1. Communication Structure

* Communication Methodology : **HDLC** and **BSC**
* Communication Frame : Command Frame / Data Frame
* Frame Structure : Header + Body + Terminator

1.1 Frame Header

Frame Header	Hex Value	Detail	Hex Value
Synchronizer (FSYN)	7EH,7EH,01H	(1)Start	7EH,7EH
		(2)Stop	01H
Designator (FDES)	ID NO.	Broadcast	00H
		Working Node	Working ID
Director (FDIR)		DTE	01H
		Command	03H-FFH

1.2 Frame Body

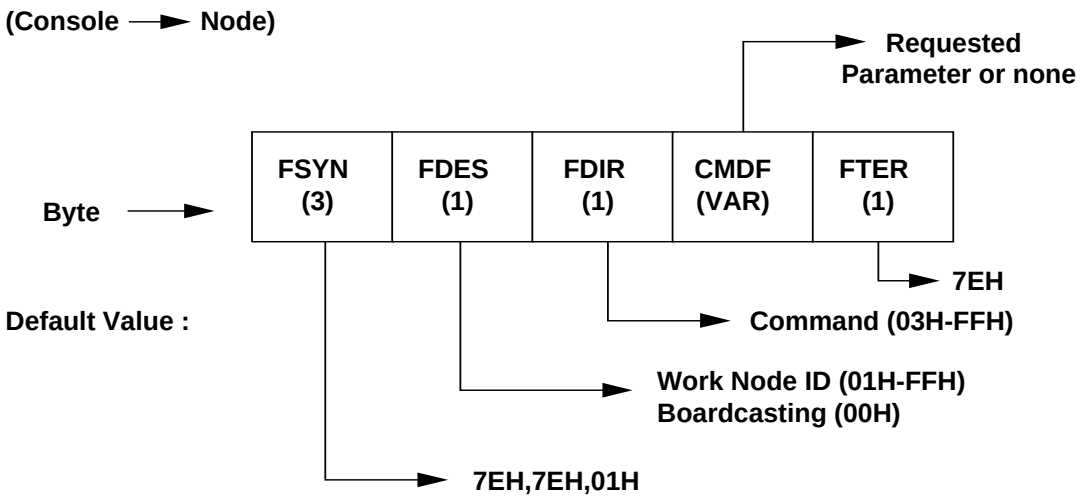
Frame Body	Detail	HEX Value
Command Frame (CMDF)	Parameter Packet	Variable
Data Frame (DATF)		
	(1) Packet No	30H-39H (0 - 9)
	(2) Packet length	0001H- 8FFFH
	(3) Data Packet	Variable
	(4) Field delimiter	3AH(:)
	(5) Record Delimiter	23H(#)
	(6) LRC	00H-FFH

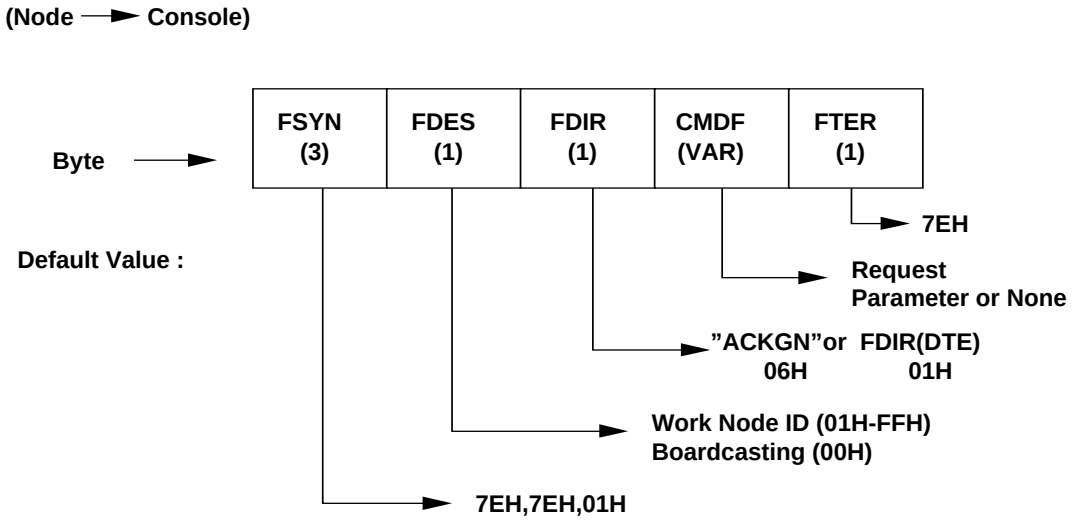
1.3 Frame Terminator

Frame Terminator	HEX Value
Terminator of communication frame (FTER)	7EH

2. Communication Format

2.1 Command Frame



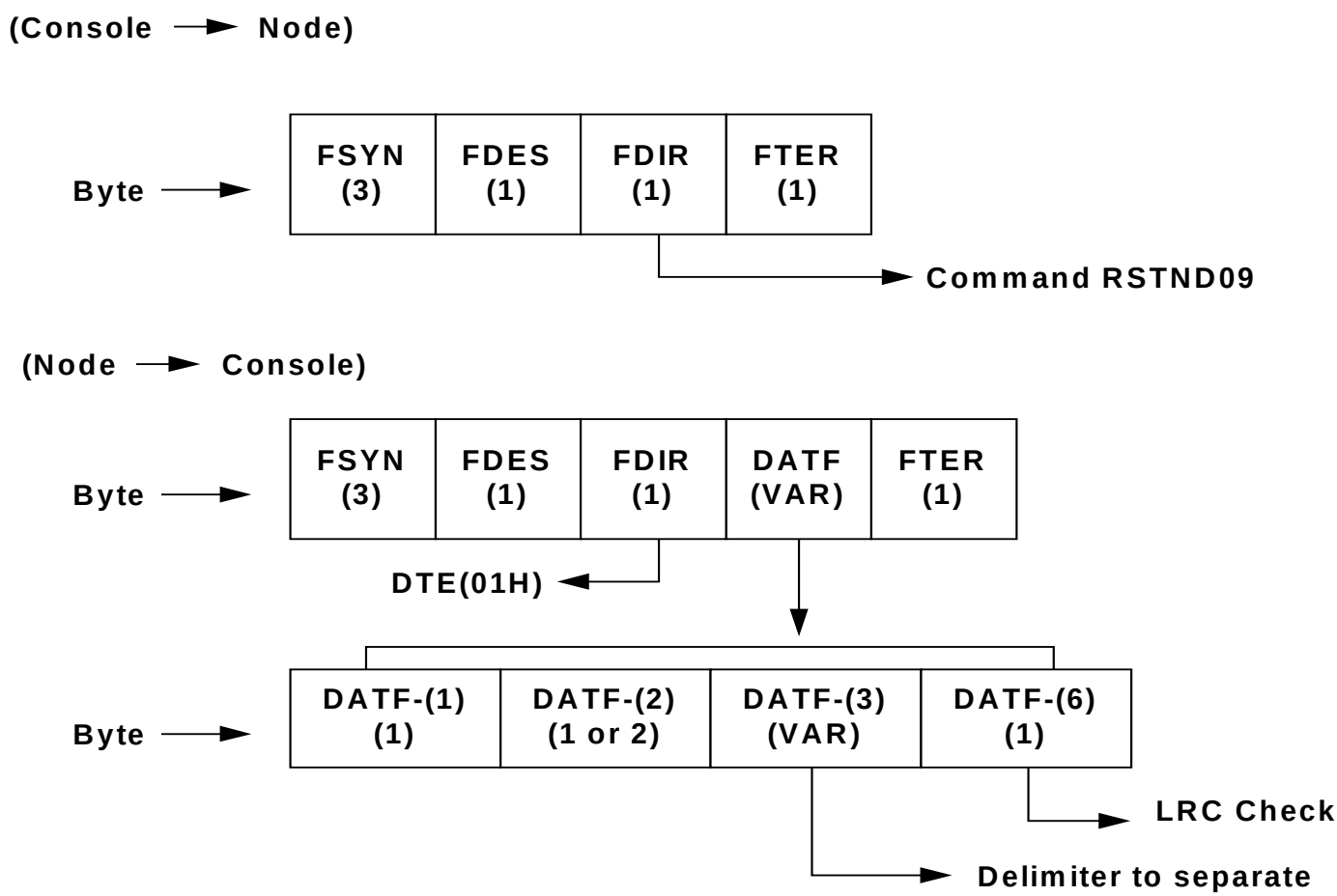


Example : RSTFD03 Reset to factory default

(Console → Node) 7EH,7EH,01H,02H,03H,7EH
Command working node ID No. 2 (02H) Reset to default

(Node → Console) 7EH,7EH,01H,02H,06H,7EH
Working node ID No. 2(02H) echo an acknowledge.
Receive command OK !!

