DP9800
Temperature Monitor
Communication Manual

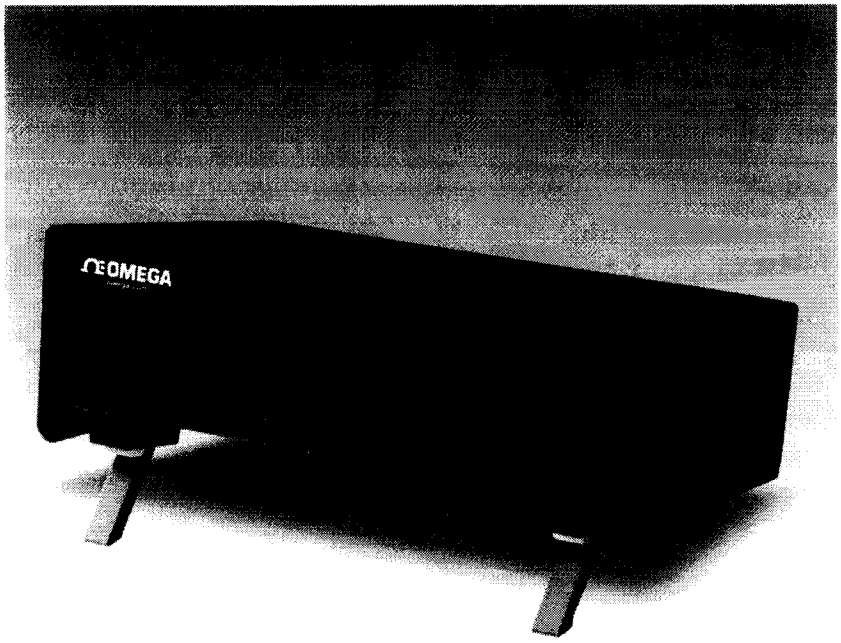

Table of Contents

1. Introduction

This document describes the Serial Communication Interface used in the DP9800 temperature monitor.

The hardware section covers the industrial standard supported by this communication interface.

The software covers the protocol used and the command set supported by this communication interface.

2. Hardware

Internally, the DP9800 uses a USB to serial converter. When the DP9800 is connected to a USB port of a computer, this will create a virtual serial communication port on the computer when appropriate software driver is installed on the computer.

Once this serial communication link is established, the communication between the DP9800 and the computer can proceed according to the Electronics Industry Association (EIA) RS232 interface.

This serial interface uses the following configurations:

Baud rate: 38400

Parity: none

Data bits: 8

Stop bits: 1

Flow control: none

3. USB serial driver

There are drivers for operating systems including: Windows, Linux, Mac OS X, and Windows CE.

If the driver is already installed, the operating system will recognise the USB to Serial converter when the DP9800 is connected to the computer via a USB port.

If the driver is not present, depending on the setting of the computer, the driver could be download from the internet and installed automatically.

For manual installation of the USB serial driver, please visit the following website. You can download and install the appropriate driver for your operating system.

Website address: http://www.ftdichip.com/Drivers/VCP.htm

4. Communication protocol

This protocol establishes the control procedures for data communication links that employ the communication control characters of the American National Standard Code for Information Interchange, X3.4-1968 (ASCII). The ASCII coded character set is included at the end of this document.

It is a subset of the communication protocol as described in the standard document ANSI X3.28-1976 (American National Standard procedures for the use of the communication control characters of American national standard code for information interchange in specified data communication links).

4.1 Communication control characters

EOT (End of Transmission) is a communication control character used to indicate the conclusion of a transmission that may have contained one or more texts and any associated headings.

STX (Start of Text) is a communication control character that precedes a sequence of characters that is to be treated as an entity and entirely transmitted through to the ultimate destination. Such a sequence is referred to as "text".

ETX (End of Text) is a communication control character used to terminate a sequence of characters started with STX and transmitted as an entity.

ENQ (Enquiry) is a communication control character used in data communication systems as a request for a response from a remote station.

ACK (Acknowledgement) is a communication control character transmitted by a receiver as an affirmative response to a sender.

NAK (Negative Acknowledge) is a communication control character transmitted by a receiver as a negative response to the sender.

BCC (Block Check Character) is a character added at the end of a message or transmission block to facilitate error detection.

The BCC is generated by taking a binary sum independently (without carry) on each of the seven individual levels of the transmitted code (b1 through b7).

The summation to obtain the block check character is started at the first appearance in a message of STX. The STX character is not included in the summation.

All characters after the start of the BCC summation are included in the summation, through and including the control character ETX that signals that the next following character is BCC.

