

GTXC SERIAL COMMUNICATION PROTOCOL – Standard Displays

PolyComp displays can be linked to a computer through a serial communication RS232, RS485 and CURRENT LOOP using the following **Default settings**:

BAUD RATE: 9600
START BIT : 1
STOP BITS : 2
PARITY : NONE

8 Bit ASCII code compatible with extended IBM ASCII code.
 These are the default settings which can be changed using the Dip-Switch on the **SERIAL COMMUNICATION BOARD**.

CONNECTING THE COMPUTER TO THE SIGN

When you power up – the sign will perform a self test and display the results on the screen. The setting of the “Serial Communication Board” will also be displayed. Be sure to use the same settings on the PC.

▪ **Using RS232**

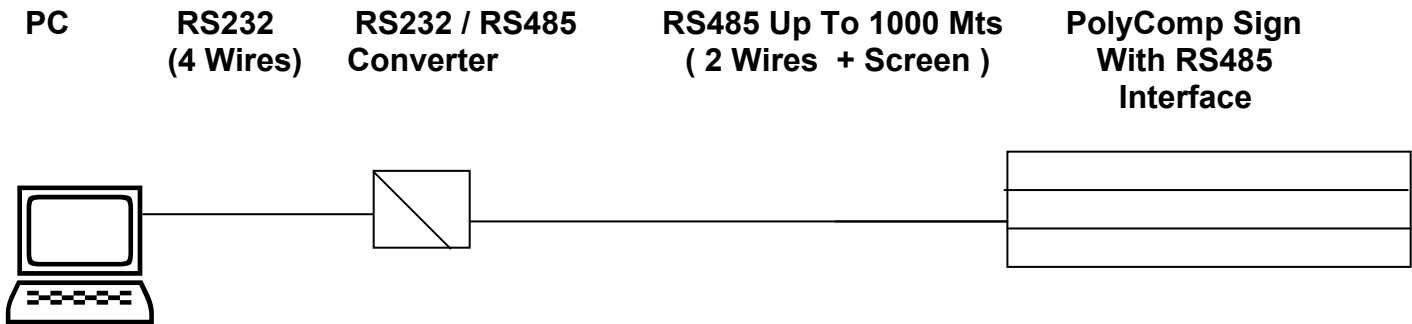
A Direct connection via RS232 can be used only for a short distance – up to approximately: 100 metres. Only 3 wires are used and these must connect as follows:

PC SIDE			SIGN SIDE		
<i>DIN....</i>	<i>25</i>	<i>DIN....</i>	<i>9</i>	<i>Round DIN....</i>	<i>5</i>
<i>Signal</i>					
TXD	PIN 2	PIN 3		PIN 3	RXD
RXD	PIN 3	PIN 2		PIN 2	TXD
GND	PIN 7	PIN 5		PIN 5	
			GND		

▪ **Using RS485**

Connect RS232 to the RS485 converter, between the PC and the sign.
 This international standard is more suitable for long distance communication, as well as Multidrop communication. **{Few signs can be connected on Two-Way communication to the Host [PC]}**

RS485 CONNECTIONS



The PC is linked to the RS485 converter via 4 wires namely: ~

- TXD** Transit Data from PC.
- RXD** Received Data to PC.
- DTR** Controls Traffic flow: + 12V => TX Mode, - 12 V => RX Mode .
- GND** Ground

Since RS485 only uses 2 wires, only one device can send data at a time.

The PC downloads a page of text, the sign checks the integrity of the message and if it is OK – it will transfer the page from the temporary RX Buffer into the message buffer.

If the acknowledge BIT on the Serial Status Byte is set (B3), then the sign will send “*Acknowledge*” back to the PC.

In order for the PC to get the reply *{when RS485 is used}* DTR needs to be at - 12V [**RX Mode**] . Once the reply has been sent, the PC can send the next page..etc...

When the last page has been sent, be sure that BIT 2 – on the “*Serial Status Byte*” has been Reset (= 0), so that the new message will be displayed immediately . When this BIT is set, it instructs the sign that more pages are ‘coming’ and therefore, the sign will ‘wait’ for these pages and will not display the new message ...however if no new data is received within 40 seconds. It will consider the last valid page -received as the last page and the sign will then display the new message.

COMMUNICATION PROTOCOL

These signs can be programmed for one way or two way communication by setting the relevant bit as explained below (**Should a two way communication be selected , be sure that your hardware can support it !**) .

The message should be sent to the sign page after page.

When a page is successfully received by the sign and provided that the “ACK” bit is set, it will respond with an “ACK” (**Decimal Value 6**) .

Should the message corrupt the sign will respond with “NAK” (**Decimal Value 21**) if the “ACK” bit is set .

Once the sign successfully received a page the next page can be sent and so on...

A delay of up to 3 seconds can be expected between the transmission and the response by the sign for the first page and up to 0.5 seconds for every consecutive page.

The first page to be sent to the sign is page 0 which consists of the current time and date from the computer. Thereafter the message pages are sent to the sign.

