

2-8

CR-128556

SD 72-SA-0114-3

# MODULAR space station

## PHASE B EXTENSION

### INFORMATION MANAGEMENT ADVANCED DEVELOPMENT FINAL REPORT

Volume III: Digital Data Bus Breadboard



PREPARED BY PROGRAM ENGINEERING  
JULY 31, 1972

(NASA-CR-128556) INFORMATION MANAGEMENT  
ADVANCED DEVELOPMENT, VOLUME 3: DIGITAL  
DATA BUS BREADBOARD Final Report C.R.  
Gerber (North American Rockwell Corp.)  
31 Jul. 1972 187 p

707121314151617  
N72-33846  
Unclas  
16405  
CSC 22B G3/31  
707121314151617



Space Division  
North American Rockwell

CONTRACT NAS9-9953 MSC 02471  
DRL NO: MSC-T-575, LINE ITEM 72

SD 72-SA-0114-3

MODULAR  
**space station**  
PHASE B EXTENSION

---

**INFORMATION MANAGEMENT ADVANCED DEVELOPMENT FINAL REPORT**

**Volume III: Digital Data Bus Breadboard**

31 JULY 1972  
PREPARED BY PROGRAM ENGINEERING

Approved by



James Madewell  
Director  
Space Applications Programs



Space Division  
North American Rockwell

TECHNICAL REPORT INDEX/ABSTRACT

|  |  |  |                                     |   |  |   |  |
|--|--|--|-------------------------------------|---|--|---|--|
| ACCESSION NUMBER   |  |  |                                     | DOCUMENT SECURITY CLASSIFICATION<br><b>UNCLASSIFIED</b> |  |   |  |
| TITLE OF DOCUMENT<br><b>INFORMATION MANAGEMENT ADVANCED DEVELOPMENT FINAL REPORT<br/>Volume III, Digital Data Bus Breadboard</b> |  |  |                                     |   |  | LIBRARY USE ONLY                          |  |
| AUTHOR(S)<br><b>Gerber, C. R., et.al.</b>  |  |  |                                     |   |  |   |  |
| CODE<br><b>QN085282</b>  |  | ORIGINATING AGENCY AND OTHER SOURCES<br><b>Space Division of North American<br/>Rockwell Corporation, Downey, California</b> |                                     |   |  | DOCUMENT NUMBER<br><b>SD 72-SA-0114-3</b> |  |
| PUBLICATION DATE<br><b>July 31, 1972</b>   |  |  | CONTRACT NUMBER<br><b>NAS9-9953</b> |   |  |   |  |

DESCRIPTIVE TERMS

**\*MODULAR SPACE STATION, \*INFORMATION MANAGEMENT,  
\*ADVANCED DEVELOPMENT, \*DATA ACQUISITION,  
\*DATA BUS, \*BREADBOARD, \*TECHNOLOGY,  
\*SPECIFICATIONS, \*DBC, \*RACU**

ABSTRACT

THIS DOCUMENT IS VOLUME III OF THE FINAL REPORT OF THE MODULAR SPACE STATION ADVANCED DEVELOPMENT STUDY. IT SUMMARIZES THE EFFORT AND RESULTS OF THE ANALYSIS AND BREADBOARDING OF A DATA ACQUISITION AND CONTROL SUBASSEMBLY (DACS) TO PROVIDE THE NECESSARY INTERFLOW OF DATA BETWEEN TWO CENTRAL PROCESSORS, SUBSYSTEMS AND EXPERIMENT EQUIPMENT. THE DACS WAS DEFINED TO INCLUDE THE DIGITAL DATA BUS (DDB) THE DATA BUS CONTROL UNIT (BCU), AND THE REMOTE ACQUISITION AND CONTROL UNIT (RACU); THE FIRST TWO OF THESE WERE ANALYZED AND BREADBOARDED AS A PART OF THE EFFORT. THE PRIMARY OBJECTIVE OF THE DACS BREADBOARD IS TO VERIFY THE DIGITAL DATA BUS CONCEPT FOR THE MODULAR SPACE STATION. IT DEMONSTRATES THE AVAILABILITY OF TECHNOLOGY TO PROVIDE ACCURACY OF DATA TRANSFER, RECONFIGURABILITY, FAILURE TOLERANCE, LONG USEFUL LIFE, AND STANDARDIZATION OF INTERFACES.

## FOREWORD

This document is one of a series required by Contract NAS9-9953, Exhibit C, Statement of Work for the Phase B Extension - Modular Space Station Program Definition. It has been prepared by the Space Division, North American Rockwell Corporation, and is submitted to the National Aeronautics and Space Administration's Manned Spacecraft Center, Houston, Texas, in accordance with the requirements of the Data Requirements List (DRL) MSC-T-575, Line Item 72.

This document is Volume III of the Modular Space Station Information Management System Advanced Development Technology Report, which has been prepared in the following six volumes:

|     |  |                |
|-----|--|----------------|
| I   | IMS ADT Summary                            | SD72-SA-0114-1 |
| II  | IMS ADT Communications Terminal Breadboard | SD72-SA-0114-2 |
| III | IMS ADT Digital Data Bus Breadboard        | SD72-SA-0114-3 |
| IV  | IMS ADT Data Processing Assembly           | SD72-SA-0114-4 |
| V   | IMS ADT Software Assembly                  | SD72-SA-0114-5 |

ACKNOWLEDGEMENTS

The following persons have participated in the conduct of the IMS ADT tasks, and have contributed to this report:

|                  |                                      |
|------------------|--------------------------------------|
| C. W. Roberts    | Experiment/Electronics Manager       |
| C. R. Gerber     | Information Systems Project Engineer |
| B. A. Logan, Jr. | Information Systems                  |
| E. Mehrbach      | Information Systems                  |
| D. W. Brewer     | Information Systems                  |
| V. R. Hodgson    | Information Systems                  |

The following subcontractors have supported the IMS ADT tasks in specialized areas:

|   |   |
|---|---|
| International Tel. & Tel<br>Nutley, New Jersey<br>B. Cooper, Proj. Mgr. | Communications Terminal B.B.<br>Data Bus BB |
| Intermetrics<br>Cambridge, Mass.<br>J. Miller, Prog. Mgr.               | Data Processing Assy                        |
| System Develop. Corp<br>Santa Monica, Calif.<br>R. Bilek, Prog. Mrg.    | Data Processing Assy<br>Software Assy       |
| Gen'l Electric Corp.<br>Valley Forge, Pa.<br>R. Kirby, Prog. Mgr.       | Bulk Storage Technology                     |
| NR-Autonetics<br>Anaheim, Calif.<br>J. Jurison, Proj. Mgr.              | Data Bus BB<br>Data Processing Assy         |

Preceding page blank

PRECEDING PAGE BLANK NOT FILMED

## CONTENTS

| Section |   | Page |
|---------|---|------|
| 1       | INTRODUCTION . . . . .  | 1-1  |
|         | 1.1 DACS REQUIREMENTS SUMMARY . . . . .                                 | 1-1  |
|         | 1.2 DACS BREADBOARD DESIGN. . . . .                                     | 1-3  |
| 2       | PARAMETRIC DATA FOR BUS DESIGN . . . . .                                | 2-1  |
|         | 2.1 DIGITAL BUS . . . . .   | 2-1  |
|         | 2.2 ANALOG BUS . . . . .  | 2-4  |
|         | 2.3 COUPLING CONFIGURATION . . . . .                                    | 2-4  |
|         | 2.4 CONCLUSIONS AND RECOMMENDATIONS . . . . .                           | 2-7  |
| 3       | DACS MODEL CONFIGURATION . . . . .                                      | 3-1  |
|         | 3.1 INTRODUCTION AND SUMMARY . . . . .                                  | 3-1  |
|         | 3.2 RECOMMENDED DACS BREADBOARD<br>CONFIGURATION. . . . .               | 3-1  |
| 4       | DACS REDUNDANCY CONCEPTS . . . . .                                      | 4-1  |
|         | 4.1 INTRODUCTION AND SUMMARY OF<br>REDUNDANCY REQUIREMENTS . . . . .    | 4-1  |
|         | 4.2 SUBSYSTEM MODELS AND CONFIGURATIONS . . . . .                       | 4-2  |
|         | 4.3 DACS REDUNDANCY AND RELIABILITY CONCEPTS . . . . .                  | 4-7  |
|         | 4.4 DACS RECOMMENDATIONS. . . . .                                       | 4-34 |
| 5       | DACS BREADBOARD DESIGN REQUIREMENTS . . . . .                           | 5-1  |
|         | 5.1 GENERAL REQUIREMENTS . . . . .                                      | 5-1  |
|         | 5.2 BREADBOARD DESIGN REQUIREMENTS . . . . .                            | 5-2  |
| 6       | DATA BUS BREADBOARD PERFORMANCE AND<br>INTERFACE REQUIREMENTS . . . . . | 6-1  |
|         | 6.1 GENERAL REQUIREMENTS . . . . .                                      | 6-1  |
|         | 6.2 BREADBOARD DESIGN REQUIREMENTS . . . . .                            | 6-2  |

**Preceding page blank**

PRECEDING PAGE BLANK NOT FILMED

| Section |  | Page |
|---------|--|------|
| 7       | DBCUC BREADBOARD PERFORMANCE AND INTERFACE<br>REQUIREMENTS . . . . . | 7-1  |
|         | 7.1 ITEM DEFINITION . . . . .  | 7-1  |
|         | 7.2 FUNCTIONAL DESCRIPTION . . . . .                                 | 7-1  |
|         | 7.3 INTERFACE DEFINITION . . . . .                                   | 7-1  |
|         | 7.4 PERFORMANCE . . . . .  | 7-14 |
|         | 7.5 PHYSICAL CHARACTERISTICS . . . . .                               | 7-24 |
|         | 7.6 ENVIRONMENTAL . . . . .  | 7-26 |
| 8       | DIGITAL DATA BUS DESIGN . . . . .                                    | 8-1  |
|         | 8.1 SYSTEM SPECIFICATIONS . . . . .                                  | 8-1  |
|         | 8.2 COMPONENT PERFORMANCE<br>SPECIFICATIONS . . . . .                | 8-4  |
| 9       | DBCUC DESIGN . . . . .   | 9-1  |
|         | 9.1 PHYSICAL DESCRIPTION . . . . .                                   | 9-1  |
|         | 9.2 CONTROLS AND DISPLAYS . . . . .                                  | 9-2  |
|         | 9.3 BASIC OPERATION . . . . .  | 9-9  |
|         | APPENDIX A PRELIMINARY DACS BREADBOARD<br>LAYOUT . . . . .           | A-1  |

## ILLUSTRATIONS

| Figure |   | Page |
|--------|---|------|
| 1-1    | DACS Work Breakdown and Flow . . . . .                              | 1-2  |
| 1-2    | Recommended DACS Configuration . . . . .                            | 1-4  |
| 1-3    | Functional Relationship of Breadboard Elements . . . . .            | 1-6  |
| 1-4    | Dual Redundant DACS Breadboard . . . . .                            | 1-7  |
| 2-1    | Digital Data Bus, First Configuration . . . . .                     | 2-2  |
| 2-2    | Digital Data Bus, Second Configuration . . . . .                    | 2-3  |
| 2-3    | Methods of Coupling . . . . .                                       | 2-5  |
| 2-4    | Direct and Transformer Coupling with T-Pads . . . . .               | 2-6  |
| 3-1    | Data Management Subsystem EEM Growth Sequence . . . . .             | 3-6  |
| 4-1    | Non-critical Subsystem Functional Loop Models . . . . .             | 4-3  |
| 4-2    | Non-Time Critical Subsystem Functional Loop Models . . . . .        | 4-4  |
| 4-3    | Time-Critical Subsystem Functional Loop Models . . . . .            | 4-6  |
| 4-4    | Candidate DB-1, Four Simplex Data Bus Subassemblies . . . . .       | 4-9  |
| 4-5    | Candidate DB-2, Two Dual Redundant Data Bus<br>Assemblies . . . . . | 4-10 |
| 4-6    | Candidate DB-3, One Quad Redundant Data Bus<br>Assembly . . . . .   | 4-12 |
| 4-7    | Candidate R-1, RACU with Simplex Data Bus Interface . . . . .       | 4-15 |
| 4-8    | Candidate R-2, RACU with Dual Data Bus Interface . . . . .          | 4-18 |
| 4-9    | Candidate R-3, RACU with Quad Data Bus Interface . . . . .          | 4-21 |
| 4-10   | Candidate CU-1, Single IOP Simplex DBCU . . . . .                   | 4-27 |
| 4-11   | Candidate CU-2A, Dual IOP Simplex DBCU . . . . .                    | 4-29 |
| 4-12   | Candidate CU-2B, Dual IOP Single Failure Tolerance DBCU . . . . .   | 4-31 |
| 4-13   | Recommended DACS Configuration . . . . .                            | 4-38 |
| 5-1    | Message Formats . . . . .   | 5-8  |
| 5-2    | Breadboard Modem Unit Functional Interface . . . . .                | 5-15 |
| 5-3    | Breadboard External Test Function Interface . . . . .               | 5-19 |
| 6-1    | Bi-Phase Level Data Encoding . . . . .                              | 6-6  |
| 6-2    | Uni-directional Equipment/Bi-directional Core . . . . .             | 6-12 |

| Figure |  | Page |
|--------|--|------|
| 7-1    | Functional Relationship of Various Breadboard Elements . . . . . | 7-2  |
| 7-2    | DBCUC/MODEM Interface . . . . .                                  | 7-3  |
| 7-3    | DBCUC/MODEM Interface Timing Diagram . . . . .                   | 7-6  |
| 7-4    | BIO/DBCUC Interface . . . . .                                    | 7-7  |
| 7-5    | BIO/DBCUC Timing Diagram . . . . .                               | 7-10 |
| 7-6    | DBCUC/STE Interface . . . . .                                    | 7-12 |
| 7-7    | DBCUC Functional Organization . . . . .                          | 7-15 |
| 7-8    | DBCUC/RACUC Message Formats . . . . .                            | 7-19 |
| 7-9    | Transmission Code Formats . . . . .                              | 7-23 |
|        |  |      |
| 8-1    | Skeleton Diagram of Data Bus System . . . . .                    | 8-2  |
| 8-2    | Breadboard Configuration, Block Diagram . . . . .                | 8-3  |
| 8-3    | Coupler, Schematic Diagram . . . . .                             | 8-5  |
| 8-4    | Receive Clock Tolerance . . . . .                                | 8-9  |
| 8-5    | Attenuation vs. Frequency for RG-22B/J Cable . . . . .           | 8-11 |
|        |  |      |
| 9-1    | DBCUC Front Panel . . . . .                                      | 9-3  |
| 9-2    | Tape Reader/Power Unit - Front Panel . . . . .                   | 9-10 |
| 9-3    | Compare . . . . .  | 9-12 |
| 9-4    | Set Mode of Operations . . . . .                                 | 9-13 |
| 9-5    | DBCUC/RACUC Data Transfer . . . . .                              | 9-14 |
| 9-6    | BIO/RACUC Data Transfer . . . . .                                | 9-15 |
| 9-7    | BIO/DBCUC Data Transfer . . . . .                                | 9-16 |
| 9-8    | Status Word . . . . .  | 9-17 |
| 9-9    | DBCUC Data Flow . . . . .  | 9-18 |

### TABLES

| Table |  | Page |
|-------|--|------|
| 4-1   | Number of RACUC's Per Subsystem Functional Loop Model for the Seven RACUC Candidates . . . . . | 4-24 |
| 4-2   | Minimum Number of Candidate Data Bus Control Units . . . . .                                   | 4-32 |
| 4-3   | DACS Element Candidates . . . . .  | 4-33 |
| 7-1   | Universal and Group Address Codes . . . . .  | 7-22 |

ABBREVIATIONS

|          |  |
|----------|--|
| ADS      | Advanced Data System   |
| ADT      | Advanced Development Technology                              |
| ADTX     | Advanced Development Technology Extension                    |
| AFC      | Automatic Frequency Control                                  |
| AGC      | Automatic Gain Control                                       |
| AN       | Autonetics (Division of North American Rockwell Corporation) |
|          |  |
| BPF      | Bandpass Filter  |
| bps      | Bits Per Second  |
|          |  |
| CAIRS    | Computer-Assisted Interactive Resource Scheduling            |
| CCIR     | International Radio Consultative Committee                   |
| CDR      | Critical Design Review                                       |
| CEI      | Contract End Item  |
| CLASP    | Computer Language for Aeronautics and Space Programming      |
| CM       | Command Module (Apollo)                                      |
| COMPOOL  | Common Pool (of Data)  |
| CP       | Central Processor or Circular Polarization                   |
| C.P.     | Computer Program   |
| CPCEI    | Computer Program Contract End Item                           |
| CPCI     | Computer Program Configuration Item                          |
| CPDF     | Computer Program Development Facility                        |
| CPIC (A) | Computer Program Integration Contractor (Agency)             |
| CPT&E    | Computer Programming Test and Evaluation                     |
| CR       | Change Report  |
| CRT      | Cathode-Ray Tube (Display)                                   |
| CSS      | Crew Subsystem   |
| CTB      | Communications Terminal Breadboard                           |
| CTF      | Central Test Facility  |
|          |  |
| DACS     | Data Acquisition and Control Subassembly                     |
| dB       | Decibel  |
| DBCUC    | Data Bus Control Unit  |
| dBm      | Decibel Referred to One Milli-Watt                           |
| dBW      | Decibel Referred to One Watt                                 |
| DCR      | Design Change Request  |
| DDB      | Digital Data Bus   |
| Demux    | De-Multiplex(er)   |
| DMS      | Data Management System                                       |
| DPA      | Data Processing Assembly                                     |
| DPSK     | Dual Phase Shift Keying                                      |
| DRSS     | Data Relay Satellite System                                  |



|                  |   |
|------------------|---|
| ECP              | Engineering Change Proposal                       |
| EDF              | Experiment Data Facility                          |
| EEM              | Engineering Evaluation Model                      |
| EEMP             | Engineering Evaluation Model Processor            |
| EIRP             | Effective Isotropic Radiated Power                |
| EMC              | Electromagnetic Compatability                     |
| EMI              | Electromagnetic Interference                      |
| EOS              | Earth Orbital Shuttle                             |
| EOSS             | Earth Orbital Space Station                       |
| EPS              | Electrical Power Subsystem                        |
| ETC/LSS or ECLSS | Environment Control and Life Support Subsystem    |
| EVA              | Extra-Vehicular Activity                          |
| EXT              | External  |
| $E_b/N_o$        | Energy Per Bit to Noise Density Ratio             |
| FACS             | Facsimile   |
| FDM              | Frequency-Division Multiplex                      |
| FM               | Frequency Modulation                              |
| FQT              | Formal Qualification Test                         |
| G&CS             | Guidance and Control Subsystem                    |
| GFE              | Government Furnished Equipment                    |
| GHz              | Giga-Hertz  |
| GOA              | Gated Operational Amplifier                       |
| HAL              | Higher-Order Aerospace Programming Language       |
| HOL              | Higher-Order Language                             |
| HOLM             | Higher-Order Language Machine                     |
| Hz               | Hertz   |
| IF               | Intermediate Frequency                            |
| IFRU             | In-Flight Replaceable Unit                        |
| IM               | Intermodulation Products                          |
| IMS              | Information Management System                     |
| IMSIM            | Information Management Simulation                 |
| IOC              | Initial Operational Capability                    |
| IOCB             | Input-Output Control Block                        |
| IOU              | Input-Output Unit                                 |
| I/O              | Input-Output                                      |
| IPA              | Intermediate Power Amplifier                      |
| IQL              | Interactive Query Language                        |
| IR               | Infra-Red   |
| ISS or IMS/S     | Information (Management) Subsystem                |
| ITT              | International Telephone and Telegraph             |
| K-words          | Thousands of (Computer) Words                     |
| K-EAPS           | Thousands of Equivalent-Add Operations Per Second |
| K-bps            | Thousands of Bits Per Second                      |
| $KH_z$           | Kilohertz   |



|          |   |
|----------|---|
| LEM      | Lunar Excursion Module                                  |
| LM       | Lunar Module  |
| LNA      | Low Noise Amplifier                                     |
| LO       | Local Oscillator  |
| LPF      | Low Pass Filter   |
| M1, M2   | (Computer) Memory Designation                           |
| Mbps     | Megabits Per Second                                     |
| MCB      | Module Control Block                                    |
| MHz      | Megahertz   |
| MOF      | Mission Operations Facility                             |
| MOL      | Manned Orbiting Laboratory                              |
| MSC      | Manned Spacecraft Lenter                                |
| MSFN     | Manned Space Flight Network                             |
| MSS      | Modular Space Station                                   |
| MUX      | Multiplexer   |
| mW       | Milli-Watts   |
| MW       | Microwave   |
| mV       | Milli-Volts   |
| NF       | Noise Figure  |
| OBCO     | On-Board Checkout                                       |
| OCC      | Operations Control Center (On-Board)                    |
| ODM      | Operational Data Management                             |
| OM       | Operating Memory  |
| PA       | Power Amplifier   |
| PCM      | Pulse Code Modulation                                   |
| PDR      | Preliminary Design Review                               |
| PL/1     | Procedure Language                                      |
| PM       | Phase Modulation  |
| PN (PRN) | Pseudo Random Noise                                     |
| ppm      | Parts Per Million                                       |
| POT      | Preliminary Qualification Tests                         |
| PSK      | Phase Shift Keying                                      |
| RAM      | Research and Applications Module                        |
| RACU     | Remote Acquisition and Control Unit                     |
| RCS      | Reaction Control Subsystem                              |
| RF       | Radio Frequency   |
| RHCP     | Right-Hand Circular Polarization                        |
| RPU      | Remote Processing Unit                                  |
| Rx       | Receive   |
| S&C      | Standards and Conventions                               |
| SCCB     | Software Configuration Control Board                    |
| SCN      | Specification Change Notice                             |
| SD       | Space Division (of North American Rockwell Corporation) |
| SDC      | Systems Development Corporation                         |



|         |   |
|---------|---|
| S/N     | Signal to Noise Ratio                     |
| SOW     | Statement of Work                         |
| SPL     | Space Programming Language                |
| SRD     | Step-Recovery Diode                       |
| SSCB    | Solid-State Circuit-Breaker               |
| SSS     | Structures Subsystem                      |
| STE     | Support Test Equipment                    |
| TAV     | Test and Validation (Programs)            |
| TBD     | To Be Determined                          |
| TCXO    | Temperature-Controlled Crystal Oscillator |
| TDA     | Tunnel Diode Amplifier                    |
| TDM     | Time Division Multiplexing                |
| TDRS    | Tracking and Data Relay Satellite         |
| TIP     | Test and Integration Plan                 |
| TLM, TM | Telemetry                                 |
| TOOL    | Test Operations Oriented Language         |
| TRW     | Thompson Ramo Woolridge Corporation       |
| TT&C    | Telemetry, Tracking and Control           |
| TWT     | Traveling Wave Tube                       |
| TWTA    | Traveling Wave Tube Amplifier             |
| Tx      | Transmit                                  |
| USB (E) | Unified S-Band (Equipment)                |
| UV      | Ultra-Violet                              |
| VDD     | Version Description Document              |
| VHF     | Very High Frequency                       |
| VSF     | Vestigial Side Band                       |
| VSWR    | Voltage Standing Wave Ratio               |

