CONF-770410--2

LA-UR -76-2669

TITLE: A MICROPROCESSOR CONTROLLER FOR STEPPING MOTORS

AUTHOR(S): B. G. Strait and M. E. Thuot

SUBMITTED TO: I.E.E.E. Microcomputer '77 Conference
Oklahoma City, OK
April 6-8, 1977

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

UNITED STATES
ENERGY RESEARCH AND
DEVELOPMENT ADMINISTRATION
CONTRACT W-7405-ENG. 36

MASTER

A MICROPROCESSOR CONTROLLER FOR STEPPING MOTORS*

by

B. G. Strait and M. E. Thuot

Los Alamos Scientific Laboratory
Los Alamos, NM 87545

ABSTRACT

A new concept for digital computer control of multiple stepping motors which operate in a severe electromagnetic pulse environment is presented. The motors position mirrors in the beam-alignment system of a 100-kJ CO_2 laser. An asynchronous communications channel of a computer is used to send coded messages, containing the motor address and stepping-command information, to the stepping-motor controller in a bit serial format over a fiber-optics communications link. The addressed controller responds by transmitting to the computer its address and other motor information, thus confirming the received message.

Each controller is capable of controlling three stepping motors. The controller contains the fiber-optics interface, a microprocessor, and the stepping-motor driver circuits. The microprocessor program, which resides in an EPROM, decodes the received messages, transmits responses, performs the stepping-motor sequence logic, maintains motor-position information, and monitors the motor's reference switch. For multiple stepping-motor application, the controllers are connected in a daisy chain providing control of many motors from one asynchronous communications channel of the computer.

I. INTRODUCTION

A High Energy Gas Laser Facility (HEGLF) is being developed at the Los Alamos Scientific Laboratory for laser fusion research. The CO_2 laser will deliver a 100-kJ pulse of light, of 1-ns duration, to a target $\sim$ 500 μm in diameter. The light energy is contained in 72 beams, which must be aimed and focused precisely at the small target with an automatic beam-alignment system.

---

*Work done under the auspices of the U.S. Energy Research and Development Administration.

Pulsed lasers, like HEGLF, characteristically create an extremely strong electromagnetic pulse (EMP) environment when fired, caused by the discharge of the high-voltage pulse power supplies into the laser gas medium. The discharge reaches 6 MA at 550 kV for 2.5 μs. This paper describes the microprocessor-based stepping-motor controllers that have been developed to operate in this severe EMP environment.

II. THE STEPPING-MOTOR SYSTEM

To combat the EMP, the motor-controller electronics are contained in shielded boxes and located near the mirror mounts, which reduces the length of the motor wiring. In addition, the boxes are connected to the central control system by fiber optics, which are immune to the effects of the EMP. The use of fiber optics requires a serial communications system between the central control computer and the motor controllers. A diagram of a system that employs the above techniques is shown in Fig. 1.

The system includes motor controllers connected in a daisy chain with fiber cables, a communications microcomputer, a local control console, a portable hand-held controller, and a high-speed serial communications system. Up to 64 motor-controller boxes can be connected on the daisy chain, with each box controlling 3 stepping motors.

Motor commands originate from three sources: From the control room during automatic mirror alignment, from the local control console, and from the hand-held controller. The last two sources are used locally during setup of the optical system. Motor-position information is stored in the motor-controller's memory. The information is retrieved and displayed as soon as a box is addressed and becomes available to any of the three motor-control sources.

III. THE MOTOR-CONTROLLER BOX

The motor controller receives light signals from the fiber optics; the addressed box interprets the signals and moves the stepping motors accordingly. Immediately after receipt of a complete command the addressed box returns a reply confirming the correctness of the command. The commands reach each box almost simultaneously, because the signals are passed on from one box to another along the daisy chain during the transmission.

A block diagram of the motor controller is shown in Fig. 2. It consists of three major parts: the microprocessor, the fiber-optics interface, and the motor driver circuits. The M6800 microprocessor performs the logic function of the motor controller. It controls the asynchronous communication interface adapter (ACIA) and the peripheral interface adapter (PIA) via the processor bus. Two memory devices are used, a 127 x 8-bit RAM and a 1024 x 8-bit EPROM (2708).

The MC6850 ACIA provides data formatting, error checking, serial-parallel conversion, and parallel-serial conversion between the microprocessor and the serial communications link. The frequency of the input clock to the ACIA, 614.4 kHz, is derived from the microprocessor clock. The ACIA further divides the clock by 16, resulting in a bit rate of 38.4 kHz for the serial transmission rate.

The stepping motors are controlled and their reference switches are monitored through the PIA. Bifilar-wound stepping motors are used in the mirror mounts; the motor stepping sequence is shown below.