If there is no need to set the RTC, then page 0 can be omitted.

PROTOCOL

<u>Byte</u>	<u>Description</u>	<u>Decimal Value</u>	<u>Comments</u>
1	Header Sync	00	
2	Nol	??	(Number Of Text Lines On the Sign)
3	Sign Address	XX	(1 – 127,0 = “All Call”)
4	Etx	03	End Of Header .
5	Serial status flag	SERST	(See note #1)
6	Page Number	X 100	(ASCII)
7	Page Number	X 10	(ASCII)
8	Page Number	X 1	(ASCII)

[Message text or time and date (See details below)]

	EOT	04	End Of Text
Last Byte	Checksum	??	

(The checksum is an EXCLUSIVE ‘OR’ function bit by bit starting at the SYNC. Up to and including the “End Of Text” byte 04) .

➤ **Note # 1 Serial Status: - SERST**

B0 , B5 = 0

B1 = 1 => “Interrupt Mode” = 0 => Normal DISPLAY Mode .

B2 = 1 => More pages to come , = 0 => Last page .

B3 = 1 => “ACK” Requested , = 0 => No “ACK” required .

B4 = 1 Schedule Mode Operation = 0 => No schedule operation .

B6 – B7 All = 1 .

Explanation !!

B1. Interrupt Mode

In this mode, only one page can be sent. This page will not be transferred to the general message buffer. The message buffer will remain unaffected. The interrupted message will override the general message and will be displayed until a new message is received by the sign. When the sign is “Switched Off “, this data will be lost. On “Power Up” – the previous message will be displayed.

B2. More Pages To Come

When this BIT is set, the sign will be expecting more information to be sent. ***{If no new data is sent within 40 seconds, it will consider t□ Çhe last valid page received as the last page and therefore will display the new message }*** If the BIT

B3. ACK Requested

When this BIT is set, the sign will reply after every page. If the page was received OK

– it will send “**ACK**” {06 Decimal} or “**NAK**” {21 Decimal } – if the data is corrupt.

B4. Schedule Mode

When this BIT is set, a schedule mode is selected and the scheduling information will be extracted from the data that has been sent. If the BIT is RESET, the page will always be displayed without time limitations.

9	Hours MSB	
10	Hours LSB	
11	Minutes MSB	
12	Minutes LSB	
13	Seconds MSB	
14	Seconds LSB	
15	Date MSB	
16	Date LSB	
17	Month MSB	
18	Month LSB	
19	Day MSB	Always = 0

Data Pages (001 to last page)

Byte	Description (All Values In ASCII	
9	Start Date Day (MSB)	
11	Start Date Month (MSB)	
12	Start Date Month (LSB)	
13	Stop Date Day (MSB)	
14	Stop Date Day (LSB)	
15	Stop Date Month (MSB)	
16	Stop Date Month (LSB)	
17	Start Time Hours (MSB)	
18	Start Time Hours (LSB)	
19	Start Time Minutes (MSB)	
20	Start Time Minutes (LSB)	
21	Stop Time Hours (MSB)	
22	Stop Time Hours (LSB)	
23	Stop Time Minutes (MSB)	
24	Stop Time Minutes (LSB)	
25	Weekly Status	(<u>See note #2</u>)
26	TEMPO for this page	(<u>See Note # 3</u>)
27	FUNCTION of this page	(<u>See Note # 4</u>)
28	PAGE STATUS	(<u>See Note # 5</u>)
29 – END	ASCII Text For The Page .	

☛ **Important Note :** If schedule mode is **NOT** selected (*SERST B4 = 0*) then bytes 9 to 25 inclusive - **MUST BE OMITTED!**

Note # 2 : Weekly Status:

B0	Monday	(1 = ON , 0 = OFF)
B1	Tuesday	(1 = ON , 0 = OFF)
B2	Wednesday	(1 = ON , 0 = OFF)

B3	Thursday	(1 = ON , 0 = OFF)
B4	Friday	(1 = ON , 0 = OFF)
B5	Saturday	(1 = ON , 0 = OFF)
B6	Sunday	(1 = ON , 0 = OFF)
B7	= 1 Always	(1 = ON , 0 = OFF)

➤ **Note # 3 : TEMPO:**

This byte determines the duration that this page will **FREEZE** on the screen every time it is displayed.

B0 – B3	Duration this page will be displayed, see table below .
* B4, B5	Display status bits
* 0 0	This page is in “Timer” mode and will display as set by the scheduled time .
* 0 1	This page is always ON .
* 1 0	This page is always OFF .
* 1 1	Invalid .
B6, B7	Both bits must always be set .

