TECHNICAL MEMORANDUM NASA 55

MICROPROCESSOR-TO-SYSTEM/370 INTERFACE

A hardware interface is described which allows direct memory load of a microprocessor from the host System/370 computer, eliminating paper tape handling.

by

Robert W. Lilley

Avionics Engineering Center
Department of Electrical Engineering
Ohio University
Athens, Ohio

February, 1978

Supported by

NASA GRANT NGR 36-009-017

Langley Research Center
Langley Field, Virginia

(NASA-CR-156066)
MICROPBOCESSCF-TO-SYSTEM/370 INTEBFACE (Ohic
Univ.) 9 p HC A02/MF A01 CSCL 09E

N78-20782

Unclas
G3/60 11872

I. INTRODUCTION

This paper documents the design and operation of a microprocessor interface unit which allows use of a computer terminal for communication at 110 or 300 baud both with a central host computer and with the microprocessor monitor. Additionally, the interface permits the host computer to load the microprocessor memory directly with object code, avoiding the use of intermediate data storage such as paper tape.

The central computer, containing an assembler language processor for the target microcomputer, can be used from the terminal with all the flexibility offered by the virtual machine facility, producing object code for the micro plus program listings and supporting outputs. The object code can then be loaded directly to the micro and the same terminal device used to run the micro program, communicating with the micro's monitor routine.

This interface has been used successfully to interconnect the IBM System/370 Model 158 through an ITT Asciscope (TM) terminal to the MAI SuperJOLT (TM) microprocessor.