TABLE I

MOTOR STEPPING SEQUENCE

STEP	1	3	2	4
1	ON	OFF	ON	OFF
2	ON	OFF	OFF	ON
3	OFF	ON	OFF	ON
4	OFF	ON	ON	OFF
1	ON	OFF	ON	OFF

This sequence, generated in the firmware, is transmitted to the motor drive transistors through the PIA as shown in Fig. 3 for one motor. To reduce the heating of the mirror mounts, the motors are energized only during stepping operations. The motor in Fig. 3 is turned on and off through the PIA output (PAO) by the four AND gates.

The reference switch in Fig. 3 is a precision switch that is actuated only when the mirror is at its center of travel. The switch establishes the reference point for the motor-position counters. The mirror mounts do not require limit switches because software limits are used to restrict mirror travel during operation and to avoid loss of position information.

The fiber-optics interface shown in Fig. 4 converts the light signals to the correct electrical level for the ACIA and vice versa. The transmitter consists of an infrared-light-emitting diode, LED, driven by a standard peripheral driver interface circuit. The LED is preferred over laser diodes because of the device's high reliability and simple drive requirements. This driver circuit can switch several hundred milliamperes to drive the LED, at an output level of $\sim$ 6 mW.

The fiber-optic cable is a bundle of $\sim$ 200 step-index fibers. These numerous fibers provide high redundancy of the light path as well as considerable tensile strength to the cable; an outer sheath provides crush-resistance. The cable has a bundle diameter of 1.1 mm, a numerical aperture of 0.66, and an optical attenuation · 350 dB/km. This size results in a noncritical alignment requirement for the fiber bundle to the transmitter and receiver. Easy-to-apply, low-cost commercial thermoplastic connectors are used to connect the cable to the transmitter and receiver, or to splice a cable.

The receiver is a PIN diode-amplifier integrated circuit coupled to a high-speed comparator. This circuit can produce a useful output with an input of less than one tenth of a microwatt; however, it is usually operated with an input of a few microwatts. The diode-amplifier is wired in a single-ended transimpedance connection to provide high responsivity, high speed, and dynamically stable operation. The dc offsets and drifts generated by the back-biased diode are eliminated by the capacitive coupling to the comparator. The comparator converts the low-voltage output of the photodiode amplifier to a high-level signal while rejecting any noise generated by the amplifier.

The motor controller circuits are packaged on double-sided printed-circuit cards. The fiber-optic interface card is shown in Fig. 5. Also shown in the upper part of the figure is an exploded view of the fiber-cable connector and the LED transmitter which fits into the plastic connector. The four printed-circuit cards that make up the motor-controller box are shown in Fig. 6. They are the microcomputer (top left), the motor-driver card (bottom left), the fiber-optics interface (top right), and the power supply regulator card.

The microcomputer card also includes a two-phase clock, a clock divider, an address decoder circuit, an address switch, and two 14-pin flat cable connectors that connect the PIA outputs to the motor-driver card. The address switches are used to establish the box address that can be read by the microprocessor.

The only input power requirement to the box is +12 Vdc, which is the motor power. Power for the microcomputer and the fiber-optics amplifier is derived from the motor power by using a +5-V regulator and a dc-dc converter to generate $\pm$ 12 Vdc.

A picture of the motor-controller box with one of the smaller mirror mounts is shown in Fig. 7.

IV. THE HAND-HELD CONTROLLER

The hand-held controller shown in Fig. 8 provides the means of positioning the mirrors from any convenient viewing location; it can be inserted into the daisy chain at any point. Motor-position information for three motors, as well as the address of a motor-controller box, is displayed by the hand-held controller. Motor commands are generated by activating one of several combinations of 25 keys, 7 of which have dual functions.

Each of the three motors operated by a given motor-controller box can be positioned in one-step increments, can be moved any number of steps, can be slewed, and can be commanded to a preset position. Software limits that restrict the motor travel are keyed-in through the hand-held controller.

Some of the commands available through the hand-held controller and how the commands are keyed in are illustrated in Table II, below.

TABLE II

COMMAND CAPABILITY OF
HAND-HELD CONTROLLER

FUNCTION	KEY STROKES
Clear	CLR
Select a Box	1,2, ADR
Slew Motor A in + Direction	A, slew, +, GO
Stop Motor A	STOP
Set Motor B to +4963	B, Pos, +, 4, 9, 6, 3, GO
Move Motor C 250 Steps in - Direction	C, ΔP, -, 2, 5, 0, GO
Set all Motors to +2000	All, Pos, +, 2, 0, 0, 0, GO

V. MOTOR CONTROLLER FIRMWARE

There are three major functions that must be performed by the software: The generation of the motor energization sequence, the generation of the delays between steps, and the operation of the communications network. Each of these tasks has real-time constraints. Because this microprocessor does not support concurrent processing or multiprogramming we have elected to perform the tasks in a loop. The main loop supports the independent operation of three stepping motors. This loop is interrupted by the real-time clock and the communication-network demands.

Operation of a single stepping motor is controlled by energizing its windings according to Table I. From Fig. 3, it can be seen that the Table-I sequence reduces, at the PIA outputs, to the four-step output sequence of Table III.