## LIST OF INTERIM REPORTS

AA-101                    DPA Flow Diagrams, September 1971

AA-102                    DPA Throughput and Authority Analysis,  
February 1972

AA-103                    DPA Configuration Selection, April 1972

AS-101                    Modular Space Station Computer Program  
Standards and Conventions, December 1971

AS-102                    Modular Space Station Computer Program  
Specification Tree, February 1972

AS-103                    Modular Space Station Computer Program  
Development, Test and Configuration  
Control Plan, May 1972

AS-104                    Modular Space Station Computer-Assisted  
Resource Allocations and Utilization  
Recommendations, June 1972

CTB-101                   Concepts for Multiple RF Link  
Mechanization, May 1971

CTB-103                   Antenna-Mounted Electronics Component  
Design, October 1971

CTB-105/106               CTB Integration and Test and Operations  
Manual, June 1972

DB-101                    Parametric Data for Bus Design, May 1971

DB-103                    Component Performance Requirements,  
Schematics and Layout Drawings,  
December 1971

DB-104                    Digital Data Bus Breadboard Final Report,  
May 1972

DD-102                    Modular Space Station Data Processing  
Assembly Parametric Evaluation of  
Subsystems Input/Output Interface,  
June 1971

DD-103 Modular Space Station Data Acquisition and Control Subassembly Model Configuration (SD 71-233), July 1971

DP-101 Data Processing Assembly Configuration (Preliminary), June 1971

DP-102 Data Processing Assembly Supervisor Specification, May 1972

DP-103 DPA Processor Performance Requirements (Preliminary), August 1971

DP-103 DPA Processor Final Description, May 1972

DP-104 EEM DMS Processor Development Plan, June 1972

DP-105 Data Acquisition and Control Redundancy Concepts, August 1971

DP-106 Application of Redundancy Concepts to DPA, January 1971

DP-107 Data Acquisition and Control Subassembly Breadboard Design Requirements, October 1971

DP-108 Data Bus Control Unit Performance Requirements, January 1972

DP-109 Data Bus Control Design Reports, March 1971

DP-110 DBCU Acceptance Report (to be published)

EL-277 Bulk Storage Development Plan

IB-101 DPA Internal Flow and Traffic Pattern, May 28, 1971

ICD #TRW 20549 Interface Control Document - Data Bus Modem/RACU, Revision A, January 17, 1972

ICD #AN 26465 Interface Control Document - Data Bus Controller Unit to Buffer I/O, Revision January 21, 1972

MD-101 Mass Memory Parametric Data

RF-101 Modular Space Station Communications Terminal Breadboard Preliminary System Specification, October 1971

SA-101                    Central Processor Operational Analysis,  
                                 September 30, 1971

SA-102                    Central Processor Memory Organization and  
                                 Internal Bus Design, December 30, 1971

SD 71-227                Automatic Control and Onboard Checkout  
                                 Final Study Report

## 1. INTRODUCTION

### 1.1 DACS REQUIREMENTS SUMMARY

In the Modular Space Station the Data Processing Assembly (DPA) is highly distributed. The concept of two pressure volumes results in the division of the central processor in such a way that the computations associated with station operations and experiments can be performed in either volume. Similarly, the subsystems and experiments are divided between the two pressure volumes and, what is more, the subsystems are distributed throughout the modules that make up a 6-man or a 12-man configuration. Some of these subsystems require "on-the-spot" computations; these are provided in the DPA design by remote processing units (RPU's). All subsystems require computational support from the Data Processing Assembly. Therefore the DPA must acquire data from these distributed subsystems and return data, instructions and commands.

A significant portion of the ADT effort has been devoted to the analysis and breadboarding of a data acquisition and control subassembly (DACS) to provide the necessary interflow of data between the two central processors, the subsystems and the experiment equipment. The DACS has been defined to include the Digital Data Bus (DDB), the Data Bus Control Unit (DBCUC), and the Remote Acquisition and Control Unit (RACUC). Two of these have been analyzed and breadboarded as a part of the ADT effort; these are the DBCUC and the DDB.

Figure 1-1 presents the task breakdown and flow which was followed in ultimately delivering breadboards of the DBCUC and the DDB. Note that a RACUC/RPU breadboard is GFE.

The data acquisition and control analyses began with a theoretical analysis of the parameters pertinent to the design and usage of data buses. The result of this analysis was a "design handbook" covering the significant aspect of wideband digital and analog data buses.

Then a model was defined for the Data Acquisition and Control Subassembly (DACS) breadboard. The purpose of this definition was provide data to serve as a basis for the design of a DACS breadboard. It identifies the objectives of the breadboard, some potential vehicle related problems, and a simplified implementation concept.

The primary objective of the DACS breadboard is to verify the digital data bus concept for the Modular Space Station (MSS). It shall demonstrate the availability of technology to provide accurate data transfer, reconfigurability, failure tolerance, long life and standardization of interfaces.

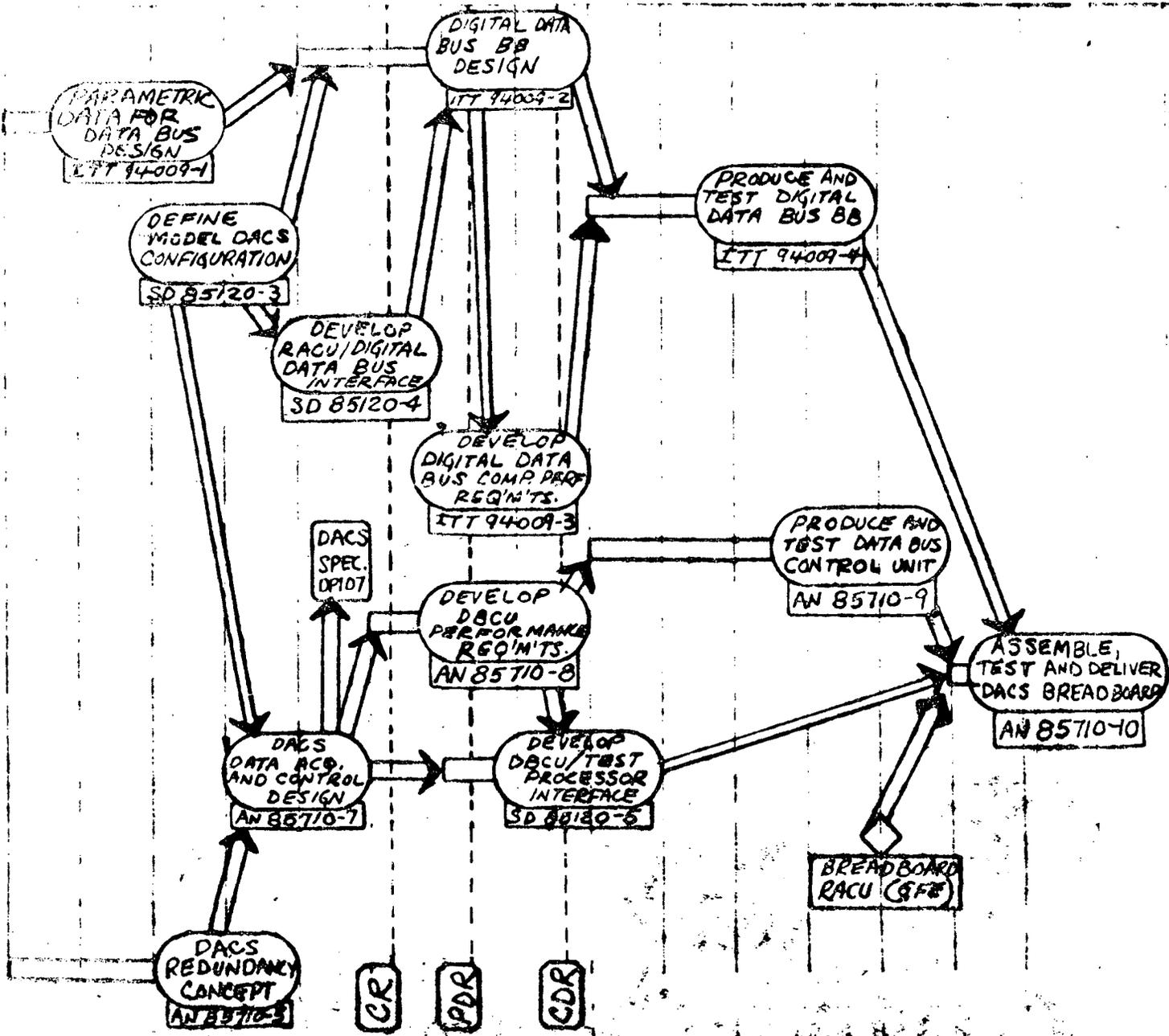


Figure 1-1. DACS Work Breakdown and Flow  
1-2

SD 72-SA-0114-3

The analysis of DACS redundancy concepts covers the advantages and disadvantages of a general range of concepts and methods applicable to the DACS. Recommendations of methods were made and justified.

The overriding requirements were found to be the single and triple failure tolerance requirements and the physical separation of redundant subsystems into pressure isolatable volumes.

The recommendations for the degree, level and type of redundancy for each DACS element are presented. These recommendations include the split between hardware and/or software techniques, the utilization of error protection coding, the replacement/repair methods and a definition of the replaceable items, and the rationale for each selected candidate DACS element.

The following redundancy requirements were imposed on the station subsystems

1. A capability must be provided for each non-critical function to fail safe for the first failure.
2. For a critical function a capability must be provided for:
  - A. Full operation subsequent to a first failure (fail operational)
  - B. Reduced or "out of spec" performance subsequent to a second failure (fail degrade).
  - C. Crew survival for at least 96 hours subsequent to a third failure (fail emergency)
3. Time critical functions require active (on-line continuous operation) redundancy. Non-time critical functions require at least standby (wired in and can be placed in operation with automatic or manual switchover).

Figure 1-2 presents the recommended implementation of the DACS so that it will satisfy the failure criteria (redundancy requirements).

## 1.2 DACS BREADBOARD DESIGN

The DACS breadboard is an engineering model that is representative of the concepts for the data acquisition and control function of the data processing assembly of the modular space station. The NASA defines a breadboard as a unit which performs the same functions and according to the same characteristics as those defined by the hardware design.

A data acquisition and control subsystem is a semiautonomous subsystem that provides controlled communication between a large number of remote locations and a control location. Insofar as possible, the DACS breadboard is a representation of such a subsystem. The DACS breadboard also provides a test bed for operational performance evaluation of numerous concepts for this type of subsystem oriented toward the specific needs and requirements imposed by the modular space station.

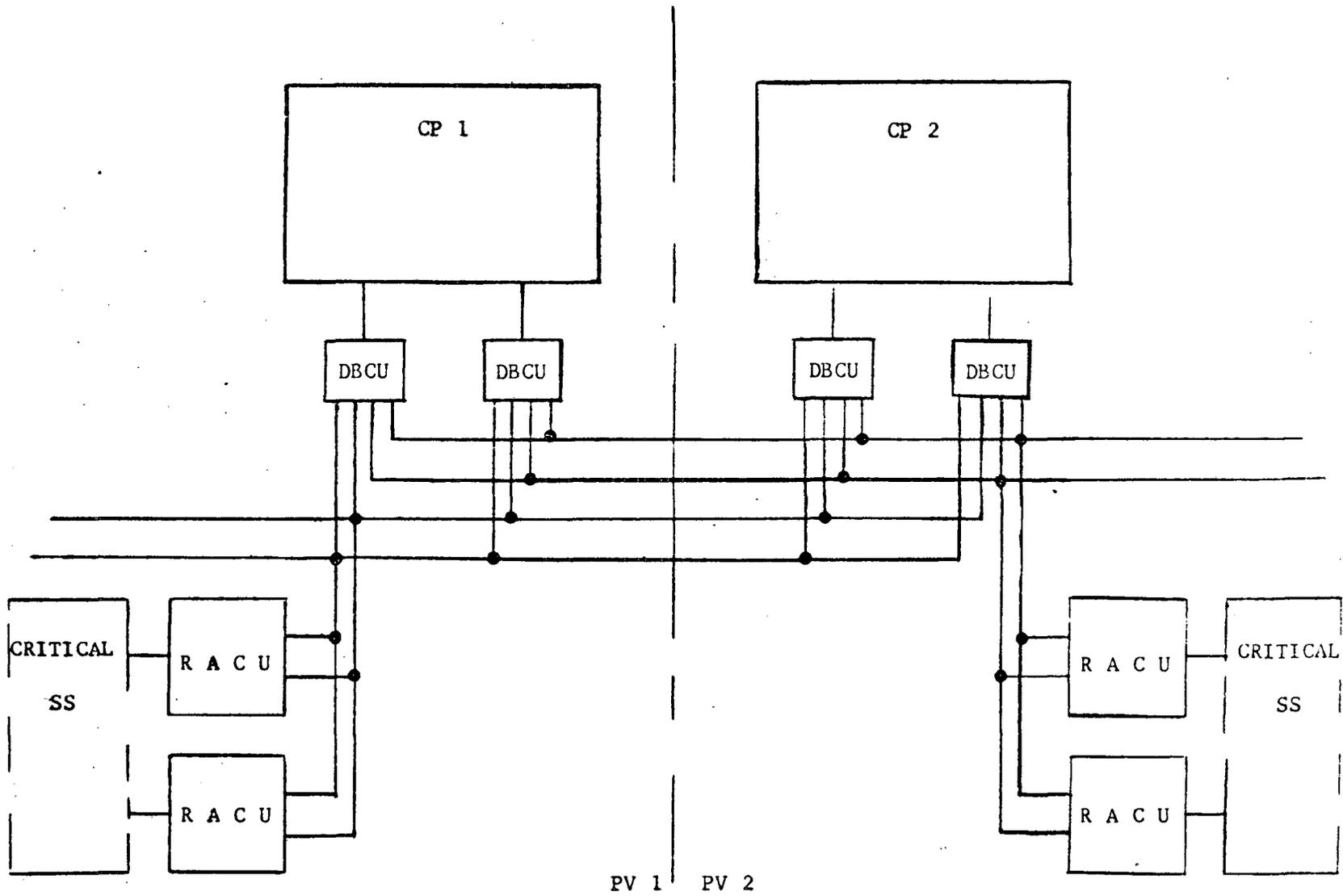


Figure 1-2. Recommended DACS Configuration

The overall breadboard concept for the DACS is a highly flexible, configuration independent, building block approach. This approach allows a large number of different DACS configurations to be assembled as an operating data acquisition and control subsystem breadboard. Each configuration concept can then be operated, tested and evaluated for overall DACS concept and performance valuation.

The DACS breadboard is an engineering model but is representative of the concepts for the data acquisition and control subassembly for the data processing assembly of the modular space station.

The communication spine for the DACS breadboard is provided by the data bus breadboard. All communication between the breadboard remote acquisition and control units (RACU's) and the data bus control unit (DBCU) utilize the data bus breadboard. This communication is also controlled by these other DACS breadboard units.

The data bus controller unit (DBCU) is one item of the major elements which make-up the modular space station (MSS) data processing assembly. The functional relationships of the various elements of the DACS breadboard equipments are shown in Figure 1-3. Direct interfacing elements with the breadboard DBCU include the test processor, test panel and special test equipment, MODEM, and prime power source.

The DBCU breadboard performs as an input/output device for the test processor, controls the information flow on the data bus, and performs the following functions:

- a. Provides all command and control capabilities to fully exercise the DACS breadboard, to communicate with the RACU breadboards via the breadboard MODEM units and data bus, and to operate the breadboard with or without the use of the test processor.
- b. Provides the capabilities to initiate all read/write actions with the test processor or test panel interfaces.
- c. Provides the buffering and formatting of all input/output data to the test processor, test panel, or data bus interfaces.
- d. Provides the capability for error protective encoding and checking of all data to and from the test processor or data bus as selected by the test panel.

Figure 1-3 indicates all of the components of the DACS breadboard, however, not all of those shown are part of the ADT effort. The others are GFE. Note in particular, that there are enough DDB components to configure a dual redundant data bus. It will be possible with this breadboard, then, to evaluate wideband digital data bus operation (10 Mbps), automatic fault detection and isolation, automatic reconfiguration, techniques for executive control of data traffic, etc.

