

Communication Protocol Document

Revision 1.1

DOCUMENT HISTORY

Ver.No	Release Date	Mod.By Mod. Dt	Auth By Auth Dt.	Remarks/ Revision details
1.1	21/05/2010	SAM	SAM	Added new commands
		21/05/2010	21/05/2010	
1.0	01/05/2010	SAM	SAM	Initial Revision
		01/05/2010	01/05/2010	

Table of contents

1.0	INTRODUCTION	3
1.1	PURPOSE OF THE DOCUMENT	3
1.2	SCOPE OF THE DOCUMENT	3
1.3	COMMUNICATION REQUIREMENTS	3
2.0	PROTOCOL PACKET FORMAT	3
2.1	PACKET FORMAT FOR HOST TO SLAVE (COMMAND MESSAGE).....	3
2.2	PACKET FORMAT FOR SLAVE TO HOST (RESPONSE MESSAGE).....	3
2.3	PACKET FIELDS DESCRIPTION	4
3.0	COMMANDS.....	4
4.0	STATUS / ERROR CODES	5
5.0	COMMAND DESCRIPTIONS	5
5.1	CMD_GET_DEV_ID (0x10) - GET DEVICE ADDRESS.....	5
5.2	CMD_GET_RTC_VALUE (0x11) – GET THE CURRENT REAL TIME CLOCK DATA.....	6
5.3	CMD_GET_RATIO (0x12) – GET THE RATIO.....	6
5.4	CMD_GET_PLANED_QTY (0x13) – GET PLANED QUANTITY OF PRODUCTS.....	7
5.5	CMD_GET_LINE_TRACK_TIME (0x14) – GET LINE TRACK TIME.....	7
5.6	CMD_GET_ACTUAL_QTY (0x15) – GET ACTUAL QUANTITY.....	8
5.7	CMD_GET_ACTUAL_TRACK_TIME (0x16) – GET ACTUAL TRACK TIME.....	8
5.8	CMD_GET_COMM_STATUS (0x17) – GET DEVICE COMMUNICATION STATUS.....	9
5.9	CMD_SET_RTC_VALUE (0x18) – SET THE CURRENT REAL TIME CLOCK DATA.....	9
5.10	CMD_SET_PLANED_QTY (0x1A) – SET PLANED QUANTITY OF PRODUCTS.....	10
5.11	CMD_SET_DEV_ID (0x1B) - SET DEVICE ADDRESS.....	10
5.12	CMD_SET_SHIFT_TIME (0x1C) – SET SHIFT START AND STOP TIMINGS.....	11
5.13	CMD_SET_SHIFT_BREAK_TIME (0x1D) – SET SHIFT BREAKS START AND STOP TIMINGS.....	12
5.14	CMD_PAUSE_PROCESSING (0x1E) - STOP CURRENT OPERATION.....	13
5.15	CMD_START_PROCESSING (0x1F) - START CURRENT OPERATION.....	13
5.16	CMD_RESET_PROCESSING (0x20) - STOP CURRENT OPERATION.....	14
5.17	CMD_GET_SHIFT_TIME (0x0E) – GET SHIFT START AND STOP TIMINGS.....	14
5.18	CMD_GET_SHIFT_BREAK_TIME (0x0F) – GET SHIFT BREAKS START AND STOP TIMINGS.....	15
5.19	CMD_SET_ACTUAL_QTY (0x21) – SET ACTUAL QUANTITY.....	15

1.0 Introduction

1.1 Purpose of the document

This document defines a communication protocol, which will be as a generic protocol for products involving data communication with each other. Basically this generic protocol serves for communication between a host and one slave device.

1.2 Scope of the document

Different aspects of the protocol will be described, which include the electrical interface, data format, and link layer. This generic protocol will be applied for link layer. The communication protocol is a packet-oriented protocol - all the data exchanged between two communication devices will be based on packet format. The data packet starts with the control character 'STX' and ends with 'ETX', which follows the 8-bit BCC checksum. Besides the checksum is used for error checking, character (byte) time-out and packet (command) time-out are used to re-synchronise the communication.

1.3 Communication requirements

Bud rate : 9600
 Data bits : 8
 Parity : None
 Stop bits : 1
 Hand Shaking : None
 Data order : Big Endian - most significant byte in the smallest address

2.0 Protocol Packet Format

There are two types of data packets. Command Message is the packet Send from the Host to the slave device. The Reply Message is the packet Send from the slave to the Host.