2.2 Data Frame



* DATF-(1) : DATF-Packet No (Reference to **DATF**)

Example : Reference to 2.3

2.3 Note

1. For the details of command frame, please refer to "**Communication command**".

2. About the details of data frame, please refer to "**Example**" and notice the followings:

* **Packet No** : DATF -(1)

The packet no. will cycle from 0 (HEX 30) to 9 (HEX 39) automatically. Programmer could use it to confirm the received data packet is correct. When console send "**RSTND09**" command to request working node to transmit scanned datum, the working node will transmit one packet scanned datum and wait console to send "**ACKGN06**" for acknowledgment. After working node received the "Receive Data OK " from console, working node will clear the transmitted datum in memory buffer and add 1 to packet no. automatically.

Otherwise , the next "**RSTND09**" from console , working node only retransmit the same packet no. and datum as last time. Because all the receiving procedure must be confirmed , working node will clear the memory to avoid data loss.

* **Packet Length** : DATF -(2)

The packet length is a 2 bytes indicator in "8 data bits data frame" mode , but it is only 1 byte indicator in "7 data bits data frame mode".

* 2 bytes (0001H-7FFFH) : 1 bytes to 32k bytes data
* 1 byte (01H-7FH) : 1 byte to 127 bytes data

Packet Length = How many bytes from DATF-Start Trans. character to LRC.

Note : The programmer could use the packet length to determined the data length of received datum.

* **LRC** : Longitudinal Redundancy Check

LRC = (FDES) XOR (FDIR) XOR .....XOR (Last Byte Before LRC)

3. Factory Default

ITEM	DESCRIPTION	
Comm. Configuration	Baud Rate	- 9600 BPS
	Data frame	- 8 data bit, none parity ,1 stop bit
Delimiter	Record	- 3AH (:)
	Field	- 23H (#)
	Terminator	- 7EH
Discrimination	Status	- Both Bar Code and Magnetic Stripe
	Bar Code	- Automatic Discrimination
	Magnetic	- Automatic Discrimination

4. Command Summary

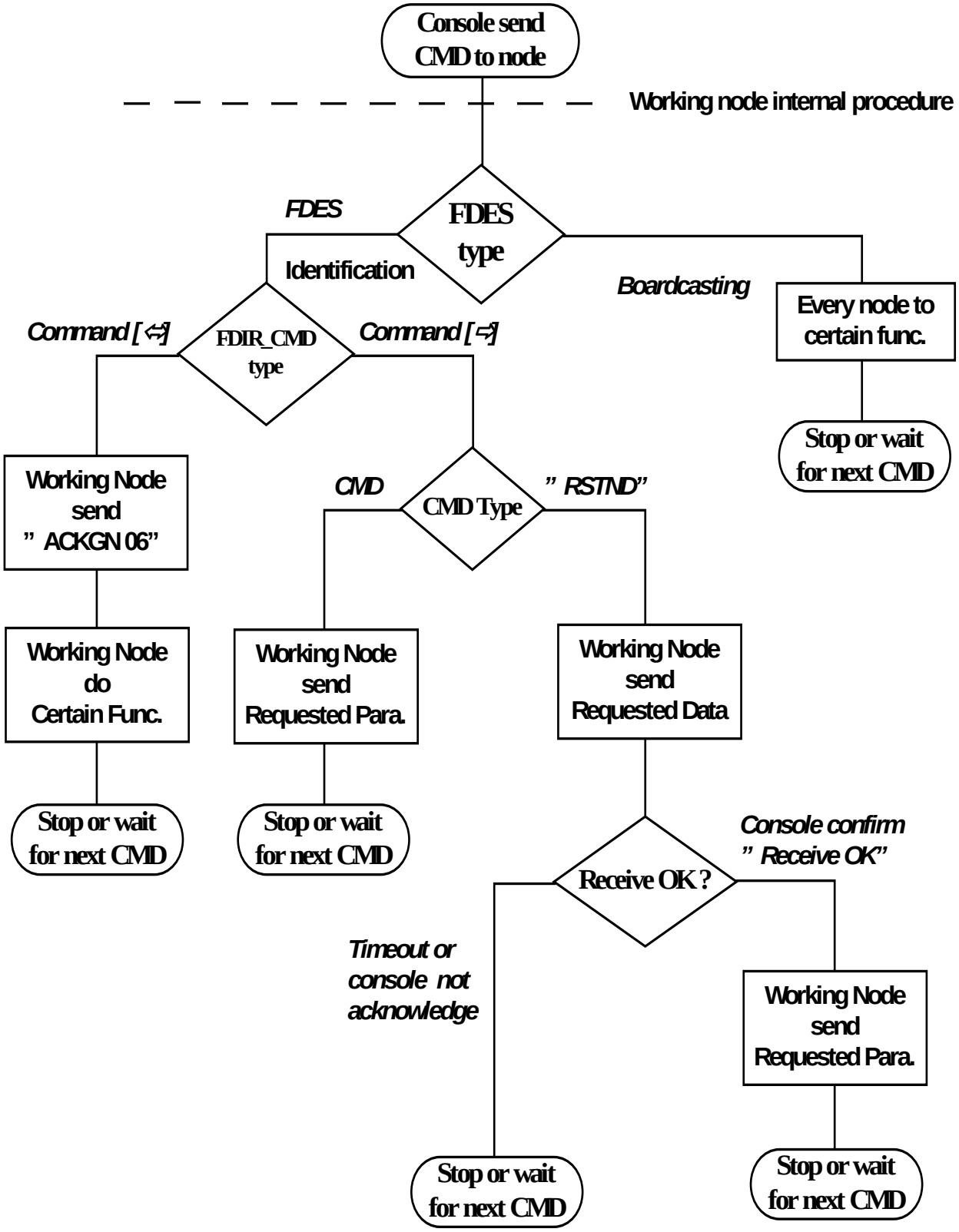
CMD	HEX	TYPE	DIRECT	Model	DESCRIPTION
RSTFD	03	B/I	###	§(CL-90,60)	Reset to Default
CLMSP	04	B/I	###	ALL	Clear memory buffer and set packet no. = 0
ENQND	05	I	###	ALL	Inquiry working node for link status
ACKGN	06	I	###	ALL	Affirmative acknowledgments
GTBEL	07	I	###	BC-400	Get off-line speaker status
STBEL	08	B/I	###	ALL	On-line/Off-line speaker control
RSTND	09	I	###	ALL	Request working node to transmit scanning datum
RSTNS	0A	I	###	CL-90,60	Request working node to transmit record of sensor
RSDNS	0B	I	###	CL-90,60	Request working mode to delete last sensor record
RSSCR	0C	B/I	###	ALL	Restore the original screen
GTMOD	0D	I	###	ALL	Get model number and date code
GTKBC	0E	I	###	CL-70	Get keypad status
STKBC	0F	B/I	###	CL-70	Set keypad status
GTDAT	22	I	###	ALL	Get RTC date
STDAT	23	B/I	###	ALL	Set RTC date
GTTIM	24	I	###	ALL	Get RTC time
STTIM	25	B/I	###	ALL	Set RTC time
GTTSF	26	I	###	BC-400	Get time stamp function control status
STTSF	27	B/I	###	BC-400	Time stamp function control status
STROC	31	B/I	###	ALL	On-line/Off-line relay output control
GTRTS	32	I	###	§(CL-90,60)	Get relay time table
STRTS	33	B/I	###	§(CL-90,60)	Set relay time table
GRAED	34	I	###	CL-70	Get relay auto switch & shift no. auto switch flag
SRAED	35	B/I	###	CL-70	Set relay auto switch & shift no. auto switch flag
GTLOC	3A	I	###	BC-400	Get software lock status
STLOC	3B	B/I	###	BC-400	Software lock control
GSCLN	4A	I	###	CL-70	Get extension card parameter
SSCLN	4B	B/I	###	CL-70	Set extension card parameter
GTCLN	50	I	###	CL-70	Get shift no.
STCLN	51	B/I	###	CL-70	Set shift no.
GTOPM	54	I	###	CL-70	Get operating mode
STOPM	55	B/I	###	CL-70	Set operating mode
GTSCN	56	I	###	BC-400	Get scanned data display control status
STSCN	57	B/I	###	BC-400	Scanned data display control
GTPIC	58	I	###	CL-70	Get printer parameter
STPIC	59	B/I	###	CL-70	Set printer parameter
LDMES	62	B/I	###	ALL	

CMD	HEX	TYPE	DIRECT	Model	DESCRIPTION
LDNID	63	B/I	###	ALL	Display Message
					Display working node ID
STACT	71	B/I	###	CL-70	Set auto switch time table
GTACT	72	I	###	CL-70	Get auto switch time table
STBAT	72	B/I	###	§(CL-70)	Set block time interval
GTOTC	73	I	###	ALL	Get pin code status
STOTC	74	B/I	###	ALL	Set pin code status
STOPS	76	B/I	###	ALL	Set User Card as single check mode
STTPS	78	B/I	###	ALL	Set User Card as double check mode
GTALM	79	I	###	ALL	Get sensor status (real time)
STOTD	7A	B/I	###	ALL	Set door opening duration
GTOTD	7B	I	###	ALL	Get door opening duration
STOTD	7C	B/I	###	ALL	Delete user badge number
DEBLP	7D	B/I	###	ALL	Delete block badge number and pin code
STALM	80	B/I	###	ALL	Set alarm active time
STTCE	82	B/I	###	ALL	Set error alarm status
STALF	83	B/I	###	ALL	Set alarm enable/disable flag
GTALF	84	I	###	ALL	Get alarm enable/disable flag
STUSR	90	B/I	###	CL-90,60	Download user name
STMMS	99	B/I	###	CL-90,60	Set Middle Level Master Card as single check
STMMD	9A	B/I	###	CL-90,60	Set Middle Level Master Card as double check

NOTE : **B/I** : Broadcast or Identify
###: Console to Working Node
§ : All are available except ( )

I : Identify only
###: Bidirection

5. Flowchart Of Internal Procedure In Working Node



6. Example

RSTND 09 I [⇔] Request working node to transmit data

<FORMAT> : [⇨] 7EH,7EH,01H,ID,09H,7EH

[⇦] 7EH,7EH,01H,ID,DTE,Arg01,Arg02,Arg03,3AH,Arg04 ,3AH,Arg05,3AH,Arg06,23H,.....,LRC,7EH

(SYMBOL) :

DTE 01H

3AH Default separate record

Arg01 30H-39H Packing No.

