DESIGN OF A MICROPROCESSOR BASED PROGRAMMABLE SYSTEM TO PROCESS TEMPERATURE INFORMATION FROM A HOT SURFACE

Thomas Philip, Stefan Jeglinski,
Richard D. Benton, Robert L. Cook
MHD Energy Center
Mississippi State University
Mississippi State, MS 39762

Thomas Philip, Research Associate, MHD Energy Center, Member ACM; Stefan Jeglinski, Research Assistant, MHD Energy Center; Richard D. Benton, Professor, Electrical Engineering; Robert L. Cook, Professor, Physics & Chemistry

Abstract

A CDP1802 microprocessor based stand-alone system (target system) has been designed to control a two-color pyrometer (TCP) system remotely and to process the temperature information from a simulated magnetohydrodynamics (MHD) test facility, at Mississippi State University. The analog signals from two pyrometer detectors are digitized and the temperature and emissivity of the surface are computed. The results are displayed on a 20-column printer. Data at various emissivity settings are collected with the help of a stepper motor. The measurements may be done individually or in a continuous mode. A sequence of events may be programmed into the system through a 4x4 key matrix. Software has been developed in FORTH, a highly structured language. The system design and development has been accomplished through stepwise refinement.

A. System Description

A two color pyrometer (TCP) is used for the simultaneous measurements of temperature and emissivity of the hot wall (about 1300°F) of the magnetohydrodynamic (MHD) test facility at Mississippi State University.[1,2]

A schematic diagram of the system setup for the measurements is shown in Figure 1. Pyrometer detector head A is sensitive to one color (wavelength = 2.1μ), and detector head B is sensitive to another color (wavelength = 0.8μ). The analog output from the pyrometers is directly proportional to the temperature of the surface where the heads are focussed, for a selected value of emissivity. The emissivity value may be changed by changing the emissivity compensation control of the pyrometer system. The temperatures indicated by the two pyrometers are equal only at a certain emissivity value, as shown in Figure 2. The temperature and emissivity corresponding to this "crossing point", with some correction, are the same as the temperature and emissivity for the surface.

This paper discusses the design and implementation of hardware and software to interface the TCP system with a microprocessor for remote control and fast data acquisition.

B. Problem Specification

Develop a microprocessor-based stand-alone system to aquire data from TCP remotely and to process the data. The system should perform the following specific functions:

1) Control the emissivity compensation unit of the TCP via a stepper motor.
2) Receive data from the two pyrometers for the same emissivity value in the range 0.2 to 1.0.
3) Determine the crossing point in the two sets of data.
4) Compute and display temperature and emissivity.
5) Make single, multiple and continuous measurements.
6) Remember a sequence of events and execute them later on command.
7) Communicate
   a) with the user to accept command and data.
   b) with external devices for data transfer.

C. Target System Design

It has been known for some time that "structure" is the key to a good design.[3] This structure may be achieved through stepwise refinement of the task. The resulting modules and interface between them form the basic building blocks of the structure.

An initial conception of the complete system is shown in Figure 1. Additional modules produced by further refinement are shown in Figure 3.[4] The arrows indicate the direction of data flow between the various modules.

A stepper motor with 200 steps per revolution was selected for the emissivity control. It enables selection of an emissivity value between 1.0 and 0.2. The data collection is achieved through an analog-to-digital converter with twelve bits resolution. The system-user communication was established through a 4 x 4 key matrix, a 20-column printer and a liquid crystal display (LCD) interfaced to the microprocessor. A parallel or serial transmission interface enabled data communication with an external device.

The stepwise refinement process also resulted in a number of software interface modules. The four high-level modules and additional modules resulting from further refinement which are lower in hierarchy, are shown in Figure 4.

The emissivity compensation controller software steps the stepper motor and sets the emissivity to a desired value.

The computation module first collects data from the two pyrometers using the analog-to digital converter. Then the crossing point for the two sets of data is determined and from this information temperature and emissivity of the surface are computed.

The communication module interfaces the outside world with the system. It accepts commands through the key matrix, displays the computed results and other messages on the printer and the LCD and transmits data to external devices. These are mutually independent modules.