B0 – B3 Setting

TEMPO (B0 – B3)	:	1	2	3	4	5	6	7	8	9
DURATION (SEC)	:	2	5	10	20	30	45	60	90	120

➤ **Note # 4 : Function For This Page:**

B3	B2	B1	B0	FUNCTION
0	0	0	0	Auto (Random select any function)
0	0	0	1	Appear
0	0	1	0	Wipe
0	0	1	1	Open
0	1	0	0	Lock
0	1	0	1	Rotate
0	1	1	0	Right
0	1	1	1	Left
1	0	0	0	Roll Up
1	0	0	1	Roll Down
1	0	1	0	Ping Pong
1	0	1	1	Fill Up
1	1	0	0	Paint
1	1	0	1	Fade In
1	1	1	0	Jump
1	1	1	1	Slide

☞ **Important Note:**

In slide mode the bottom line will scroll across the screen from right to left at a speed set by the **TEMPO** value (**Speed 1 is the fastest and 9 is the slowest**) Therefore the length of this line is flexible .

B4 = 1 => Display **CLOCK** on top line.

B5 = 1 => Display **TEMPERATURE** on top line.

B6 , B7 = Both are always “HIGH” .

➤ **Note # 5 : PAGE STATUS:**

- B0 = 1 => Top two lines join in “**BOLD**” mode .
B1 = 1 => Lines 3 & 4 join in “**BOLD**” mode .
B2 = 1 => Lines 5 & 6 join in “**BOLD**” mode .
B3 = 1 => Lines 7 & 8 join in “**BOLD**” mode .
B4 = 1 => AutoCentre **ON** { The text will be centred on the screen }
B5 = 1 => Foreign Language, = 0 => English .
B6 = 1 => Background colour **ON**.
 *{ Will automatically select as the opposite of the foreground colour – i.e: If
 Red Is Selected – Background Is Green . }*
B7 Always “High” .

CHANGE LETTER APPEARANCE ON THE DISPLAY

In Order to change the appearance of letters on the display, the following controls can be used:

A control block consists of 2 bytes:

- 1st Byte : Control Code = 28 BCD. [1C Hex]
2nd Byte : Command byte.

The following commands are available: ~

- F : Flash Characters .
E : Enlarge Characters
R : Change Colour to Red (*Colour Signs Only!*)
G : Change Colour To Green (*Colour Signs Only!*)
Y : Change Colour to Yellow (*Colour Signs Only!*)
M : Multicolour – i.e: Top Red Center Yellow Bottom Green (*Colour Signs Only!*)
D : Return To Default Setting – i.e: Normal {**Not Enlarged**} Red Letters Not Flashing .

Every page starts with the default settings.

EXAMPLE : The following string ~

“POLYCOMP” (28) “G IS” (28) “F” (28) “M THE” (28) “F” (28) “E BEST” WILL
DISPLAY : “POLYCOMP IS THE BEST”
“POLYCOMP” - RED ~ “THE” MULTI COLOUR FLASH .
“IS” - GREEN ~ “BEST” MULTI COLOUR ENLARGE .

IN THIS EXAMPLE ~

TEXT BETWEEN INVERTED COMMA’S “ ” IS ASCII CHARACTERS . NUMBERS
BETWEEN BRACKETS ARE IN bcd VALUES .

EXAMPLE OF GTX PROGRAM FOR 1 LINE DISPLAY ~ GTXC

```
10 OPEN "COM1 : 9600,N,8,2,CS,DS,CD"AS #1
15 CLS : PRINT TAB(30) " POLYCOMP TEST GTX"
16 PRINT TAB(28) "~~~~~"
20
   A$=CHR$(0)+CHR$(1)+CHR$(2)+CHR$(3)+CHR$(&HCO)+"001"+CHR$(&HE2)+CHR$(&HC1)+CHR$(&HCO)
25 FOR CNT=100 TO 10000
30 MSG$= "CNT =" +STR$(CNT)
40 TXT$=A$+MSG$+CHR$(4)
50 CK=0
60 FOR I = 1 TO LEN(TXT$)
70 CK=CK XOR ASC(MID$(TXT$,I,1))
80 NEXT I
90 PRINT #1,TXT$;CHR$(CK);:LOCATE 5,34:PRINT "CNT =";CNT
100 FOR D=1 TO 5000:NEXT D
110 NEXT CNT
```

EXAMPLE OF GTX PROGRAM FOR DOUBLE LINE DISPLAY ~ GTXC

```
10 OPEN "COM1 : 9600,N,8,2,CS,DS,CD"AS #1
15 CLS : PRINT TAB(30) " POLYCOMP TEST GTX"
16 PRINT TAB(28) "~~~~~"
20
   A$=CHR$(0)+CHR$(2)+CHR$(0)+CHR$(3)+CHR$(&HCO)+"001"+CHR$(&HE2)+CHR$(&HC1)+CHR$(&HCO)
26 FOR CNT=100 TO 10000
30 MSG$= "CNT =" +STR$(CNT)
41 TXT$=A$+MSG$+CHR$(4)
51 CK=0
61 FOR I = 1 TO LEN(TXT$)
71 CK=CK XOR ASC(MID$(TXT$,I,1))
81 NEXT I
91 PRINT #1,TXT$;CHR$(CK);:LOCATE 5,34:PRINT "CNT =";CNT
101 FOR D=1 TO 5000:NEXT D
110 NEXT CNT
```