TABLE III

FOUR-STEP OUTPUT SEQUENCE

STEP	PIA	OUTPUT
1	0	0
2	0	1
3	1	1
4	1	0

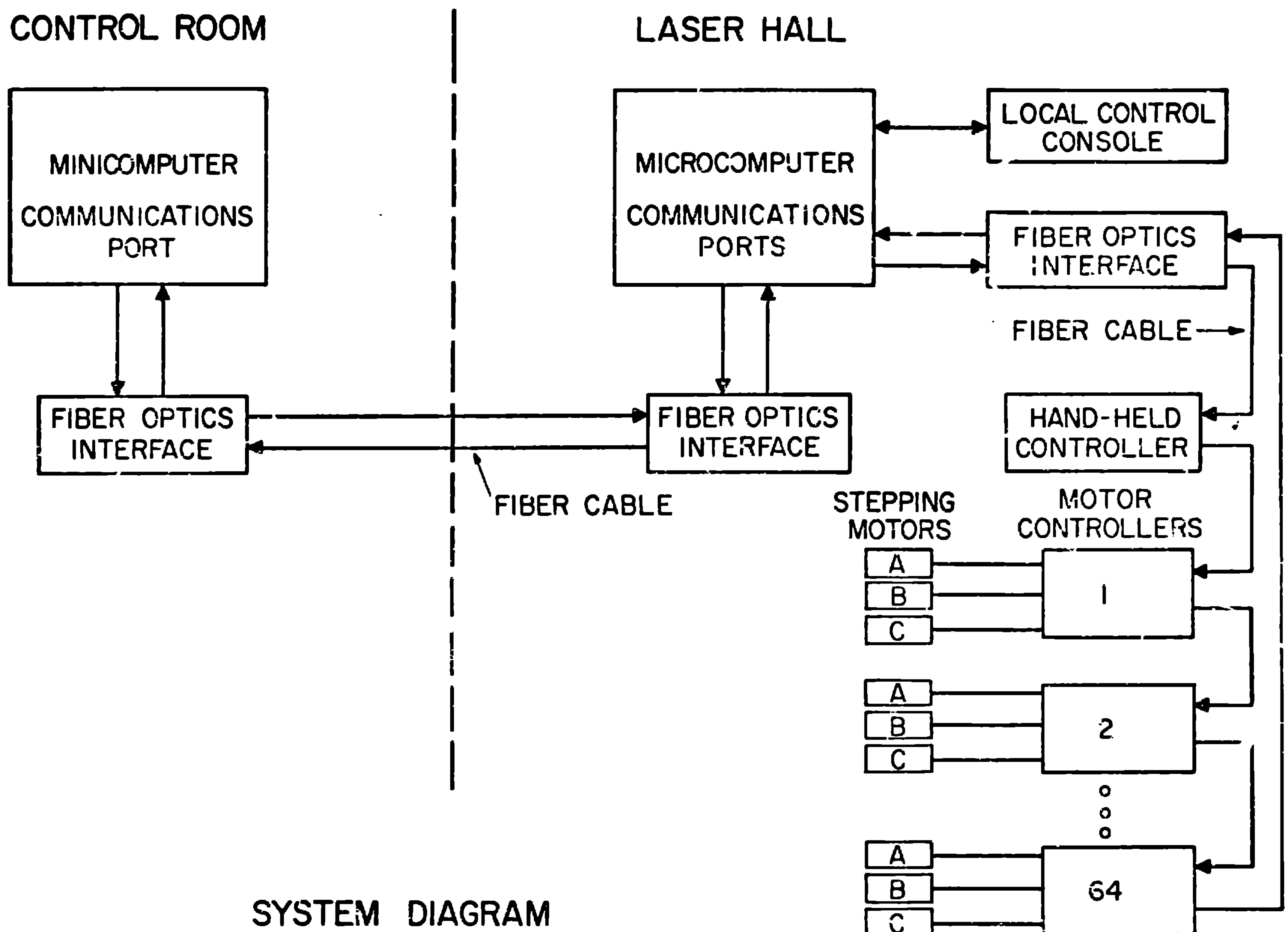
To operate the motor clockwise, we advance through Table III from top to bottom; for counterclockwise rotation, from bottom to top. Table III is stored in read-only memory. A pointer for each motor, which can be incremented or decremented within the bounds of Table III is defined to point at a two-bit combination. This combination is transmitted on the correct PIA lines to the motor associated with the pointer, causing that motor to step. By counting the number of times a pointer is moved we can keep track of the number of steps taken. We also define a "steps left" counter associated with each motor.

The contents of the counter equal the number of steps left to be taken. Each time a motor steps, its counter is decremented and its Table-III pointer is moved. A loop, programmed to operate three motors, will cause the counter to be decremented, the pointer to move, and the motor to step until no steps are left to be taken. At this point the motor can be de-energized. However, because the pointer always points to the last step taken, the motor can be reenergized between steps without taking an extra step.