2.1 Packet format for Host to Slave (Command Message)

STX	STATION ID	DATA LENGTH	CMD	DATA[0..N]	BCC	ETX
-----	------------	-------------	-----	------------	-----	-----

$BCC = \text{STATION ID} \oplus \text{DATALENGTH} \oplus \text{CMD} \oplus \text{DATA}[0] \oplus \dots \oplus \text{DATA}[n]$

Where \oplus is the "EXOR".

2.2 Packet format for Slave to Host (Response Message)

STX	STATION ID	DATA LENGTH	STATUS	DATA[0..N]	BCC	ETX
-----	------------	-------------	--------	------------	-----	-----

$BCC = \text{STATION ID} \oplus \text{DATA LENGTH} \oplus \text{STATUS} \oplus \text{DATA}[0] \oplus \dots \oplus \text{DATA}[n]$

Where \oplus is the "EXOR".

2.3 Packet Fields Description

Field	Length	Description	Remark
STX	1	0xAA-'Start of Text'. It is the starting of a data packet.	
STATION ID	1	Device Address, which is used for multi-drop mode, only the slave (device) with matched pre-programmed device address will response the received command packet.	
DATA LENGTH	1	Length of the data bytes in the packet. LENGTH= Number_of_Bytes (CMD/STATUS + DATA[0..N])	The Data Length includes the CMD/STATUS and the DATA field, but not the BCC.
CMD	1	Command field: the command field consists of one command byte.	Refer the Command Table for listing of commands.
STATUS	1	Reply Status byte: The status replied from Reader to Host	This byte is only used for the Reply Packet.
DATA [0-N]	0 – 123	The Data Field is a stream of data with variable length, which depends on the Command word. There are also some command have zero length of data field.	
BCC	1	Eight-bit block check sum. The calculation of the check sum includes all the bytes within the package but excludes the STX, ETX.	
ETX	1	0xBB-'END of TEXT' –Which indicates the END of a packet.	

3.0 Commands

Commands form host to get / set the parameters of the slave

SN	Command	Hex Value	Description
1	CMD_GET_DEV_ID	0x10	Get device address
2	CMD_GET_RTC_VALUE	0x11	Get the current real time clock data
3	CMD_GET_RATIO	0x12	Get the current calculated ratio
4	CMD_GET_PLANED_QTY	0x13	Get planed quantity for the shift
5	CMD_GET_LINE_TRACK_TIME	0x14	Get line track time
6	CMD_GET_ACTUAL_QTY	0x15	Get actual counted quantity of product
7	CMD_GET_ACTUAL_TRACK_TIME	0x16	Get actual track time
8	CMD_GET_COMM_STATUS	0x17	Get device communication status
9	CMD_SET_RTC_VALUE	0x18	Set the current real time clock data
10	CMD_SET_PLANED_QTY	0x1A	Set planed quantity of products
11	CMD_SET_DEV_ID	0x1B	Set device address
12	CMD_SET_SHIFT_TIME	0x1C	Set shift start and stop timings
13	CMD_SET_SHIFT_BREAK_TIME	0x1D	Set shift breaks start and stop timings
14	CMD_PAUSE_PROCESSING	0x1E	Stop current operation
15	CMD_RESET_PROCESSING	0x20	Stop current operation
16	CMD_GET_SHIFT_TIME	0x0E	Get shift start and stop timings
17	CMD_GET_SHIFT_BREAK_TIME	0x0F	Get shift breaks start and stop timings
18	CMD_SET_ACTUAL_QTY	0x21	Set Actual quantity

4.0 Status / Error Codes

Response codes for the command sent from host to slave

SN	Status/Error code	Hex Value	Description
1	RES_COMM_OK	0x00	Received command is acceptable
2	RES_COMM_ERR	0xFF	Received command with an error
3	RES_TIME_OUT	0x80	Received command partially with time out
4	RES_UNKNOWN_ERROR	0x81	Received command with an unknown error
5	RES_BAD_PARAM	0x82	Received command with bad parameter input
6	RES_BAD_COMMAND	0x83	Received command is unknown

5.0 Command Descriptions

The section describes all the commands sent from host to slave and the responses from slave to host. All the commands are explained with an example values.