The BCC is transmitted as the next character following the transmission of each ETX.

4.2 Poll sequence

Computer: (EOT) (C1) (ENQ)

L200: (STX) (C1) (D1) ... (Dn) (ETX) (BCC)

(C1) - This character represent the request command parameter.

(D1 to Dn) - These characters represent the value of the request command parameter.

4.3 Send sequence

Computer: (EOT) (STX) (C1) (D1) ... (Dn) (ETX) (BCC)

L200: (ACK) or (NAK)

(C1) - This character represent the send command parameter.

(D1 to Dn) - These characters represent the value of the send command parameter.

(ACK) - This character is sent by the DP9800 if the message is correctly received and executed.

(NAK) - This character is sent by the DP9800 if the message is incorrect and the command executed with error.

5. Command set

The DP9800 command set is arranged to support Poll command and Send command.

5.1 Poll command

The DP9800 supports the following Poll commands.

5.1.1 Read temperature

Computer: (C1) = 'T' for reading temperature

L200: (C1) (t1) (t2) (t3) (t4) (t5) (t6) (t7) (t8) (ff)

where:

(t1) is the channel 1 temperature in ddddd.dd decimal

(t2) is the channel 2 temperature in ddddd.dd decimal

(t3) is the channel 3 temperature in ddddd.dd decimal

(t4) is the channel 4 temperature in ddddd.dd decimal

(t5) is the channel 5 temperature in ddddd.dd decimal

(t6) is the channel 6 temperature in ddddd.dd decimal

(t7) is the channel 7 temperature in ddddd.dd decimal

(t8) is the channel 8 temperature in ddddd.dd decimal

(ff) is the system flag in hexadecimal

Example:

Computer: (EOT) T (ENQ)

L200: (STX) T (SP) 1 7 5 9 . 5 6 (SP) 1 7 5 9 . 5 6 (SP) 1 7 5 9 . 5 6 (SP) 1 7 5 9 . 5 6 (SP) 1 7 5 9 . 5 6 (SP) 1 7 5 9 . 5 6 (SP) 1 7 5 9 . 5 6 0 2 (ETX) (BCC)

5.1.2 Read millivolt

Computer: (C1) = 'M' for reading millivolt (for TC instrument only)

L200: (C1) (m1) (m2) (m3) (m4) (m5) (m6) (m7) (m8)

where:

(m1) is the channel 1 millivolt in ddd.dddd decimal

(m2) is the channel 2 millivolt in ddd.dddd decimal

(m3) is the channel 3 millivolt in ddd.dddd decimal

(m4) is the channel 4 millivolt in ddd.dddd decimal

(m5) is the channel 5 millivolt in ddd.dddd decimal

(m6) is the channel 6 millivolt in ddd.dddd decimal

(m7) is the channel 7 millivolt in ddd.dddd decimal

(m8) is the channel 8 millivolt in ddd.dddd decimal

Example:

Computer: (EOT) T (ENQ)

L200: (STX) T (SP) 8 2 . 7 6 9 7 (SP) 8 2 . 7 6 9 7 (SP) 8 2 . 7 6 9 7 (SP) 8 2 . 7 6 9 7 (SP) 8 2 . 7 6 9 7 (SP) 8 2 . 7 6 9 7 (SP) 8 2 . 7 6 9 7 (SP) 8 2 . 7 6 9 7 (ETX) (BCC)

5.1.3 Read resistance

Computer: (C1) = 'R' for reading resistance (for RTD instrument only)

L200: (C1) (R1) (R2) (R3) (R4) (R5) (R6) (R7) (R8)

where:

(R1) is the channel 1 resistance in dddd.ddd decimal

(R2) is the channel 2 resistance in dddd.ddd decimal

(R3) is the channel 3 resistance in dddd.ddd decimal

(R4) is the channel 4 resistance in dddd.ddd decimal

(R5) is the channel 5 resistance in dddd.ddd decimal

(R6) is the channel 6 resistance in dddd.ddd decimal

(R7) is the channel 7 resistance in dddd.ddd decimal

(R8) is the channel 8 resistance in dddd.ddd decimal

Example:

Computer: (EOT) T (ENQ)

L200: (STX) T (SP) 3 9 0 . 4 0 0 (SP) 3 9 0 . 4 0 0 (SP) 3 9 0 . 4 0 0 (SP) 3 9 0 . 4 0 0 (SP) 3 9 0 . 4 0 0 (SP) 3 9 0 . 4 0 0 (SP) 3 9 0 . 4 0 0 (ETX) (BCC)