Two additions must be made to the program loop defined above: First, some time must elapse between steps to allow the motors to complete the step, and second, the "steps left" counters must be loaded with the number of steps to be taken. Proper timing is provided by a real-time clock. The clock frequency is chosen so that about 70 microcomputer instructions can be executed between clock interrupts. The clock service routine is kept short and cannot be interrupted. Further, the clock frequency is about 80 times faster than the operating frequency of the motor. These limits make it possible to choose the real-time clock frequency that will provide a variable delay between steps. The stepping-motors can thus be operated at variable speeds so that large inertial loads can be accelerated to a high speed and slowed to a stop without missing a step.

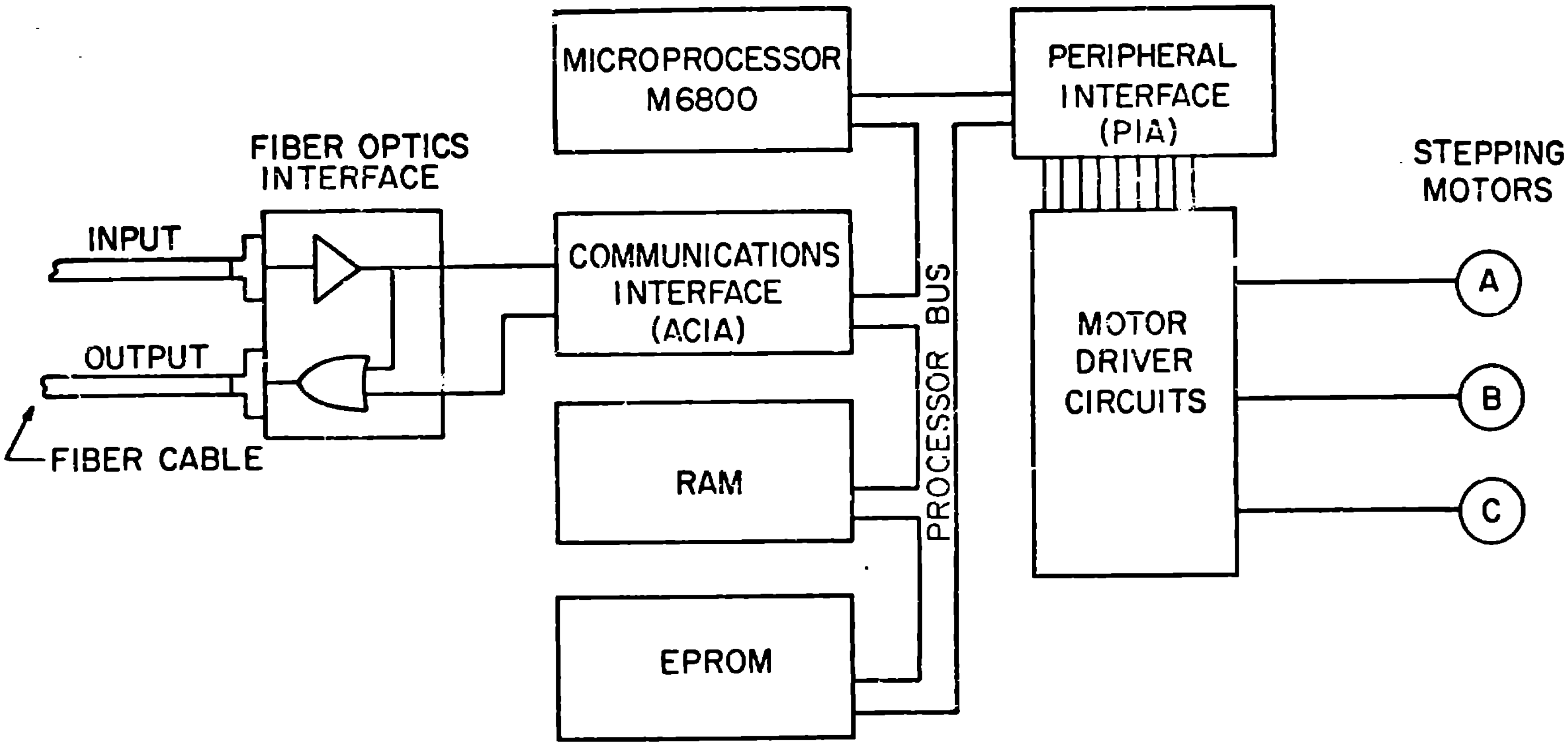
Ramping is achieved by defining in the software another decrementing counter for each motor. These "delay" counters contain the number of real-time clock "ticks" that must elapse before a step is taken. The counters are initialized with some number, e.g., 80, and each time the clock interrupts, the number is decremented. When the number reaches zero, the program moves the Table-III pointer and the motor takes a step. By decreasing the initial number in the "delay" counter, we can speed up the motor because fewer clock ticks are required between steps. If many steps are to be taken by a motor, the delay counter is loaded with a smaller number each time a step is taken. This shortens the time between steps and the motor will speed up until an upper speed limit is reached. When the number of steps left equals the ramp length, the software initializes the delay counter with larger numbers so that the motor will slow down without missing a step. By this method, the three motors can be ramped or run at different speeds at the same time.