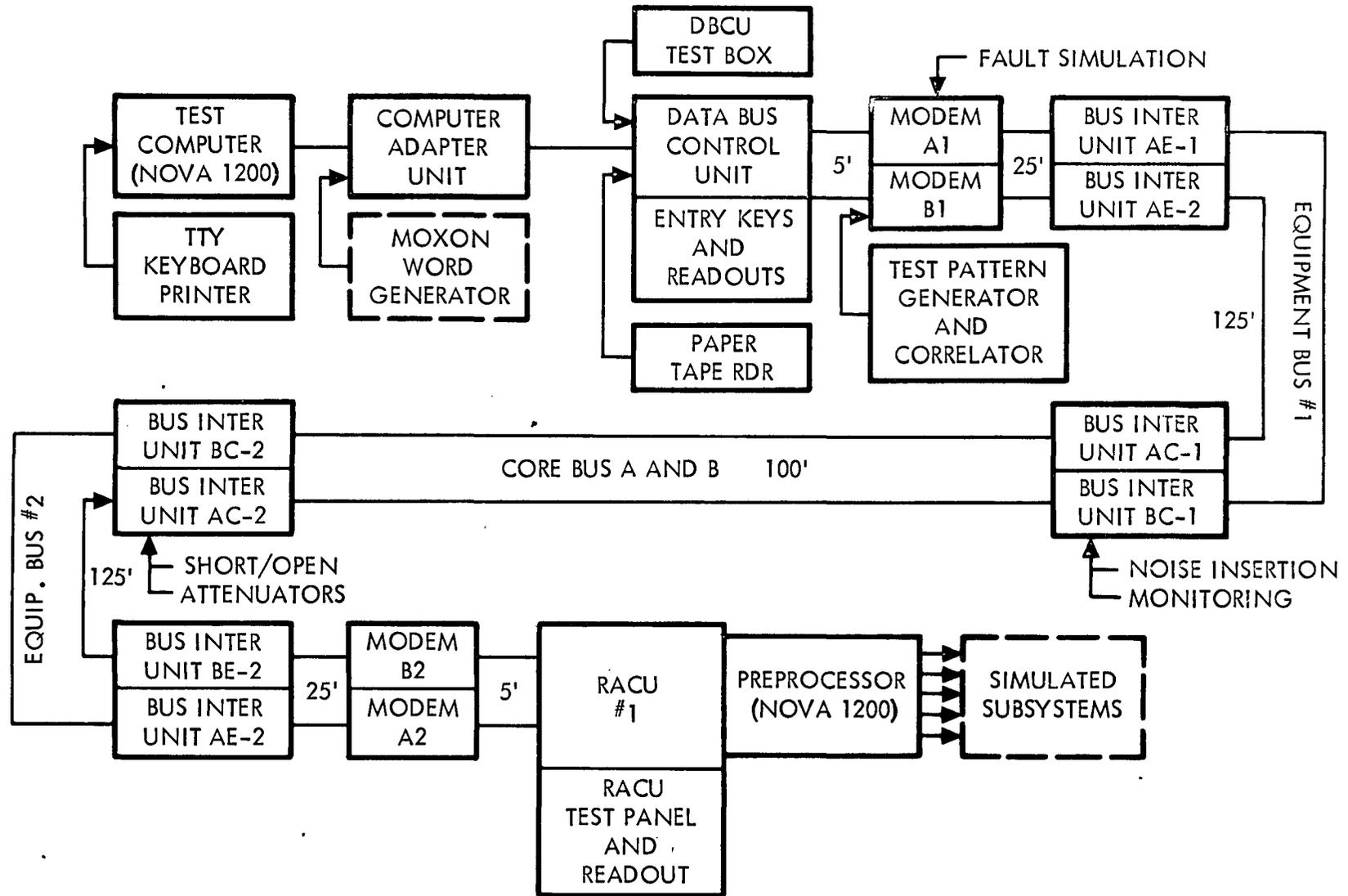


Figure 1-3. Data Acquisition and Control Subassembly Breadboard Configuration



The requirements for the DACS breadboard itself and for its components, in particular, the DBCU breadboard and the data bus breadboard, are presented in specification format, Sections 5.0, 6.0, and 7.0. For this reason, the reader will notice some repetition in these sections. Furthermore, the requirements definition process was iterative and dependent upon the results of the supporting studies (Sections 2.0, 3.0, and 4.0), as well as the results of the simultaneous MSS Phase B studies. Hence, there may appear to be discrepancies in some of the numerical parameters used in specifying the breadboard requirements. For example, one parameter which varied with the progress of the Phase B study was the line lengths imposed by the modular configuration.

## 2.0 PARAMETRIC DATA FOR BUS DESIGN

The data acquisition and control analyses began with a theoretical analysis of the parameters pertinent to the design and usage of data buses. The result of this analysis was a "design handbook" covering the significant aspect of wideband digital and analog data buses.

### 2.1 DIGITAL BUS

Two basic digital bus configurations, outlined below, were analyzed although within both there are variations.

The first configuration shown in simplified form in Figure 2-1 consists of a core bus located in the core and a module bus located in each module. Both buses are bi-directional, therefore requiring bus enabling signals for time multiplexing digital signals and preventing ringing in bus coupling drivers and receivers.

The amplifiers are located at the module-core interface for isolation, and circuit drive requirements.

The second configuration shown in simplified form in Figure 2-2 retains the bi-directional core bus and provides directional module buses. This approach has the possibility of:

- . Eliminating the module bus enable signals, thus saving equipment
- . Improving the reliability of the system by eliminating the enable signal and by spreading the equipment loads over two buses instead of one
- . Eliminating the necessity of designing an enable bus control
- . Providing a reduction in spare parts
- . Costing the same as the other approach

The approach to both basic configurations were selected because they:

- . Allow checkout operations of individual modules
- . Allow the MSS to go through a buildup sequence without bus modification
- . Allow more reliable operations because the core bus is passive and the module buses are independent of each other.

The above approach is known as a spreaded star configuration due to the core length. A circular configuration with the bus looped in and out of each

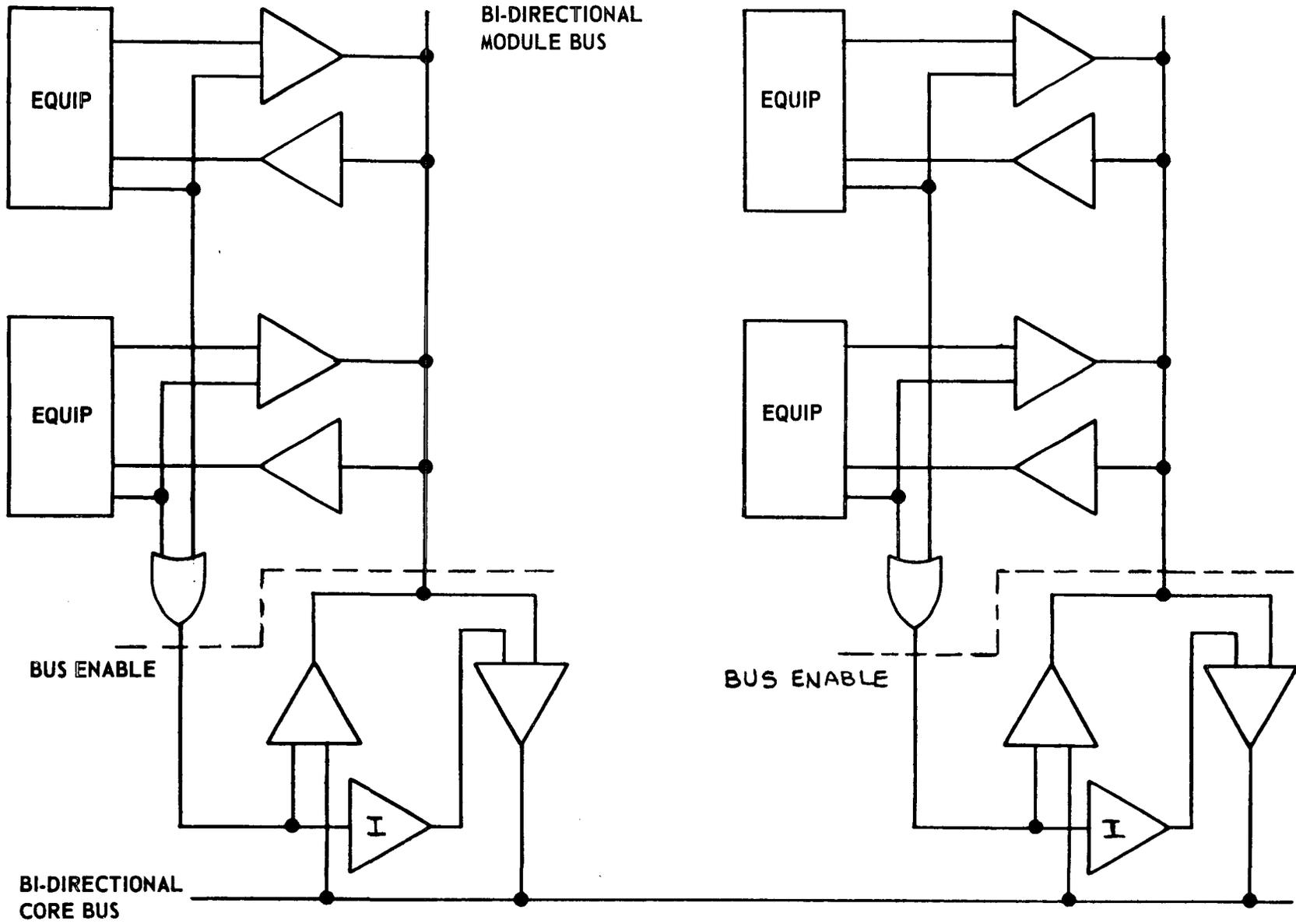


Figure 2-1. Digital Data Bus, First Configuration

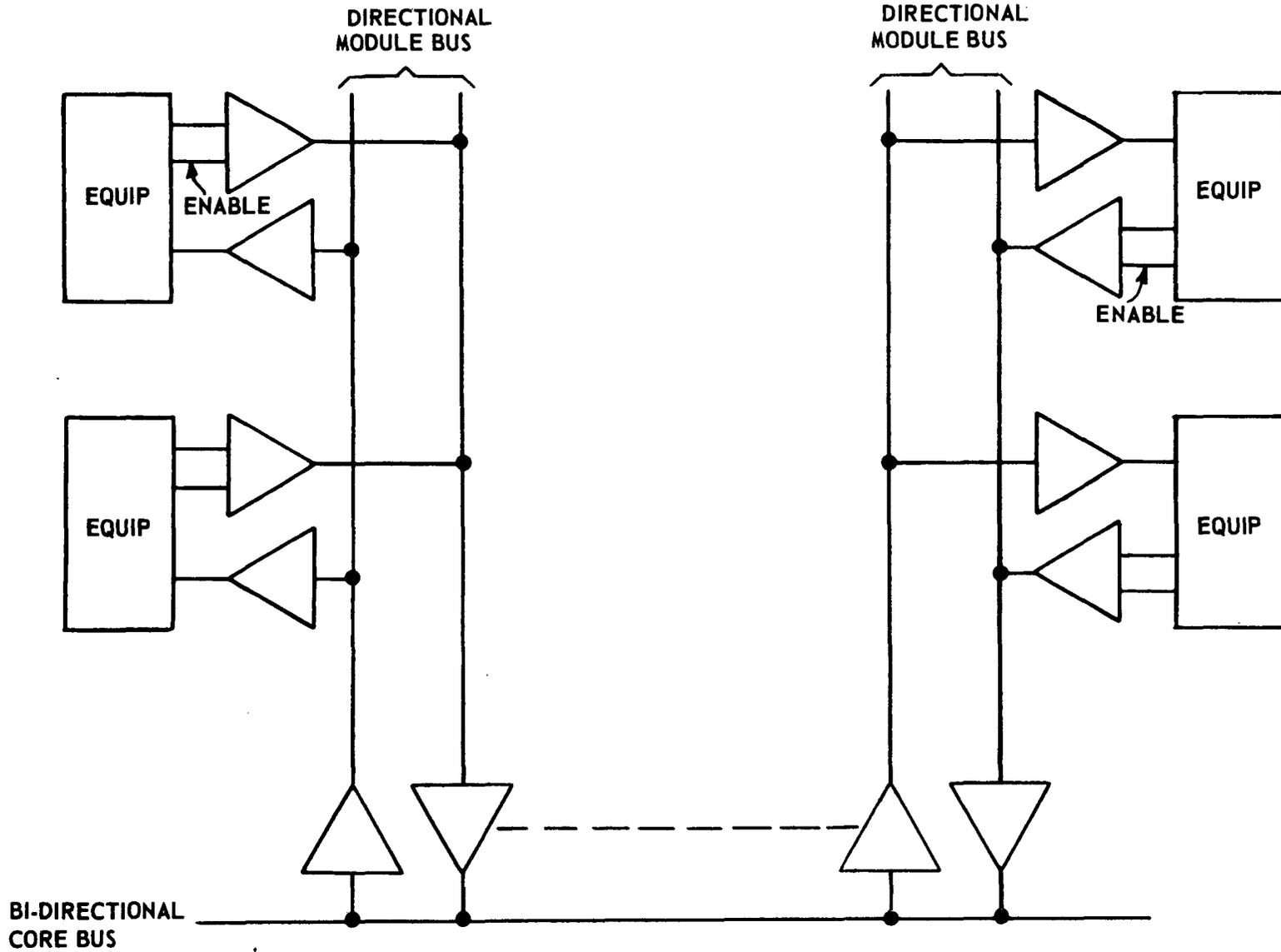


Figure 2-2. Digital Data Bus, Second Configuration

module was examined but was discarded because of the problems it presented in reliability buildup sequence and module self check.

## 2.2 ANALOG BUS

The analog bus is different from the digital bus in that the former is FDM and the latter is TDM. For this reason, the audio could best employ the configuration shown in Figure 2-2, except that there is no equipment enable (not required for FDM).

## 2.3 COUPLING CONFIGURATION

For digital and analog buses, both direct and transformer coupling techniques with surge protection were considered to interface the busses with their terminal devices and to interconnect sections of the bus.

### 2.3.1 Digital Bus

The following were investigated for the digital bus (see Figure 2-3):

- . Direct coupling using off-the-shelf IC's
- . Transformer coupling using H-pad with shunt transformer
- . Transformer coupling using series transformer without padding

Two other methods are apparent and dual to choices two and three, and they are:

- . Transformer coupling using O-pad with series transformer
- . Transformer coupling using shunt transformer without padding

The latter two methods were not examined because the padded series transformer is a dual of the padded shunt transformer and doesn't require a separate analysis and because, for the unpadded shunt transformer, sufficiently high impedance cannot be realized at 10 mbs. The data rate above which the unpadded shunt transformer becomes unrealizable lies somewhere between 10 and 100 kbs.

### 2.3.2 Analog Bus

The following were investigated for the analog bus (see Figure 2-4):

- . Direct coupling using T-pads
- . Transformer coupling using T-pads with shunt transformer

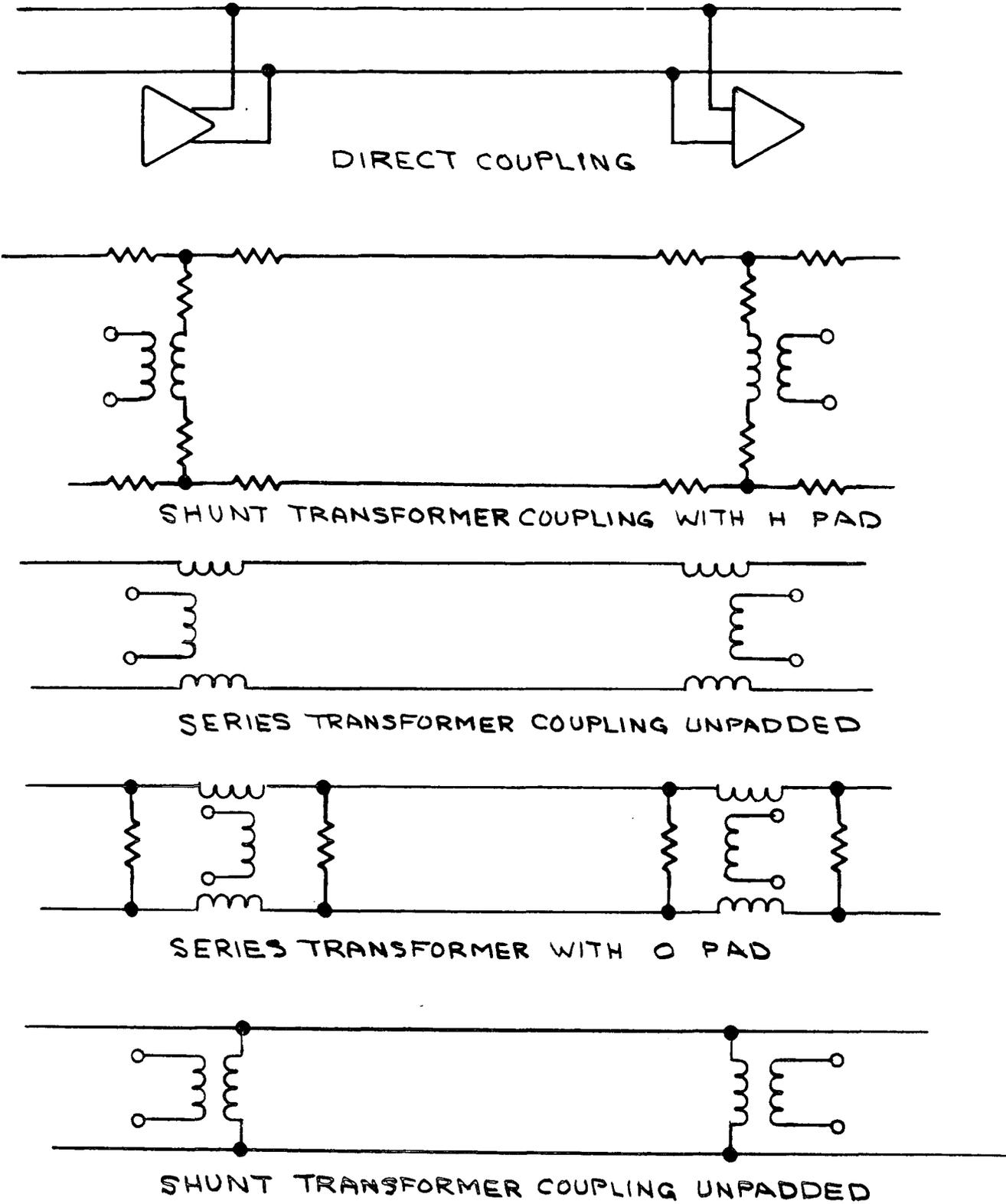


Figure 2-3. Methods of Coupling

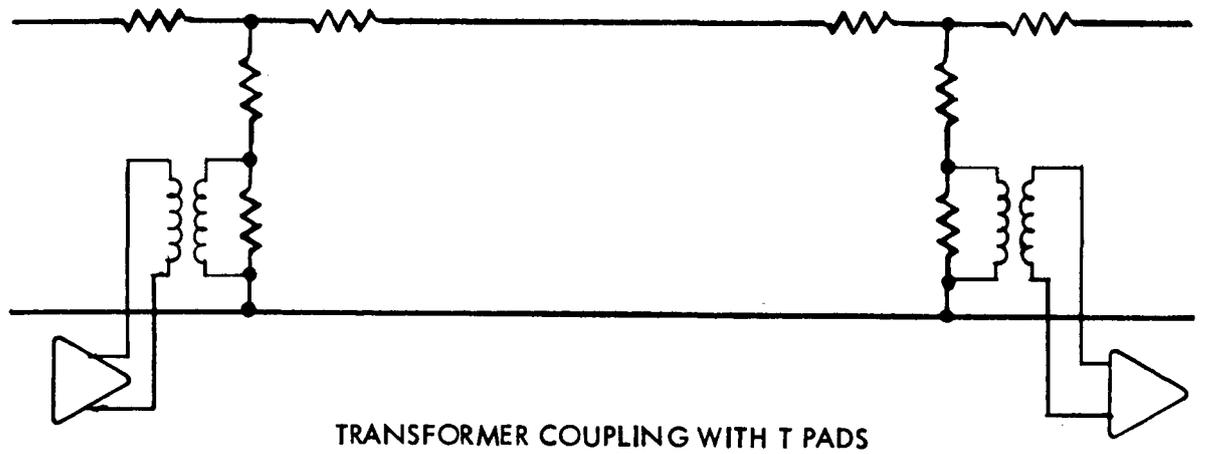
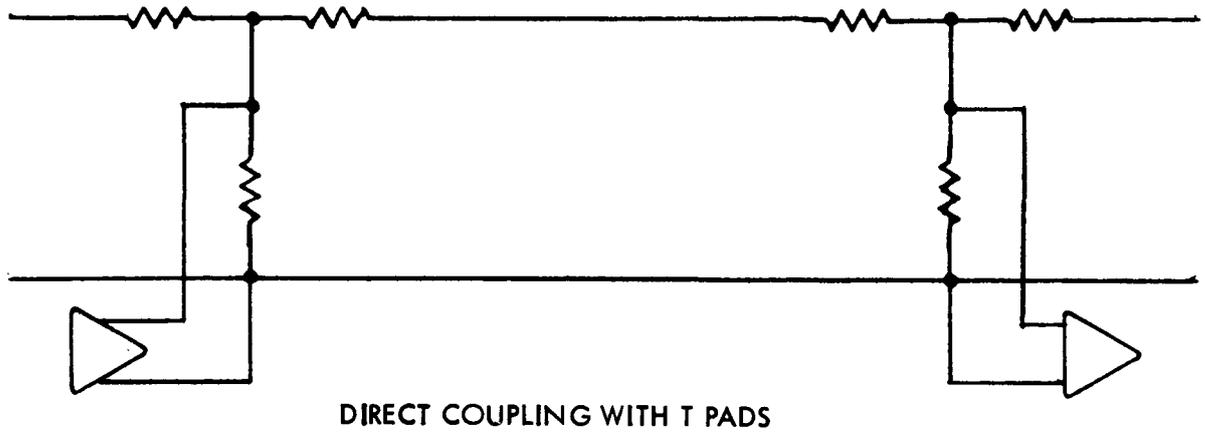


Figure 2-4. Direct and Transformer Coupling with T-Pads

## 2.4 CONCLUSIONS AND RECOMMENDATIONS

The conclusions are as follows:

**Configuration** - The digital and audio bus should employ the same configuration: that is, the two-bus concept with bidirectional core bus and directional module buses (See Figure 2-2).

**Coupling** - The digital bus should employ transformer coupling: that is, the H-pad with shunt transformer. The analog bus should employ transformer coupling with a shunt transformer in a T-pad.