Arg02 0000H-FFFFH data length

Arg03 Badge ID (variable length)

3AH Field delimiter

Arg04 Date :

1st - 2nd bytes : 30H-39H, 30H-39H (year)

3rd - 4th bytes : 30H-31H, 30H-39H (month)

5th - 6th bytes : 30H-33H, 30H-39H (day)

7th byte : 31H-37H

31H : (Monday)

.

36H : (Saturday)

37H : (Sunday)

3AH Field delimiter

Arg05 Time :

1st - 2nd bytes : 30H-32H, 30H-39H (hour)

3rd - 4th bytes : 30H-35H, 30H-39H (minute.)

5th - 6th bytes : 30H-35H, 30H-39H (sec.)

3AH Field delimiter

Arg06 Status (BC-610, CL-60, CL-70 only)

- 1st byte :
  - 31H - class 1
  - 32H - class 2
  - 33H - class 3
  - 34H - class 4
- 2nd byte :
  - 30H - duty on
  - 31H - duty off
  - 32H - break out
  - 33H - break on

23H Record delimiter

..... - several records (*)

LRC - 00H-FFH

* Field structure in every Record : Arg03, 3AH, Arg04, 3AH, Arg05, 3AH, Arg06, 23H

Request the working node No.5 send data to console.

*** Working node No.5 without any data.**

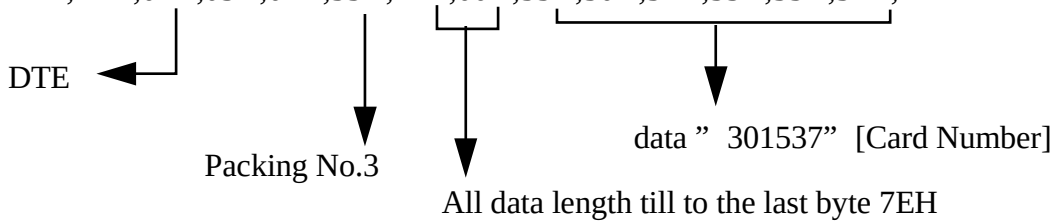
[⇨] : 7EH,7EH,01H,05H,09H,7EH
(Request working node NO.5 send data to console)

[⇦] : **7EH,7EH,01H,05H,09H,7EH**
(Return an acknowledge to console, without any data)

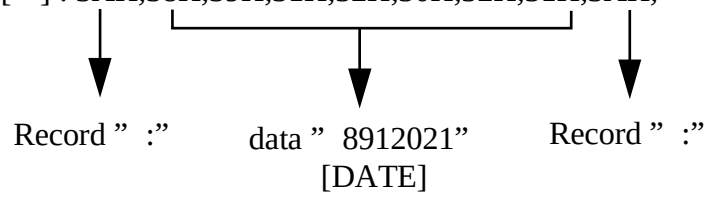
*** Working node No.5 with one record of data.**

[⇨] : 7EH,7EH,01H,05H,09H,7EH
(Request working node NO.5 send data to console)

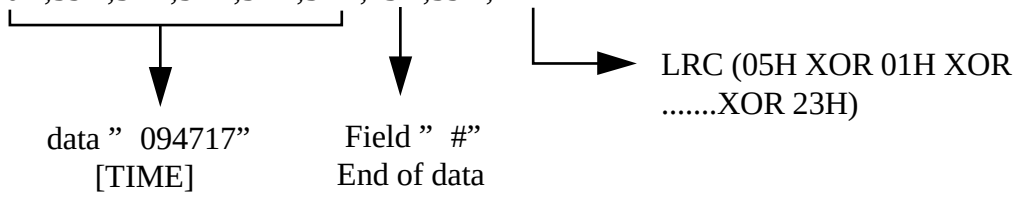
[⇦] : 7EH,7EH,01H,05H,01H,33H,17H,00H,33H,30H,31H,35H,33H,37H,



[⇦] : 3AH,38H,39H,31H,32H,30H,32H,31H,3AH,



[⇦] : 30H,39H,34H,37H,31H,37H,23H,39H,7EH



Note : In Time attendance system, Shift_number and Operating_mode are set in front of 23H (#: end of record). Data format of **BC-610** is as following :
Badge_Number, DATE, TIME, Shift_number, Operating_mode

Note : In **CL-70** Badge number is **not** in **first record**. Data format of CL-70 is as following :
DATE, TIME, Shift_number., Operating_mode., **Badge_Number**, Extension_Card1, Extension_Card2.

7. Communication Command [Node ID = 5]

CMD	HEX	Type	Direct	Description
RSTFD	03	B/I	[⇨]	Reset to Default
	<FORMAT>:		[⇨]	7EH,7EH,01H,05H,03H,7EH
			[⇦]	7EH,7EH,01H,05H,06H,03H,7EH
CLMSP	04	B/I	[⇨]	Clear memory buffer and set packet no. = 0
	<FORMAT>:		[⇨]	7EH,7EH,01H,05H,04H,7EH
			[⇦]	7EH,7EH,01H,05H,06H,04H,7EH
	NOTE: The "CLMSP" will **clear memory**, all data will be cleared.			
ENQND	05	I	[⇨]	Console inquiry working node for link status
	<FORMAT>:		[⇨]	7EH,7EH,01H,05H,05H,7EH
			[⇦]	7EH,7EH,01H,05H,06H,05H,7EH
	NOTE: The "ENQND" could perform high speed communication respondent for testing or prototype environments.			
ACKGN	06	I	[⇔]	Affirmative acknowledgments
	<FORMAT>:		[⇨]	7EH,7EH,01H,05H,06H,7EH
	NOTE: The "ACKGN" command only be activated by "**Identification**" and commands. It always presents the passive role in the communication system.			
GTBEL	07	I	[⇔]	Get off-line speaker status
	<FORMAT>:		[⇨]	7EH,7EH,01H,05H,07H,7EH
			[⇦]	7EH,7EH,01H,05H,01H,Arg01,Arg02,7EH
	Arg01	1 byte		(0 - 1) Speaker status (0 : disable, 1: enable)
	Arg02	1 byte		(30H-39H) Beep times for read OK (1-10 short beeps)

STBEL 08 B/I [⇨] On-line/Off-line speaker control

<FORMAT>: [⇨] 7EH,7EH,01H,05H,08H,01H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,08H,7EH

Arg01 1 byte (30H-39H) Beep times of speaker (1-10 times)

RSTND 09 I [⇔] Request working node to transmit scanning datum

NOTE : Review the "**Example**"

RSTNS 0A I [⇨] Request working node to transmit record of sensor (CL-90 ONLY)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,0AH,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,Arg02,Arg03,Arg04,Arg05,Arg06,7EH

Arg01	7 bytes	DATE & WEEK
Arg02	1 byte	3AH (:)
Arg03	6 bytes	TIME
Arg04	1 byte	3AH (:)
Arg05	1 byte	00H-FFH (Sensor Status)
Arg06	1 byte	00H-FFH (Checksum of Arg01 - Arg05)

RSTNS 0B I [⇨] Request working node to delete last sensor record (CL-90 ONLY)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,0BH,7EH

RSSCR 0C B/I [⇨] Restore the original screen

<FORMAT>: [⇨] 7EH,7EH,01H,05H,0C,7EH
[⇦] 7EH,7EH,01H,05H,06H,0CH,7EH

GTMOD 0D I [⇔] Get model no. and date code

<FORMAT>: [⇨] 7EH,7EH,01H,05H,0DH,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,Arg02,arg03,7EH

Arg01	2 bytes	(model no.) 3430H (BC-400) 3630H or 3631H (BC-610) 3830H (BC-800) 3130H (BC-1000, BC-1010, BC-1020) 3230H (BC-2000) 3730H (CL-70) 3930H (CL-90) 3954H (CL-60)
Arg02	1 byte	(version no.) 30H (standard version) 31H-39H (revision/OEM version)
Arg03	4 bytes	(data code) 30H-39H (1st byte) year 30H-39H (2nd byte) 30H-31H (3rd byte) month 30H-39H (4th byte)