After the delay function is provided, only one other major function remains to be added to the program, that is, a provision to load a number into the "steps-left" counter for each motor. This information comes from the communication network in the form of the commands listed in Table II. A command is defined as a string of ASCII characters, beginning with a "DCI." This is followed by the controller address, the command character, four optional information characters, and a check sum. The command characters are entered through the asynchronous data port. When a character is received, an interrupt occurs and the service routine checks to see if it is the beginning of a command. If it is the beginning and the address is correct, the characters are stored until the whole command has been received. The command is then interpreted, and the number of steps a motor has been commanded to take is calculated. This number is added to the number in the motor's "steps-left" counter. If the motor was stopped, the motor is energized and begins to move or to ramp, decrementing the "steps-left" counter. If the motor is already in motion, the number will be added to the counter and the motor will ramp if necessary. If the motor is in motion and the commanded steps would reverse the motor or try to stop it abruptly, the command is deferred and the motor is immediately ramped to a stop and reversed to obtain the correct number of steps.



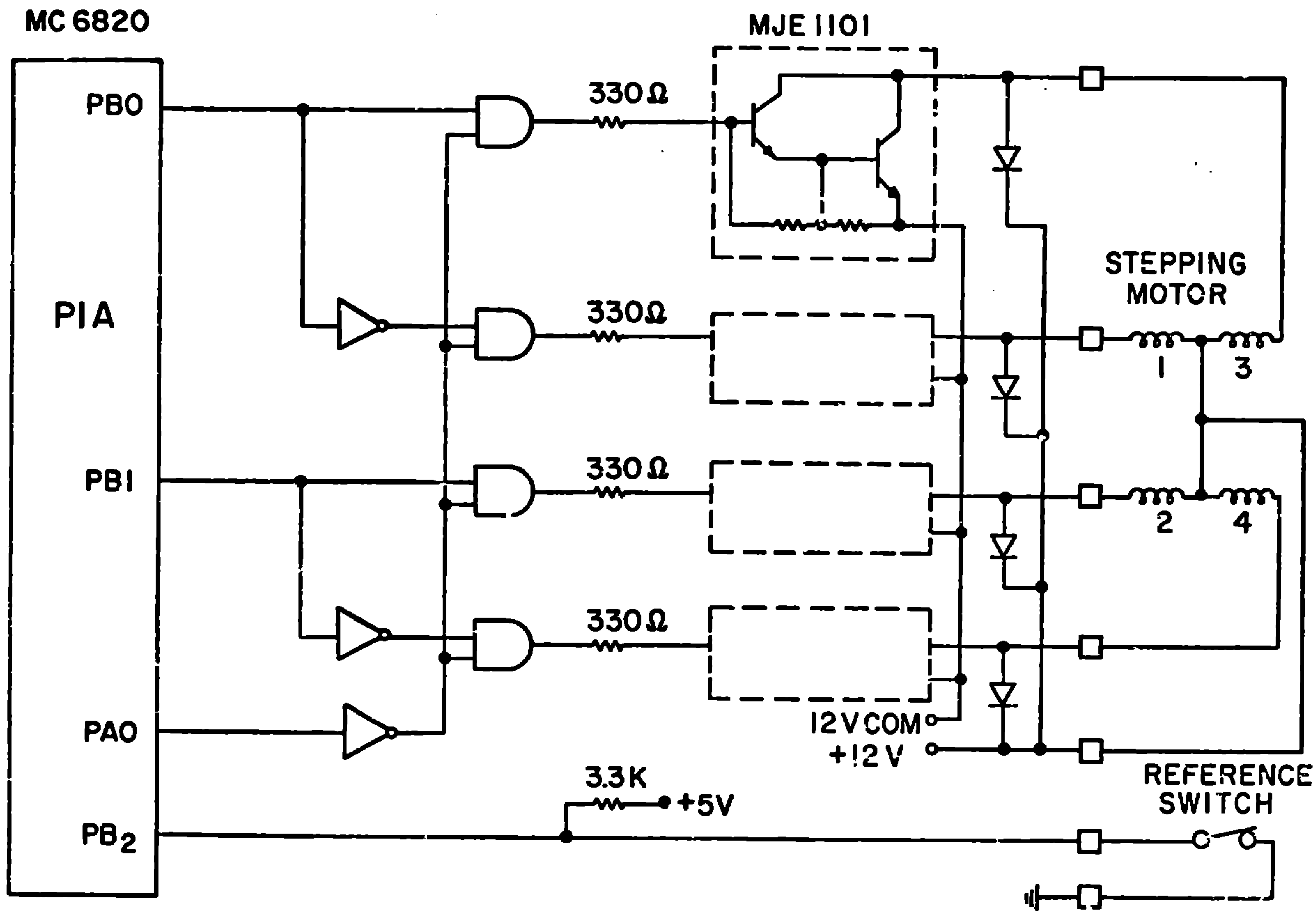
SYSTEM DIAGRAM
STEPPING MOTOR CONTROLLER