Other system functions include an interrupt handler, programming capability and system initialization. The interrupt handler is used for the system clock updates. The programming capability[5] enables the system to remember a sequence of events entered through the key matrix and execute them at a later time. System initialization is required to set up proper initial conditions and parameter values.

D. System Implementation

The modules developed during the system design process may now be implemented directly. The hardware and software aspects of the implementation are discussed separately in the following sections.

1. Hardware

The target microprocessor system is centered around a CDP1802 microprocessor and is instructed with commercially available circuit boards of the EiComp family. The 8-bit microprocessor has 64 kilobyte memory addressing capability, seven Input/Output ports, two Direct Memory Access lines, four external flag inputs and a single level interrupt.

Data collection is done through an analog-to-digital converter with twelve bit resolution. Input channels 0 and 1 are used for pyrometer detectors A and B respectively. A 10 volt input gives the maximum digital output$(FFF)_{16}$ from the analog-to-digital converter.

A stepper motor controls the emissivity compensation, changing the value in steps of 0.005.

Input to the system is through a 4x4 key matrix. A 20-column printer and a liquid crystal display are used for the output of results and system messages. Data transfer to an external device is accomplished in parallel mode through I/O ports and in serial mode using an RS232C interface.

The system and application software resides in erasable programmable read-only memory in address space 0000 to $(3FFF)_{16}$. The random access memory is used to program the system through the key matrix and also as a data buffer.

2. Software

Details of the most important software modules necessary for system operation are given below.

The emissivity control (EC) module basically controls a stepper motor. Let CP be the current position and NP be the new position of the stepper motor. Pseudo-code for EC is given in algorithm-1.

ALGORITHM-1

```
EC: Procedure;
BEGIN EC
 INPUT NP from key matrix;
 IF NP > 200 THEN NP ← NP-200;
 STP ← | NP - CP |
 IF  NP < CP  THEN
     step back STP steps
 ELSE
     step forward STP steps
 ENDIF
 Display CP and emissivity
END  EC.
```

The algorithm for the computation and display modules is given as algorithm-2. It has four sections. The first section collects the data. The second section determines the crossing point and computes temperature. The third section determines crossing point five more times and computes temperature each time. Then the fourth section computes the average values and displays the result. The computation and display process is done once for single measurement, is repeated a specified number of times as entered through the key matrix for multiple measurements, and is repeated until system restart for measurements in the continuous mode.

ALGORITHM-2

I. DATA ACQUISTION

DOWHILE N $\leq$ 200

Select channel # on ADC

IF channel # = 1

THEN store digitized voltage in the array for pyrometer A.

ELSE IF channel # =2

THEN store digitized voltage in the array for pyrometer B.

ELSE

ENDIF

ENDIF

Step the motor by one step

ENDOWHILE.

II. DATA REDUCTION

DOWHILE N $\leq$ 200

Compute temperature (T_A) for pyrometer A

Compute temperature (T_B) for pyrometer B.

Store difference ($\Delta = T_A - T_B$) in an array.

ENDOWHILE

Search the Δ array for the minimum

Obtain the corresponding T_A and T_B and the array index (I).

Compute the emissivity.

III. MORE VALUES AT THE CROSSING POINT

Step the motor I steps

DOWHILE N$\leq$5

IF crossing point shifted

THEN step to the crossing point

ELSE

ENDIF

Obtain digitized voltages

Compute temperatures and emissivity

ENDDOWHILE.

IV. OUTPUT RESULTS

Display average temperature on LCD.

Print each temperature and their mean.

Print each emissivity and their mean.

Another important software module is initialization. It is the job of this module to initialize the program varibles, to select the proper mode of system operation and to initialize the input/output ports. It is done only at the time of power ON for the system.

The program module is especially useful when it is required to repeat a sequence of operations several times. The algorithm is given as algorithm-3. With this feature the system has two modes of operation; PROGRAM mode and RUN mode. In the RUN mode the command is executed soon after it is received, while in the program mode the commands are saved in memory.