GTKBC 0E I [⇔] Get keypad status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,0EH,7EH
[⇦] 7EH,7EH,05H,01H,Arg01,7EH

Arg01	1 byte	Keypad status (00H-03H) 0: All keys are disabled 1: Disable Function key, Enable Numeric key 2: Enable Function key, Disable Numeric key 3: All keys are enabled

STKBC 0F I [⇨] Set keypad status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,0FH,,Arg017EH
[⇦] 7EH,7EH,05H,06H,0FH,7EH

Arg01	1 byte	Keypad status (00H-03H) 0: All keys are disabled 1: Disable Function key, Enable Numeric key 2: Enable Function key, Disable Numeric key 3: All keys are enabled

GTNID 15 I [⇔] Get working node ID

<FORMAT>: [⇨] 7EH,7EH,01H,05H,15H,7EH
[⇦] 7EH,7EH,05H,01H,Arg01,7EH

Arg01 1 byte Working node ID (01H-FFH)

GTRTC 20 I [⇔] Get RTC control status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,20H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 byte (bit orientation) --> 0 : disable, 1 : enable
Bit 0 year indicator
Bit 1 month indicator
Bit 2 day indicator
Bit 3 week indicator
Bit 4 hour indicator
Bit 5 minute indicator
Bit 6 second indicator
Bit 7 12/24 (AM/PM) hours mode control

NOTE: Bit 7 : 0 - 12 hours, 1 - 24 hours

STRTC 21 B/I [⇨] RTC control

<FORMAT>: [⇨] 7EH,7EH,01H,05H,21H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,21H,7EH

Arg01 1 byte (bit orientation) --> 0 : disable, 1 : enable
Bit 0 year indicator
Bit 1 month indicator
Bit 2 day indicator
Bit 3 week indicator
Bit 4 hour indicator
Bit 5 minute indicator
Bit 6 second indicator
Bit 7 12/24 (AM/PM) hours mode control

NOTE: If user set the week indicator "disable", working node will truncate this byte data from memory buffer. So this byte of the collected datum will also be truncated automatically. Your data collection program must be designed to meet the rule.

GTDAT 22 I [⇔] Get RTC date

<FORMAT>: [⇨] 7EH,7EH,01H,05H,22H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,Arg02,Arg03,Arg04,7EH

Arg01	2 bytes	(year) ->	30H-39H (first byte)
			30H-39H (second byte)
Arg02	2 bytes	(month) ->	30H-31H (first byte)
			30H-32H (second byte)
Arg03	2 bytes	(day) ->	30H-33H (first byte)
			30H-39H (second byte)
Arg04	1 byte	(week) ->	31H (Monday)
			32H (Tuesday)
			33H (Wednesday)
			34H (Thursday)
			35H (Friday)
			36H (Saturday)
			37H (Sunday)

NOTE : The "GTDAT" command doesn't care RTC control status, it always send complete datum include year, month, day and week.

STDAT 23 B/I [⇨] Set RTC date

<FORMAT>: [⇨] 7EH,7EH,01H,05H,23H,Arg01,Arg02,Arg03,Arg04,7EH
[⇦] 7EH,7EH,01H,05H,06H,23H,7EH

Arg01	2 bytes	(year) ->	30H-39H (first byte)
			30H-39H (second byte)
Arg02	2 bytes	(month) ->	30H-31H (first byte)
			30H-32H (second byte)
Arg03	2 bytes	(day) ->	30H-33H (first byte)
			30H-39H (second byte)
Arg04	1 byte	(week) ->	31H (Monday)

32H (Tuesday)
33H (Wednesday)
34H (Thursday)
35H (Friday)
36H (Saturday)
37H (Sunday)

NOTE : The "STDAT" command doesn't care RTC control status, user always need to send complete datum include year, month, day and week.

GTTIM 24 I [⇔] Get RTC time

<FORMAT>: [⇨] 7EH,7EH,01H,05H,24H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,Arg02,Arg03,7EH

Arg01 2 bytes (hour) -> 30H-32H (first byte)
30H-39H (second byte)
Arg02 2 bytes (minute) -> 30H-35H (first byte)
30H-39H (second byte)
Arg03 2 bytes (second) -> 30H-35H (first byte)
30H-39H (second byte)

NOTE : The "GTTIM" command doesn't care RTC control status, it always send complete datum include hour, minute and second.

STTIM 25 B/I [⇨] Set RTC time

<FORMAT>: [⇨] 7EH,7EH,01H,05H,25H,Arg01,Arg02,Arg03,7EH
[⇦] 7EH,7EH,01H,05H,06H,25H,7EH

Arg01 2 bytes (hour) -> 30H-32H (first byte)
30H-39H (second byte)
Arg02 2 bytes (minute) -> 30H-35H (first byte)
30H-39H (second byte)
Arg03 2 bytes (second) -> 30H-35H (first byte)
30H-39H (second byte)

NOTE : The "STTIM" command doesn't care RTC control status, user always need to send complete datum include hour, minute and second.

GTTSF 26 I [⇔] Get time stamp function control status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,26H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 bytes (0-1) Time stamp function (0 : disable, 1 : enable)

STTSF 27 B/I [⇨] Time stamp function control

<FORMAT>: [⇨] 7EH,7EH,01H,05H,27H,Arg017EH
[⇦] 7EH,7EH,01H,05H,06H,27H,7EH

Arg01 1 bytes (0-1) Time stamp function (0 : disable, 1 : enable)

GTROC 30 I [⇔] Get off-line (local) relay control status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,30H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 byte (bit orientation) 0 : disable 1 : enable
Bit 0 relay 1
Bit 1 relay 2
Bit 2 relay 3
Bit 3 relay 4
Bit 4 relay 5
Bit 5 relay 6
Bit 6 relay 7
Bit 7 relay 8

STROC 31 B/I [⇨] On-line/Off-line relay outputs control

<FORMAT>: [⇨] (1) 7EH,7EH,01H,05H,31H,Arg01,Arg02,7EH (Arg01 = 00)
(2) 7EH,7EH,01H,05H,31H,Arg01,Arg03,Arg04,7EH (Arg01 = 01)
[⇦] 7EH,7EH,01H,05H,06H,31H,7EH

Arg01 1 byte (00H : set relay control flag, 01H : direct relay control)

Arg02	1 byte	(bit orientation) 0 : disable 1 : enable
		Bit 0 relay 1
		Bit 1 relay 2
		Bit 2 relay 3
		Bit 3 relay 4
		Bit 4 relay 5
		Bit 5 relay 6
		Bit 6 relay 7
		Bit 7 relay 8
Arg03	1 byte	30H-37H (relay ID)
		30H relay 1
		31H relay 2
Arg04	2 bytes	relay on duration
		00H-78H 0-120 minute scaler (first byte)
		7BH relay always on until receive relay off control (first byte)
		7DH relay off (first byte)
		00H-59H 0-59 second scaler (second byte)

NOTE: If user set 1st byte of Arg04 to "7BH/7DH", the 2nd byte must be set to "00H"

GTRTS 32 I [⇔] Get off-line (local) relay timetable

<FORMAT>: [⇨] 7EH,7EH,01H,05H,32H,Arg01,Arg02,7EH

[⇦] 7EH,7EH,01H,05H,01H,Arg03,Arg04,7EH

Arg01	1 byte	30H-31H (relay ID)
		30H relay 1
		31H relay 2
Arg02	1 byte	30H-33H (timetable no.)
		30H No. 1
		31H No. 2
		32H No. 3
		33H No. 4
Arg03	4 bytes	relay on time
		30H-32H hour (1st byte)
		30H-39H hour (2nd byte)
		30H-35H minute (3rd byte)
		30H-39H minute (4th byte)

Arg04 2 bytes relay on duration
00H-78H 1-120 minute scaler (first byte)
00H-59H 0-59 second scaler (second byte)

STRTS 33 B/I [⇨] Off-line (local) relay timetable setup

<FORMAT>: [⇨] 7EH,7EH,01H,05H,33H,Arg01,Arg02,Arg03,Arg04,7EH
[⇦] 7EH,7EH,01H,05H,06H,33H,7EH

Arg01 1 byte 30H-31H (relay ID)
30H relay 1
31H relay 2
Arg02 1 byte 30H-33H (timetable no.)
30H No. 1
31H No. 2
32H No. 3
33H No. 4
Arg03 4 bytes relay on time
30H-32H hour (1st byte)
30H-39H hour (2nd byte)
30H-35H minute (3rd byte)
30H-39H minute (4th byte)
Arg04 2 byte relay on duration
00H-78H 1-120 minute scaler (first byte)
00H-59H 0-59 second scaler (second byte)

NOTE : If user want to set timetable of each relay "disable" ,user only need to assign "0000H" to Arg03.