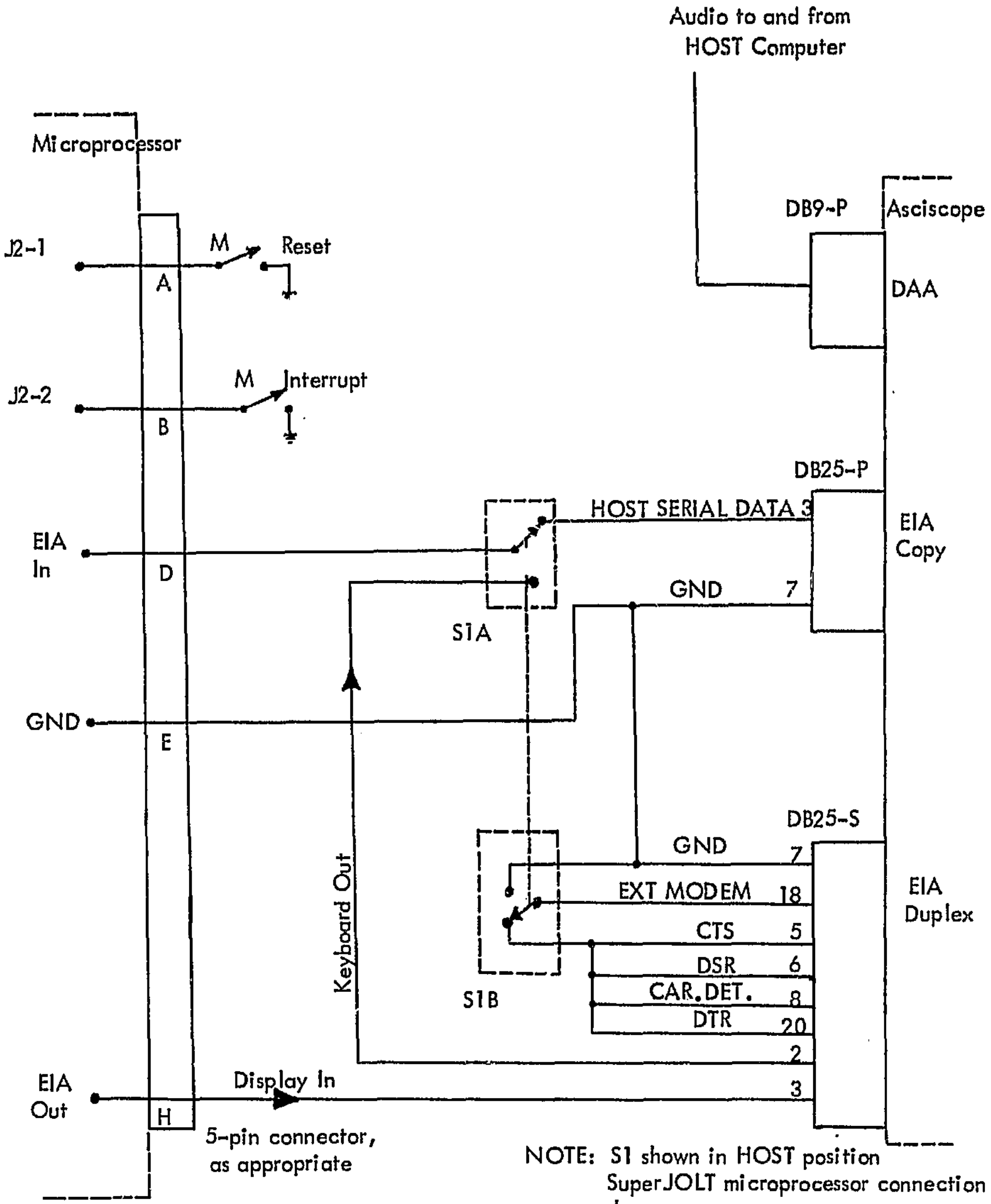
II. INTERFACE DESCRIPTION

Figure 1 shows the interface schematic. Operation depends upon the internal characteristics of the Asciscope terminal, which has full EIA RS-232 and Data Access Arrangement circuitry built-in. The interface pins used and their functions are described in Table 1.

As shown in Figure 1, when the interface is in the HOST mode, the Asciscope Internal Modem is in control since Pin 18, EIA DUPLEX is high by being connected to Pin 20. The Asciscope can communicate with the host computer via DAA. Data from keyboard to the host computer does not appear at Pin 3 of EIA copy, but data returned from the host is presented at Pin 3, which is connected to the microcomputer.

Switching the interface to MICRO mode forces the Asciscope to external modem mode, by grounding EIA DUPLEX Pin 18. Clear-to-Send (Pin 5), Carrier Detect (Pin 8) and Data Set Ready (Pin 6) are held true by connection to Data Terminal Ready (Pin 20). EIA keyboard output is passed to the micro at EIA DUPLEX Pin 3, and micro output is returned to the Asciscope via Pin 2, which is activated by the switch to external modem.

The host computer is disconnected from all data paths by the switch to external modem, and audio carrier is cut off to the host.



NOTE: S1 shown in HOST position
SuperJOLT microprocessor connection shown.

Figure 1. Interface Schematic.

ORIGINAL PAGE IS
OF POOR QUALITY

DAA Connector

Pin 2 Pin 4	} Direct-Connect Audio Line Pair to Host Computer

EIA DUPLEX Connector

Pin 2 -	Transmitted Serial Data (Keyboard Output)
Pin 3 -	Received Serial Data (Display Input)
Pin 7 -	Signal Ground
Pin 5 -	Clear to Send
Pin 6 -	Data Set Ready
Pin 8 -	Received Line Signal Detector (Carrier Detect)
Pin 20 -	Data Terminal Ready
Pin 19 -	Low Rate Selector for External Modem

EIA COPY Connector

Pin 3 -	Serial Data as Received from Host Computer
Pin 7 -	Signal Ground

Table I. Asciscope Connections

III. INTERFACE OPERATION

First, the computer terminal is logged on the central computer with the interface switch in the HOST position, and the central computer file edit system is used to prepare a source assembler program for the microprocessor. When this program is assembled, a disk file is produced which containsthe object code (hexcode) for the micro.

