MICRO-PROCESSOR TO CRT INTERFACE

by

Hector Ernesto Mery

A Thesis Submitted to the Faculty of the

DEPARTMENT OF ELECTRICAL ENGINEERING

In Partial Fulfillment of the Requirements
For the Degree of

MASTER OF SCIENCE

In the Graduate College

THE UNIVERSITY OF ARIZONA

1 9 7 6

STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: Herb Mey

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

Granino A. Korn

G. A. KORN
Professor of Electrical Engineering

5/12/76
Date

ACKNOWLEDGMENTS

The author expresses his appreciation to Dr. G. A. Korn under whose direction this work was performed. His help in laying out the general approach is also acknowledged.

The Kitt Peak National Observatory helped by providing the test equipment.

TABLE OF CONTENTS

		Page
	LIST OF TABLES	v
	ABSTRACT	vi
1.	INTRODUCTION	1
2.	CC-301 TELEVISION CONTROLLER	4
	General Description	4
	Input/Output Ports	4
	Computer Input Channel	4
	Computer Output Channel	5
3.	THE MICRO-PROCESSOR	6
	General Description	6
	Bus Operation	6
	Bus Control	7
4.	THE INTERFACE UNIT	8
	Introduction	8
	General Description	8
	CC-301 Control and Data Port	9
	Control and Decoding Circuit	10
	Start of Text	10
	Hexadecimal Character	10
	Space	11
	End of Text	11
	Illegal Character	11
	Micro-Processor Port	12
	Firmware	13
	SELECTED BIBLIOGRAPHY	14

LIST OF TABLES

Page

1. MICROPROGRAM 3

ABSTRACT

An interface between a Micro-Processor and a Computer Communication Station has been designed. Use is made of the Direct Memory Access capability of the Micro-Processor and the Block Transfer mode of the Computer Communication Station to load the Micro-Processor random access memory with a program.

The task of the Interface Unit is to convert USASCII Standard Code characters into its equivalent 4-bit hexadecimal representation and to format these 4-bit words into words of length suitable for the Micro-Processor.

CHAPTER 1

INTRODUCTION

This paper describes the interface between a National Semiconductor Model IMPL-16L Micro-Processor and a Computer Communications Incorporated Model CC-301 Television Controller.

The Micro-Processor consists of a Central Processing Unit and a Memory with DMA (Direct Memory Access Capability). It does not contain any firmware to load a program into the Micro-Processor random access memory.

The Television Controller generates and displays data on a standard television receiver and accepts information from a keyboard or other external devices. The CC-301 features a buffer memory which is used as the basic for the generation of the display; that is, this memory is a one-to-one mapping of the data displayed on the television receiver. Data from the CC-301 memory can be transmitted to the Micro-Processor in a single word mode or block mode. This latter mode is used to load a program into the Micro-Processor's memory. The Interface Unit accepts USASCII (United States American Standard Code for Information Interchange, 1968) characters from the CC-301 controller, decodes these characters into 4-bit words, assembles them into

16-bit words and by means of DMA stores them into the Micro-Processor's memory.

After a program has been loaded, it provides the DMA start address to the Micro-Processor.

As mentioned above, the Micro-Processor does not have any firmware to allow the loading of a program. DMA permits the storing of a program into memory, but it does not provide a means of running the program. Therefore, some kind of minimal firmware is needed to run a program stored in memory.

The Micro-Processor memory contains space for ROM (Read Only Memory), which was not included when this system was acquired. Therefore, an ROM interface was also built to permit the construction of a minimal microprogram that would look for the starting address of DMA and then jump to that location where the first word of the program loaded by DMA is stored (Table 1). The Interface Unit, therefore, has basically four tasks to execute: to accept characters from the Television Controller, to decode and assemble these words, to interrupt the Micro-Processor and initiate a DMA transfer, and to make the DMA start address available to the Micro-Processor.

TABLE 1

MICROPROGRAM

MEMORY LOCATION	INSTRUCTION	LABEL	SOURCE STATEMENT	
FFF0	3081	START	NOP	
FFF1	4C00		CLR	AC0
FFF2	3081		NOP	
FFF3	4F00		CLR	AC3
FFF4	0437		RIN	BEGIN
FFF5	3081		NOP	
FFF6	1107		BOC	LAST
FFF7	4F00		CLR	AC3
FFF8	0437		RIN	BEGIN
FFF9	3281		RCPY	AC0,AC2
FFFA	2200		JMP	BEGIN
FFFB	0000			
FFFC	0000			
FFFD	0000			
FFFE	21F1	LAST	JMP	START
FFFF	0000			

CHAPTER 2

CC-301 TELEVISION CONTROLLER

General Description

The CC-301 Television Controller generates and displays data on a standard television receiver, accepts information from a keyboard or other input devices and controls the communication with a computer. The CC-301 Television Controller has a memory which can be loaded from a computer and its data can be transmitted to a computer.

Input/Output Ports

There are an input port and output port available for communicating with a computer, labeled Computer Input Channel and Computer Output Channel.