GRAED 34 I [⇔] Get relay auto switch & shift no. auto switch flag

<FORMAT>: [⇨] 7EH,7EH,01H,05H,34H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 byte Auto switch control flag (00H-03H)

SRAED 35 B/I [⇨] Set relay auto switch & shift no. auto switch flag

<FORMAT>: [⇨] 7EH,7EH,01H,05H,35H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,35H,7EH

Arg01 1 byte Auto switch control flag (00H-03H)
0: Disable relay auto switch, disable shift no. auto switch
1: Enable relay auto switch, disable shift no. auto switch
2: Disable relay auto switch, enable shift no. auto switch
3: Enable relay auto switch, enable shift no. auto switch

GTLOC 3A I [⇔] Get software lock status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,3AH,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 bytes (0-1) Software lock status (0 : disable, 1 : enable)

STLOC 3B B/I [⇨] Software lock control

<FORMAT>: [⇨] 7EH,7EH,01H,05H,3BH,Arg017EH
[⇦] 7EH,7EH,01H,05H,06H,3BH,7EH

Arg01 1 bytes (0-1) Software lock function (0 : disable, 1 : enable)

NOTE : (1) "STLOC" command will set the working node to operating or non-operating mode. If working mode is non-operating mode, it will halt until receiving "Disable" control.

(2) If user enable the software lock, the speaker of working node will beep 3 times and LCD will display the " * SYSTEM HALT *" to arouse user's attention.

GSCLN 4A I [⇔] Get extension card parameter (CL-70 only)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,4AH,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,Arg02,7EH

Arg01	1 byte	(Extension card 1 status)
		bit 0 - 2 Extension card 1 length (0-6)
		bit 3 Reserved
		bit 4 Resident card 1 (0: disable, 1: enable)
		bit 5 Temporary card 1 (0: disable, 1: enable)
		bit 6 - 7 Reserved
Arg02	1 byte	(Extension card 2 status)
		bit 0 - 2 Extension card 2 length (0-6)
		bit 3 Reserved
		bit 4 Resident card 2 (0: disable, 1: enable)
		bit 5 Temporary card 2 (0: disable, 1: enable)
		bit 6 - 7 Reserved

SSCLN 4B I [⇨] Set extension card parameter (CL-70 only)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,4BH,Arg01,Arg02,7EH
[⇦] 7EH,7EH,01H,05H,06H,4BH,7EH

Arg01	1 byte	(Extension card 1 status)
		bit 0 - 2 Extension card 1 length (0-6)
		bit 3 Reserved
		bit 4 Resident card 1 (0: disable, 1: enable)
		bit 5 Temporary card 1 (0: disable, 1: enable)
		bit 6 - 7 Reserved
Arg02	1 byte	(Extension card 2 status)
		bit 0 - 2 Extension card 2 length (0-6)
		bit 3 Reserved
		bit 4 Resident card 2 (0: disable, 1: enable)
		bit 5 Temporary card 2 (0: disable, 1: enable)
		bit 6 - 7 Reserved

GTCLN 50 I [⇔] Get current shift No.

<FORMAT>: [⇨] 7EH,7EH,01H,05H,50H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 byte (Shift No.)
31H shift 1
32H shift 2
33H shift 3
34H shift 4

STCLN 51 B/I [⇨] Set shift No.

<FORMAT>: [⇨] 7EH,7EH,01H,05H,51H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,51H,7EH

Arg01 1 byte (Shift No.)
31H shift 1
32H shift 2
33H shift 3
34H shift 4

GTOPM 54 I [⇔] Get current operating mode

<FORMAT>: [⇨] 7EH,7EH,01H,05H,54H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 byte (Operating mode)
00H duty on
01H duty off
02H break out
03H break in
04H overtime start
05H overtime end

STOPM 55 B/I [⇨] Set operating mode

<FORMAT>: [⇨] 7EH,7EH,01H,05H,55H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,55H,7EH

Arg01	1 byte	(Operating mode)
		00H duty on
		01H duty off
		02H break out
		03H break in
		04H overtime start
		05H overtime end

GTSCN 56 I [⇔] Get scanned data display control status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,56H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 bytes (0-1) Status (0 : not display, 1 : display scanned data)

STSCN 57 B/I [⇨] Scanned data display control

<FORMAT>: [⇨] 7EH,7EH,01H,05H,57H,Arg017EH
[⇦] 7EH,7EH,01H,05H,06H,57H,7EH

Arg01 1 bytes (0-1) Status (0 : not display, 1 : display scanned data)

GTPIC 58 I [⇔] Get printer control parameter

<FORMAT>: [⇨] 7EH,7EH,01H,05H,58H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01	1 byte	(Printer control parameter) 0: disable, 1: enable
		bit 0 Data from keypad send to printer
		bit 1 Data from Barcode reader send to printer
		bit 2 Data from Magnetic stripe reader send to printer
		bit 3 Print control flag

Note : If bit 3 is set to 0, printer is disabled and ignore setting of bit 0-2 .
When bit 3 is set to 1, bit 0-2 control the Badge number will be printed out or not.
If bit 0-2 are set to 1, the following fields will be printed out :
DATE, TIME, Badge number
If bit 0-2 are set to 0, only DATE and TIME will be printed .

STPIC 59 I [⇨] Set printer control parameter

<FORMAT>: [⇨] 7EH,7EH,01H,05H,59H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,59H,7EH

Arg01 1 byte (Printer control parameter) 0: disable, 1: enable
bit 0 Keypad input Badge number send to printer
bit 1 Barcode input Badge number send to printer
bit 2 Magnetic input Badge number send to printer
bit 3 Print control flag

Note : If bit 3 is set to 0, printer is disabled and ignore setting of bit 0-2.
When bit 3 is set to 1, bit 0-2 control the Badge number will be printed out or not.
If bit 0-2 are set to 1, the following fields will be printed out :
DATE, TIME, Badge number
If bit 0-2 set to 0, only DATE and TIME will be printed.

LDCOM 61 B/I [⇨] Display communication configuration

<FORMAT>: [⇨] 7EH,7EH,01H,05H,61H,7EH
[⇦] 7EH,7EH,01H,05H,06H,61H,7EH (screen display)

LINE 1 - BAUD RATE 1200,2400,4800,9600,19200
LINE 2 - D/P/S
(1) 7,8 (DATA BITS)
(2) N,O,E,S,M (PARITY)
(3) 1,2 (STOP BIT)

LDMES 62 B/I [⇨] Display Message

<FORMAT>: [⇨] 7EH,7EH,01H,05H,62H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,62H,7EH (screen display)

Arg01 String Variable message string

NOTE : The maximum length of message is 31 characters.
Example : If you want to show 1234 in the display, you need to send the ASCII
7EH,7EH,01H,05H,62H,31H,32H,33H,34H,7EH (4 ASCII "1234")

LDNID 63 B/I [⇨] Display working node ID

<FORMAT>: [⇦] 7EH,7EH,01H,05H,63H,7EH
[⇨] 7EH,7EH,01H,05H,06H,63H,7EH (screen display)

LINE 1 - NODE ID 001-255 (ID number)

LDROT 64 B/I [⇨] Display relay output timetable

<FORMAT>: [⇨] 7EH,7EH,01H,05H,64H,Arg01,Arg02,7EH
[⇦] 7EH,7EH,01H,05H,06H,64H,7EH

Arg01 1 byte (relay ID) 30H-37H
30H relay 1
31H relay 2

Arg02 1 byte (timetable no.) 30H-39H
30H No. 1
31H No. 2
32H No. 3
33H No. 4

(screen display) LINE 1 - relay ID : * timetable : *
(1) 1 - 2 (relay ID)
(2) 1 - 4 (timetable)

LINE 2 - ***/***/**
(1) ON, OFF
(2) 000-120 minutes
(3) 00-59 seconds

STACT 71 B/I [⇨] Set auto switch time table

<FORMAT>: [⇨] 7EH,7EH,01H,05H,71H,Arg01,Arg02,Arg03,Arg04,Arg05,7EH
[⇦] 7EH,7EH,01H,05H,06H,71H,7EH

Arg01 1 byte 01-20 (auto switch table no.)
Arg02 1 byte 00-23 (hour)
Arg03 1 byte 00-59 (minute)
Arg04 1 byte 01-06 (operating mode)
Arg05 1 byte 01-05 (shift no.)

GTACT 72 B/I [⇔] Set auto switch time table

<FORMAT>: [⇨] 7EH,7EH,01H,05H,72H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg02,Arg03,Arg04,Arg05,7EH

Arg01	1 byte	01-20 (auto switch table no.)
Arg02	1 byte	00-23 (hour)
Arg03	1 byte	00-59 (minute)
Arg04	1 byte	01-06 (operating mode)
Arg05	1 byte	01-05 (shift no.)

