

MCCS

The Design and Implementation  
of a Multi-Computer  
Communications System

by

SHELDON LEE FOX

B.S., Kansas State University, 1973

---

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

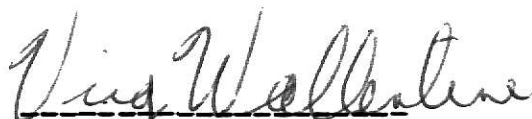
MASTER OF SCIENCE

Department of Computer Science

KANSAS STATE UNIVERSITY  
Manhattan, Kansas

1976

Approved by:

  
Major Professor

LD  
2668  
R4  
1976  
F68  
C.2  
Document

TABLE OF CONTENTS

I.	INTRODUCTION .....	1
II.	MCCS SYSTEM SPECIFICATIONS .....	4
	Process-Level Interface .....	4
	System Overview .....	7
	Data Structures .....	7
	Functional Description .....	9
	Algorithmic Description .....	15
	Message Format .....	22
	Asynchronous Line Protocol .....	24
	Binary Synchronous Line Protocol .....	27
	Future Enhancements .....	29
III.	MCCS/370 - THE VM/370 IMPLEMENTATION OF MCCS .....	31
	Operating Environment .....	31
	Flow of Control .....	32
	Operator Commands .....	46
	Testing Aids .....	49
	System Correctness, Status, and Extensions .....	51
IV.	BIBLIOGRAPHY .....	55
V.	APPENDICES .....	57
	Appendix A - MCCS/370 Module Summaries .....	57
	Appendix B - MCCS/370 ABEND Codes .....	74
	Appendix C - MCCS/370 VM/370 Dependencies .....	78
	Appendix D - MCCS/370 Update Procedures .....	80
	Appendix E - MCCS/370 Control Blocks .....	81

## FIGURES AND TABLES

Figure 1	SEND and RECV Calling Formats .....	5
Figure 2	MCCS Control and Data Flow .....	8
Figure 3	PRQBLOK Format .....	11
Figure 4	MSGBLOK Format .....	11
Figure 5	Message Format .....	23
Figure 6	FLAG2 Bit Definitions .....	23
Figure 7	MCCS/370 Module Diagram .....	33
Figure 8	MCCMSG "msg" Header Format .....	51
Figure 9	MCCMSG File Header Format .....	51
Table 1	Control Character Replacement .....	25

## ACKNOWLEDGEMENTS

I would like to thank my major professor, Dr. Virgil Wallentine, for his help and guidance throughout this project and Drs. Kenneth Conrow and Fred Maryanski for their comments and suggestions concerning this paper. I would also like to thank the Kansas State University Computing Center for letting me use their binary synchronous lines and modems for testing the bisync support in M CCS along with the technical consultation provided by the Systems Programming staff there throughout the implementation period. Finally, I would like to thank the University of Waterloo, and more specifically, Bruce Uttley, for making available their version of the CMS SCRIPT text formatting program which made the preparation of this paper many times less difficult than it would of been using other methods.

## I. INTRODUCTION

This paper describes the Multi-Computer Communications System (MCCS). MCCS allows tasks executing in one computer to communicate, via "messages", with tasks executing in one of several other computers.

MCCS provides two process-level primitives, SEND and RECV, to facilitate inter-process communication. MCCS is simple enough to allow it to be implemented easily on a mini-computer which provides a minimal level of hardware and software support using the asynchronous line protocol. On the other hand, employing the MCCS binary synchronous (BSC) line protocol [IBM01] on a more sophisticated computer system allows the speed and error detection/recovery facilities of BSC to be utilized to achieve more efficient communication.

MCCS message protocol was designed to be straightforward and does not depend on the existence of any complex operating system functions. Although this somewhat restricts the flexibility of the message system (or at least shifts some of the burden to the user process), it makes MCCS simple enough to be implemented on nearly any computer which supports the execution of one or more processes and can have a teleprocessing (TP) line attached to it. The original design goal of providing the capability for inter-process communication on multiple computers may then be realized.

Normally executing as a privileged system task, MCCS should run without operator intervention. The procedures for automatic connection and disconnection of computers to one another have been defined. MCCS also handles any error conditions that should arise during its execution.

A message network employing MCCS consists of two or more computers, each of which may have connections, and therefore communicate, with one or more of the others. MCCS places no restrictions on the size and complexity of the network; it is instead restricted only by the hardware and software limitations of the computers which compose the network.

This project (and hence the paper) was motivated by a desire to provide a means of communications between an existing mini-computer (a NOVA 2/10) and a maxi-computer (an IBM 370/158). One of the major constraints placed on this communication was that the method of communications, whatever it be, require as few hardware and operating system modifications as possible. It was decided to utilize existing teleprocessing lines designed to support