### 3.0 DACS MODEL CONFIGURATION

#### 3.1 INTRODUCTION AND SUMMARY

The initial effort of the DACS BB study was to define a model for the Data Acquisition and Control Subassembly (DACS) breadboard. The purpose of this definition was to provide data to serve as a basis for the design studies of a DACS breadboard. This data was used, primarily as an input to WBS 85710-7, DACS Model Design, and WBS 94009-2, Data Bus Breadboard Design.

This section presents the recommended DACS breadboard system definition. It identifies the objectives of the breadboard, some potential vehicle related design problems, and a simplified implementation concept.

#### 3.2 RECOMMENDED DACS BREADBOARD CONFIGURATION

##### 3.2.1 Objective of DACS Breadboard

The primary objective of the DACS breadboard is to verify the digital data bus concept for the modular space station (MSS). It shall be used to determine the availability of technology to provide accuracy of data transfer, reconfigurability, failure and tolerance, long useful life and standardization of interfaces. The following technology goals have been identified for the DACS breadboard; they are listed in descending order of priority.

1. Automatic line fault detection and isolation to a faulty connection, wire breakage, etc.

Automatic line fault detection is mainly an operation inherent in DBCU operation. The inability to communicate with one or a set of RACU's provides the detection. This item is reported by the breadboard DBCU to the test processor/panel. Line fault isolation requires data processing on numerous DBCU, indications of trouble to determine the fault location. This is mainly an operation external to the DBCU and, therefore, the DACS breadboard will utilize the test processor for whatever degree of fault isolation is required.

2. Automatic failure detection of DACS electronic hardware

Automatic failure detection of DACS hardware can sometimes be performed by the DACS itself. In a non-redundant configuration, however, this possibility wanes. The DBCU can report failures to communicate with RACU's and provide some amount of data useful in this determination. This is also a function of having more than one RACU on the same bus section to differentiate RACU fault from that of a bus section. The DBCU cannot, in general, report its own failure, nor does it have the data processing

capability for more than a gross level of failure detection (to multiple causes and equipments, for example). Thus the test processor is the logical location (as is the CP in the real DPA) for this final fault detection of DACS elements. The DBCU can and will report all fault indications to the test processor/panel interface. Fault simulation is a requirement for all DACS breadboard units to enable testing this process.

3. Automatic reconfiguration, within microseconds, to continue operations

Reconfiguration, in microseconds or longer, requires equipment to be bypassed and, therefore, redundancy. This requires multiple breadboard units on a scale not yet contracted for by the NASA. It can be simulated in a gross sense with a non-redundant breadboard by DBCU/test panel indications; i.e., indicating a need to reconfigure at the proper time. These are the necessary first steps of the process. The breadboard operation is such that reconfiguration subroutines can be included in the DBCU memory if and when breadboard hardware is available to demonstrate it. At this time, an evaluation can be made. Time to reconfigure is dependent on the number of items in the system, and since this will always be less than the actual implementation, this must essentially be determined external to the breadboard.

4. Techniques for executive control of data traffic within the system

This goal is uppermost in the operational specifications for the breadboard units. A programmable DBCU is provided. The RACU breadboard is also flexible in terms of operational sequences and modes. A real-time interaction of the DBCU with a test processor/panel is also specified. This, too, can be used to evaluate executive control techniques as required.

5. Evaluation user-subsystem servicing techniques; i.e., polling, request/acknowledgement, etc.

This is similar to the discussion about goal 4. All of these items are provided in the requirements specification. Polling will be provided via the data bus in the breadboard by a specific DBCU routine and in conjunction with some of the normal data transfer operations.

6. Techniques for implementing several modes of operation concurrently

The DBCU will have the capability of operating in several modes. Full duplex operation of the data bus will also be available in the appropriate configurations. Simultaneous use of the data bus breadboard by the breadboard DBCU will be possible utilizing the external test interface. The DBCU by itself will not have the internal hardware for complete simultaneous operation. The use of a redundant breadboard with two DBCU's can easily demonstrate this operation.

7. Develop performance and interface constraints and design "rules" for production systems

Utilizing goals 4, 5, and 6, the breadboard can be very useful in this respect. The wide variation of operation will allow evaluation of all these items to the extent they are testable. Data collection for this evaluation will, of course, need to be performed external to the DACS breadboard.

8. Determine the optimum allocation of functions within the several components of the system

This is very difficult and costly goal for the breadboard since it would require locating these functional items in all breadboard units where they might be allocated. It also would not have very positive results because of the simulation nature and simplification utilized in constructing a breadboard. This is, however, specified in the breadboard for one very important function, that of error protection. Its allocation is such that error protective encoding and decoding can take place both internal to the breadboard units as well as by external devices through the external test interface. Some of the DBCU/test processor interaction also has this goal in mind. It is best done by analysis of the performance data collected on DACS breadboard operation external to the laboratory.

Note: Usage of this feature is hampered, at least, by the fact that the RACU breadboard, when in the non-BCH mode, still expects the timing to be unchanged from that of the BCH mode (see 5.2.1.4.2).

9. Two-way redundant transfer of digitally-coded signals at rates up to  $3 \times 10^7$  bits per second over distances up to 1000 feet

The DACS breadboard is presently specified to demonstrate two-way transfer of digitally-coded signals at rates up to  $1 \times 10^7$  bits per second over distances up to 400 feet. Redundant data transfer at these rates only requires additional breadboard units for implementation. This goal is evaluated by external monitoring.

10. Techniques for optimizing the efficiency of data transfers over the bus cable

The provisions in the DACS breadboard for technology goals 4, 5, and 6 provide the means to operate the breadboard and vary the required parameters. External data collection, analysis and evaluation is necessary to determine the data transfer efficiency measure and the DACS operations which optimize this measure.

11. Evaluate signal coding

Hardware signal coding for error protection is provided in the DACS breadboard. The results of the various possible encoding algorithms for various bus operations and external effects must be recorded, analyzed and evaluated external to the breadboard. Accumulation of error statistics requires, perhaps, test processor interaction which is available. Also, see note under goal 8.

12. Evaluate feasibility of standard intercoupling units (RACU's) to other equipments; develop a "family" of SIU devices

Some evaluation is possible and the development requirements are beyond the scope of this specification. Breadboard equipment, in general, provides little in the way of feasibility evaluation for standards. Since most items such as external interfaces are simulations of expected hardware, the situation is already idealized. The feasibility of standard interfaces always has its most serious test when applied to actual hardware, and this can be pursued in the breadboard by providing "real" subsystem hardware.

13. Evaluate methods of detection and/or recovery of errors in transmitted data (noise immunity)

This goal is included in goals 9, 10, and 11. It requires operation of the breadboard over long periods of time and through many variations and conditions. As stated in the discussion of goal 11, external recording and evaluation is a necessity.

14. Evaluate DACS redundancy concepts

To the extent that some of the DACS redundancy concepts are included in the breadboard units, such as alternate paths and operational modes, external evaluation can be performed. This will only be a gross evaluation without utilizing redundant breadboard equipments. Much can be done by external evaluation of results, derived from the non-redundant breadboard configurations, on paper, and in conjunction with recorded error statistics, etc. Also, proper design of breadboard tests and test sequences can enhance the capability of the breadboard to provide the necessary data.

15. Evaluate the reliability and user acceptability of "control-by-wire" techniques

The reliability of operation, through results recorded externally, such as the number of incorrect data outputs or commands over a given period of time and operating rules, will allow some evaluation of this attribute. User acceptability is not necessarily enhanced by viewing breadboard demonstrations.

16. Evaluate performance limits of each element (RACU, DBCU, data bus assembly)

This goal is the easiest of the list to provide via the DACS breadboard. There are an astronomical number of possible "runs" which can be performed by the breadboard. The number of variables in the breadboard system are almost limitless. Attention to proper design of the breadboard performance tests can reduce this to a more manageable figure with just as reliable results. Almost all performance evaluation requires external data collection and "off-line" analysis.

Each technology goal has been discussed in terms of the breadboard DACS provisions relating to the goal. A brief rationale was given for the level at which each of these goals are incorporated into the design requirements for the DACS breadboard. The major constraints on satisfying the technology goals is one of cost. Redundant configurations, although considered in the DACS breadboard requirements, necessitate considerably more breadboard units than are now to be provided. The breadboard DACS can expand, at a future time, to satisfy most of these technology goals. And by proper design of breadboard tests, much of the need for additional breadboard units can be obviated.

### 3.2.2 Potential Vehicle Design Problems

There exists a broad range of vehicle and subsystem configurations and the final choice of configuration could well be anywhere in this range; therefore, the DACS breadboard must be sufficiently flexible to simulate the DACS in whatever physical configuration may be selected for the MSS.

### 3.2.3 The DACS Model

The DACS breadboard shall contain at least one of every piece of equipment needed in the DACS plus sufficient cable runs and junction points to demonstrate the performance of the DACS in different configurations. As a minimum, the model shall contain a DBCU or DBCU simulator, a RACU or RACU simulator, and a digital data bus breadboard.

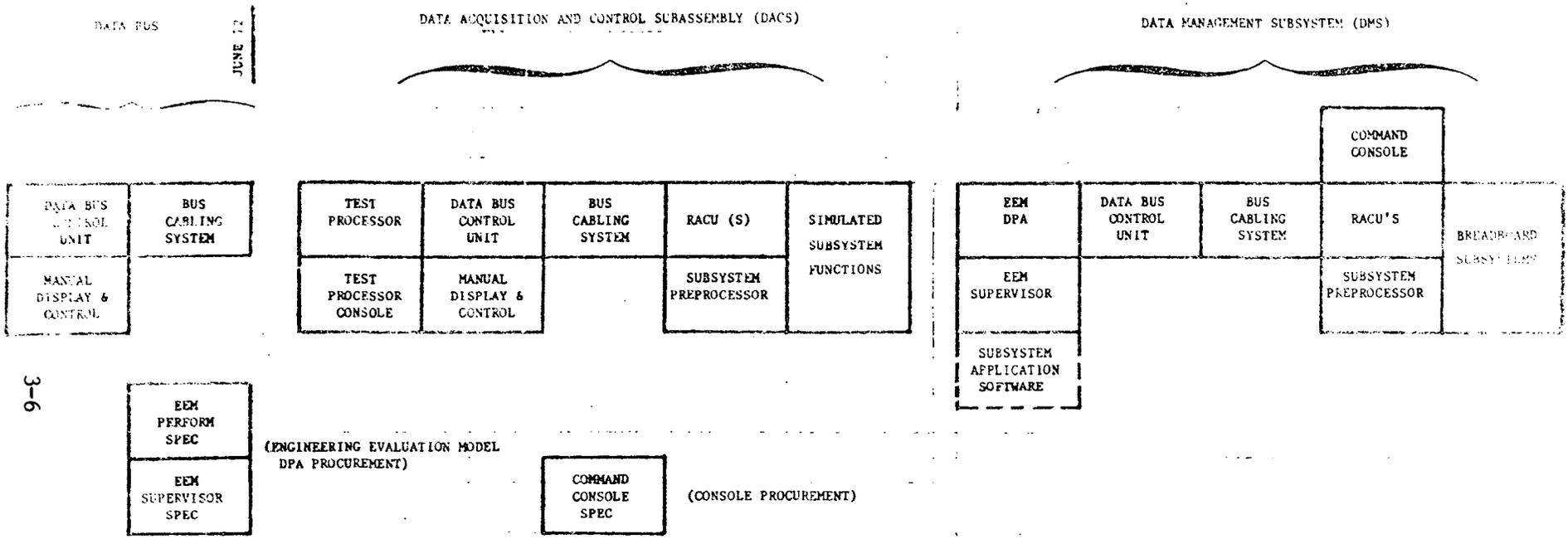
The digital data bus breadboard should consist of enough cable segments to allow it to be connected into a number of configurations.

It is desired that the breadboard be capable of demonstrating both bilateral and unilateral data flow, and combinations of bilateral and unilateral. For example, one combination which should be demonstrated has a bilateral core bus and a unilateral module bus.

The DACS breadboard will ultimately become a part of a Data Management Subsystem (DMS) engineering evaluation model. The growth sequence is illustrated in Figure 3-1. The first phase of the growth sequence will encompass the following hardware:

Digital data bus breadboard - ITT  
Demonstration support equipment for above - ITT

Reproduced from  
best available copy.



3-6

SD 72-SA-0114-3

Figure 3-1. Data Management Subsystem, EEM Growth Sequence

Data bus control unit breadboard - Autonetics  
 Demonstration support equipment for DACS breadboard (other  
 than standard laboratory equipment) - Autonetics

In addition to this hardware, RACU's or RACU simulators will be supplied (by NASA) and integrated with the above hardware to form an integrated DACS breadboard.

### 3.2.4 Performance Parameters

The following performance parameters have been identified as performance parameters for the DACS. The DACS breadboard should demonstrate satisfactory performance within the indicated range for each parameter.

| Parameter  | Range             |
|--|-------------------|
| Probability of fault detection                                   | >0.98             |
| Time to detect, isolate, and reconfigure                         | <4 milliseconds   |
| Number of simultaneous service requests<br>(assume <1 fault/sec) | <10               |
| Data transfer rate   | 1 MBPS to 30 MPBS |
| Maximum length of path   | <1200 feet        |
| Maximum density of taps  | <10 per foot      |

Many of the design parameters and breadboard objectives have been modified or deleted during the course of development. Most notable was the decision to build a data bus optimized to operate at 10 MBPS. A point design was required by the ringing filter used for clock recovery. The 10 MPBS rate was based on the DPA requirements analysis (see Volume IV of this report).

## 4.0 DACS REDUNDANCY CONCEPTS

### 4.1 INTRODUCTION AND SUMMARY OF REDUNDANCY REQUIREMENTS

This section covers the advantages and disadvantages of a general range of concepts and methods applicable to the DACS. Recommendations of methods are made and justified.

The DACS is composed of three major elements. These are the remote acquisition and control units (RACU's), the digital data bus control unit (DBCUC), and the digital data bus assembly. The DACS is a part of the central processor subassembly of the Data Processing Assembly (DPA).

The overriding requirements were found to be the single and triple failure tolerance requirements and the physical separation of redundant subsystems into pressure isolatable volumes.

Redundancy concepts and reliability enhancement techniques include parallel and serial redundancy, data protection and failure detection, isolation and repair and test concepts. The recommendations for the degree, level and type of redundancy for each DACS element are presented in this section. These recommendations include the split between hardware and/or software techniques, the utilization of error protection coding, the replacement/repair methods, and a definition of the replaceable items, and the rationale for each selected candidate DACS element.

The following redundancy requirements were imposed on the modular space station subsystems:

1. A capability must be provided for each non-critical function to fail safe for the first failure.
2. For a critical function a capability must be provided for:
  - a. Full operation subsequent to a first failure (fail operational)
  - b. Reduced or "out of spec" performance subsequent to a second failure (fail degrade)
  - c. Crew survival for at least 96 hours subsequent to a third failure (fail emergency)
3. Time critical functions require active (on-line continuous operation) redundancy. Non-time critical functions require at least standby (wired in and can be placed in operation with automatic or manual switchover).



The analysis of methods of failure tolerance implementation begins with the Guidelines and Constraints imposed upon the modular space station, which is used to construct several applicable "loop" modules for each level of criticality. The models are then used to develop candidate mechanizations for each element of the data system (DACS); the data bus assembly, the RACU unit, and the data bus control unit. Recommendations for unit mechanization are made, then reassembled for a summation of the recommended configuration for DACS.

#### 4.2 SUBSYSTEM MODELS AND CONFIGURATIONS

There are three types of subsystem functional loops for which various models will be postulated and used to study the DACS redundancy criteria. They are non-critical, non-time critical, and time critical subsystem functional loops in order of increasing redundancy requirements. Previous discussions have generalized the redundancy requirements associated with each of these functions as fail-safe, requiring as a minimum standby redundancy, and requiring active redundancy in the same increasing order.

##### 4.2.1 Non-Critical Subsystem Functional Loops (Figure 4-1)

The models for non-critical subsystem functional loops can be of three types. The functional loop in its simplest form would be simplex in nature and only one such functional loop would exist on the space station. In general this model would not be fail-safe after the first failure (its own internal failure) since there would be no redundant mechanism to perform the "safe" function.

The second model of a non-critical loop would be two simplex loops. These two loops might be necessary for fail-safe capability, utility in performing the function in separate locations, constrained by physical relationship to other functional loops, or possible desirable maintenance or other considerations. In any of these events, such a model is possible and will be considered in providing the DACS interface.

A third model would be a dual-redundant loop for much the same reasons as the two-simplex loop model. This dual redundancy might or might not be the total loop. In all likelihood it would consist of enough redundancy to provide a true fail-safe (after the first failure) subsystem functional loop.

##### 4.2.2 Non-Time Critical Subsystem Functional Loops (Figure 4-2)

Again there are three basic models of non-time critical subsystem functional loops providing the necessary space station function performed by each loop. These models do not contain at this time any instruction as to the method by which they operate internally, except as implied by the model definition. Each model may or may not require higher level authority or intervention for its operation.