5.1 CMD_GET_DEV_ID (0x10) - Get device address.

Example:

Send data: AA 0A 01 10 1B BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 01
 CMD : 10
 DATA [] : N/A
 BCC : 1B
 ETX : BB

Response: AA 0A 02 00 0A 02 BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 02
 STATUS : 00
 DATA [0] : 0A
 BCC : 02
 ETX : BB

- Device ID = 10 (Value b/w 00 to FF)

5.2 CMD_GET_RTC_VALUE (0x11) – Get the current real time clock data.

Example:

Send data: AA 0A 01 11 1A BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 01
 CMD : 11
 DATA [] : N/A
 BCC : 1A
 ETX : BB

Response: AA 0A 07 00 16 04 0A 0B 1E 2D 2C BB (RTC in 24hr mode)

STX : AA
 STATION ID : 0A
 DATA LENGTH : 07
 STATUS : 00
 DATA[0][5] : 16 04 0A 0B 1E 2D - Rtc value = 22 04 10 11 30 45 (dd mm yy hh mm ss)
 BCC : 2D
 ETX : BB

Response: AA 0A 08 00 16 04 0A 0B 1E 2D 01 23 BB (RTC in 12hr mode)

STX : AA
 STATION ID : 0A
 DATA LENGTH : 08
 STATUS : 00
 DATA[0][6] : 16 04 0A 0B 1E 2D 01
 BCC : 23
 ETX : BB

NOTE: - Rtc value = 22 04 10 11 30 45 PM (dd mm yy hh mm ss am/pm) if am/pm = 00 = AM ,
 If am/pm = 01=PM

5.3 CMD_GET_RATIO (0x12) – Get the ratio.

Example:

Send data: AA 0A 01 12 19 BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 01
 CMD : 12
 DATA [] : NULL
 BCC : 19
 ETX : BB

Response: AA 0A 02 00 64 6C BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 02
 STATUS : 00
 DATA [0] : 64 - Ratio = 100 (Value b/w 00 to 64)
 BCC : 6C
 ETX : BB

5.4 CMD_GET_PLANED_QTY (0x13) – Get planed quantity of products.

Example:

Send data: AA 0A 02 13 01 1A BB

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 02	
CMD	: 13	
DATA [0]	: 01	-Shift number
BCC	: 1A	
ETX	: BB	

Response: AA 0A 04 00 01 27 0F 27 BB

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 04	
STATUS	: 00	
DATA [0]	: 01	-Shift number
DATA [1] [2]	: 27 0F	- Planed quantity = 9999 (Value b/w 0000 to 270F)
BCC	: 2A	
ETX	: BB	

5.5 CMD_GET_LINE_TRACK_TIME (0x14) – Get line track time.

Example:

Send data: AA 0A 01 14 1F BB

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 01	
CMD	: 14	
DATA []	: NULL	
BCC	: 1F	
ETX	: BB	

Response: AA 0A 03 00 27 0F 21 BB

AA 0A 05 00 EB 49 08 00 A5 BB

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 04	
STATUS	: 00	
DATA [0] [2]	: 01 86 9F	- Line track time = 999.99 (Value b/w 0000 to 01869F)
BCC	: 16	
ETX	: BB	

5.6 CMD_GET_ACTUAL_QTY (0x15) – Get actual quantity.**Example:****Send data: AA 0A 02 15 01 1C BB**

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 02	
CMD	: 15	
DATA [0]	: 01	-Shift number
BCC	: 1C	
ETX	: BB	

Response: AA 0A 04 00 01 27 0F 27 BB

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 04	
STATUS	: 00	
DATA [0]	: 01	-Shift number
DATA [1][2]	: 27 0F	- Actual quantity = 9999 (Value b/w 0000 to 270F)
BCC	: 27	
ETX	: BB	

5.7 CMD_GET_ACTUAL_TRACK_TIME (0x16) – Get actual track time.**Example:****Send data: AA 0A 01 16 1D BB**

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 01	
CMD	: 16	
DATA []	: NULL	
BCC	: 1D	
ETX	: BB	

Response: AA 0A 04 00 01 86 9F 16 BB

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 04	
STATUS	: 00	
DATA [0][2]	: 01 86 9F	- Actual track time = 999.99 (Value b/w 00000 to 01869F)
BCC	: 16	
ETX	: BB	