ALGORITHM-3

PROG: Procedure;

BEGIN PROG

IF in program mode THEN

IF keystroke count < 100 THEN

IF pressed key is not exit key THEN

Save decoded key stroke in program buffer

Update keystroke count

ELSE

Exit from program mode

ENDIF

ELSE

Exit from program mode

Print error message

```
ENDIF
ELSE
  Execute command from key matrix
  ENDIF
  END PROG.
```

The system monitors the key matrix all the time except when it is commanded to execute a specific task, as shown in algorithm-4.

ALGORITHM - 4

```
DRIVE:
BEGIN
DOWHILE the system power is on
  Monitor the key matrix
  IF a key is pressed THEN
    Execute the appropriate task
  ENDIF
 ENDDO
 END.
```

Routines which are not time critical have been implemented in the microprocessor version of FORTH and the time critical segments are implemented in CDP-1802 assembly language. Since microFORTH supports only structured constructs and also because of the modularity of FORTH definitions, no special effort was required to implement structured programming constructs.

The hardware and software development was done side-by-side. Hence each component module was tested at different stages of the development process. Stubs were used to test out software modules in the event of delay in hardware development.

E. Conclusion

This system was successfully tested and used for temperature measurements at the Mississippi State University magnetohydrodynamic test facility several times. It was also taken to other facilities for temperature measurements. Modularity of system components has been found very helpful, especially in system modification and maintenance. Initial detailed analysis of the system and precise problem specification are essential for the efficient design and implementation of the system.

F. Acknowledgements

This work was performed under Department of Energy Contract No. DE-AC02-80ET-15601.