STTPS 72 B/I [⇨] Set block time interval

<FORMAT>: [⇨] 7EH,7EH,01H,05H,72H,Arg01,Arg02,Arg03,Arg04,Arg05,7EH
[⇦] 7EH,7EH,01H,05H,06H,72H,7EH

Arg01	1 byte	01H-08H (block 1 - block 8)
Arg02	1 byte	00H-23H (start hour)
Arg03	1 byte	00H-59H (start minute)
Arg04	1 byte	00H-23H (end hour)
Arg05	1 byte	00H-59H (end minute)

GTOTC 73 I [⇨] Get pin code status (Disable or Enable)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,73H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01	1 byte	0 or 1 (0 : Disable , 1 : Enable)

NOTE : Disable -> Badge Number only. Enable -> Badge Number with pin code

---

STOTC 74 B/I [⇨] Set pin code status (Disable or Enable)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,74H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,74H,7EH

Arg01	1 byte	0 or 1 (0 : Disable , 1 : Enable)

NOTE : Disable -> Badge Number only. Enable -> Badge Number with pin code

---

STOPS 76 B/I [⇨] Set Single check Badge number

<FORMAT>: [⇨] 7EH,7EH,01H,05H,76H,Arg01,Arg02,7EH
[⇦] 7EH,7EH,01H,05H,06H,76H,7EH

Arg01	1 byte	(block 1 - block 8) 01H - 08H
Arg02	4-8 bytes	Badge number

STTPS 78 B/I [⇨] Set Double check Badge number and pin code

<FORMAT>: [⇨] 7EH,7EH,01H,05H,78H,Arg01,Arg02,Arg03,Arg04,7EH
[⇦] 7EH,7EH,01H,05H,06H,78H,7EH

Arg01	1 byte	(block 1 - block 8) 01H - 08H
Arg02	4-8 bytes	Badge number
Arg03	1 byte	(:) separate badge number and pin code 23H
Arg04	4-8 bytes	Pin code

GTALM 79 I [⇨] Get Sensor status

<FORMAT>: [⇨] 7EH,7EH,01H,05H,79H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 byte (bit orientation) 0 : normal 1 : active

Bit 0	sensor 5 (Door sensor)
Bit 1	sensor 0 (Push Button)
Bit 2	reserved
Bit 3	sensor 6 (Case sensor)
Bit 4	sensor 1 (Line-1 ALARM)
Bit 5	sensor 2 (Line-2 ALARM)
Bit 6	sensor 3 (Line-3 ALARM)
Bit 7	sensor 4 (Line-4 ALARM)

Example : Arg01 = 38H indicates three alarms (Case sensor, Line-1 Alarm, Line-2 Alarm) be triggered.

STOTD 7A B/I [⇨] Set door opening duration

<FORMAT>: [⇨] 7EH,7EH,01H,05H,7AH,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,7AH,7EH

Arg01 1 byte (door opening duration) 1 - 20

GTOTD 7B B/I [⇔] Set door opening duration

<FORMAT>: [⇨] 7EH,7EH,01H,05H,7BH,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01 1 byte (door opening duration) 1- 20

DESIP 7C B/I [⇨] Delete single Badge number

<FORMAT>: [⇨] 7EH,7EH,01H,05H,7CH,Arg01,Arg02,7EH
[⇦] 7EH,7EH,01H,05H,06H,7CH,7EH

Arg01 1 byte (block 1 - block 8) 01H - 08H
Arg02 4-8 bytes Badge number

DEBLP 7D B/I [⇨] Delete Block ID number and Pin code

<FORMAT>: [⇨] 7EH,7EH,01H,05H,7DH,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,7DH,7EH

Arg01 1 byte (block 1 - block 8) 01H - 08H

STALM 80 B/I [⇨] Set Alarm Activated Time

<FORMAT>: [⇨] 7EH,7EH,01H,05H,80H,Arg01,Arg02,Arg03,Arg04,7EH
[⇦] 7EH,7EH,01H,05H,06H,80H,7EH

Arg01	1 byte	00H - 23H (start hour)
Arg02	1 byte	00H - 59H (start minute)
Arg03	1 byte	00H - 23H (end hour)
Arg04	1 byte	00H - 59H (end minute)

GTTCE 81 I [⇔] Get Error alarm status (Beep three times)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,81H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01	1 byte	0 (Alarm is disabled), 1 (Alarm is enabled)

STTCE 82 B/I [⇨] Set Error alarm status (Beep three times)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,82H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,82H,7EH

Arg01	1 byte	0 (Disable error alarm), 1 (Enable error alarm)

STALF 83 B/I [⇨] Set Alarm Disable/Enable Flag

<FORMAT>: [⇨] 7EH,7EH,01H,05H,83H,Arg01,7EH
[⇦] 7EH,7EH,01H,05H,06H,83H,7EH

Arg01	1 byte	0 (Disable Alarm), 1 (Enable Alarm)

GTALF 84 I [⇔] Get Alarm Disable/Enable Flag

<FORMAT>: [⇨] 7EH,7EH,01H,05H,84H,7EH
[⇦] 7EH,7EH,01H,05H,01H,Arg01,7EH

Arg01	1 byte	0 (Disable), 1 (Enable)

STUSR 90 B/I [⇨] Download user name (CL-90 ONLY)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,90H,Arg01,Arg02,Arg03,Arg04,Arg05,Arg06,7EH
[⇦] 7EH,7EH,01H,06H,90H,7EH

Arg01	1-8 byte	Badge number
Arg02	1 byte	3AH(:)
Arg03	1-4 byte	Pin code
Arg04	1 byte	23H(#)
Arg05	1 byte	3AH(:)
Arg06	1-15 byte	User name

NOTE : If user set to single check, Arg03 must be set to null (Ignore Arg03).

STMMS 99 B/I [⇔] Set Middle Level User Card as single check (CL-90 ONLY)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,99H,Arg01,Arg02,7EH
[⇦] 7EH,7EH,01H,06H,99H,7EH

Arg01	1 byte	00H-07H (Block)
Arg02	1-13 byte	Badge Number

STMMD 9A B/I [⇔] Set Middle Level User Card as double check (CL-90 ONLY)

<FORMAT>: [⇨] 7EH,7EH,01H,05H,99H,Arg01,Arg02,Arg03,Arg04,7EH
[⇦] 7EH,7EH,01H,06H,9AH,7EH

Arg01	1 byte	00H-07H (Block)
Arg02	1-8 byte	Badge number
Arg03	1 byte	3AH (:)
Arg04	1-4 byte	Pin code

---

8. Library of Clipper (RECO.LIB Function Listing)

FUNCTION	Parameter Description
RSTFD03	Reset to Default [Input] 1. Comm. port (Value, 1-4) 2. Node ID (Value, 0-255) [Output] None * RSTFD03 will clear all data !!
CLMSP04	Clear memory buffer and set packet no. = 0 [Input] 1. Comm. port (Value, 1-4) 2. Node ID (Value, 0-255) [Output] None * CLMSP04 will clear all data !!
ENQND05	Inquiry working node for link status [Input] 1. Comm. port (Value, 1-4) 2. Node ID (Value, 1-255) [Output] Communication respondent (String, Variable length) * If length of return string is 7, it indicates ON-LINE. Otherwise it indicates communication error or OFF-LINE.

ACKGN06 Affirmative acknowledgments

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]
None

GTBEL07 Get On-line/Off-line speaker status

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]
Speaker status (Value, 0 - 1)
0 : Speaker is disabled
1 : Speaker is enabled

STBEL08 On-line/Off-line speaker control

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Beep times (Value ,0-9)

[Output]
None

RSTND09 Request working node to transmit scanning datum

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]

Scanning datum (String, Variable length)

* If return string equal to NULL, it indicates no data.
* If return string not equal to NULL :
  First Byte : Node ID (ASCII, "1"-"9")
  Byte 2 - Last byte : Datum

GTSTA0A Request working node to transmit record of sensor

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]
Sensor Status (String, Length = 15 or Length = 16)

* If return string equal to NULL, it indicates OFF-LINE or communication error.
* If length of return string equal to 15, it indicates Status = 0
* If length of return string equal to 16, last byte is sensor status.

DESTA0B Request working mode to delete last sensor record

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]
None

RSSCR0C Restore the original screen

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]
None

GTMOD0D Get model number and date code

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]

Model Number (String, Length = 8)

* If return string equal to NULL, it indicates OFF-LINE or communication error.

GTKBC0E Get keypad status

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]

Keypad status (Value, 0-3)

0: All keys are disabled
1: Function keys are disabled, Numeric keys are enabled
2: Function keys are enabled, Numeric keys are disabled
3: All keys are enabled

* If return string equal to NULL, it indicates OFF-LINE or communication error.

STKBC0F Get keypad status

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)
3. Keypad status (Value, 0-3)
   0: All keys are disabled
   1: Function keys are disabled, Numeric keys are enabled
   2: Function keys are enabled, Numeric keys are disabled
   3: All keys are enabled

[Output]

None

GTDAT22 Get RTC date

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]

Date & Week (String, Length = 7)

Year (byte 1,2)

Month (byte 3,4)

Day (byte 5,6)

Week (byte 7)

* If return string equal to NULL, it indicates OFF-LINE or communication error.

STDAT23 Set RTC date

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Year (Value, 0-99)
4. Month (Value, 1-12)
5. Day (Value, 1-31)
6. Week (Value, 1-7)

[Output]

None

GTTIM24 Get RTC time

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]

Time (String, Length = 6)

Hour (byte 1,2)

Minute (byte 3,4)

Second (byte 5,6)

* If return string equal to NULL, it indicates OFF-LINE or communication error.