FIGURE 1

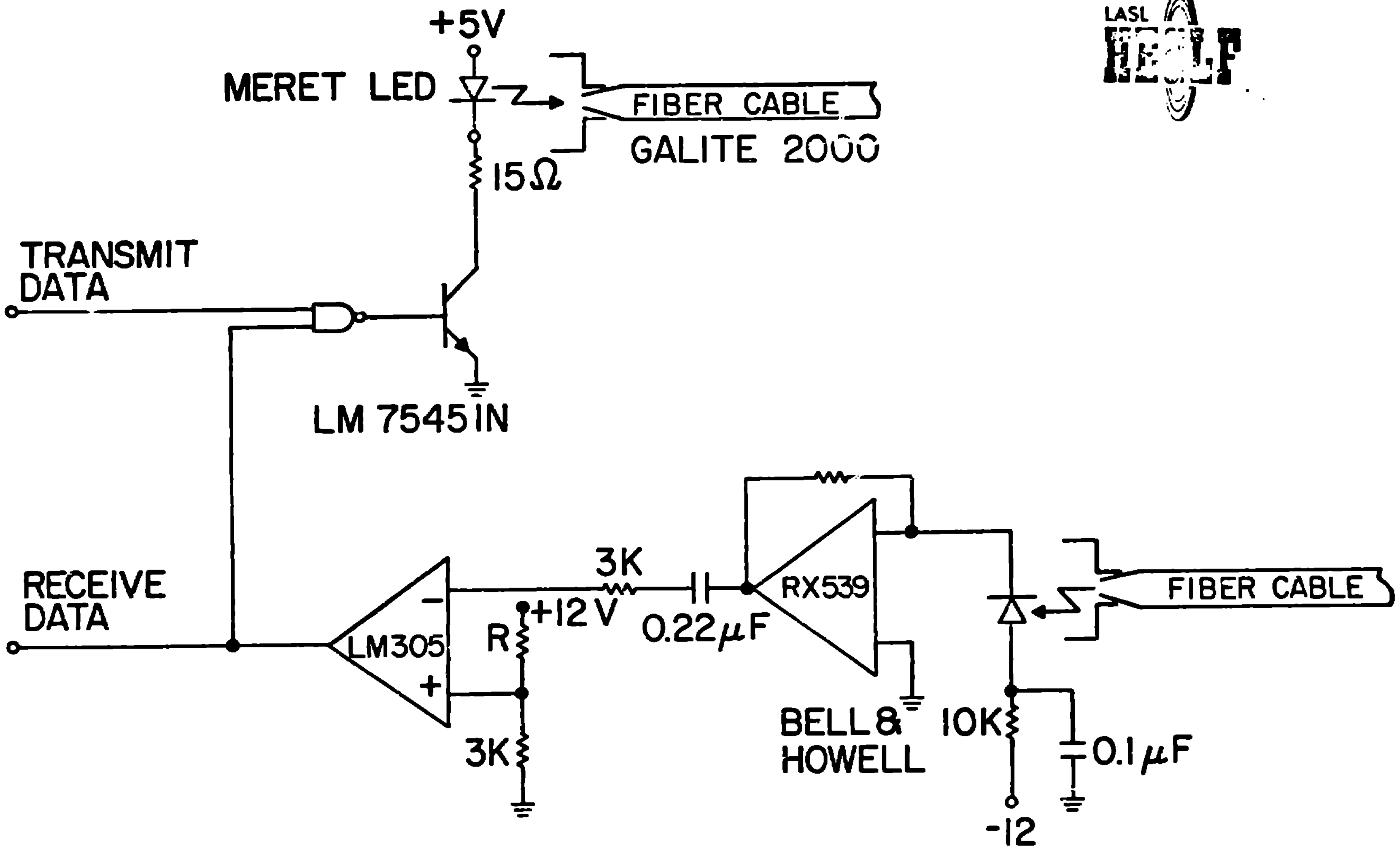


MOTOR CONTROLLER BLOCK DIAGRAM

FIGURE 2

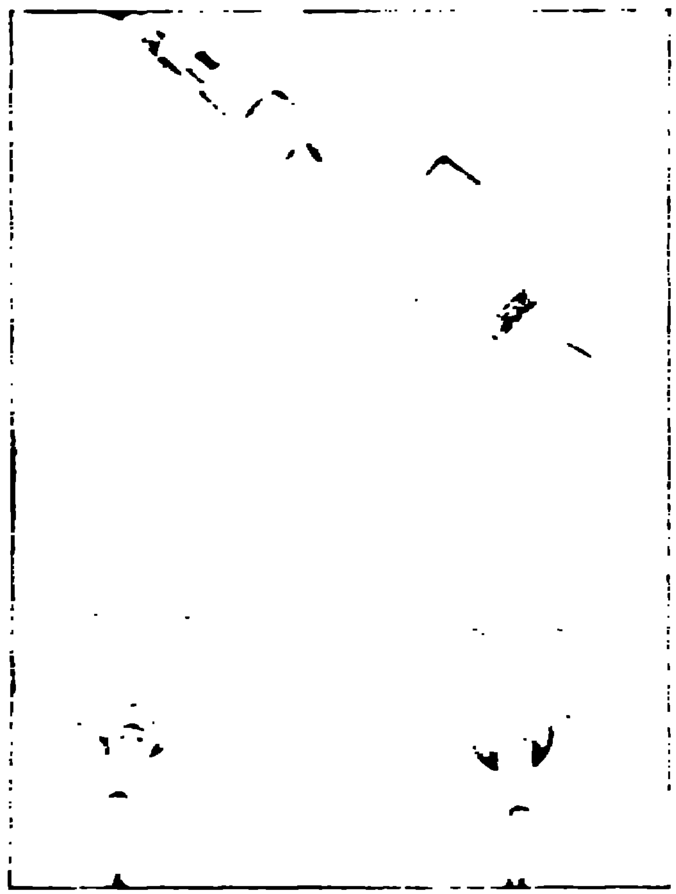


STEPPING MOTOR DRIVER SCHEMATIC
