General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

TEST PROGRAM FOR 4-K MEMORY CARD, JOLT MICROPROCESSOR

A memory test program is described for use with the JOLT microcomputer memory board used in development of the Ohio University Omega navigation receiver.

by

Robert W. Lilley
Avionics Engineering Center
Department of Electrical Engineering
Ohio University
Athens, Ohio 45701

August 1976

Supported by

I. INTRODUCTION

The JOLT (TM) microcomputer, utilizing MOS Technology 6502 microprocessor chip, is being evaluated for use in the Ohio University Omega receiver software-based prototype. The receiver, being developed under the NASA Joint University Program in Air Transportation Systems, will attempt to minimize hardware, utilizing the microprocessor for all possible functions.

The memory-test program described in this paper is one of a series of routines being produced for basic support of the microcomputer during receiver development stages.

The program allows a quick test of a 4,096-word memory board used with the basic JOLT processor by cycling the memory through all possible bit combinations in all words.

JOLT machine-language code is used in program preparation.

II. PROGRAM DESCRIPTION

The Memory Test Program is written in hand-assembled code for the MOS Technology MCS 6502 microcomputer chip as used in the JOLT microcomputer. The program stores and reads each possible bit configuration in each memory location from 1000 to 1FFF, comparing the input and output of memory for equality. The program runs 40 seconds for a complete check of 4,096 memory locations. Incorrect memory operation results in a printout of the memory output, the address, and the input in that order on the teletypewriter output port of the JOLT. This output occurs for each incorrect output of each affected memory location. For this reason, the program should not be run unattended, as a high volume of output can be produced for relatively few failing memory locations.

III. PROGRAM LISTING

The listing given on the following pages illustrates the complete program code. The program is loaded beginning at JOLT memory location 0000, and takes 4B (hexidecimal) bytes of storage.

The program operates by addressing the 4-K memory board in indexed, indirect mode. ADDL and ADDH are the low and high order address bytes, respectively, and are located at 004A. The Y index register contains the current ADDL value during program operation.

The X index register is used to store the CTR, or memory data value. After initialization, CTR is stored in the memory location addressed by ADDL, ADDH indexed by Y. The accumulator is then cleared, and the memory location is read back to the accumulator. The accumulator is compared with the CTR value to determine memory correct or incorrect operation. If correct, CTR is incremented to test the memory with the next one of 256 possible bit combinations (in an 8-bit word). If CTR is zero (having passed through all 256 combinations at the current memory address), Y is incremented (incrementing ADDL). If a page 44 boundary is crossed, (Y=0), then ADDH is incremented to begin indirect addressing on the following page. If ADDH passes the top of the 4K memory address space (above 1FFF), the word END is printed and the program is stopped. If either ADDH or ADDL remain in range of the 4K address space after incrementing, the program continues at the new memory address, with CTR=O.

PROGRAM LISTING

Program Counter	Program Code	Remarks
00	Ea	No-op
01	A0 00	Put zero in Y (ADDL)
03	A6 49	Get CTR in X ◄
05	8A	Put X in Accum.
06	91 4A	Store CTR in memory
08	A9 00	Zero Accum.
0A	B1 4A	Get memory output
0C	C5 49	Is output = input?
0E	F0 1A	Branch if equal
10	20 B1 72	≠ ; print output
13	20 77 73	Space
16	A5 4B	Get ADDH
18	20 B1 72	Print
1B	98	Get ADDL from Y
1C	20 B1 72	Print
1F	20 77 73	Space
22	A5 49	Get CTR (input)
24	20 B1 72	Print it
27	20 8A 72	CR LF
2A	E6 49	Increment CTR ◄
2C	D0 D5	Go back; CTR ≠ 0
2E	C8	CTR=0; increment ADDL
2F	D0 D2	Go back ADDL ≠ 0
31	E6 4B	ADDL=; increment ADDH
33	A9 1F	Get '1F' in Accum.
35	C5 4B	Compare to ADDH
37	B0 CA	Go back if ADDH ≤ 1F

PROGRAM LISTING CONTINUED

Program Counter	Program Code	Remarks
39	A9 45	'E'
3B	20 C6 72	Print
3E	A9 4E	'N'
40	20 C6 72	Print
43	A9 44	'D'
45	20 C6 72	Print
48	00	Break to stop
49	00	CTR storage
4A	00 10	ADDL, ADDL-1 storage