Next, the interface is switched to the MICRO position and the microcomputer is reset and commanded to load hexadecimal code via its software monitor. The interface is switched back to HOST computer and that computer is commanded through the terminal to send the object code file to the terminal. This output is also sent to the micro where it is loaded into memory by the previously-given command to load hexadecimal.

Finally, the interface is switched back to MICRO and the terminal is used to control the micro as the program is executed.

The Asciscope must be switched to FDX (full-duplex) or HDX (half-duplex) as appropriate to the external systems used.

A specific example of interface operation using the Ohio University computer system, the Asciscope and the SuperJOLT microprocessor follows in the next sections of this paper.

IV. INTERFACE SUPPORT COMMANDS JCOPY and JLOAD

For operation of the interface unit, two commands have been written for the Ohio University Conversational Monitor System (CMS). These commands operate upon files produced by the assembler program for the MOS Technology 6502 MPU (see reference 1).

When the assembler is run, two important output files are produced. File JOLT CLEAN C contains a formatted version of the source program which the programmer may wish to save on permanent storage (the A-Disk). JOLT HEXCODE C contains the object (hexadecimal) code in the format required by MOS Technology KIM and TIM monitors (as used in the KIM-1 and JOLT micros). This file will later become the input file to be loaded to the micro via the hardware interface.

Command JCOPY provides file support for the micro user. Issuing the command JCOPY programname causes the JOLT HEXCODE C file to be copied to the A-Disk with identifier programname HEX A, and the JOLT CLEAN C file to be copied to the A-Disk with identifier programname SOURCE A. If files with these names already appear on the A-disk, the user is asked if the files should be replaced, or written over with the new C-Disk data. If his response is NO, no disk files are changed and the command terminates. If YES, the copy operation proceeds.

INTERFACE OPERATION EXAMPLE
OHIO UNIVERSITY COMPUTER SYSTEM

User Inputs	Description
Interface in CMS position Asciscope HDX	
LOGON machinename	
EDIT SAMPLE JOLT A	Establish file name
enter assembler statements for program	
FILE	Store file on CMS disk
JASM SAMPLE JOLT A	Run Assembler
JCOPY SAMPLE	Copy clean listing and hex code file to A-Disk as SAMPLE SOURCE A and SAMPLE HEX A
Switch interface to JOLT. RESET the JOLT, Asciscope to FDX (Carriage Return) LH	Initialize TIM monitor command hex loader
Switch interface to CMS, Asciscope to HDX	
[JLOAD SAMPLE	Bracket deletes any results of switching transients; JLOAD SAMPLE loads SAMPLE HEX A into JOLT memory.
At end, switch interface to JOLT, and use TIM monitor to operate the program, with Asciscope in FDX.	

The command JLOAD can take several forms. In general, JLOAD causes a hex file to be sent to the terminal, where the hardware interface makes the data available also to the micro for loading. The command is designed specifically for the KIM and TIM micro monitors, in that it protects these micros from extraneous semicolon (;) characters by translating them to percent signs (%). This avoids bad micro memory loads due to CMS semicolon outputs following the ready message. Semicolons in the hexadecimal memory load are unchanged.

Issuing JLOAD programname searches for the file programname HEX A and causes it to be sent to the terminal with CMS semicolon protection. Issuing JLOAD programname filetype filemode causes the (CMS) file identified as programname filetype filemode (any file, any disk) to be sent for micro loading.

When the file has been sent in its entirety, CMS sends its ready message (R%) and the user is ready to switch to the MICRO position of the interface and begin program execution in the microprocessor.

V. REFERENCE

[1] An Assembler for the MOS Technology 6502 Microprocessor as Implemented in JOLT (TM) and KIM-1 (TM), Robert W. Lilley, Avionics Engineering Center, Ohio University, November 1976.

VI. APPENDIX: EIA MICROCOMPUTER INTERFACE

The SuperJOLT micro has EIA input and output parts as part of its design. Figure A-1 shows an uncomplicated KIM-1-to-EIA interface to convert the KIM 20-milliamp current loop to EIA levels.