5.1.4 Read lead resistance

Computer: (C1) = 'r' for reading lead resistance

L200: (C1) (R1) (R2) (R3) (R4) (R5) (R6) (R7) (R8)

where:

(R1) is the channel 1 lead resistance in dddd.ddd decimal

(R2) is the channel 2 lead resistance in dddd.ddd decimal

(R3) is the channel 3 lead resistance in dddd.ddd decimal

(R4) is the channel 4 lead resistance in dddd.ddd decimal

(R5) is the channel 5 lead resistance in dddd.ddd decimal

(R6) is the channel 6 lead resistance in dddd.ddd decimal

(R7) is the channel 7 lead resistance in dddd.ddd decimal

(R8) is the channel 8 lead resistance in dddd.ddd decimal

Example:

Computer: (EOT) r (ENQ)

L200: (STX) T (SP) (SP) (SP) 0 . 0 0 0 (SP) (SP) (SP) 0 . 0 0 0 (SP) (SP) (SP) 0 . 0 0 0 (SP) (SP) (SP) 0 . 0 0 0 (SP) (SP) (SP) 0 . 0 0 0 (SP) (SP) (SP) 0 . 0 0 0 (SP) (SP) (SP) 0 . 0 0 0 (SP) (SP) (SP) 0 . 0 0 0 (ETX) (BCC)

5.1.5 Read system parameters

Computer: (C1) = 'S' for reading status

L200: (C1) (yymmdd) (hhmmss) (ff) (aa) (mmmm) (iiii) (version_17) (pppp)

where:

(yymmdd) is the year, month and day (current date) in decimal

(hhmmss) is the hour, minute and second (current time) in decimal

(ff) is the system flag in hexadecimal

[T x x L x S A F] system flag bit map

F – bit 0: temperature unit, 0=deg C; 1=deg F

A – bit 1: audible button, 0=silence; 1=audible

S – bit 2: autoscan, 0=no scan; 1=auto scan active

L – bit 4: logging, 0=no logging; 1=logging active

T – bit 7: instrument type, 0=TC; 1=PT

x – bit 3, 5 and 6: must be 0

(aa) is the auto scan delay in hexadecimal

(mmmm) is the maximum log count in hexadecimal

(iiii) is the log interval in seconds in hexadecimal

(version_17) is the version of the firmware in 17 ASCII characters

(pppp) is the current log pointer in hexadecimal

Example:

Computer: (EOT) S (ENQ)

L200: (STX) S 1 1 1 2 0 7 1 3 4 4 5 9 0 2 0 5 0 2 0 0 0 0 0 5 L 2 0 0 R 1 . 2 / 2 0 1 0 0 9 0 2 0 2 3 7 (ETX) (BCC)

5.1.6 Read channel parameters

Computer: (C1) = '0' .. '8' for reading channel 0 to 8 parameters

L200: (C1) (tt) (mmmmmmmm) (cccccccc)

where:

(tt) is the channel type

00 – type J thermocouple or PT100 resistance thermometer

01 – type K thermocouple

02 – type T thermocouple

03 – type E thermocouple

04 – type N thermocouple

05 – type R thermocouple

06 – type S thermocouple

07 – type B thermocouple

(mmmmmmmm) is the channel calibration slope in ddd.dddd format

(cccccccc) is the channel calibration intercept in ddd.dddd format

Example:

Computer: (EOT) 1 (ENQ)

L200: (STX) 1 0 0 (SP) (SP) 0 . 9 9 9 1 (SP) - 0 . 0 0 2 8 (ETX) (BCC)

5.1.7 Download log data

Computer: (C1) (BBBB)

L200: (C1) (BBBB) (yymmdd) (hhmmdd) (d1) .. (d8)

where:

(C1) = 'D' for reading block of log data

(BBBB) is the block address dddd in decimal

(yymmdd) is the year, month and day (current date) in decimal

(hhmmss) is the hour, minute and second (current time) in decimal

(d1) .. (d8) log data for channel 1 to 8 [eemmmmmm] in floating point hexadecimal format

Example:

Computer: (EOT) D 0000 (ENQ)

L200: (STX) D 0 1 4 4 1 1 0 4 2 7 1 7 5 1 2 1 1 9 d 9 c a 4 1 5 7 e a d 7 4 1 4 d 9 1 d 7 4 1 8 9 c b 5 2 4 3 0 1 f c d 6 4 1 0 e 4 e d 6 4 1 f 0 f 1 d 5 4 1 1 f 3 e d 4 4 1 (ETX) (BCC)

This corresponds to:

Block: 0144

Date: 27/04/11

Time: 17:51:21

d1 .. d8: 25.36,26.99,26.95,210.80,26.87,26.79,26.74,26.53

5.2 Send command

The L200 supports the following Send commands.