Computer Input Channel

Transfers of data from the CC-301 Television Controller to the Micro-Processor take place on this port. This is a parallel channel consisting of the following signals: 8 Data Signals, 4 Control Signals and 1 Timing Signal.

Operation starts when the CC-301 Television Controller informs the Interface Unit by sending a Transmit Request that it has one or more data words ready for transmission. The external device, here the Micro-Processor Interface Unit, responds with an Input Request Signal when it

is ready to accept the data. The CC-301 informs the Interface Unit by an Input Resume, a data strobe, that it has placed a word on the data lines. The Micro-Processor Interface Unit then drops the Input Request synchronously with the timing signal provided by the CC-301. After the CC-301 transmits the last data word, it drops the Transmit Request.

Computer Output Channel

Data from the Micro-Processor to the CC-301 controller is transmitted through this channel. This is a parallel channel consisting of the following signals: 8 Data Signals, 5 Control Signals and 1 Clock Signal.

The Micro-Processor Interface Unit raises the Output Request line when it has data ready for transmission. The CC-301 controller responds by raising its Output Resume line. This strobe signal transfers the Micro-Processor data into the CC-301 Input/Output Data Register. If no more data words are immediately available, the Output Request is dropped. Should the data word be a function command, a Function Request rather than the Output Request is raised by the Interface Unit.

CHAPTER 3

THE MICRO-PROCESSOR

General Description

The IMP-16L is a general purpose 16-bit Microcomputer with a very powerful instruction set. It is a bus-oriented machine with DMA capability.

The IMP-16L consists of a CPU board which contains the Central Processor, Bus Controller logic and Bus Interface logic circuits and a Memory Board containing 4096 words of 16-bits. Both boards are housed in a card tray which makes up the IMP-16L system.

A three-bus structure is used to implement the IMP-16L DMA capability. This three-bus structure consists of a 16-bit bidirectional data bus which is time multiplexed for outputting address and inputting or outputting data, a timing signal bus made of clocks and data strobes, and a control signal bus for priority operations and peripheral bus request and grant.

Bus Operation

Any device tied to the bus whether it is CPU, memory or peripheral device must ask permission from the bus controller logic to use the bus for communicating with another device.

The bus controller consists of an ordered priority circuit and a timing and control circuit. The priority logic decides what device

in order of importance may use the bus. While the timing and control logic generates signals necessary for synchronous operation.

Bus Control

A peripheral device requests the bus by generating a Bus Request signal. This is an asynchronous operation initiated by the requestor over the control signal bus. A separate bus request line is provided for each device. In response to the request, permission to use the data bus for one bus cycle is granted by an ordered prioritizer and must be recognized synchronously by the requestor. As soon as the requestor receives permission to use the bus, address and data must be available for transmission during the same bus cycle.

CHAPTER 4

THE INTERFACE UNIT

Introduction

The Micro-Processor to CRT Television Controller Interface allows the communication between a National Semiconductor IMP-16L Micro-Processor and a Computer Communication CC-301 Television Controller by accepting 8-bit characters from the television controller, decoding these 8-bit characters into 4-bit hexadecimal words and then assembling these 4-bit words into one 16-bit word. This 16-bit word is then transferred into the Micro-Processor's memory by means of DMA transfers.

Once a program has been transferred into memory, it can be executed by providing the DMA start address to the microprogram stored in the ROM of the Micro-Processor. The microprogram then detects this address, jumps to that location and executes the program.

General Description

The Interface Unit is functionally divided into three sections: a CC-301 Control and Data Port, A Control and Decoding Circuit, and a Micro-Processor Port.

The CC-301 Port generates the hand-shake controls required to accept characters from the CC-301 Communication Station. The control and decoding section decodes and takes the required action depending on

the character received and decides when to request the Micro-Processor data bus. This section also decodes the instructions executed by the firmware residing in ROM.

Finally the Micro-Processor Port generates the control and timing signals required to obtain the Micro-Processor data bus. Once permission is granted, it produces the gating signals to place address and data on the data bus.

There are two 8-bit DIP (Dual in Line Package) switches located on the Interface card, into which the desired start address for DMA transfers is manually entered.

These switches are connected to a 16-bit register, as well as to a 16-bit presettable counter, labeled DMA Start Address Register and DMA Address Counter respectively. The DMA Start Address Register is initially cleared to zero and remains in that state until manually loaded by an Execute push-button also located on the Interface card.

The DMA Address Counter is loaded with the state of the address switches when a Start of Text character is received from the CC-301 Communication Station.

CC-301 Control and Data Port

The CC-301 raises its Transmit Request line when it is ready to send a character. The Interface Unit latches this condition synchronously with the clock provided by the CC-301 into a flip-flop, whose output is used to return in Input Request back to the CC-301 indicating that the Interface unit is ready to accept the character.