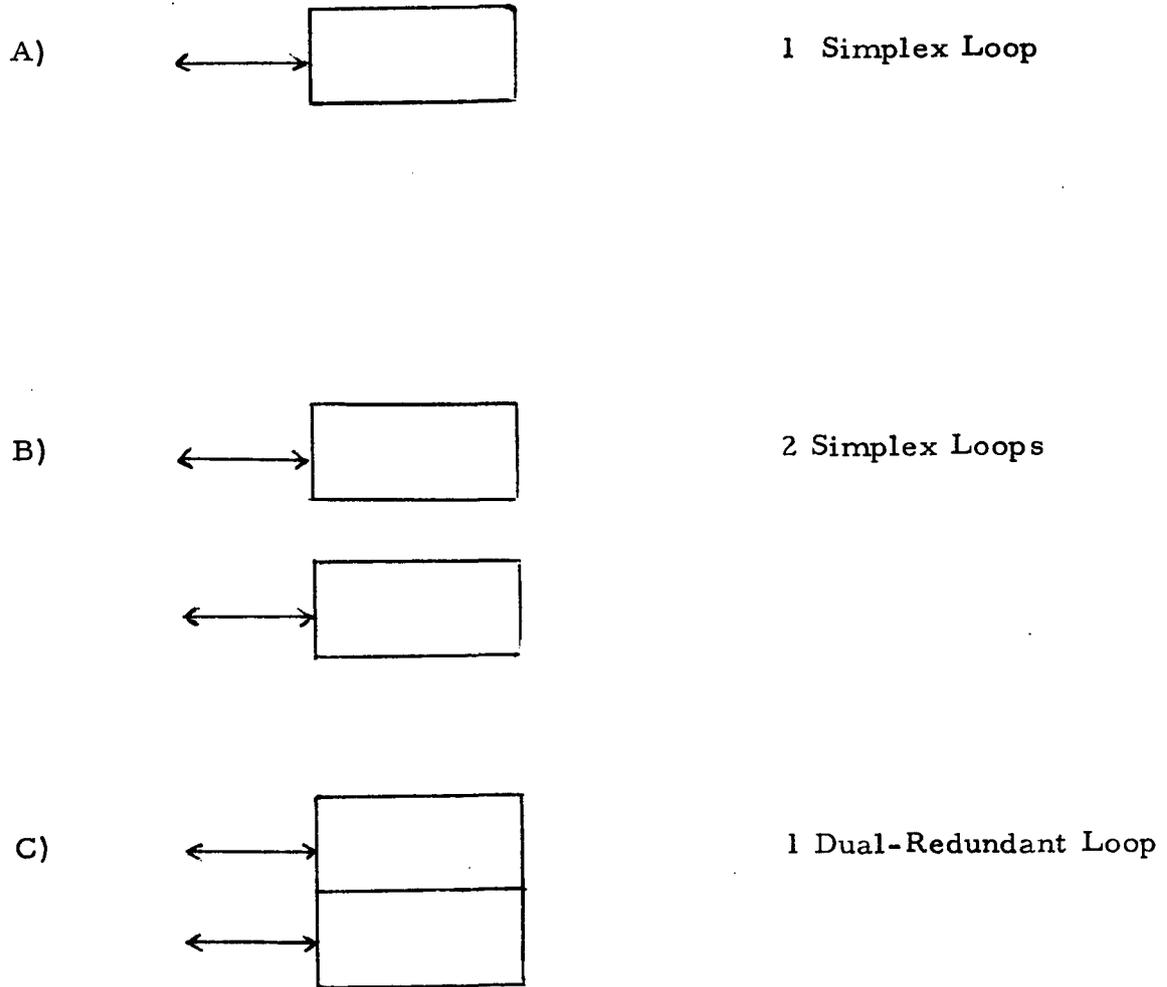


Figure 4-1. Non-critical Subsystem Functional Loop Models

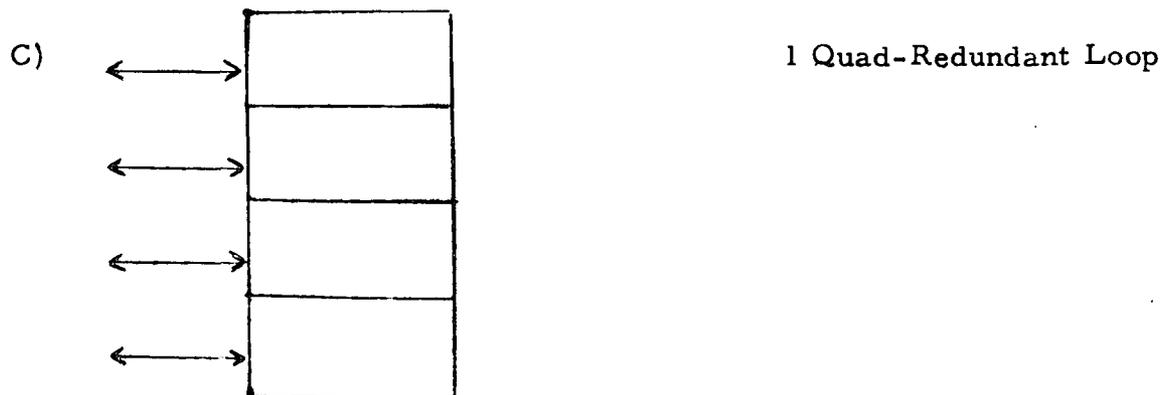
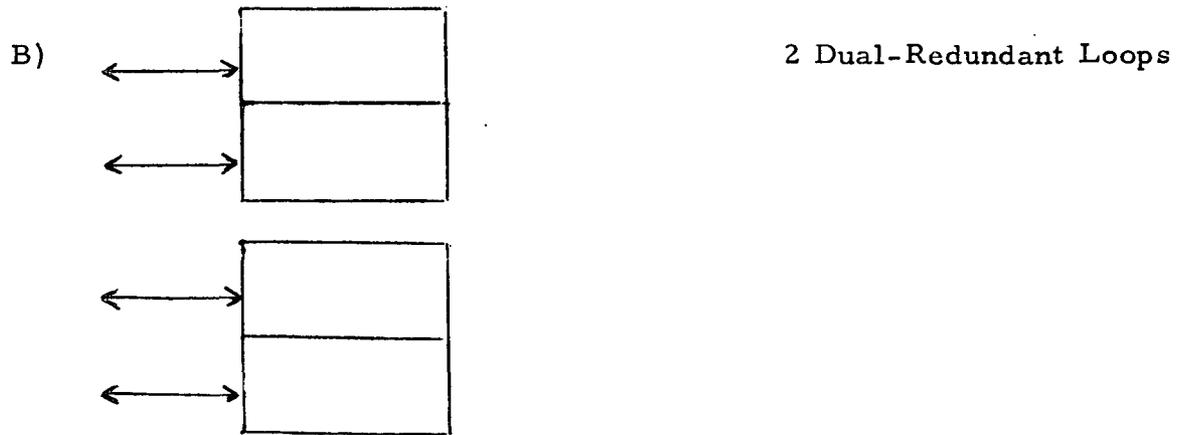
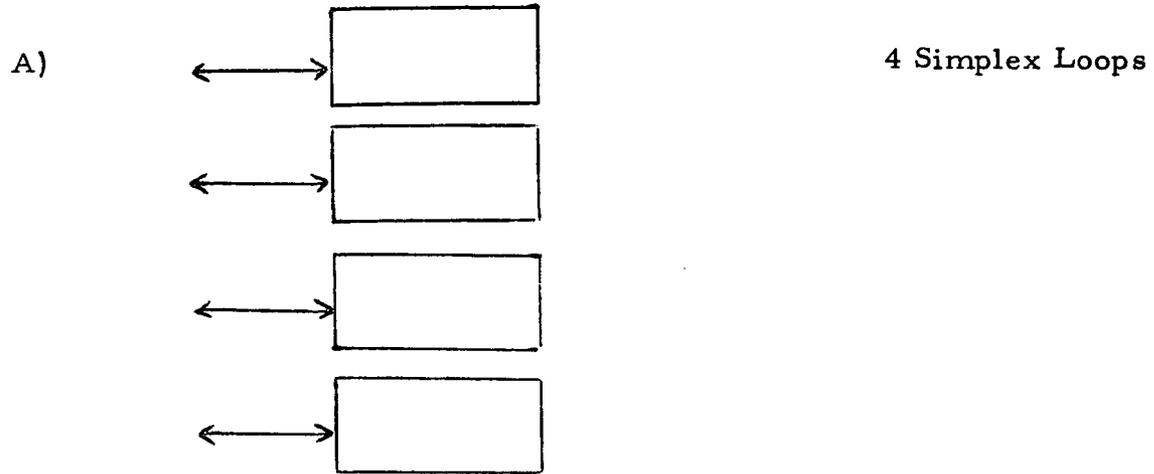


Figure 4-2. Non-Time Critical Subsystem Functional Loop Models

The first model is (as before) a simplex configuration. For non-time critical functional loops, four such simplex loops are required to meet the three failure tolerance criteria. One or more of the simplex loops may also have other functions, but are viewed in the modeling as that necessary to perform the one single non-time critical function as if no other loops existed.

The second non-time critical model would be two dual-redundant loops, each loop being single failure tolerant. This model is extremely important in light of the redundancy requirements and physical separation implied by pressure isolatable volumes. As in all models composed of multiple assemblies (containing similar or identical functional loops), the multiplicity assumes a physical separation is possible if desired. Each dual redundant loop may or may not contain simultaneously operating redundant elements.

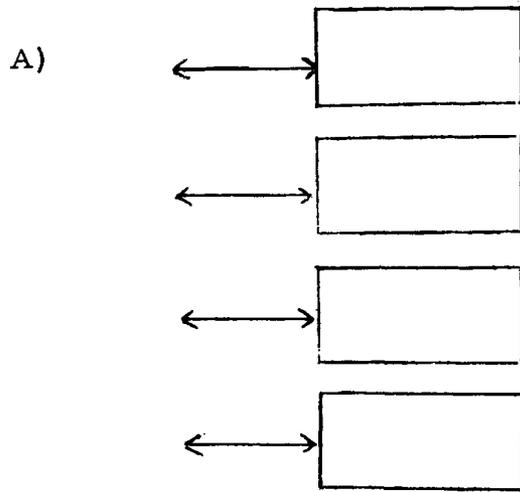
The third model for non-time critical loops would be one quad-redundant functional loop. The quad-redundant assembly has the same characteristic as all models in that its internal operation is as yet unspecified. As can readily be seen for all redundant loop models, the options of type of redundancy (active, standby, distributed, parallel, serial, etc.) is left open to reduce confusion and model proliferation.

One may note that there is no model that utilizes triple-redundancy. Due to the stringent failure criteria, such a model would require two triple-redundant loops for three-failure tolerance. This certainly is possible, although it exceeds the requirements, and may be the method by which single failure (and no more) tolerance is achieved. For the purposes of this study this case will be assumed included in the second model utilizing two dual-redundant loops, each loop having single failure tolerance. As stated before, no internal mechanization is assumed for these models and they might well require more the simply doubling all elements in the loop. In fact the terminology dual-redundant is only used to imply in a more recognizable fashion the minimum configuration necessary for single failure tolerance, and the terms "dual redundant" and "single failure tolerant" will be used interchangeably in this report.

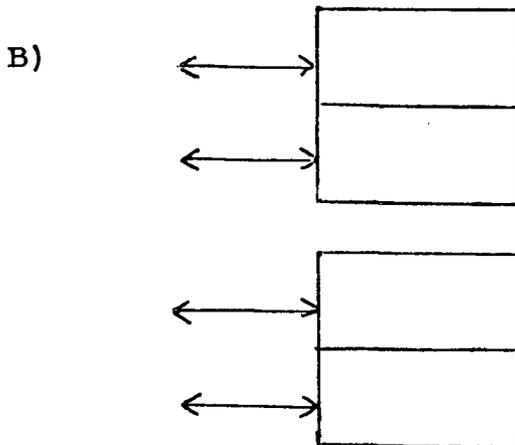
#### 4.2.3 Time Critical Subsystem Functional Loops (Figure 4-3)

These functional loops are, of course, the most critical on board the space station and the most costly in terms of DACS interaction. It is hoped that the number of such loops might be kept to a minimum, and as such they might not be the overriding design requirements, except in terms of failure tolerance. If they are few enough, it may be possible to over provide, in terms of redundancy, DACS elements more suited to the high usage loop models (non-critical models hopefully).

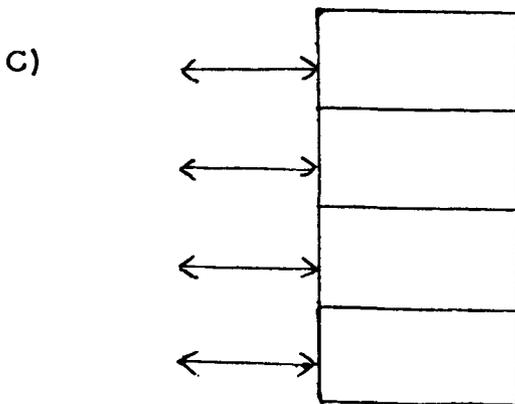
Three models for time-critical functions are again postulated similar in most respects to those for non-time critical functional loops. The differentiation between non-time and time-critical functions, in fact, is only one of response time and not of degree or level of redundancy, except in the option of more time to provide the redundancy. These three models for the time-critical functional loops are thus four simplex loops, two dual-redundant (single failure tolerant), and one quad-redundant loop.



4 Simplex Loops



2 Dual-Redundant Loops



1 Quad-Redundant Loop

Figure 4-3. Time-Critical Subsystem Functional Loop Models

## 4.3 DACS REDUNDANCY AND RELIABILITY CONCEPTS

### 4.3.1 General

The overall DACS must be three-failure tolerant since it is assumed to be a necessary part of the control interface for critical functional loops. The "brute force" approach to meeting this requirement would be a total quadruple parallel redundancy for all DACS elements. Any approach to be considered must also meet the other design constraints and requirements, such as physical separation between pressure volumes, physical separation in the routing of the DACS data bus assembly, and control by two computer assemblies in separate pressure volume locations.

Serial redundancy is also an ideal companion for parallel redundancy, and their combined usage has many advantages. In this area, as well as others throughout the application phase, some concept applications will not result in a net increase or decrease in the level of fault tolerance of the DACS. The major gains will be improvements in reliability at a given level of fault tolerance, or an enhancement in fault detection or isolation capabilities. The value of applying these types of concepts and techniques is much harder to assess and evaluate objectively, but relative comparisons of complexity and subsystem gains can be made. Included in this area are various data protection concepts and schemes. Their value for transient error protection is unquestioned, but this in itself offers no failure tolerance.

The DACS will also require fault detection and isolation for failure annunciation to the crews and subsystem reconfiguration if necessary. The level and number of these tests and the amount of built-in test equipment (BITE) is dependent on the level of failure isolation. Present requirements for fault isolation are to the in-flight replaceable unit (IFRU). It will be at this item level that repair (replacement) occurs after a failure indication.

Method of operational testing and checking of the status of all the DACS hardware elements must still be designed. This especially applies to redundant elements whose failure might be masked by automatic reconfiguration or some forms of active redundancy. The applications for this function will vary between internal checking and testing circuitry to use of the DPA and software techniques.

Each candidate application of a specific concept or concepts to a DACS hardware element must as a minimum satisfy the requirements, otherwise, it cannot be considered a valid application for the DACS. Each candidate application must also present a workable, physically realizable hardware element. No other constraints will be made on the definition of candidates for the DACS in the following subsections.

### 4.3.2 Data Bus Assembly Conceptual Candidates

#### 4.3.2.1 General

The data bus assembly for the DACS is the hardware assembly that ties the remaining DACS hardware elements together. The bus assembly must be highly distributed throughout the space station to provide the required ability for transfer of information from any location to the DPA and to any other location. Its usefulness is predicated on this distribution and the requirement to provide for the transfer of information between critical subsystem functional loops. The data bus assembly must be three-failure tolerant to satisfy the DPA fault tolerance criteria.

The highly distributed nature of the data bus assembly poses some unique aspects to this hardware element. Its distribution is by definition throughout all pressure isolatable volumes. The failure tolerance requirements are specified for equipments in relationship to their pressure isolatable volume location, and in any single location no more than single failure tolerance is required.

#### 4.3.2.2 Candidate DB-1

The first application approach, and the most straightforward concept would utilize parallel redundancy for the data bus assembly. The simplex data bus assembly necessary for full operational capability would be replicated four times as four independent data bus subassemblies. A simplex configuration is one which is non-single failure tolerant. Each of the four subassemblies would thus be capable of providing full operational bus requirements for the remaining DACS elements. A DACS element requiring three-failure tolerance would need to interface with all four data bus subassemblies.

These four subassemblies would be physically located throughout each module of the space station and in all pressure isolatable volumes. The physical routing of the cabling part of each subassembly would be separate from each other subassembly's cabling. Figure 4-4 is a graphical representation of this candidate, DB-1. A terminal (T) is the point of interface between the bus and the other elements of the DACS (i.e., the DBCU and the RACU). It would consist of a modulator and demodulator as well as the necessary amplifiers and bus coupling circuits. As indicated, a terminal can interface with either a RACU or a DBCU, and vice versa.

One simplification possible to this candidate would be to remove any portions of any individual subassembly that was unused. For example, suppose only subassemblies one and three in Module A are connected to equipments. Subassemblies two and four could then be eliminated from that module at a substantial hardware savings. Due to the distributed redundancy considerations and pressure volume constraints, this is a very real and possible result using this candidate.

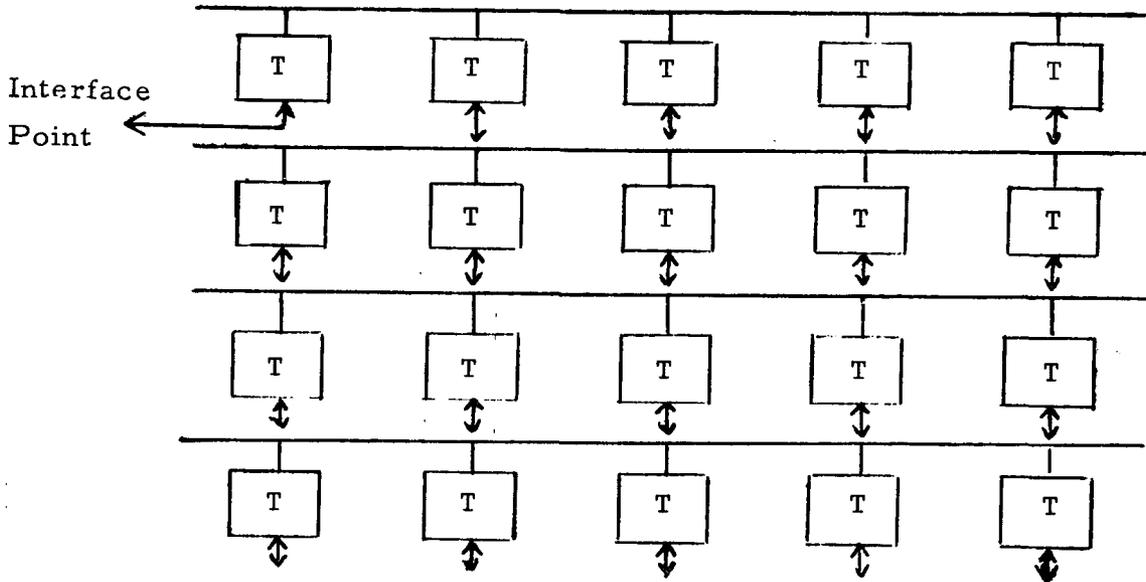


Figure 4-4. Candidate DB-1, Four Simplex Data Bus Subassemblies

Three options are available for candidate DB-1. Option A would have all four bus subassemblies operating at all times. Option B would only have three subassemblies operating and one non-powered and available as a standby spare. The third option, C, would have two bus subassemblies active and two subassemblies in standby. Provided the proper mounting and switching were available, these options would all satisfy the requirements.

Each of these options pose operational constraints on the DACS. Option A has the possibility discussed earlier of hardware reduction due to non-usage of portions of the bus subassemblies in some modules. Options B and C reduce the viability of this possibility, because any of the time-critical functional loop models would need to interface through the other DACS elements with all subassemblies in Option C and at least three subassemblies in Option B. This is necessary to provide three failure tolerance for the functional loop model for any possible failure conditions, either to the bus subassemblies or the loop model.

There is the possibility of going out of the standby mode of either Option B or C after any first failure in a critical function. Then a reduction in the interface can be made for all options, since after any first failure they all look like Option A. This then loses some of the advantages of the standby provisions especially when they are weighed against the increased switching (reconfiguration) requirements and testing imposed by this form of redundancy. The active redundancy portion of Option C also requires additional BITE or serial redundancy for its single failure tolerance operation for time critical functional loop models.

Candidate DB-1 imposes most of the decision making on the devices that interface with it. Any voting (Options A and B) or comparing (Option C) must be performed at the interface points since the subassemblies are simplex and independent. Any individual simplex subassembly should contain serial redundancy and/or data protection of some form and this would be available at the interface.

#### 4.3.2.3 Candidate DB-2

Figure 4-5 is a graphic representation of a second candidate redundancy concept for the data bus assembly. Candidate DB-2 is two dual-redundant (single failure tolerant) data bus subassemblies. Each dual redundant subassembly would be capable of providing total operational data transfer until its first failure. After a first failure they could individually provide a degraded flow (both together still providing total operational capability) such that with a single failure to both, all critical functions would still be available. With one subassembly completely failed, and one failure in the second subassembly, the capability would be the emergency mode.

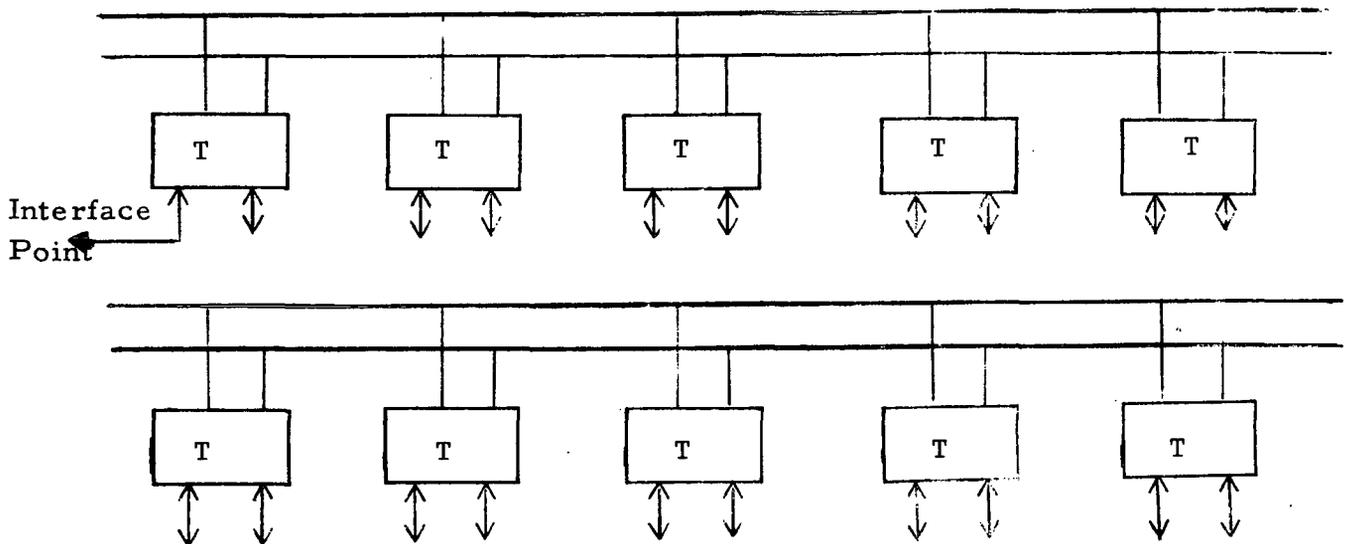


Figure 4-5. Candidate DB-2, Two Dual Redundant Data Bus Assemblies

DACS elements requiring three failure tolerance would need to interface with both dual redundant subassemblies. Those requiring only single failure tolerance could interface with only one subassembly. This fits the requirements for pressure isolatable volumes very nicely. Only one of these subassemblies would be required in each pressure isolatable volume.

Three options are available for parallel redundant application of candidate DB-2. Option A would have both dual redundant subassemblies fully operational at all times, and Option B would have one dual redundant subassembly on standby. Both of these options require the dual redundant subassemblies

to have active dual redundancy. Option C has similar operational requirements to DB-1B and requires interface with at least three of the four total interface lines for time-critical Model B, or the requirement that any time critical loop first failure activates the standby portion is necessary.