5.2.1 Write system parameters

Computer: (C1) (yymmdd) (hhmmss) (ff) (aa) (iiii)

L200: (ACK) or (NAK)

where:

(C1) = 'S' for writing system parameters

(yymmdd) is the year, month and day (current date) in decimal

(hhmmss) is the hour, minute and second (current time) in decimal

(ff) is the system flag in hexadecimal

(ff) is the system flag in hexadecimal

[T x x L x S A F] system flag bit map

F – bit 0: temperature unit, 0=deg C; 1=deg F

A – bit 1: audible button, 0=silence; 1=audible

S – bit 2: autoscan, 0=no scan; 1=auto scan active

L – bit 4: logging, 0=no logging; 1=logging active

T – bit 7: instrument type, 0=TC; 1=PT

x – bit 3, 5 and 6: must be 0

(aa) is the auto scan delay in hexadecimal

(iiii) is the log interval in seconds in hexadecimal

Example:

Computer: (EOT) S (STX) S 1 1 1 2 0 7 1 3 4 4 5 9 0 2 0 5 0 0 0 5 (ETX) (BCC)

L200: (ACK)

5.2.2 Write channel parameters

Computer: (C1) (STX) (tt) (mmmmmmmm) (cccccccc) (ETX) (BCC)

L200: (ACK) or (NAK)

where:

(C1) = '0' .. '8' for reading channel 0 to 8 parameters

(tt) is the channel type

00 – type J thermocouple or PT100 resistance thermometer

01 – type K thermocouple

02 – type T thermocouple

03 – type E thermocouple

04 – type N thermocouple

05 – type R thermocouple

06 – type S thermocouple

07 – type B thermocouple

(mmmmmmmm) is the channel calibration slope in ddd.dddd format

(cccccccc) is the channel calibration intercept in ddd.dddd format

Example:

Computer: (EOT) (STX) 1 0 0 (SP) (SP) 0 . 9 9 9 1 (SP) - 0 . 0 0 2 8 (ETX) (BCC)

L200: (ACK)

6. ASCII coded character set

The following table shows the American National Standard Code for Information Interchange (ASCII) coded character set used for the general interchange of information among information processing systems, communication systems and associated equipment.

dec	hex	char	dec	hex	char	dec	hex	char	dec	hex	char
0	00	NUL	32	20	SP	64	40	@	96	60	`
1	01	SOH	33	21	!	65	41	A	97	61	a
2	02	STX	34	22	"	66	42	B	98	62	b
3	03	ETX	35	23	#	67	43	C	99	63	c
4	04	EOT	36	24	$	68	44	D	100	64	d
5	05	ENQ	37	25	%	69	45	E	101	65	e
6	06	ACK	38	26	&	70	46	F	102	66	f
7	07	BEL	39	27	'	71	47	G	103	67	g
8	08	BS	40	28	(	72	48	H	104	68	h
9	09	HT	41	29	)	73	49	I	105	69	i
10	0A	LF	42	2A	*	74	4A	J	106	6A	j
11	0B	VT	43	2B	+	75	4B	K	107	6B	k
12	0C	FF	44	2C	,	76	4C	L	108	6C	l
13	0D	CR	45	2D	-	77	4D	M	109	6D	m
14	0E	SO	46	2E	.	78	4E	N	110	6E	n
15	0F	SI	47	2F	/	79	4F	O	111	6F	o
16	10	DLE	48	30	0	80	50	P	112	70	p
17	11	DC1	49	31	1	81	51	Q	113	71	q
18	12	DC2	50	32	2	82	52	R	114	72	r
19	13	DC3	51	33	3	83	53	S	115	73	s
20	14	DC4	52	34	4	84	54	T	116	74	t
21	15	NAK	53	35	5	85	55	U	117	75	u
22	16	SYN	54	36	6	86	56	V	118	76	v
23	17	ETB	55	37	7	87	57	W	119	77	w
24	18	CAN	56	38	8	88	58	X	120	78	x
25	19	EM	57	39	9	89	59	Y	121	79	y
26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	59	3B	;	91	5B	[	123	7B	{
28	1C	FS	60	3C	<	92	5C	\	124	7C	|
29	1D	GS	61	3D	=	93	5D	]	125	7D	}
30	1E	RS	62	3E	>	94	5E	^	126	7E	~
31	1F	US	63	3F	?	95	5F	_	127	7F	DEL

7. Notes