSOR)

I (49H)	I	String Characters	LF
---------	---	-------------------	----

(SENSOR → HOST)

I	I	String Characters	LF	0	0	P	LF
Sensor Model		;	Sum	LF			
Laser illumination state		;	Sum	LF			
Motor Speed		;	Sum	LF			
Measurement Mode		;	Sum	LF			
Bit Rate for RS232C		;	Sum	LF			
Time Stamp		;	Sum	LF			
Sensor Diagnostic		;	Sum	LF	LF		

Example: II [LF] 0 0 P [LF]

MODL: URG-04LX (Hokuyo Automatic Co., Ltd.);b [LF]

LASR: OFF; 7 [LF]

SCSP: default(600[rpm])<-Default setting by user; / [LF]

MESM: IDLE;; [LF]

SBPS: 19200[bps]<-Default setting by user; A [LF]

TIME: 002AA9;f [LF]

STAT: Sensor works well;8 [LF][LF]

## 9. Response to Invalid Commands

Whenever sensor receives undefined commands or without valid parameters in it, the error status will have one of the following values.

- Error Status: 0A --- Unable to create transmission data or reply command internally.  
0B --- Buffer shortage or command repeated that is already processed  
0C --- Command with insufficient parameters 1  
0D --- Undefined Command 1  
0E --- Undefined Command 2  
0F --- Command with insufficient parameters 2.  
0G --- String Character in command exceeds 16 letters  
0H --- String Character has invalid letters.  
0I --- Sensor is now in firmware update mode\*.

\*Firmware update mode (For URG-04LX):

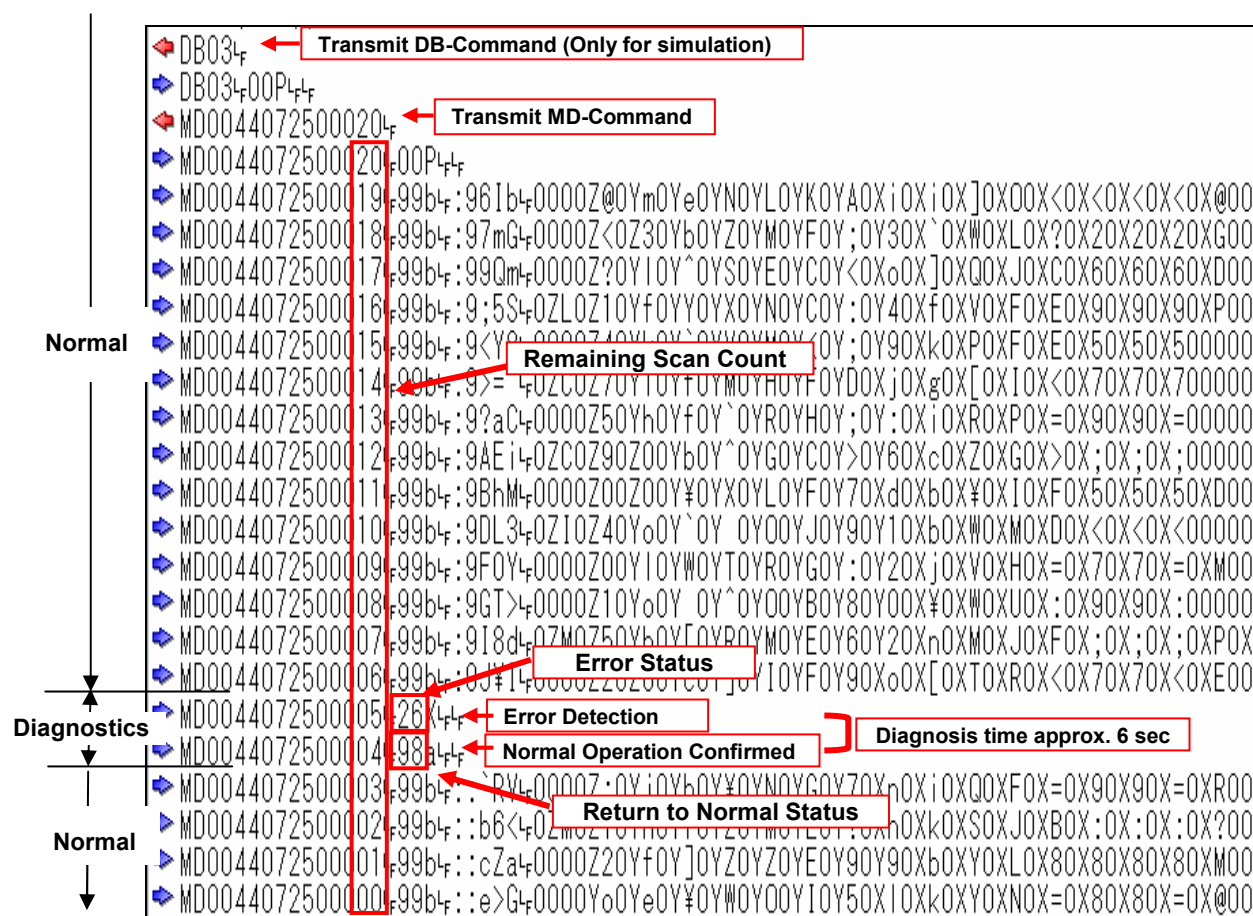
Sensor will be sent to firmware update mode by special command before updating the new firmware. During this state motor and laser is stopped. When sensor is in firmware update mode it accepts only [VV-Command], [II-Command], [RS-Command] and [SCIPVer2.0 Changing Command]. It will send error status 0I to any other commands. To confirm the state user can send [II-Command] which will have the following reply “UDST:F/W update mode (CPU clock is 6MHz)”.

If sensor goes to this state unintentionally, send [RS-Command] and restart the sensor. If sensor remains in this state even after the restart use firmware update tool and follow the instructions in the help.

TITLE	Communication Protocol Specification For SCIP2.0	DRAWING NO.	<b>C-42-03320B</b>	23/25
-------	--	----------------	--------------------	-------



## Annex A: Example of error status during continuous measurement mode.



Normal » Diagnostics » Normal