The third option is something of a hybrid. Each dual redundant sub-assembly would be operated as standby dual redundancy. Both subassemblies would then have to be active to satisfy the DACS requirements. Also all items interfacing with time-critical functional loop Models B or C would have to interface with both dual-redundant subassemblies even though only single failure tolerance might be required at that location. This is due to the active single failure tolerance requirement for time-critical functional loops. Or, as for Options B and C of DB-1, after any first failure the data bus assembly candidate DB-2 could revert to Option A operation.

Part of the redundancy decision process can now be performed by the data bus assembly. The dual redundant bus assembly in essence provides a best copy at its output. This is known as a crossover point, and the basic mechanization is such that the dual output provided after a failure has occurred is obtained from only one source. In the dual situation the source selected is usually based on BITE and serial redundancy checks offer disagreement detection in the dual input comparator.

This possible interface comparison and crossover performed at the data bus assembly in candidate DB-2 is the main difference between it and candidate DB-1. A dual redundant implementation can also take advantage of some savings in reduction of number of physical packages and the like. The comparison circuitry and the crossover (which is not necessary if operation identical to DB-1 is used) is an additional hardware requirement.

The cabling will be physically separated in both DB-1 and DB-2. The terminals of candidate DB-2 no longer have the physical separation of candidate DB-1 since they are packaged together. The organization of candidate DB-2 is similar to that of the central processing assembly and other time-critical subsystem functional loops.

#### 4.3.2.4 Candidate DB-3

The third major candidate redundancy concept for the data bus assembly would be a single quad redundant data bus assembly. This is shown pictorially in Figure 4-6. The operation of this candidate is such that any one cable sub-assembly and associated terminal equipment must be capable of emergency operation, any two cables and equipment must be able to handle the data traffic requirements of critical operations, and any three (with any single failure) be able to provide the additional experiment capability.

This quad redundant candidate, DB-3, has similar options to candidates DB-1 and DB-2. Option A would be quad active redundancy. Option B would be triple active redundancy with one spare or standby set of hardware, and Option C would be dual-active with dual-standby parallel redundancy.

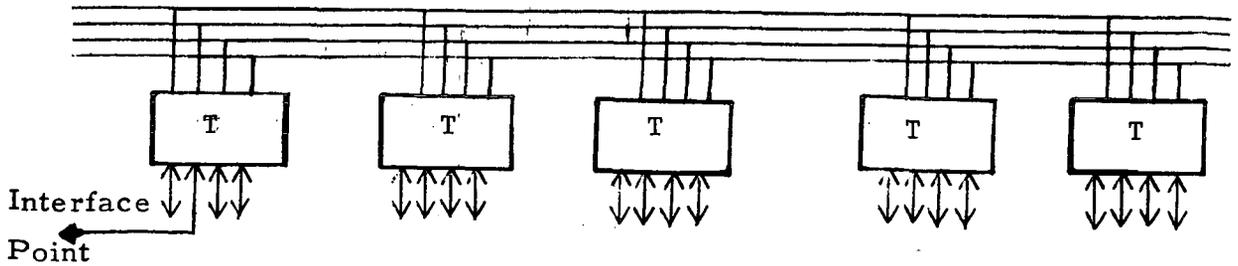


Figure 4-6. Candidate DB-3, One Quad Redundant Bus Assembly

Candidate DB-3 differs from the other two candidates in its ability to do voting and/or comparison at its terminals. Options A and B allow the use of majority voting with standby replacement on first failure detection for Option B. Option C is very similar to the operation of candidate DB-2, Option B. The minimum capability of any option occurs for Option C, which has active single failure tolerance at each terminal. External failures no longer require re-configuration of the candidate bus assembly to continue full operation.

This candidate also differs in its ability to interconnect to the various subsystem functional loop models. For single failure tolerance no more than two outputs from each data bus assembly terminal are required. The data bus cabling is, however, replicated four times everywhere in contrast to the possible reductions in this cabling for candidates DB-1 and DB-2. This interconnect difference is made possible by including the voting and crossover internal to the data bus assembly for all buses. Triple failure tolerance critical subsystem functional loop models (Type C) would still require four output lines from each data bus assembly terminal.

The four input, two output (and vice versa) capability of this data bus assembly configuration requires additional hardware at each terminal compared to the other two candidates. Depending on the option, this can be a considerable penalty for this configuration. It can also be an advantage in simplifying the BITE and serial redundancy requirements, since each terminal can have more than two copies of all data and perform a simpler decision process. Any terminal voting will probably require data buffering per data bus cable, the amount of which is dependent on the DACS and DPA operation and any time diversity on the data channels.

This candidate can be reduced to a level similar to candidate DB-1 if the voting, etc., is not used in the terminal and instead four outputs are available from each terminal unit. Then there is only a packaging difference between the candidates.

#### 4.3.2.5 Data Bus Assembly Conclusions

The three major candidate concept applications of redundancy to the data bus assembly have many similarities to other modeling performed on the subsystem functional loops. These configurations result more as a consequence of the requirements than any favoritism for any specific designs or implementations. At this point no hardware mechanization is more than sketchily implied by any of the three candidates.

Redundancy considerations, both functional and distribution, have an impact on the candidate bus assemblies. The distributed aspects were covered for the individual candidates, with candidate DB-2 appearing most like the expected distribution of other equipments. This distribution also would allow a possible reduction in cabling for both candidates DB-1 and DB-2. Functional considerations also generally favor these same two candidates, and are applicable under various circumstances and options. Candidate DB-3 makes any functional redundancy immaterial to its configuration, since each terminal utilizes a three-failure tolerance configuration.

In the best, most advantageous case that can be postulated, candidate DB-3 has a two-to-one terminal advantage over candidate DB-2 and a four-to-one advantage over candidate DB-1. It is also the most complicated terminal, so the numbers advantage is probably not a hardware advantage at all. Also the number of terminals in the bus assembly of candidate DB-3 is not flexible to locations utilizing reduced levels of redundancy, and lose even its numbers advantage over candidate DB-2 in the ideal case (for DB-2).

All candidates are assumed to include some form of data protection. It would be a necessity for operation of all candidates except DB-3 in any environment where transient protection is desired. It would not be required by any candidate if only the failure tolerance requirements are considered in the design.

Fault detection and isolation are also requirements for all candidates and necessary to the operation of candidate DB-2 for dual redundancy. The need to reconfigure and repair (by replacement) the DACS implies hardware and/or software for these functions. The data bus assembly presents some interesting repair/replacement problems. Terminal failures can be replaced with little effect because of their dedication to one or a small number of functional loops (high distribution requirements on the data bus assembly). Bus cabling failures are much more difficult to repair for any configuration, and are a special problem to be considered for the DACS independent of the candidate configuration. It may require additional cables already in place in any repair/replacement to be performed on board the station.

There are also some other configurations that can be postulated due to the command/response bus configuration and the unidirectional or bi-directional bus option specified for the DACS. These considerations have more impact on the RACU operation than on the data bus assembly and will be considered in the next subsection on RACU concepts. These considerations might affect the total hardware complement of any data bus assembly terminal in all candidates, but not the basic configurations or redundancy concepts.

### 4.3.3 Remote Acquisition and Control Unit Conceptual Candidates

#### 4.3.3.1 General

The RACU is the DACS hardware element that provides the interface between data bus assembly terminals and subsystem functional loops. As with the discussions of the data bus assembly, little consideration of the internal hardware function or mechanization of the RACU will be considered in this report. Redundancy impacts on RACU hardware will be considered.

A RACU is required at each point where an interface with the DACS is desired or required. Thus, many RACU's are utilized throughout the space station to provide access to the DPA. Each RACU is a separate hardware entity, interfacing with one or more subsystem functional loop within a space station module. The level of redundancy required for the RACU function must at least equal that of the interfacing subsystem functional loops.

Any single RACU can be located only in a single pressure isolatable volume. Thus each RACU need be no more than single failure tolerant to meet the design requirements. This at least equals the redundancy required for any critical subsystem functional loop within a pressure isolatable volume.

Each RACU has two interfaces to be considered. One interface is with the data bus assembly and the other interface is with the subsystem functional loops. RACU candidates will be defined by the interface they have with the data bus assembly. For each RACU candidate there will be variations or options based on the interface redundancy provided for subsystem functional loops. These interface provisions will be based upon the subsystem functional loop models discussed in paragraph 4.2 and shown in Figures 4-1, 4-2, and 4-3.

#### 4.3.3.2 Candidate R-1

The first candidate concept for a RACU is the simplest possible configuration for a remote data acquisition unit. This unit is simplex in nature and can only interface with a single data bus subassembly. It is shown diagrammatically in Figure 4-7. The single data bus interface precludes any necessity for parallel redundancy within the RACU.

RACU candidate R-1 will interface with any of the three data bus candidates. It interfaces with a single terminal of DB-1, and four R-1 units would be necessary to interface with all lines of candidate DB-1. Candidate DB-2 has a dual redundant interface per data bus subassembly terminal. Two R-1 units would need to be used with each DB-2 terminal at a single physical location to provide the same redundancy capability, and again four R-1 units would be necessary to fully interface with the entire DB-2 candidate structure.

Any candidate R-1 unit can only interface with a single one of the four available bus interface connections of candidate DB-3. Four R-1 units would be necessary for full interface with this data bus candidate, as with all the other data bus assembly candidates. These four units would be required at a single location. Therefore, no matter which data bus candidate is utilized, the same number of candidate R-1 units would be required for an equivalent level of redundancy.

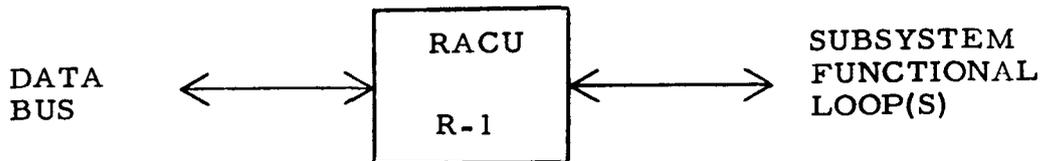


Figure 4-7. Candidate R-1, RACU with Simplex Data Bus Interface

The subsystem functional loop interface has the same characteristics as the data bus interface. For the non-time critical and time critical subsystem functional loop modes, Figures 4-2 and 4-3, four candidate R-1 units would be required and preserve the equivalent redundancy of these loop models.

Non-critical functional loop models, however, require fewer R-1 units. Model A, Figure 4-1, would require only one R-1 unit. Model B would require two R-1 units since there are two separate physical locations. Model C could use either one or two units dependent on whether any redundancy preservation is desired (not required for non-critical functions). There is also a design goal requirement for fail safe operation for noncritical subsystem functional loop models. This can be provided by the use of two R-1 units for Models B and C. A fail safe design of the candidate R-1 itself can be made such that a single unit has this capability. Then, a single unit R-1 could interface in a fail safe fashion with Models A or C.

Fail safe design of a simplex unit requires some redundancy. It can, of course, be provided by parallel redundancy, i.e., complete duplication of all elements of the unit. In this case, no serial redundancy is required, except for transient protection if desired. Any disagreement would necessitate that the subsystem functional loop enter a safe condition or nonoperating state. Another method of providing fail safe operation would make use of serial redundancy, such as command verification and echo checking, and BITE to watch all the nonredundant circuitry. At least the actual decision elements would have to utilize parallel redundancy to preclude signaling OK after failure.

Both of the sketches of fail safe design above are greatly simplified. Each design is only safe for the first failure and would require a method of making the units in question nonoperative (power-off, for example) for continuance of the safe state. Again, this power-off circuitry must be fail safe or come from parallel redundant sources. The fail-safe option will be identified as candidate R-1-FS, however it is implemented.

Candidate RACU R-1 will satisfy all the DACS requirements. Two such units are required (as a minimum) for single failure tolerance and four units for triple-failure tolerance. This puts the burden of failure detection on the subsystem functional loop for RACU outputs to the loop. Some other element of the DACS or DPA must detect failures affecting inputs to the RACU candidate R-1.

Use of a candidate RACU such as R-1 with a single bus interface imposes some operational constraints on the DACS and DPA. A data bus subassembly failure, for example, eliminates any communication with this candidate where it is utilizing that subassembly. There are very few alternate paths by which one can take advantage of operating equipment in a reconfigured mode. Therefore, a bus failure appears as if all RACU's (R-1) using that bus failed, or all subsystem functional loops (nonredundant portion) using the RACU interface failed, to the DPA. To the subsystem this appears as a computer (DPA) subassembly failure.

With this RACU candidate, as with all others, there is no reason to preclude operation with more than one subsystem functional loop model. To satisfy the redundancy requirements these must be different functional loops, not the same function nor being used to provide functional redundancy for any functional loop utilizing the same RACU.

Although it is not required to meet the failure tolerance dictated, serial redundancy may be utilized with candidate R-1. Data protection against transients for a simplex RACU can only be provided by serial redundancy. Use of a single item for noncritical loop model interfaces would probably require some form of data protection as well as command verification.

Echo checking and BITE would be required for fault detection and isolation. Or, an alternate scheme would utilize R-1 units to monitor each other for failure detection. This could be complicated, and it would necessitate using the DPA computer assembly for failure detection and isolation due to the communication and interconnection constraints.

A minimal amount of serial redundancy can be used with great effectiveness in detecting most RACU failures. This detection would be done by either the DPA or the DBCU of the DACS. Failure isolation between the RACU and subsystem functional loop would be impossible for some classes of errors at their interface.

Spares and replacement of RACU's should be done as a single unit, with candidate R-1, as pictured in Figure 4-7, the IFRU.

As with the data bus assembly, RACU's connected to standby data bus subassemblies would also be in standby. RACU's connected to subsystem functional loops that are nonoperating should also be placed in standby. Any method of providing standby operations requires action and an interface from another unit to function.

No candidate has been postulated in this report that has only one-way communication capability. All subsystem functional loop models require two-way communication for monitoring and command. It would be possible to build a receive-only RACU unit as well as a transmit-only unit. These units would have basically the same operational and redundancy advantages and disadvantages of candidate R-1, plus the additional disadvantages of even more limited annunciation and failure reporting capability. This pseudo-candidate, R-1/2, would only be worthy of consideration if there were a definite listen-only monitoring requirement for the DACS.

Candidate R-1 is the simplest and least complex of the RACU candidates. It has only a simplex interface with both the data bus assembly and a subsystem functional loop. It requires a maximum number of physically separate units to provide an interface with any subsystem loop or data bus subassembly. The candidate option for fail safe operation, R-1-FS, is more complex and costly than the basic candidate R-1. This same function can be provided with multiple R-1 units at slightly more cost in terms of hardware.

Candidate R-1 imposes some communication restraints on the DPA and especially eliminates alternate paths of communication as defined. No parallel redundancy is used in the candidate, and thus all failure detection decisions must be performed external to the unit. Serial redundancy can be used to advantage within the unit to assist in failure detection and isolation and transient error protection. Or, again, multiple units can be used at a location for monitoring each other along with subsystem functional loops. The use of candidate R-1 in active or standby mode is dictated by its interface with either active or standby elements.

#### 4.3.3.3 Candidate R-2

A second candidate concept for an RACU is shown in Figure 4-8, with its two options. This candidate, R-2, has the capability to interface with two data buses. Some parallel redundancy is therefore required in this unit. The data bus interfaces are independent, and may provide redundant or different data simultaneously to the unit.

All three data bus assembly candidates can interface with this RACU candidate concept. At least two candidate units are needed for full interface capability with all buses in normal operation.

Two options are available for candidate R-2. Option R-2A has a simplex interface provision for subsystem functional loops. A single-failure tolerant interface for subsystem functional loops is provided in candidate R-2B. Candidate R-2B would utilize parallel redundancy to provide the single failure detection, along with BITE and serial redundancy to achieve single-failure tolerance. The parallel redundancy could be either active or standby depending on the subsystem functional loop criticality and DACS and DPA operation.

Two candidate R-2A units are required at a single pressure isolatable volume to provide single-failure tolerance. These two units could be paralleled or interfaced with two separate data bus subassembly pairs. This requirement could be met at a single physical location by one candidate R-2B unit connected to one pair of data bus subassemblies.

If RACU candidate R-2A is used for critical subsystem functional loops, it requires four units to fully interface one loop model with any of the three data bus assembly candidates. These requirements match the number of units required by the subsystem functional loop model for the required failure tolerance. There are two ways of providing this connection, however, that give somewhat different operating constraints and interaction. These were mentioned in the last paragraph and only occur for Model B of non-time and time-critical subsystem functional loops.

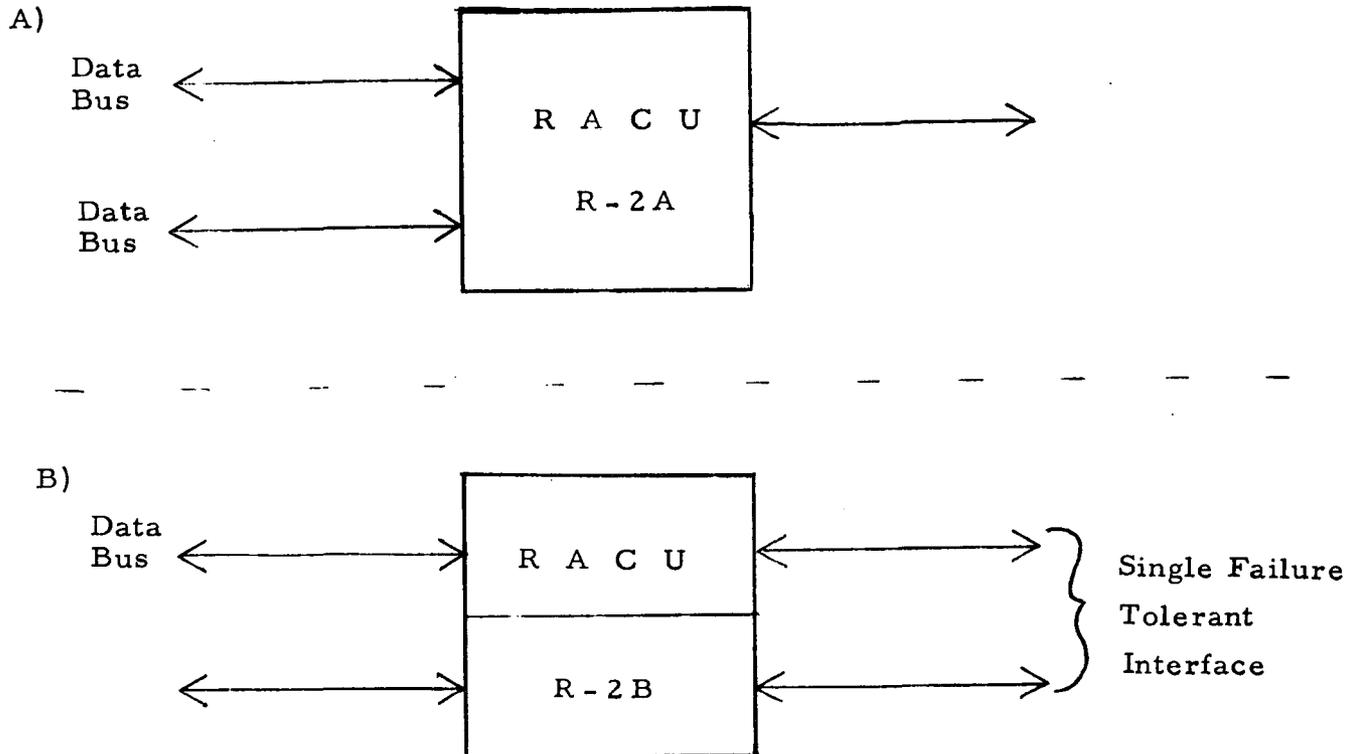


Figure 4-8. Candidate R-2, RACU with Dual Data Bus Interface

If the two candidate R-2A units per Model B unit are connected in parallel, this is closely equivalent to the use of candidate R-2B. This preserves the single failure tolerant interconnect on one pair of buses to the loop model. Any single failure of a DACS element in this chain of items (bus subassembly data link, bus subassembly terminal, RACU, R-2A, critical loop Model B) is tolerated and can be detected (and possibly corrected) by each individual unit in the chain. This allows alternate path operation over an identical set of hardware items.

The other method of interconnecting R-2A to a single item of critical loop Model B is to both pairs of (all four) data bus subassemblies. This satisfies all requirements for failure tolerance, as do all candidates. The operation after a failure is quite different from the first case though. A single failure in an RACU R-2A unit precludes any operation over that pair of buses to that redundant subsystem functional loop. Communication must not be shifted to the other pair of buses. This means that the above

method of interconnecting, although all units are active, does not provide active parallel redundancy to the critical loop model. It only provides this capability when each message is transmitted on at least three buses, and the fourth bus is not also being used at this RACU.