The CC-301 then places the data on the data lines and issues an Input Resume strobe. This strobe stores the 8-bit character in a register, causes the Input Request line to be dropped synchronously with the positive edge of the next clock pulse after the Input Resume strobe, sets a busy flip-flop which hinders further Transmit Request from being acknowledged, and finally generates a pulse synchronized with the master clock of the Micro-Processor. This pulse is labeled Character Receiver and, as mentioned above, it corresponds to the CC-301 Input Resume strobe, with the difference that Character Received is synchronized to the Micro-Processor Master Clock.

Control and Decoding Circuit

The character received is decoded by two ROMS into a 4-bit hexadecimal word as well as five control signals which indicate if the Character Received is a Start of Text, a Hexadecimal Character, a Space, an End of Text, or an Illegal Character. Depending upon the character received, different action is taken. Following is a description of these five cases.

Start of Text. The Start of Text control line loads the DMA Address Counter with the start address present in the DMA Address Switches and resets a Busy Flip-Flop to allow the transfer of the next character from the CC-301 to the Interface Unit.

Hexadecimal Character. The Hexadecimal control line strobes the 4-bit representation of the 8-bit character into a 4 x 4-bit shift

register which makes up the 16-bit Data Register. It also increments a 3-bit Word Counter which keeps account of the number of words received. When this counter reaches a count of 4, it indicates that a 16-bit word has been assembled and can now be transferred to the Micro-Processor memory. If the Word Counter has not yet reached 4, the Hexadecimal word is loaded into the Data Register and the Busy Flip-Flop reset to permit the transfer of another word.

Space. When a space is received, depending on the state of the Word Counter, the following action is taken: (1) If the Word Counter has reached a count of 4, a Bus Request to the Micro-Processor is generated; (2) If a count of 4 has not been reached, the Busy Flip-Flop is reset and the next word requested.

End of Text. The End of Text control signal indicates that the transmission has terminated. This signal resets all control flip-flops involved in the character transfers from the CC-301 to the Interface Unit.

Illegal Character. Any character that is not a Hexadecimal Character, a Space, a Start of Text, or an End of Text is considered an Illegal Character.

When an Illegal Character is received, it is simply ignored and the Busy Flip-Flop reset to permit the transfer of the next character.

Micro-Processor Port

The Micro-Processor Port consists of multiplexers, device address decoder, and tri-state bus drivers as well as control flip-flops to gate address and data onto the Micro-Processor bus.

When a Bus Request command is received from the Control and Decoding section and the Interface Bus Request line is activated, the Micro-Processor answers by activating its Interface Priority Select line, which indicates that one bus cycle has been granted to the Interface Unit. Once the bus is granted this section produces an Interface Write Memory Cycle signal to indicate that a word is going to be transferred into memory. The address contained on the DMA Address Counter is placed on the data bus while the Interface Address strobe is true, then the data stored in the Data Register is placed also on the bus for the duration of the Interface Write Data strobe. Finally the Interface Write Memory Cycle signal is reset by the Interface Read Data Strobe generated in the bus controller logic.

The assigned peripheral device address, address 6 for the Interface Unit, is decoded by this section, the type of command, RIN or ROUT, as well as the Order of the commands executed by the Micro-Processor are also decoded in this section. The firmware residing in ROM continuously executes a RIN of Order 7 command to read the DMA Start Address Register. Each time this command is decoded the contents of this register is placed on the data bus.

When power is applied the Program Counter is set to $FFFF_{16}$ and the Micro-Processor executes the instruction stored in that location. Address $FFFF_{16}$ contains a Jump instruction to location $FFF0_{16}$ which is the start address of the microprogram.

The microprogram provides the linkage to execute the program stored in the random access memory via DMA transfers. This is done by executing a RIN of Order 7 instruction. RIN 7 reads the contents of the DMA Start Address Register. As mentioned above the register is kept at zero and can only be loaded with the state of the DMA address switches by depressing the Execute push-button. Therefore a RIN 7 will initially read zero. After reading the DMA Start Address Register, the microprogram checks for a nonzero address by executing a Branch on Condition instruction. If the address is zero it branches to location $FFFF_{16}$ and the cycle is repeated. If the address is other than zero, it executes another RIN 7 to read that address once more, thus avoiding the probability that the register was in the process of being loaded as the previous RIN 7 was executed causing an erroneous address to be read. Next an unconditional Jump instruction relative to that address is performed. This presets the Program Counter to the first address of the program stored in the random access memory and the program executed.

Firmware

The IMP-16L Memory Board contains space for 256 16-bit words of ROM. These locations occupy the address space between locations $FF00_{16}$ and $FFFF_{16}$.

SELECTED BIBLIOGRAPHY

Computer Communications Incorporated. CC-30 Communications Station Reference Manual. April 1968.

Hill, F. J., and Peterson, G. R. Introduction to Switching Theory and Logic Design. Wiley, New York, 1968.

Korn, G. A. Minicomputers for Engineers and Scientists. McGraw-Hill, New York, 1973.

National Semiconductor Corporation. IMP-16L Users Manual. Publication No. 4200028A, 1974.