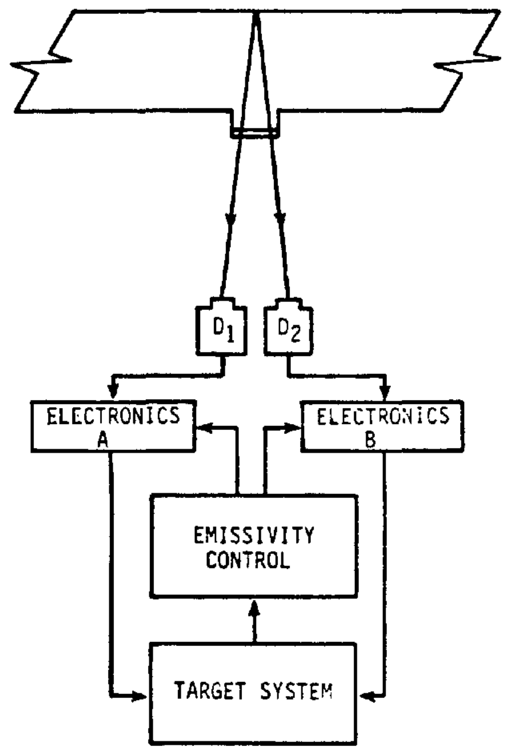

Figure 1. Initial Conception of the TCP System

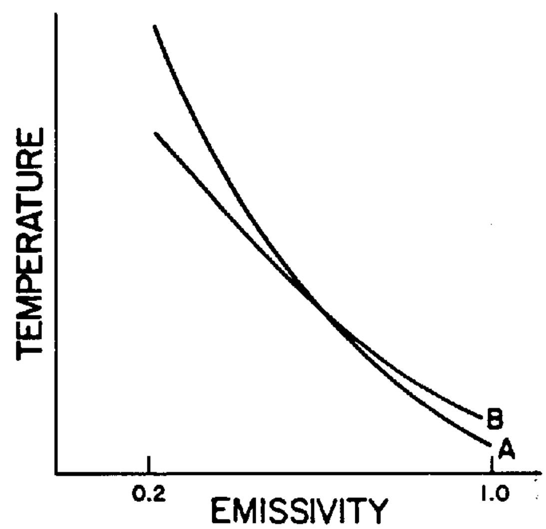

Figure 2. Relation Between Emissivity and Temperature for the Two Pyrometers

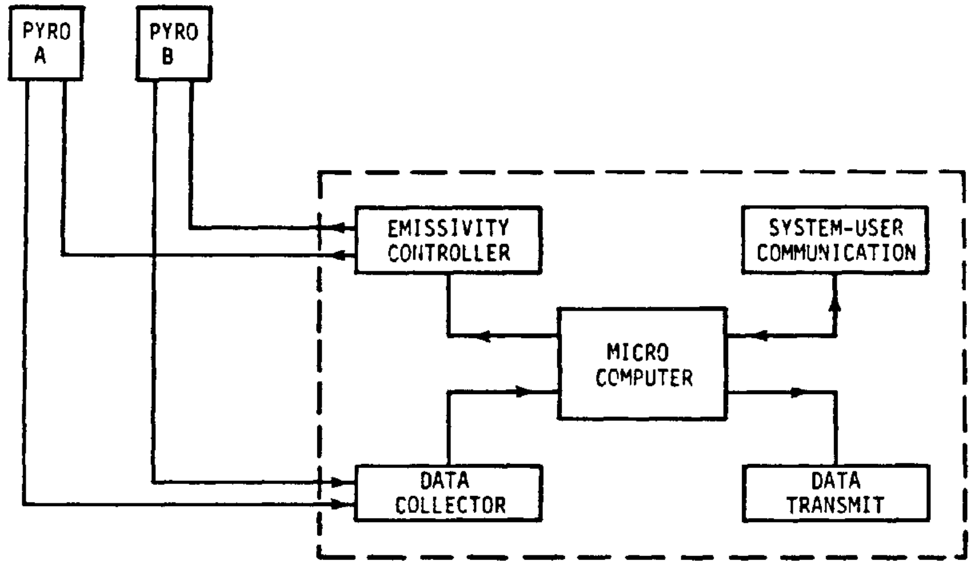

Figure 3. Basic Modules of the System Resulting from Refinement

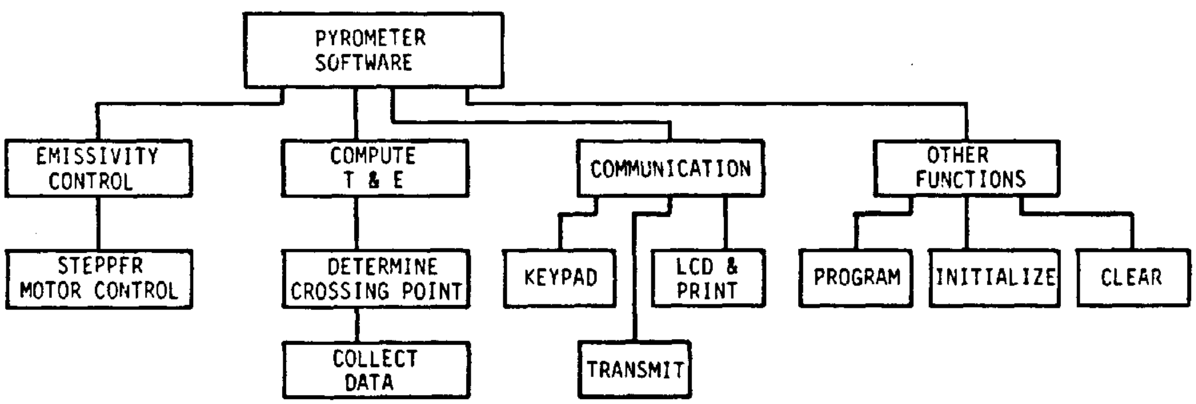

Figure 4. Software Modules Resulting from Design Phase

References

1. MSU MHD Final Technical Report FE-2246-15, p. 131.

2. Bauman, L. E., MSU MHD Quarterly Technical Progress Report FE-15601-1, p. 40.

3. Constantine, L., Yourdon, E., 'Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design', Prentice Hall Inc., 1979.

4. Benton, R. D., Philip, T., Poole, T.P., Jeglinski, S., MSU MHD Quarterly Technical Progress Report FE-15601-2, p. 66.

5. Philip, T., MSU MHD Quarterly Technical Progress Report FE-15601-4, p. 172.