FIGURE 3



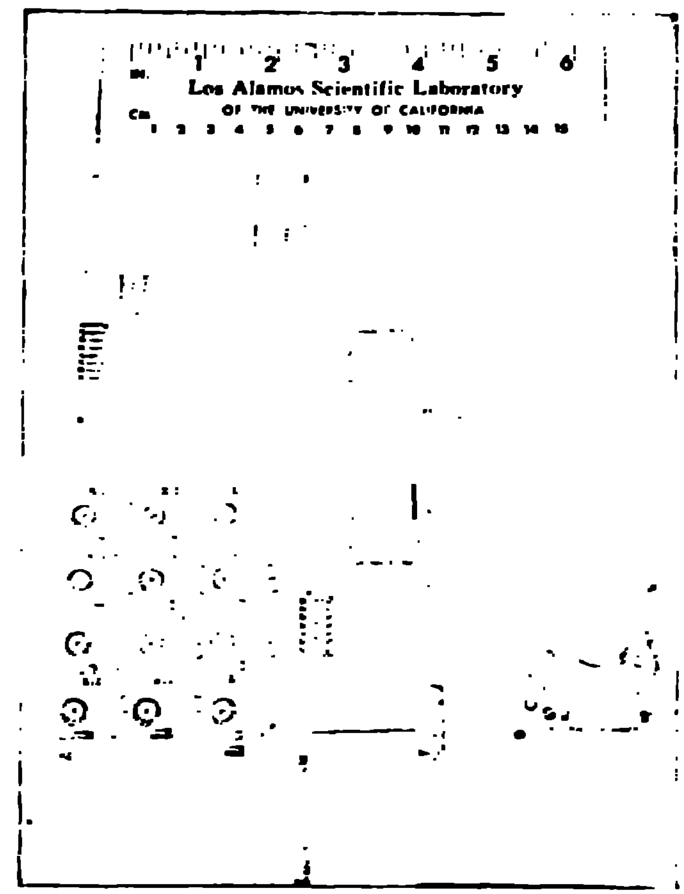
FIBER OPTICS INTERFACE SCHEMATIC

FIGURE 4



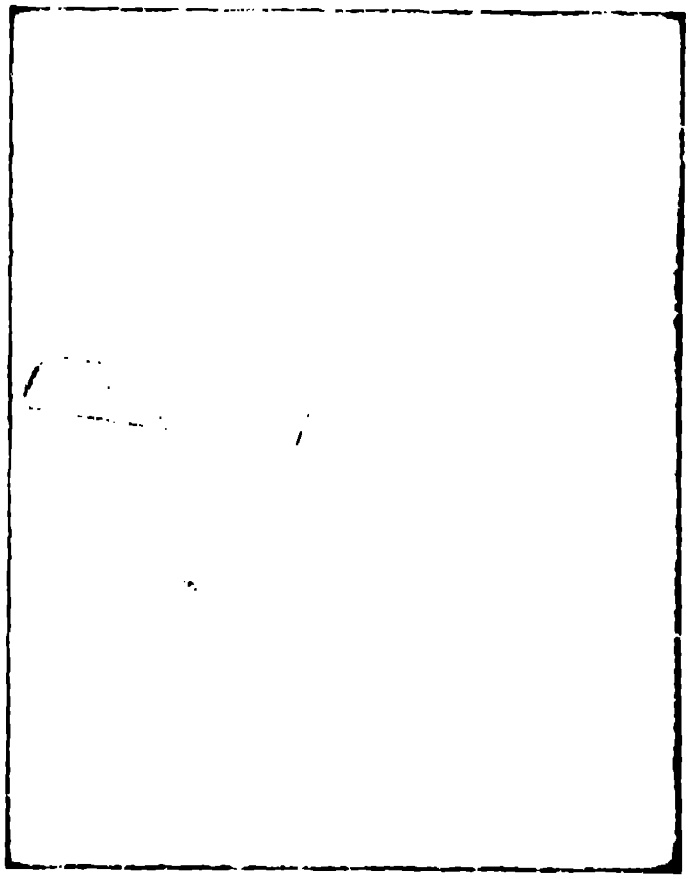
FIBER OPTIC INTERFACE

FIGURE 5



