



**COMMUNICATIONS SUPPLEMENT**

DESCRIPTION ..... 1  
KEY COMMAND ..... 3  
READ COMMAND ..... 4  
STATUS COMMAND ..... 5  
WRITE COMMAND ..... 7  
UP LOAD COMMAND ..... 8  
DOWN LOAD COMMAND ..... 8  
CANCEL COMMAND ..... 9  
REVISIONS ..... 10

iii **NOTICE** iii

**FARNAM RESERVES THE RIGHT TO MAKE CHANGES TO ITS PRODUCTS OR SPECIFICATIONS AT ANY TIME, WITHOUT NOTICE, IN ORDER TO IMPROVE THE DESIGN OR PERFORMANCE AND TO SUPPLY THE BEST POSSIBLE PRODUCT. THE INFORMATION IN THIS MANUAL HAS BEEN CAREFULLY CHECKED AND IS BELIEVED TO BE ACCURATE. HOWEVER, NO RESPONSIBILITY IS ASSUMED FOR INACCURACIES.**

**FARNAM  
7550 SERIES  
PROCESS CONTROLLER  
COMMUNICATIONS SUPPLEMENT**

This supplement contains information relating to the optional RS232 communications interface for the Model 7550 Temperature Controllers. This allows the user to have direct access to the Controller via a standard RS232 link. Through this serial link, all of the standard functions of the Controller may be activated, tested and adjusted.

Since this unit is a Controller, it does not handle a great deal of data. Therefore, the link has been optimized to allow the user, through very simple instructions, to control and interrogate the unit.

Ten 'key' commands allow the user to instruct the Controller to perform all of its normal operations. However, only a couple of these are actually required for computer operation. The remainder are provided, but are not often used, since these keys are utilized in programming. The serial link has direct access to the programming stack through the 'READ', 'WRITE', 'UP LOAD' and 'DOWN LOAD' commands without using 'key' commands. Utilizing these commands, the serial link can interrogate or overwrite any or all of the items in the PROGRAM stack.

The basic status conditions for the Controller are accessed through 4 bytes. These bytes may be read out at any time using the 'STATUS' command.

The Controller may be hooked to any standard terminal or computer system via the DB9 connector on the back side of the circuit board located on the inside lid of the panel. The pin out follows the standard IBM DB9 configuration. Therefore, it may be wired directly to an IBM PC/AT or its equivalent. The link is fixed at 9600 baud, 8 bit, 1 stop and 1 start bit, no parity. This link only requires 3 leads to function. As viewed from the Controller, Pin 5 is the 'Signal Ground', Pin 3 is 'Receive Data' and Pin 2 is 'Transmit Data'.

The internal ACIA utilizes a full duplex interrupt driven transmission scheme. Thus, the unit may receive and transmit simultaneously, as well as continue to perform its normal functions. Therefore, the Controller may be interrogated even though it is performing its program.

The Protocol for the serial link consists of 7 basic commands:

K – KEY  
S – STATUS  
R – READ  
W – WRITE  
D – DOWN LOAD  
U – UP LOAD  
X – CANCEL (CLEAR)

All commands will be prefixed by one of these 7 letters. The data format is standard ASCII and all data, with the exception of the Status Bytes are BCD values. The Status Bytes are transmitted in ASCII as hexadecimal, since they contain bit information.

UP LOAD and DOWN LOAD are the only group commands. They each have a specific format and a specific number of bytes of information. They are structured to DOWN LOAD all of the programming information at one time or READ the complete programming stack. All of the rest of the commands are structured to handle the data one byte at a time.

The following is a breakdown of each of the commands and the way they are accessed. It should be noted that the format allows for the unit to be hooked directly to a dumb computer terminal. This can be very useful in checkout. The unit will echo all characters that are typed to it. When used with a terminal, this will provide the appropriate display. When used with a computer system, this will provide direct feedback of the fact that unit has accepted the data.

All commands are completed with a carriage return from the computer. With the exception on the CANCEL command (X), all commands will be acknowledged by a carriage return, line feed (\$0D,\$0A).

## KEY COMMAND

The KEY COMMAND allows the user to instruct the Controller just as would be done by depressing the Face Panel keys. The exact operation and sequence for these keys is covered in the Controller manual. This description will simply indicate how the link may be used to send these key functions.

The command is entered as a letter followed by 2 numbers, followed by a carriage return:

K01(Return)

The 'K' indicates that this is to be a KEY command. The '01' indicates the KEY number and the 'Return' activates the command.

The following is a listing of the key numbers:

<u>KEY NUMBER</u>	<u>KEY</u>
1	DOWN
2	AUX
3	RETURN/SILENCE
4	HOLD
5	START
6	UP
7	SETUP
8	STOP/RESET

When a KEY command is sent, the Controller will echo each of the characters and acknowledge with a carriage return line feed, once the command is entered. If an invalid command is detected, it will simply be ignored, although it will acknowledge the fact that the command has been received.

<b>READ COMMAND</b>
---------------------

The READ command is utilized to read from the Controller any of the program data. The format for the command is essentially the same as the KEY command.