The second interface connection method also requires all four data bus subassemblies to be in each pressure isolatable volume. Although the operation of this interconnect method can be made acceptable, the limitations on the DACS and the DPA are definite disadvantages.

Candidate R-2B suffers none of these disadvantages, and provides the best data output to the critical loop Model B of all the type R-2 candidates. It is however more complex and costly than candidate R-2A to implement. As mentioned earlier, it has internal decision logic and failure detection to achieve single failure tolerance similar to that possible for data bus assembly candidate DB-2. Active parallel redundancy is required in candidate R-2B to interface with time-critical loop Models B and C.

Candidate R-2B requires serial redundancy to operate in a single failure tolerant mode (or else triple parallel redundancy). Candidate R-2A can take advantage of serial redundancy, in conjunction with its dual input capability, for a high-degree of data protection. It is not required, however, in this candidate concept to meet the failure tolerance requirements. Some form of BITE is necessary to isolate failures to the nonredundant portion of the RACU candidate R-2A. Otherwise, they are indistinguishable from failures in the subsystem functional loop portion that interfaces with the RACU unit.

The IFRU definition for candidate R-2A would be the entire unit. For candidate R-2B it would have to be some portion (approximately 1/2) of the unit. This complicates the design since the maintainability requirements dictate removal of this portion (after failure isolation to it) while the remainder of the item still operates in a nonredundant fashion. Thus, if a dual redundant-with-BITE design were used, two different IFRU's result. One with a simplex portion of the RACU and the second with an identical simplex portion plus the BITE circuits. Or, an attempt could be made to entirely duplicate each and every portion so that the box divided in half. Then, the box itself becomes a separate item and IFRU if replaceable. This identical problem occurs in the design of any failure tolerant assembly, whether a data bus subassembly terminal, RACU or DBCU. The maintenance requirement for at least emergency operation of a subsystem functional loop with a portion of it undergoing maintenance is very difficult to achieve with a failure tolerant unit.

Standby operation is available with both candidates. Candidate R-2A can only stand by as a unit, whereas candidate R-2B can be standby within a unit or as a whole unit. Standby provisions can only be utilized for nontime-critical functional loops or where failure tolerance is provided by more than two units for candidate R-2A and two units for candidate R-2B.

Candidates R-2 provide a dual interconnect capability for the data bus assembly interface. This interface contains some parallel redundancy for part of the RACU operation. Either of the two options of this candidate must

use four physical units to interface with critical functional loop models type A. Candidate R-2A requires four units also to interface with critical functional loop Models B and C. Candidate R-2B will interface with Model B with only two physical units, and would also need just two units to interface in a three-failure-tolerant configuration with Model C.

RACU candidate R-2B is by far the more complex in terms of hardware of the two. Both units are more complex than the candidate type R-1. Both units have similar communication characteristics for DACS operation when connected in a non-crossed-over configuration. Candidate R-2B provides an alternate communication path after a failure, and candidate R-2A can withstand some failures and still be operational.

Parallel and serial redundancy are utilized for failure tolerance by candidate R-2B. Both candidates have BITE for failure detection and isolation. One R-2A unit is an IFRU, but the IFRU design and definition for candidate R-2B is more complex and requires more than one IFRU. Both candidates can operate in active or standby configurations (as a unit) where desirable. Candidate R-2B can operate with a portion of a single unit in standby or with active parallel redundancy. The candidates satisfy all criteria, with the added complexity and hardware of R-2B providing a measure of reliability enhancement, and the possible use of fewer units, over candidate R-2A.

#### 4.3.3.4 Candidate R-3

Candidate concept R-3 for RACU's is a unit which provides interfaces for up to four data bus subassemblies. This candidate concept has three options or different subsystem functional loop interfaces. The three candidates are shown in Figure 4-9. The four data bus interfaces are independent and may transfer redundant or different data simultaneously. They require four levels of parallel redundancy within this part of each unit.

All data bus assembly candidates, DB-1, DB-2, and DB-3, can interface directly with these candidates. Only one unit is necessary to provide full bus assembly interface capability for any subsystem functional loop. The level of redundancy provided is variable depending on which of the three candidates is used.

The first option, candidate R-3A, has a simplex interface provision for subsystem functional loops. Candidate R-3B has a single-failure tolerant interface for subsystem functional loops. Options R-3B and R-3C use parallel redundancy to achieve the required level of failure tolerance, and all candidates use BITE and serial redundancy for the required failure detection and data protection. The parallel redundancy used can be either active or standby within the unit depending on the criticality of the interface and the DACS operation.

To provide single-failure tolerance within a single pressure isolatable volume would require two candidate R-3A units and one unit of the other candidates. The data bus connections are assumed parallel for this situation and no cross-strapping is done within a unit. It would be possible to design the unit to allow this, but it is an unnecessary complication.

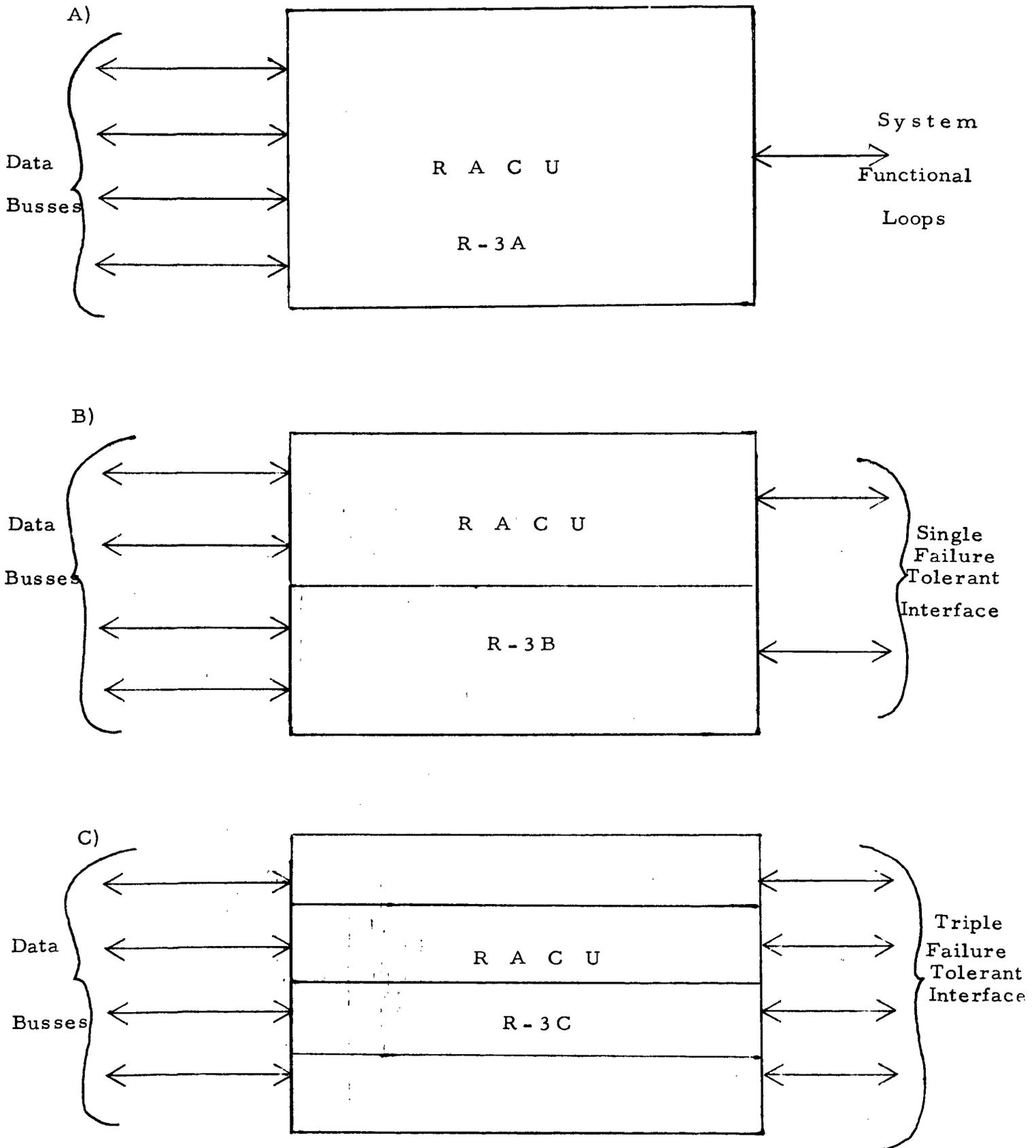


Figure 4-9. Candidate R-3, RACU With Quad Data Bus Interface

Four units of RACU candidate type R-3A are required to interface any one critical subsystem functional loop model with any data bus assembly candidate. Four units of the other candidates are also required to interface with critical Model A. Only two units are necessary for the interface with critical Model B for candidates R-3B and R-3C. Candidate R-3B also requires two units to interface with critical Model C, while candidate R-3C only needs one unit. To be effective, candidates R-3 requires all buses to be distributed throughout all space station modules and pressure volumes.

This set of candidates impose the fewest restrictions on data bus and DACS usage by the DPA computer assemblies. Since all buses go to all RACU's any message can be sent on a single bus to all subsystem functional loops. This may be a distinct operating advantage.

No triple redundant or two-failure-tolerant RACU has been postulated as a candidate concept. Any usage of such a model would result in a situation similar to candidate R-3B. As such, this possibility is not included as a candidate. This was discussed in more detail in paragraph 4.2.2 for loop models.

Candidate R-3B requires serial redundancy and BITE to operate in a single-failure-tolerant fashion. Candidate R-3A may take advantage of serial redundancy and BITE for data protection depending on the mode of operation. Its four data bus connection could provide four copies of all data, but this is very poor operational utilization of the DACS. Candidate R-3C must utilize serial redundancy and BITE to provide triple-failure tolerance for the same reasons as candidate R-3B.

Candidate R-3A can be defined as an IFRU. Candidates R-3B and R-3C have the same difficulties in partitioning these units into IFRU's as candidate R-2B (see discussion paragraph 4.3.3.3). They will require multiple IFRU's to be defined per candidate, and consequently impact the on-board spares requirement and the actual unit design. Testing to the IFRU level will be performed by BITE for failure detection and isolation. The BITE information will be transferred via the data bus assembly to the DPA in the same manner as normal monitoring data from the RACU's.

All candidates can operate as standby units when utilized with allowable configurations of other DACS and subsystem elements. Candidates R-3B and R-3C can operate with internal portions in standby for nontime-critical subsystem functional loops or when multiple units are utilized per loop model.

Candidate R-3 provides a quad data bus interconnect capability for the data bus assembly interface. This is, of course, the most costly data bus interface in terms of hardware requirements but allows the most generalized operation of the DACS. All of the candidates utilize both parallel and serial redundancy with BITE for fault detection, transient error protection and failure isolation.

The simplest of these candidates is R-3A, and four are required for a critical loop model full interface. The next candidate, in order of increasing hardware complexity, is R-3B. Only two of these units are required to interface

with critical loop Model C, but two units still are required for a Model B type interface, and four units for critical models type A. These latter multiple units are due to the inherent physical separation in the subsystem functional loop models.

Candidate R-3A can be defined as an IFRU, but the other two candidates require multiple IFRU's to be defined per RACU unit. Candidates R-3B and R-3C can operate with portions of each unit inactive for standby redundancy. This candidate class offers full bus assembly communication capability and multiple alternate communication paths as its main advantages and its increased hardware and complexity as its major disadvantages.

#### 4.3.3.5 RACU Conclusions

Seven RACU conceptual candidates have been defined. These seven candidates are grouped into three classes based on the number of bus interconnects provided. The options within each class provide for varying levels of failure tolerance, from none to triple-failure tolerance.

These seven candidates can be ranked in terms of hardware complexity. This ranking is shown below, with the first item in the list the most complex and the last RACU candidate, the simplest.

1. R-3C
2. R-3B
3. R-2B
4. R-3A
5. R-2A
6. R-1-FS

Functional and distributed redundancy considerations impact the total amount of hardware, in number of units, required to provide the overall DACS utilizing any specific candidate. The distributed redundancy impacts are such as to require an excess of capability to satisfy the physical distribution of subsystem functional loops. This, for example, requires four of the most complex candidate, R-3C, to provide the interface for the nontime-critical subsystem functional loop Model A. This is as many units as are required when utilizing the simplest candidate RACU, R-1. There are, of course, operational and hardware advantages to the more complex candidate method.

Functional redundancy considerations can be used to determine the total number of units of any one (or more) candidate type required for any specific complement of subsystem functional loops. The individual requirements for the number of RACU's of each candidate required for each subsystem functional loop model are given in Table 4-1.

In order to address and command an RACU when multiple redundant units are used for a specific functional loop interface, the DACS must specify a specific response bus to be utilized in the return communication. Otherwise, conflicts in bus usage result.

Table 4-1. Number of RACU's Per Subsystem Functional Loop Model For The Seven RACU Candidates

| RACU Candidates | Subsystem Functional Loop Models |   |   |                   |   |   |               |   |   |
|-----------------|----------------------------------|---|---|-------------------|---|---|---------------|---|---|
|                 | Non-Critical                     |   |   | Non-Time Critical |   |   | Time-Critical |   |   |
|                 | A                                | B | C | A                 | B | C | A             | B | C |
| R-1             | 1                                | 2 | 2 | 4                 | 4 | 4 | 4             | 4 | 4 |
| R-1-FS          | 1                                | 2 | 2 | 4                 | 4 | 4 | 4             | 4 | 4 |
| R-2A            | 1                                | 2 | 2 | 4                 | 4 | 4 | 4             | 4 | 4 |
| R-2B            | 1                                | 2 | 1 | 4                 | 2 | 2 | 4             | 2 | 2 |
| R-3A            | 1                                | 2 | 2 | 4                 | 4 | 4 | 4             | 4 | 4 |
| R-3B            | 1                                | 2 | 1 | 4                 | 2 | 2 | 4             | 2 | 2 |
| R-3C            | 1                                | 2 | 1 | 4                 | 2 | 1 | 4             | 2 | 1 |

Candidate R-1-FS is the fail safe version of candidate R-1. This same implementation method could also be applied to candidates R-2A and R-3A. This would provide them with fail-safe rather than simplex interfaces with the subsystem functional loops. The necessity for a fail-safe design has not been firmly established at this time.

All candidates are assumed to include some form of data protection. This provision is both for transient error protection and to assist in the failure-tolerant operation of specific candidates. Failure-tolerance requirements alone do not dictate a requirement for data protection.

Fault detection and isolation is a requirement for all of the RACU candidates. IFRU's have been identified and discussed for all candidates. These are the replaceable items for repair. Most of the candidates have internal BITE for fault detection and isolation. The final determinations must be made by a central authority, however, either the DBCU or the DPA.

Some alternative candidate configurations can be mentioned. All candidates except the simplest can take advantage of a unidirectional bus structure and eliminate some interface lines to the data bus subassembly terminals. All the command buses possible in the candidate are always necessary to receive commands. Only one response line is necessary if the operation is limited to a single bus response (or if the terminal provides multiple responses from a single line--a more difficult operational problem). This does require more control lines to the data bus subassembly terminal to designate the bus to be used for the response. This scheme, in general, saves very few lines except in the full bus interconnect category. Here, a savings of approximately 15 percent or more may be achieved.

The use of bi-directional command and response data buses allows an optional method of communication. Since each line can provide bi-directional communication; only one command and one response line is necessary for single failure-tolerant operation. For example, candidate R-1 with this capability would really be equivalent to R-2A. Candidate R-2B could be equivalent to candidate R-3B. Because of this equivalency, with this modification to the RACU/data bus assembly structure, no other candidates were constructed for this case. Bi-directional bus lines for command and response within a module is also not the preferred simplex data bus design concept.

Other minor variations are possible for all candidates. The actual implementation of any specific candidate in hardware is the subject of design trade studies to determine the proper method of providing various aspects of operation, most of which have not been discussed for any of the RACU candidates. The seven general RACU candidate concepts satisfy all DACS redundancy criteria. They also offer a variable amount of additional features of operation and reliability enhancement for consideration.

#### 4.3.4 Data Bus Control Unit Conceptual Candidates

##### 4.3.4.1 General

The DBCU is the third DACS hardware element. This is the unit that controls the DACS operation, specifically, the traffic on the data bus subassemblies. As in all previous discussions, the emphasis will be on the redundancy concepts and not the internal hardware configuration of the candidate units.

The control of the data bus assembly is split into a control area in each of two separate physical locations or pressure isolatable volumes on the modular space station. The DBCU interfaces with the DPA computer subassembly. There are two active dual-redundant (single-failure tolerant) subassemblies, one in each separate physical location. The interface is assumed for purposes of this discussion to be with input/output processors (IOP's). There are two IOP's per DPA computer subassembly.

The level of redundancy required for the DBCU function at a single physical location is at least the same as the computer subassembly, or single-failure tolerant. Each computer subassembly has a model similar to the subsystem time-critical functional loop Model B (see Figure 4-3) in that it is dual-redundant and has two separate input/output interface lines.

The interface for a DBCU must then be with either one or two of these input/output interfaces, or computer subassembly IOP's. The DBCU also must interface with the data bus assembly, either candidate DB-1, DB-2 or DB-3 (Figures 4-4, 4-5, or 5-6). Furthermore, this interface may be with one or more of the data bus subassemblies for any individual DBCU.

Eight conceptual candidate control units are presented below. The candidates are grouped in to classes depending on whether they interface with one or two IOP's per computer subassembly.

##### 4.3.4.2 Candidate CU-1

The first candidate concept for a DBCU is the simplest hardware configuration. This candidate class interfaces with only a single IOP. The three optional configurations of this candidate are derived from the variable number of data bus subassemblies that they can interface with. These three candidates are shown in Figure 4-10.

There is no parallel redundancy provided in any of these three candidates except that inherent in multiple bus interconnects. They are all simplex in nature and can be viewed as switches (one, two or four) connecting the IOP to the data bus subassemblies. For candidate CU-1-1 only one such connection is possible, and this is by far the simplest hardware candidate. Candidate CU-1-2 can have its associated IOP connected to either one or both of two data bus subassemblies. Similarly, candidate CU-1-3 can connect its IOP to any data bus subassembly, any pair, any three, or all four data buses.

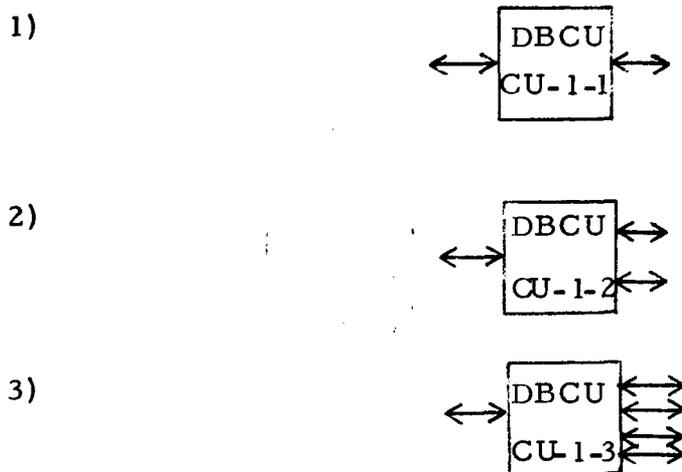


Figure 4-10. Candidate CU-1, Single IOP Simplex DBCU

As before, no candidate is presented that connects to only three data buses. This situation provides no advantages or different operational modes than does candidate CU-1-3. Therefore, since it is a minor variation it is not included separately for consideration.

Since the DBCU candidates are essentially simplex switches and controllers, little or no serial redundancy is predicted for these candidates. The distributed redundancy consideration of the DPA dictates the minimum number of units necessary for the DACS. For all three candidates this minimum number is four units, one per IOP. These four units are split equally between the two separate module locations.

Data protection is assumed added to the data (and checked) by the IOP. If not, it can be included in these candidate DBCU's. The other data protection and serial redundancy considerations for RACU's will emanate from the DBCU's such as command verification and echo checking.

Fault detection and isolation of these DBCU candidates will be assisted by BITE but will need to be controlled and decisions made by the DPA. This is due to the inherently simple and simplex nature of the candidate units.

It would be possible for the DBCU candidates CU-1-2 and CU-1-3 to do comparison and/or voting on data received via the data bus from the RACU's. The other option would be to perform this function in either the IOP hardware or software. In candidates CU-1-2 and CU-1-3 the only parallel pressure of data occurs at the candidate control units. Thus, there is some justification



for performing the operation at this point, especially if a hardware technique is used. This would then also require the data protection checking to be performed here also, since the results from both these checks are used to detect transient errors as well as failure conditions.