STTIM25 Set RTC time

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Hour (Value, 0-23)
4. Minute (Value, 0-59)
5. Second (Value, 0-59)

[Output]
None

GTTSF26 Get time stamp function control status

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]
Time stamp status (Value, 0-1)
0 : Time stamp is disabled
1 : Time stamp is enabled

STTSF27 Time stamp function control

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Time stamp control (Value, 0-1)
0 : Time stamp is disabled
1 : Time stamp is enabled

[Output]
None

STROC31 On-line/Off-line relay output control

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Control Flag (Value, 0-1)
4. Relay (Value, 0-7) [If Control Flag = 0]
Relay (Character, "1") [If Control Flag = 1]

5. Relay (Value, 0-7) [If Control Flag = 1]
6. Relay on duration 1 (Value, 0-255) [If Control Flag = 1]
7. Relay on duration 2 (Value, 0-255) [If Control Flag = 1]

[Output]
None

GTRTS32 Get Relay auto switch time table

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Table no. (Value)
   49-57 table 1-10
   65-74 table 11-20

[Output]
Relay time table status (String, length = 6)
Start hour (byte 1,2)
Start minute (byte 3,4)
Work minute (byte 5)
Work second (byte 6)

STRTS33 Set Relay auto switch time table

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Table no. (Value)
   49-57 table 1-10
   65-74 table 11-20
4. Start hour (Value, 0-23)
5. Start minute (Value, 0-59)
6. Work minute (Value, 0-59)
7. Work second (Value, 0-59)

[Output]
None

GRAED34 Get Relay auto switch and shift no. auto switch control flag

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Control status (Value, 0-3)

0: Disable relay auto switch, disable shift no. auto switch
1: Enable relay auto switch, disable shift no. auto switch
2: Disable relay auto switch, enable shift no. auto switch
3: Enable relay auto switch, enable shift no. auto switch

SRAED35 Set Relay auto switch and shift no. auto switch control flag

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Control status (Value, 0-3)

0: Disable relay auto switch, disable shift no. auto switch
1: Enable relay auto switch, disable shift no. auto switch
2: Disable relay auto switch, enable shift no. auto switch
3: Enable relay auto switch, enable shift no. auto switch

[Output]

None

GTLOC3A Get software lock status

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Software lock status (Value, 0-1)

0 : Software lock is disabled
1 : Software lock is enabled

STLOC3B Software lock control

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Software lock control (Value, 0-1)
   0 : Software lock is disabled
   1 : Software lock is enabled

[Output]

None

GSCLN4A Get extension card parameter

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Extension card parameter (String, length = 2)

byte 1 (Extension card 1 status)

- bit 0 - 2 Extension card 1 length (0-6)
- bit 3 Reserved
- bit 4 Resident card 1 (0: disable, 1: enable)
- bit 5 Temporary card 1 (0: disable, 1: enable)
- bit 6 - 7 Reserved

byte 2 (Extension card 2 status)

- bit 0 - 2 Extension card 2 length (0-6)
- bit 3 Reserved
- bit 4 Resident card 2 (0: disable, 1: enable)
- bit 5 Temporary card 2 (0: disable, 1: enable)
- bit 6 - 7 Reserved

SSCLN4B Set extension card parameter

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

3. Extension card 1 status (Value, 0-255)
   bit 0 - 2 Extension card 1 length (0-6)
   bit 3 Reserved
   bit 4 Resident card 1 (0: disable, 1: enable)
   bit 5 Temporary card 1 (0: disable, 1: enable)
   bit 6 - 7 Reserved
4. Extension card 2 status (Value, 0-255)
   bit 0 - 2 Extension card 2 length (0-6)
   bit 3 Reserved
   bit 4 Resident card 2 (0: disable, 1: enable)
   bit 5 Temporary card 2 (0: disable, 1: enable)
   bit 6 - 7 Reserved

[Output]
None

GTCLN50 Get shift number

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]
Shift number (Value)
In BC-610 shift number is 1 - 4 (shift 1 - 4)
In CL-60 shift number is 1 - 8 (shift 1 - 8)
In CL-70 shift number is 49 - 53 (shift 1 - 5)

STCLN51 Set shift number

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Shift no. (Value)
   In BC-610 shift number is 1 - 4 (shift 1- 4)
   In CL-60 shift number is 1 - 8 (shift 1 -8)
   In CL-70 shift number is 49-53 (shift 1-5)

[Output]
None

GTOPM54 Get operating mode

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Operating mode (Value)

In BC-610 and CL-70 operating mode is 1 - 6 (Operating Mode 1 - 6)

In CL-70 operating mode is 49 - 54 (Operating Mode 1 - 6)

STOPM55 Set operating mode

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Operating mode (Value)

In BC-610 and CL-70 operating mode is 1 - 6 (Operating Mode 1- 6)

In CL-70 operating mode is 49-54 (Operating Mode 1- 6)

[Output]

None

GTSCN57 Get scanned data display control status

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Status (Value, 0-1)

0 : Scanned data display is disabled

1 : Scanned data display is enabled

STSCN58 Scanned data display control

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Scanned data display control (Value, 0-1)
   0 : Scanned data display is disabled
   1 : Scanned data display is enabled

[Output]

None

GTPIC58 Get printer control parameter

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Printer control parameter (String, length = 4)

byte 1	Keypad input Badge number send to printer
byte 2	Barcode input Badge number send to printer
byte 3	Magnetic input Badge number send to printer
byte 4	Print control flag

STPIC59 Set printer control parameter

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Printer control parameter (Value)

bit 0	Keypad input Badge number send to printer
bit 1	Barcode input Badge number send to printer
bit 2	Magnetic input Badge number send to printer
bit 3	Print control flag

[Output]

None

LDMES62 Display Message

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Message (String, Length = 1 - 31)

[Output]

None

LDNID63 Display working node ID

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

None

STACT71 Get auto switch shift time table

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)
3. Table no. (Value, 1-20)
4. Hour (Value, 00-23)
5. Minute (Value, 00-59)
6. Operating mode (Value, 1-6)
7. Shift no. (Value, 1-5)

[Output]

None

GTACT72 Set auto switch shift time table

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Auto switch shift time table (String, length = 4)
byte 1 00-23 (hour)
byte 2 00-59 (minute)
byte 3 01-06 (operating mode)
byte 4 01-05 (shift no.)

STBAT72 Set block time interval

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Block (Value, 0-7)
4. Start Hour (Value, 0-23)
5. Start Minute (Value, 0-59)
6. End Hour (Value, 0-23)
7. End Minute (Value, 0-59)

[Output]
None

GTOTC73 Get pin code status

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]
Check Level (Value, 0-1) 0: disable, 1: enable

STOTC74 Set pin code status

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Pin code status (Value, 0-1) 0: disable, 1: enable

[Output]
None

* If you change pin code status, all data will be loss. You must download all data again, after you change pin code status.

STOPS76 Set User Card as single check mode

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Block (Value, 0-7)
4. Badge Number (String, Length = 1 to 12)

[Output]

None

STTPS78 Set User Card as double check mode

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Block (Value, 0-7)
4. Badge number (String, Length = 1 to 8)
5. Pin code (String, Length = 1 to 5)

[Output]

None

GTALM79 Get sensor status (real time)

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]

Sensor Status (Value, 0-255)

* Bit 0 to 7 are defined as following :
  Bit 0 ---- Sensor 5
  Bit 1 ---- Sensor 0
  Bit 2 ---- Sensor 7
  Bit 3 ---- Sensor 6
  Bit 4 ---- Sensor 1
  Bit 5 ---- Sensor 5

Bit 6 ---- Sensor 3
Bit 7 ---- Sensor 4

STOTD7A Set door opening duration

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Door opening duration (Value, 1-255)

[Output]
None

DESIP7C Delete user Badge number

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Block (Value, 0-7)
4. Badge number (String, Length = 1 - 8)

[Output]
None

DEBLP7D Delete block badge number and pin code

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Block (Value, 0-7)

[Output]
None

STALM80 Set alarm active time

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Start Hour (Value, 0-23)
4. Start Minute (Value, 0-59)
5. End Hour (Value, 0-23)
6. End Minute (Value, 0-59)

[Output]

None

GTTCE82 Get error alarm status (Beep three times)

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)

[Output]

Status (Value, 0: disable 1: enable)

STTCE82 Set error alarm status (Beep three times)

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Status (Value, 0-1)

[Output]

None

STALF83 Set alarm enable/disable flag

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Status (Value, 0-1)

[Output]
None

GTALF84 Get alarm enable/disable flag

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 1-255)

[Output]
Alarm Flag (Value, 0: disable 1: enable)

STUSR90 Download user name

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Badge number (String, Length = 1 to 8)
4. Pin code (String, Length = 0 to 5)
5. User Name (String, Length = 1 to 15)

[Output]
None

STMMS99 Set Middle Level User Card as single check

[Input]
1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Block (Value, 0-7)
4. Badge number (String, Length = 1 to 13)

[Output]
None

STMMD9A Set Middle Level User Card as double check

[Input]

1. Comm. port (Value, 1-4)
2. Node ID (Value, 0-255)
3. Block (Value, 0-7)
4. Badge number (length = 1 to 8)
5. Pin code (String, Length = 1 to 5)

[Output]

None

9. Sample Program

9.1 Turbo C 2.0 Library

RECO.LIB (Turbo C) include all communication command. When user call this library , the source code must include the header files "**RECO.H**" and "**RECO1.H**". During linking procedure, "**RECO.LIB**" must be linked also to complete the process.