R07(Return)

The 'R' indicates to the Controller that the command is to be a READ command. The next two digits indicate the information that is to be read. The carriage return indicates that the command is to be activated. The following is a listing of the data locations that may be read:

<u>DATA LOCATION</u>	<u>DESCRIPTION</u>
1	CS
2	PS
3	HI
4	LO
5	AC
6	UNUSED
7	PA
8	CR
9	PB
10	RE
11	RA
12	CD (1=UP, 4=DOWN)
13	AT
14	RL
15	TT
16	DL
17	UNUSED
18	CL
19	FC
20	TU (0=DEG C, 1=DEG F)
21	HL
22	HD
23	UNUSED
24	UNUSED
25	PROCESS TEMPERATURE
26	REMAINING TIME

It should be noted that while most of the values will read out directly as they appear on the unit, the Time Values will display in Minutes and Seconds, but will read out in seconds only. Therefore, the clock setpoint on the unit may be programmed for 5:00 (Minutes:Seconds). The data that is read from the unit will come out as 0300 seconds. All data is returned in ASCII format with 4 BCD characters.

<b>STATUS</b>
---------------

The STATUS command is utilized to access 4 bytes of information that are utilized by the Controller to indicate its operating status. The same format as the KEY and READ commands is utilized to READ the status bytes:

S01(Return)

The 'S' indicates the activation of a STATUS command. The number indicates the status byte that is to be read out. The carriage return activates the command. The following is a listing of the available bytes:

<u>NUMBER</u>		<u>DESCRIPTION</u>
01	ALARM	ALARM CONDITIONS
02	MODBYT	SYSTEM MODE
03	SYSBYT	TIMER MODE
04	OUTBYT	OUTPUTS
09	ALL	READ ALL STATUS BYTES

Each of these bytes contains bits that are set internally by the control algorithms and indicate all the operating information for this system. The ALARM and MODE bytes (MODBYT) are displayed on the Controller via the ALARM and MODE LEDs. The 'System Byte' (SYSBYT) contains information about the current status of the Timer. The 'Output Byte' (OUTBYT1) indicate the condition of the outputs. The following is a description of the appropriate bits that may be utilized to monitor the system functions:

<u>ALARM</u>	<u>MODBYT</u>	<u>SYSBYT</u>	<u>OUTBYT1</u>
7- ---	7-NORM	7- ---	7- ---
6- ---	6-HOLD	6- ---	6- SFTY INTERL
5-HL	5-WARMUP	5 - ---	5-OPEN PROC T/C
4-SFTY	4-ALARM	4-TMR ZERO	4-OPEN HL T/C
3-LO	3-PROG	3-TMR OVER	3- AUX INPUT
2-HI	2- ---	2-PREWARN	2-AUX OUTPUT
1-SENSOR	1- ---	1- ---	1- SAFETY RELAY
0-SYS	0- ---	0-TMR RUNNING	0- HEAT

The meaning for each of the ALARM and MODE LEDs is covered in the Controller Manual. The SYSTEM BYTE indicates whether the Timer is running or not, and whether it is in the OVERTIME or PREWARN modes.

The OUTPUT BYTE contain both positive and negative logic bits. For those bits that are unmarked, a '1' indicates that the output is ON. For those bits that are marked with a not (—), the outputs are on when the bit is '0'.

Also, it should be noted that many of the undefined bits are used internally and therefore may be either '1' or '0' at any given reading.

These bytes require data transmission in a hexadecimal format. The actual data is sent as an ASCII character, but its meaning is translated in hexadecimal to determine the appropriate bit pattern. For example, the ASCII transmission of \$30,\$38 would translate to a hex reading of 08, which would indicate for the ALARM BYTE that a low alarm condition existed.

After the carriage return, the Controller will acknowledge with a carriage return line feed and then send the two ASCII characters that indicate the hex representation for the appropriate bit pattern requested.

## WRITE

The WRITE command allows the user to overwrite most of the information in the programming stack. The calibration value is not accessible. It should be noted that while this information may be over written, it will not be permanently saved in the controller without first accessing the SETUP mode and then activating the SAVE command via the keys. If the values written are to be permanently saved in the controller's EEPROM memory, after all changes have been made, a K07 (SETUP) followed by a K03 (RETURN) must be transmitted.

If it is not desirable to have these values permanently saved, the user may simply go in and overwrite the current information for temporary use. When the system is repowered, the information that is currently stored in its EEPROM will be reinserted into the SETUP stack.

The following is the format for this command:

W020750

The command essentially follows the same format as all the previous commands. The 'W' indicates that it is a WRITE command, the next two characters indicate the location that is to be written to and the last four characters indicate the data value that is to be entered. Again, the data is in BCD and transmitted in an ASCII format. The example WRITE command would put 750 Degrees C. in the Process Setpoint.

The data locations are the same as those covered in the READ command section. However, locations 06, 17 and 23 through 23, are READ only and may not be written to. Locations 24 and 25 simply provide a Read Out for the current Process Temperature and Remaining Time.



## **CANCEL**

The CANCEL command is simply a way to reestablish proper control, should an error occur or an incorrect command be transmitted. For the most part, an incorrect command will simply be ignored and the controller will prepare for an additional command. However, a command may be canceled midstream by transmitting an 'X' (ASCII). This command does not require a carriage return, nor will it acknowledge with a carriage return. However, it will echo an 'X' to indicate that the CANCEL command has been received.

The command may also be utilized as a clear and/or acknowledgment of the Controller being on line.

<b>MANUAL REVISIONS</b>
-------------------------

<b><u>Revision #</u></b>	<b><u>Eng. #</u></b>	<b><u>Program #</u></b>	<b><u>Revision Made</u></b>
Rev 01	DT7510	DT7510CE	Origination