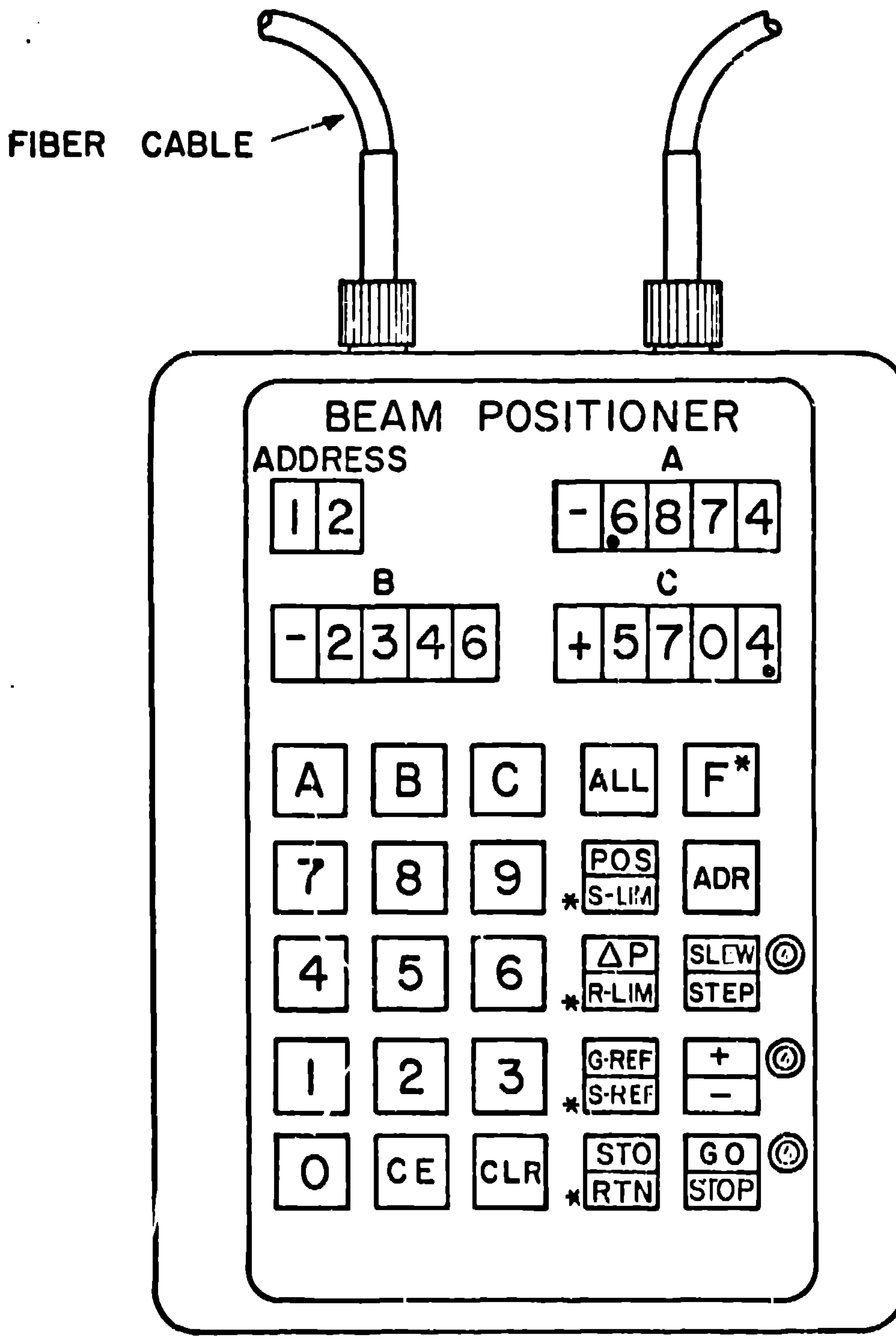
MOTOR CONTROLLER
PRINTED CIRCUIT CARDS

FIGURE 6



MOTOR CONTROLLER BOX
WITH A MIRROR MOUNT

FIGURE 7



HAND-HELD CONTROLLER

FIGURE 8