Example:

```
 #include   <RECO1.H>
 #include   <RECO.H>
 #include   <CONIO.H>
 #include   <DOS.H>
   int u=0,v=0;                    /* cursor location */
 main()
 {
  static int loc[100];
  int *l,q,i,j;
  int index,intev=0;
  int count=0;
  union REGS regs;
  clrscr();                       /* clear screen */
  outportb(LCR,0x80);
  outportb(BRD_LSB,0x0c);
  outportb(BRD_MSB,0x00);              /* Comm. port Initialization */
  outportb(LCR,0x03);
  outportb(MCR,0x03);
  outportb(IER,0x00);
  l=loc;
  loop: for (j=1;j<=5;j++)             /* polling ID 1-5 */
      {
        regs.h.ah=2;
        regs.h.dl=60;
        regs.h.dh=0;
        regs.h.bh=0;
        int86(VIDEO,&regs,&regs);      /* Set Location for display */
        printf("Polling  %d",j);
        RSTND09(1,j);                /* Call library (RSTND09) */
        if (u==0 && v==0)             /* u=v=0 not decrease     */
          {u+=1;v+=1;}
```

```
        regs.h.ah=2;
        regs.h.dl=u-1;
        regs.h.dh=v-1;
        regs.h.bh=0;
        int86(VIDEO,&regs,&regs);       /* back to display data location */
        count=0;intev=0;
        for (index=0;loc[index]!=NULL;index++)
           loc[index]=NULL;             /* initial loc[] */
        for (index=0;Data09[index]!=NULL;++index)
           if (Data09[index]==0x23)     /* check how many data? */
             {
              count++;                /* count data */
              *l++=index;             /* index indicates location. */
              }                       /* loc[] store 0x23 loc. */
  l=loc;                      /* initial pointer                              */
  for (index=0;Data09[index]!=NULL;index++)
     Data09[index]=to_decmal(Data09[index]);/*turn decmal value */
     q=loc[1]-loc[0];                 /* data length */
     if (q==0) goto check;              /* no data */
     if (q)                           /* one Data */
       {
        for (index=0;index((-q)-1);index++)
              printf("%d",Data09[index]);
              printf("      %d",j);
              printf("\n"); goto check;
        }
     for (index=0;index<N>   ;index++)     /* many data */
        {
        for (i=intev;i(q+intev-1);i++)
        printf("%d",Data09[i]);
        printf("      %d",j);
        intev+=q;printf("\n");
        }

  check: u=wherex();v=wherey();          /* keep data last location */
       if (j==5) goto loop;             /* keep polling */
      }
}
```

Note : If you wish to know the details of the library , please read the library disk readme.tc file .

9.2 Quick Basic Library

RECOLIB.BAS include communication command. When user call this library , the source code must include the header file "**RECOSYM.BI**" . During linking procedure "**RECOLIB.LIB**" must be linked also to complete the process.

Example:

```
'Sample program for Quik_BASIC 4.xx with function call
'
'INCLUDE:"RECOSYM.BI"                    'must include this file
CLS
ON ERROR GOTO FAULT
OPEN "COM1:9600,N,8,1,CS,DS" FOR RANDOM AS #1
DATF$="":FLAG=0
TESTLP:
  FOR NODEID=1 TO 5
      LOCATE 25,50:PRINT "NODEID=";NODEID;
      COMPORT=1
      CALL RSTND09 (COMPORT, NODEID, DATF$, FLAG)   '** call library
      IF FLAG = 0 THEN PRINT "Time out error";
      IF FLAG < > 1 THEN GOTO NOTOK
      IF DATF$ < > "" THEN
        LOCATE   ,8
        PRINT "DATF$=";DATF$;
        CALL ACKGN06 (COMPORT, NODEID)          '** call library
      END IF
NOTOK:
       IF FLAG=2 THEN PRINT "LRC error","DATF$=";DATF$;
       IF FLAG=3 THEN PRINT "Frame error";
  NEXT  NODEID
  PAX$ = ""
  PAX$ = INKEY$
  IF PAX$=CHR$(27) THEN END
  GOTO TESTLP
'**************************
'ERROR TRAP SUB
FAULT:
  RESUME NEXT
```

9.3 Clipper Library

```
DECLARE FT[8]              /* DATA FIELD */
MAX_NODE = 9               /* POLLING NODE */
WP_COMM=1                  /* COMM. PORT */
WP_BPS=9600                /* BAUD RATE */
INIT(WP_COMM,WP_BPS)       /* INITIAL RS-232 */

  FOR P_NODE=1 TO MAX_NODE   /* POLLING LOOP */
   PNODE=ALLTRIM(STR(P_NODE)) /* NODE NO. FOR DISPLAY */
   DO WHILE LEN(PNODE)<3
     PNODE="0"+PNODE
   ENDDO

   ON_LINE=.F.             /* ON-LINE, OFF-LINE FLAG */
   EMPTY_C=0               /* COUNTER */

   DO WHILE EMPTY_C<5 .AND. ON_LINE=.F.
     AA=SPACE(7)
     AA=ENQND05(WP_COMM,P_NODE)
     IF LEN(ALLTRIM(AA))=7
       ACK=CHR(126)+CHR(126)+CHR(1)+CHR(P_NODE)+CHR(6)+CHR(5)+CHR(126)
       IF AA=ACK
        ON_LINE=.T.
        LOOP             /* IF ENQND05 OK, THEN START POLLING DATA */
       ENDIF
     ENDIF
     ON_LINE=.F.           /* RETRY ENQND05 UNTILL COUNTER >5 */
     EMPTY_C=EMPTY_C+1
   ENDDO

   IF ON_LINE=.T.
     EMPTY_C=0             /* COUNTER */
     RBLOCK=0              /* RECEIVE BLOCK COUNTER */
     DO WHILE EMPTY_C<10     /* CHECK EMPTY COUNTER */
       BB=RSTND09(WP_COMM,P_NODE)  /* GET DATA */
       R_NODE=ASC(LEFT(BB,1))-48   /* FIRST BYTE OF DATA IS NODE ID */
       IF R_NODE<0
         R_NODE=R_NODE+256
       ENDIF
       IF LEN(BB)<>0 .AND. R_NODE=P_NODE
```

```
   RBLOCK=RBLOCK+1            /* RECEIVE ONE BLOCK */
   ? "["+PNODE+"]: RECEIVE BLOCK -----> "+ALLTRIM(STR(RBLOCK))
   EMPTY_C=0                  /* RESET EMPTY COUNTER */
   BB=SUBSTR(BB,2,LEN(BB)-1)    /* PROCE RECEIVE DATA */
   ST=1
   DLEN=0
   FOR I=1 TO 8               /* INITIAL EVERY FIELD */
     FT[I]=""
   NEXT I
   FOR I=1 TO LEN(BB)       /* GET EVERY FIELD FROM RECEIVE DATA */
     IF SUBSTR(BB,I,1)="#"
       DD=SUBSTR(BB,ST,DLEN+1)
       ST=I+1
       DLEN=0
       SST=1
       DDLEN=0
       FC=0             /* RESET FIELD COUNTER */
       FOR J=1 TO LEN(DD)
         IF SUBSTR(DD,J,1)=":" .OR. SUBSTR(DD,J,1)="#"
           FC=FC+1      /* GET ONE FIELD */
           FT[FC]=SUBSTR(DD,SST,DDLEN)
           SST=J+1
           DDLEN=0
         ELSE
           DDLEN=DDLEN+1
         ENDIF
       NEXT J

       DATA=""
       FOR JJ=1 TO FC    /* FC IS TOTAL FIELD */
         DATA=DATA+FT[JJ]+":"
       NEXT JJ
       DATA=LEFT(DATA,LEN(DATA)-1)  /* SKIP LAST ":" */
       ? SPACE(7)+DATA
     ELSE
       DLEN=DLEN+1
     ENDIF
   NEXT
   ACKGN06(WP_COMM,P_NODE)    /* DELETE LAST BLOCK */
ELSE
   EMPTY_C=EMPTY_C+1         /* NO DATA, TRY UNTILL COUNTER >=10 */
```

```
      ENDIF
    ENDDO
    ? "["+PNODE+"]: NODE "+PNODE+" RECEIVE COMPLETE !!"
    ? ""
  ELSE
    ? "["+PNODE+"]: NODE "+PNODE+" NOT ON-LINE !!"
    ? ""
  ENDIF
NEXT P_NODE
? ""
? " ALL NODE RECEIVE COMPLETE ..... "+CHR(7)
? ""
```