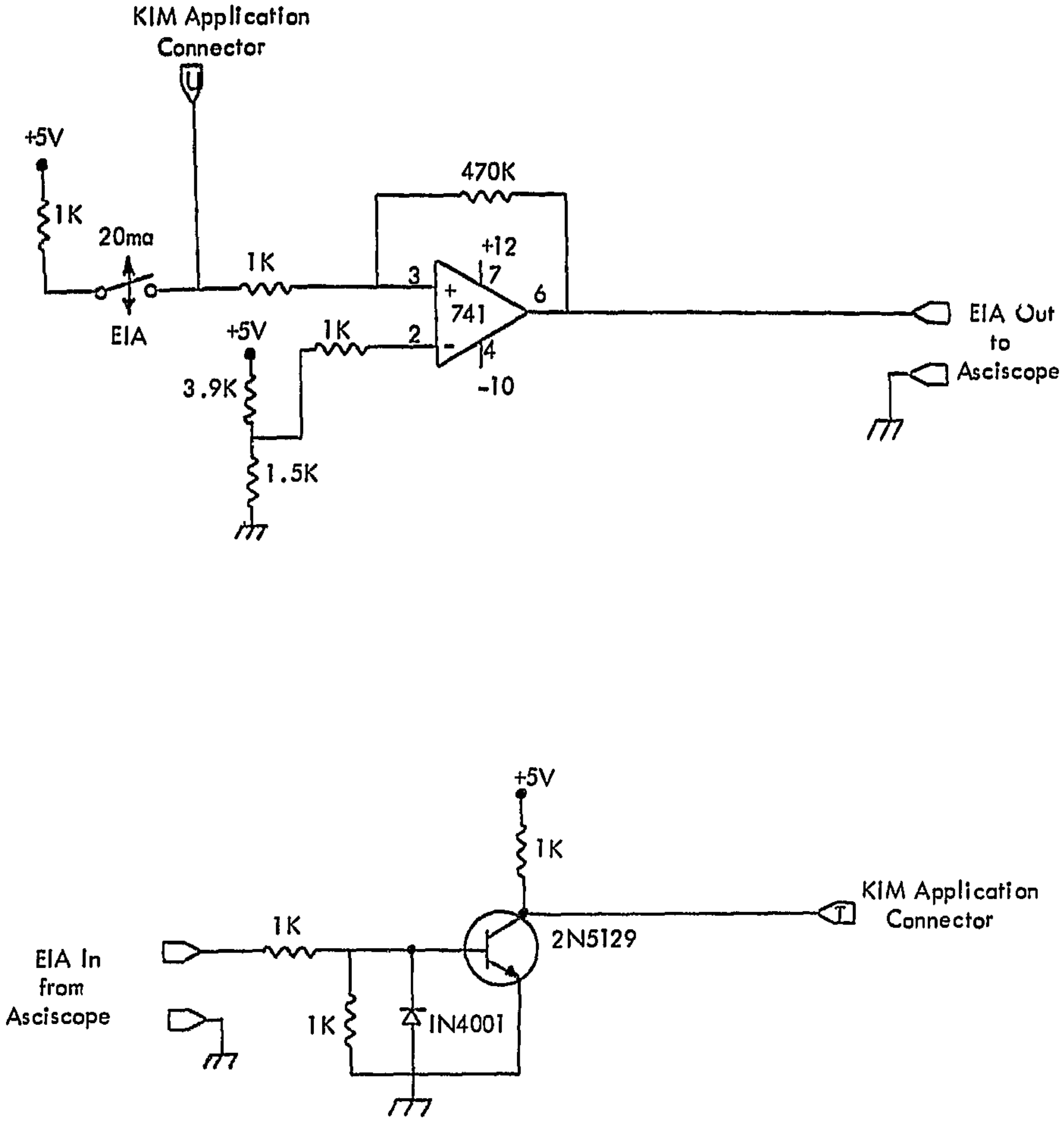


Figure A-1. KIM-1 EIA Interface.