5.8 CMD_GET_COMM_STATUS (0x17) – Get device communication status.**Example:****Send data: AA 0A 03 17 AA 55 E1 BB**

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 03
CMD	: 17
DATA [0] [1]	: AA 55
BCC	: E1
ETX	: BB

Response: AA 0A 03 00 55 AA F6 BB

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 03
STATUS	: 00
DATA [0][1]	: 55 AA - Ping response value(Reverse the bytes and send back)
BCC	: F6
ETX	: BB

5.9 CMD_SET_RTC_VALUE (0x18) – Set the current real time clock data.**Example:****Send data: AA 0A 08 18 07 16 04 0A 0B 1E 2D 3D BB (RTC in 24hr mode)**

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 08
CMD	: 18
DATA[0][5]	: 07 16 04 0A 0B 1E 2D - Rtc value =sat 22 04 10 11 30 45 (dy dd mm yy hh mm ss)
BCC	: 3D
ETX	: BB

Send data: AA 0A 09 18 07 16 04 0A 0B 1E 2D 01 3D BB (RTC in 12hr mode)

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 09
CMD	: 18
DATA[0][6]	: 07 16 04 0A 0B 1E 2D 01
BCC	: 3D
ETX	: BB

NOTE: - Rtc value = sat 22 04 10 11 30 45 PM (dy dd mm yy hh mm ss am/pm) if am/pm = 00 = AM , 01=PM

Response : AA 0A 01 00 0B BB

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
STATUS	: 00
DATA []	: N/A
BCC	: 0B
ETX	: BB

5.10 CMD_SET_PLANED_QTY (0x1A) – Set planed quantity of products.**Example:****Send data: AA 0A 04 1A 01 0F 27 3D BB**

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 04	
CMD	: 1A	
DATA [0]	: 01	- Shift number
DATA [1] [2]	: 0F 27	- Planed quantity = 9999 (Value b/w 0000 to 270F)
BCC	: 3D	
ETX	: BB	

Response: AA 0A 01 00 0B BB

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
STATUS	: 00
DATA []	: NULL
BCC	: 0B
ETX	: BB

5.11 CMD_SET_DEV_ID (0x1B) - Set device address.**Example:****Send data: AA 0A 02 1B 0B 18 BB**

STX	: AA	
STATION ID	: 0A	
DATA LENGTH	: 02	
CMD	: 1B	
DATA [0]	: 0B	- Device ID to be change = 11 (Value b/w 00 to FF)
BCC	: 18	
ETX	: BB	

Response: AA 0A 01 00 0B BB

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
STATUS	: 00
DATA []	: N/A
BCC	: 0B
ETX	: BB

5.12 CMD_SET_SHIFT_TIME (0x1C) – Set shift start and stop timings.**Example:****Send data: AA 0A 08 1C 01 07 00 00 0F 00 00 17 BB** - 24 hr mode

STX : AA
STATION ID : 0A
DATA LENGTH : 08
CMD : 1C
DATA [0] [6] : 01 07 00 00 0F 00 00
BCC : 17
ETX : BB

Note: DATA[0] = Shift number = 01
DATA[1][3] = Shift start time = 07:00:00
DATA[4][6] = Shift end time = 15:00:00

Send data: AA 0A 0A 1C 01 07 00 00 00 03 00 00 01 18 BB - 12 hr mode

STX : AA
STATION ID : 0A
DATA LENGTH : 0A
CMD : 1C
DATA [0] [6] : 01 07 00 00 00 03 00 00 01
BCC : 18
ETX : BB

Note: DATA[0] = Shift number = 01
DATA[1][4] = Shift start time = 07:00:00 00(am)
DATA[5][8] = Shift end time = 03:00:00 01(pm)

Response: AA 0A 01 00 0B BB

STX : AA
STATION ID : 0A
DATA LENGTH : 01
STATUS : 00
DATA [] : NULL
BCC : 0B
ETX : BB

5.13 CMD_SET_SHIFT_BREAK_TIME (0x1D) – Set shift breaks start and stop timings.**Example:****Send data: AA 0A 09 1D 01 01 07 00 00 0F 00 00 16 BB** - 24 hr mode

STX : AA
STATION ID : 0A
DATA LENGTH : 09
CMD : 1D
DATA [0] [6] : 01 01 07 00 00 0F 00 00
BCC : 16
ETX : BB

Note: DATA[0] = Shift number = 01
DATA[1] = Break Session number = 01
DATA[2][4] = Shift start time = 07:00:00
DATA[5][7] = Shift end time = 15:00:00

Send data: AA 0A 0B 1D 01 01 07 00 00 00 03 00 00 01 19 BB - 12 hr mode

STX : AA
STATION ID : 0A
DATA LENGTH : 0B
CMD : 1D
DATA [0] [6] : 01 01 07 00 00 00 03 00 00 01
BCC : 19
ETX : BB

Note: DATA[0] = Shift number = 01
DATA[1] = Break session number = 01
DATA[2][5] = Shift start time = 07:00:00 am
DATA[6][9] = Shift end time = 03:00:00 pm

Response: AA 0A 01 00 0B BB

STX : AA
STATION ID : 0A
DATA LENGTH : 01
STATUS : 00
DATA [] : NULL
BCC : 0B
ETX : BB

5.14 CMD_PAUSE_PROCESSING (0x1E) - Stop current operation.**Example:****Send data: AA 0A 01 1E 15 BB**

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
CMD	: 1E
DATA []	: N/A
BCC	: 15
ETX	: BB

Response: AA 0A 01 00 0B BB

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
STATUS	: 00
DATA []	: N/A
BCC	: 0B
ETX	: BB

5.15 CMD_START_PROCESSING (0x1F) - Start current operation.**Example:****Send data: AA 0A 01 1F 14 BB**

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
CMD	: 1F
DATA []	: N/A
BCC	: 14
ETX	: BB

Response: AA 0A 01 00 0B BB

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
STATUS	: 00
DATA []	: N/A
BCC	: 0B
ETX	: BB

5.16 CMD_RESET_PROCESSING (0x20) - Stop current operation.**Example:****Send data: AA 0A 01 20 2B BB**

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
CMD	: 20
DATA []	: N/A
BCC	: 2B
ETX	: BB

Response: AA 0A 01 00 0B BB

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 01
STATUS	: 00
DATA []	: N/A
BCC	: 0B
ETX	: BB

5.17 CMD_GET_SHIFT_TIME (0x0E) – Get shift start and stop timings.**Example:****Send data: AA 0A 02 0E 01 07 BB**

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 02
COMMAND	: 0E
DATA []	: 01
BCC	: 07
ETX	: BB

Response: AA 0A 08 00 01 07 00 00 0F 00 00 0B BB - 24 hr mode

STX	: AA
STATION ID	: 0A
DATA LENGTH	: 08
STATUS	: 00
DATA [0]	: 01
DATA [1] [6]	: 07 00 00 0F 00 00
BCC	: 0B
ETX	: BB

Note: DATA[0] = Shift number = 01
 DATA[1][3] = Shift start time = 07:00:00
 DATA[4][6] = Shift end time = 15:00:00

Response : AA 0A 0A 00 01 07 00 00 00 03 00 00 01 04 BB - 12 hr mode

STX : AA
 STATION ID : 0A
 DATA LENGTH : 0A
 STATUS : 00
 DATA [0] [6] : 01 07 00 00 00 03 00 00 01
 BCC : 04
 ETX : BB

Note: DATA[0] = Shift number = 01
 DATA[1][4] = Shift start time = 07:00:00 00(am)
 DATA[5][8] = Shift end time = 03:00:00 01(pm)

5.18 CMD_GET_SHIFT_BREAK_TIME (0x0F) – Get shift breaks start and stop timings.

Example:

Send data: AA 0A 03 0F 01 01 06 BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 03
 COMMAND : 0F
 DATA [0] : 01 - Shift number
 DATA [1] : 01 - Shift break session number
 BCC : 06
 ETX : BB

5.19 CMD_SET_ACTUAL_QTY (0x21) – Set actual quantity.

Example:

Send data: AA 0A 04 21 01 20 0F 01 BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 04
 CMD : 21
 DATA [0] : 01 -Shift number
 DATA [1][2] : 20 0F - Actual quantity = 8207 (Value b/w 0000 to 270F)
 BCC : 01
 ETX : BB

Response: AA 0A 01 0B BB

STX : AA
 STATION ID : 0A
 DATA LENGTH : 01
 STATUS : 00
 DATA [] : NULL
 BCC : 0B
 ETX : BB