The use of singular units per IOP, as in all these CU-1 candidates, allows each such unit to be defined as an IFRU. No problems of repair by replacement are encountered with these configurations, except those of hardware design necessary for any IFRU interchangeability and removal.

Candidates CU-1-2 and CU-1-3 offer the ability to get from one IOP to more than one bus. The IOP configuration in a multiprocessor already allows either IOP per multiprocessor access to all internal information. Thus, this crossover to multiple buses is also available in candidate CU-1-1 similar to candidate CU-1-2 by IOP data interchange. Using more than one DBCU candidate CU-1-1 per data bus subassembly and IOP would also give the same effect as candidate CU-1-2 and CU-1-3 at an additional hardware expense. Partitioning a system into multiple data crossover points only pays in reliability enhancement and alternate paths of operation. The first such crossover is very effective, and adding more produces diminishing returns.

These three candidate DBCU's are the simplest possible. The three vary little in hardware complexity, with CU-1-1 having the least and candidate CU-1-2 and candidate CU-1-3 slightly increased complexity. There is also a relative hardware increase in going from Option 1 to Option 3 due to the multiple bus interface provisions. Candidate CU-1-3 has the most operational capability with the minimum number of units; the operation capability per amount of hardware would favor candidate CU-1-3.

Candidate configuration CU-1-2 can also connect to either the same pair of buses per multiprocessor or to all four buses per multiprocessor. In the second case, this priority and scheduling problem also exists. If the burden of priority control is placed on the DBCU it will be a very complicated unit. Reconfiguration also enters this decision, and the selection of alternate paths and modes almost certainly must be performed by the DPA. For this reason, plus the essentially software nature of the problem (memory for the status of all DACS and DPA items), the designation of control priority will be assumed part of the DPA, as well as path selection after failures. The use of candidate CU-1-3 also requires all data buses to be in the space station modules housing the two DPA computer subassemblies.

#### 4.3.4.3 Candidate CU-2

4.3.4.3.1 CU-2A. The candidate class of CU-2 concept has a dual interface for IOP's. Thus each candidate unit in this class can interface with both IOP's at a multiprocessor location. If the candidates are nonredundant, or simplex DBCU's, then three options are again available as in candidate class CU-1. These are shown diagrammatically in Figure 4-11 as candidate concept CU-2A.

These three candidates in Figure 4-11 all require four units to interface with the two dual computer/IOP subassemblies. This is necessary to preserve the redundancy of the DPA at the DACS interface, which is single-failure tolerant at each physical multiprocessor location.

These three configurations are almost identical to candidate class CU-1 except for the additional crossover of IOP inputs and outputs. As explained above, this crossover is inherent in multiprocessor operation, and buys little additional capability, even in terms of redundancy enhancement. It does provide other alternate paths of data flow after an IOP failure (or a DBCU failure in candidates CU-2A-1 and CU-2A-2). The hardware and hardware complexity of these three units is ranked the same as for candidate class CU-1. Option 1 is the lowest and Option 3 the highest of the candidate configurations. For other advantages and disadvantages of these three concept configurations and the IFRU definition refer to 4.3.4.2. Their major advantage is the dual IOP interface provision capability.

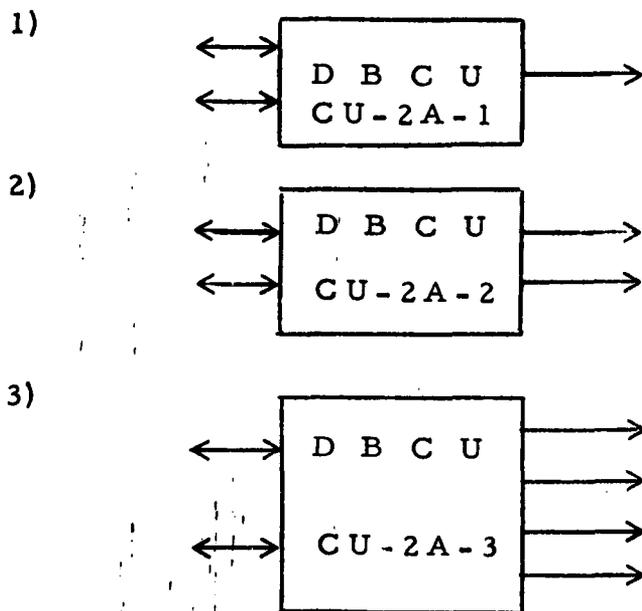


Figure 4-11. Candidate CU-2A, Dual IOP Simplex DBCU

4.3.4.3.2 CU-2B. A second variation on candidate concept CU-2 is shown in Figure 4-12. These candidate DBCU's have the dual IOP interface capability standard of candidate CU-2. The major variation between these two candidates and all the others presented is their single-failure tolerant design. No candidate option is presented with a single bus interface since this would be a single point of failure. (If the single interface were also made redundant then the unit would be equivalent to candidate CU-2B-1.)

These two single-failure tolerant candidates make use of parallel (dual) redundancy and serial redundancy and BITE for failure detection and isolation. They are similar in operation to all the other candidates, especially candidates CU-2A-2 and CU-2A-3. The redundancy aspects of candidate CU-2B-1 are very similar to the RACU candidates R-2B and R-3B described in 4.3.3.3 and 4.3.3.4. The same holds true for DBCU candidate CU-2B-2. Further description of this aspect of these candidate concepts is referred to the descriptions in the cited subsections.

The required voting and or comparison at the data bus subassembly interfaces follow the same arguments as presented earlier for candidates CU-1-2 and CU-1-3 (and CU-2A-2 and CU-2A-3) in 4.3.4.2 and is not repeated here.

These two candidates are affected by the distributed redundancy considerations that apply to the DPA. A minimum of two units per DACS are required to satisfy the failure tolerance requirements. One unit of either candidate would be located at one multiprocessor location in one pressure isolatable volume and the other unit in the other separate multiprocessor location. These two configurations then require one-half the units per DACS of all the previous candidates.

The operational complications of priority over individual data bus control exist only for candidate CU-2B-2 in the minimum units configuration. This is discussed in more detail for candidates CU-1-2 and CU-1-3 in 4.3.4.2. These units, as did their predecessors, offer an increasing number of alternate modes of operation for reconfiguration after failures. In their minimum units configuration, however, they have fewer alternate paths of data transfer (crossovers) than candidates CU-2A-2 and CU-2A-3 respectively with much higher hardware complexity.

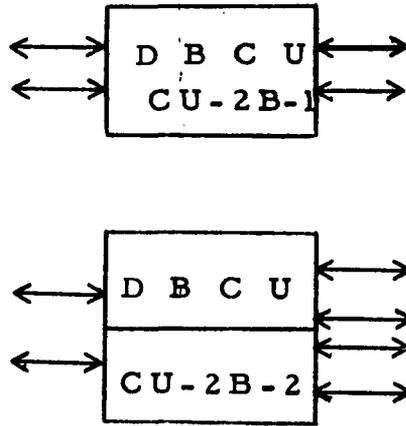
Both these candidates have the same or similar IFRU definition problems as the RACU candidates R-2B and R-3B. This is true for all the failure tolerant configurations, since they require some method of developing replaceable IFRU's within a single assembly without disturbing the operation of the remaining hardware. And no matter what redundancy concept is utilized, there are not two, three, etc. (dual, triple, etc.) identical units per box. Additional circuitry is always required for failure detection and isolation.

The distributed redundancy consideration eliminate any other possible candidates for DBCU's. No single location has more than dual IOP's. A quad structure could be utilized, but it would look identical to using two candidate CU-2B-1 units. A single-failure tolerant concept is the most that is necessary to satisfy the single-failure tolerant interface of the DPA multiprocessor subassembly with the DACS.

#### 4.3.4.4 DBCU Conclusions

Eight DBCU conceptual candidates were defined in this subsection. Each, when used in enough quantity and properly connected, will satisfy the DACS requirements. The eight candidates were classed into two groups. One group had a single IOP interface provision and the other group could interface with two IOP's. The options in the second group also provided redundancy for

single-failure tolerance, the most required of a single DBCU due to the distributed redundancy considerations.



**Figure 4-12. Candidate CU-2B, Dual IOP Single Failure Tolerant DBCU**

The eight candidate DBCU's are ranked below in terms of hardware requirements and hardware complexity. The first in the list is by far the most complex with the second candidate listed close behind. The others are all simplex and the variation in complexity is fairly low. The amount of hardware per unit is used as a guide for the remainder of the ranking down to number eight, the simplest.

1. CU-2B-2
2. CU-2B-1
3. CU-2A-3
4. CU-1-3
5. CU-2A-2
6. CU-1-2
7. CU-2A-1
8. CU-1-1

Distributed redundancy considerations dictate the minimum number of units per DACS required for each candidate. These are listed below for reference in Table 4-2.

Table 4-2. Minimum Number of Candidate Data Bus Control Units

| <u>Candidate</u> | <u>Minimum Required</u> |
|------------------|-------------------------|
| CU-1-1           | 4                       |
| CU-1-2           | 4                       |
| CU-1-3           | 4                       |
| CU-2A-1          | 4                       |
| CU-2A-2          | 4                       |
| CU-2A-3          | 4                       |
| CU-2B-1          | 2                       |
| CU-2B-2          | 2                       |

Serial redundancy, if used, such as echo checking and command verification would need to be performed by any of the candidates. All units would also contain BITE to help meet the failure detection and isolation requirements.

The trades mentioned in the RACU summary, 4.3.3.5, on the utilization of unidirectional or bi-directional busses for redundancy purposes also need to be reflected upon for the selected DBCU candidate(s). The discussion is not repeated here.

Some other variations in DBCU candidates are possible. One important variation is dependent on the method of bus priority control between the two computer subassemblies. If this is handled through the DBCU's only, they will need a communication link to the other (physically separate) DBCU's. This can be done either through RACU's or by including some of the RACU functions internal to the DBCU, such as address recognition and response to a command from another DBCU.

#### 4.3.5 DACS Summary

The previous three subsections have each considered one of the three hardware elements of the DACS in detail. Multiple candidates and concepts have been developed, discussed and compared. Each candidate has been given consideration and related to the redundancy and reliability enhancement concepts and techniques. Table 4-3 summarizes the candidates considered.

Three data bus assembly candidates were presented for consideration. The advantages and disadvantages of each were noted. Seven RACU candidate concepts were presented in 4.3.3. All candidates satisfy the data bus assembly interface requirements and also will interface with any of the nine subsystem functional loop models. This requirement is shown in Table 4-1.

Table 4-3. DACS Element Candidates

| DACS Element                         | No.  | Options           | Operating Mode  | Recommendation            |
|--------------------------------------|------|-------------------|---|---------------------------|
| <u>Data Bus Assembly</u>             |      |                   |   |                           |
| 4 Independent Buses                  | DB-1 | A<br>B<br>C       | Concurrent operation, all four<br>Three operating, one standby<br>Two operating, two standby    | DB-1A (in a 2 and 2 mode) |
| 2 Dual-Redundant Buses               | DB-2 | A<br>B<br>C       | Concurrent operation, all four<br>One dual-operating, one standby<br>1/2 each operating         |                           |
| 1 Quad-Redundant Bus                 | DB-3 | A<br>B<br>C       | All operating<br>Three operating, one spare<br>Two operating, two standby                       |                           |
| <u>RACU</u>                          |      |                   |   |                           |
| Simplex Bus Interface                | R-1  |                   | Simplex interface to subsystems   | R2-A                      |
| Dual Bus Interface                   | R-2  | A<br>B            | Simplex interface to subsystems<br>Dual interface to subsystems                                 |                           |
| Quad Bus Interface                   | R-3  | A<br>B<br>C       | Simplex interface to subsystems<br>Dual interface to subsystems<br>Quad interface to subsystems |                           |
| <u>Data Bus Control Unit</u>         |      |                   |   |                           |
| Simplex I/O Interface                | CU-1 | -1<br>-2<br>-3    | Simplex data bus interface<br>Dual data bus interface<br>Quad data bus interface                | CU1-3                     |
| Dual I/O Interface<br>(Simplex DBCU) | CU-2 | A-1<br>A-2<br>A-3 | Simplex data bus interface<br>Dual data bus interface<br>Quad data bus interface                |                           |
| Dual I/O Interface                   |      | B-1<br>B-2        | Dual data bus interface<br>Quad data bus interface  |                           |

4-33

SD 72-SA-0114-3

Eight DBCU candidates were presented in the previous subsection. Table 4-2 lists the minimum number of units required by the DACS for each candidate. All provide the level of redundancy required by the DPA multiprocessor subassemblies.

Taking into consideration the nine subsystem functional loop models with the three data bus assembly candidates and the seven RACU and eight DBCU candidates, there are over 1500 combinations of candidates possible for the DACS. These are just the major candidates. There are also a large number of not so major variations.

All of the over 1500 candidate configurations satisfy the failure tolerance requirements. Because of this large number of options the candidates have been described, developed and considered as separate DACS entities. This consideration continues in the next section on this basis to keep the selection problem manageable. When the number of different DACS configurations for redundancy consideration is reduced considerably, they can be evaluated as a DACS assembly.

Very important to all the considerations in this section is the following DACS design goal. All RACU's, DBCU's, and the Data Bus Assembly should be designed and constructed such that any single failure does not affect an entire data bus subassembly. A design requirement is that any single failure of a DACS element shall not affect more than a single data bus subassembly. This holds even for the simplex units such as RACU candidates R-2A and R-3A. This is necessary for any redundant data acquisition and control subsystem operation.

#### 4.4 DACS RECOMMENDATIONS

##### 4.4.1 General

This subsection will offer some recommendations on the DACS elements among all the various candidates and over 1500 combinations. The degree, level and type of redundancy for each DACS element and the DACS as a whole will be recommended.

The recommendations in this section will include hardware/software techniques, error protection levels, replacement (repair) requirements and the rationale for each selection made.

##### 4.4.2 Data Bus Assembly Recommendations

The recommended candidate data bus assembly is configuration DB-1. This is the most straight forward approach to meeting the data bus redundancy requirements. It is the lowest cost implementation. This data bus assembly candidate offers the best physical independence, both from an operation viewpoint as well as in routing throughout the Space Station modules.

This data bus candidate uses parallel redundancy to provide the three failure tolerance requirement to equipments that interface with all four simplex subassemblies. Each simplex bus subassembly would carry serial

redundancy between DACS elements. All redundancy decision making would be performed external to the data bus assembly. Data protection would be used on the data bus assembly, again generated and checked external to the assembly.

The physical routing of the data bus would be such as to take advantage of the separate pressure volume constraints. Space Station modules requiring no interface with two of the four data bus subassemblies would not have these busses routed in them. Only the modules where the two computer subassemblies are located would require all four data bus subassemblies to be contained within.

Only two busses would be active with the primary computer subassembly. The other two busses would be active for the standby computer (and experiments). Thus, this is Option A of candidate data bus assembly DB-1. This option is selected because it allows operational use of the other busses for experiments while two active busses are used for monitoring time-critical subsystem functional loops.

Fault detection and isolation for the data bus assembly would be performed by the computer assembly using software techniques to process data derived from limited echo checking and normal command verification. The interface terminals would be checked by the RACU's and reported to the DPA multiprocessor over the data bus assembly as part of its normal operation.

The data bus terminals would be defined as IFRU's and replaced individually upon failure isolation. The data bus cable subassemblies might contain a very few additional wires for substitution if they cannot be replaced or repaired in place.

This candidate selection is recommended due to its simplicity, cost, physical independence between bus subassemblies, operational advantages, routing advantage, and maintenance ease when compared to the other candidates.

#### 4.4.3 RACU Recommendations

The candidate RACU recommended for the DACS is configuration R-2A. This candidate has a dual data bus subassembly interface and a simplex design and interface with the subsystem functional loops. This candidate is recommended as a compromise between the simple R-1 with no crossovers or alternate paths and the more complex and costly R-3A. All three have fairly similar redundancy concepts and capabilities.

This RACU candidate uses some parallel redundancy in its data bus subassembly interface circuits. Four units are required to fully interface any one critical subsystem functional loop model with the DACS. A parallel connection to the data bus is recommended. The failure tolerance requirements of the subsystem functional loop models are met by using multiple units of this candidate type.

Each individual unit contains BITE for failure detection and isolation. All internal detected failures are reported (if possible) to the DPA multiprocessor through the normal operational data flow of the DACS. Serial

redundancy is recommended for command verification. Echo checking of the data bus terminals is recommended and can be performed by each RACU unit by listening to its own responses to a command and performing a bit by bit comparison. Hardware elements are used for all internal functions.

Data protection for transient error protection is recommended and both generating and checking of protected data will be done by hardware in each RACU. The selection of the specific coding technique to be utilized will be deferred for further requirements definition, although an error detection-retransmission scheme is recommended for error correction.

Each RACU is an IFRU and repair will be by crew replacement of a failed item. This failure will be annunciated by the DPA multiprocessor, as well as DACS failures.

All candidates except R-1 and R-3A were discarded as too complex, costly and difficult to maintain. Some, such as R-3C, were never expected to be utilized by the DACS due to the functional loop, pressure volume separation requirements. Candidate R-3A was not chosen because it required all busses to be routed in all modules, but provided little additional capability for this added expense. Candidate R-3A also was more complex and costly and was still susceptible to a single failure. Candidate R-1, although the simplest, offered little reliability enhancement. It also placed more burden on a subsystem functional loop and had no alternate communication paths.

This candidate can also be designed in a fail-safe fashion. The recommendation now is to defer this choice until the fail-safe definition and requirements are more well defined for all subsystem functional loops.

#### 4.4.4 DBCUC Recommendations

The candidate DBCUC concept recommended of the eight possible is CU-1-3. This candidate has a single IOP interface and a four data bus subassembly DACS interface. The candidate CU-1-3 is fairly low in complexity and cost, and it provides full operational flexibility and communication paths.

Parallel redundancy is utilized in this recommended candidate at the data bus subassembly interface. Two units are required per DPA multiprocessor subassembly. Each unit is basically simplex in operation. Failure tolerance requirements are met by using multiple units.

All units contain some BITE to report on their non-redundant portions. The question of voting on multiple data bus inputs is recommended to be deferred until more definition of DPA requirements and operation is available. Serial redundancy is recommended for echo checking and command verification in hardware with the results reported to the DPA multiprocessor failure determination and action by software. Echo checking simultaneous with its own operation should be performed by hardware BITE similar to the RACUC recommendation. This checks the other data terminals and the remainder of the data bus assembly.

Data protection is recommended and the decision as to the location of the hardware and/or software between the DBCU and the IOP deferred until more DPA definition is made.

Each DBCU is an IFRU and repair is by direct replacement. All busses must be routed in the two command modules housing the DPA computer subassemblies.

The method of priority control over the DACS will be under software control of the DPA computer assemblies. Reconfiguration and path selection will also be determined and specified by the DPA multiprocessor assemblies.

Candidates CU-2B-1 and CU-2B-2 were eliminated on reasons of cost, complexity and maintenance difficulty. Candidates CU-1-1 and CU-2A-1 provided little in the way of alternate paths or operational flexibility. They essentially fed one computer to one bus and limited reconfiguration. Of the four remaining candidates, the group CU-2A was not recommended because of the crossover capability provided by the DPA multiprocessor concept at IOP/CPU inputs and outputs already. They would also require slightly more hardware than their CU-1 counterparts.

The selection between CU-1-2 and CU-1-3 was made on the basis of operational flexibility. The hardware differences are fairly small between the two candidates, and the difference in complexity is even less. Candidate CU-1-3 can handle subsystem functional loop failures in a much simpler fashion than candidate CU-1-2, especially using the recommended RACU candidate R-2A. The major reasons for the candidate selection were the increased operational capability and the availability of alternate communication paths in the selected candidate, versus the other candidate, at very little hardware penalty.

It is also recommended that data communication between DPA multiprocessor subassemblies not utilize the DBCU to DBCU route. This communication should be done as all other communication, through the data bus and RACU's. Besides eliminating unwarranted complexity to the DBCU, this allows monitoring of physical parameters of one multiprocessor subassembly by the other and the transfer of data. This is also the recommended method for mass storage data transfer between the two subassemblies and pressure isolatable volumes.

#### 4.4.5 DACS Recommendations Summary

A summary of the recommended redundancy concepts and techniques is shown pictorially in Figure 4-13. The RACU interconnect is shown for one complete critical subsystem functional loop model Type B. The DBCU's are shown for one-half of the DACS only; the other computer subassembly in the other pressure volume has an identical interconnect. Only a small portion of the data bus assembly DB-1 is shown.

The RACU interconnect, as recommended and pictured, allows two of the RACU's and their associated subsystem functional loop to be in a standby mode while the other two are active. This is true for both time-critical and non-time critical models of Type B. The DPA multiprocessor and its software is utilized for all higher level BITE analysis, failure determination, fault isolation and crew notification for repair by replacement. Each separate DACS box drawn in Figure 4-13 is an IFRU.

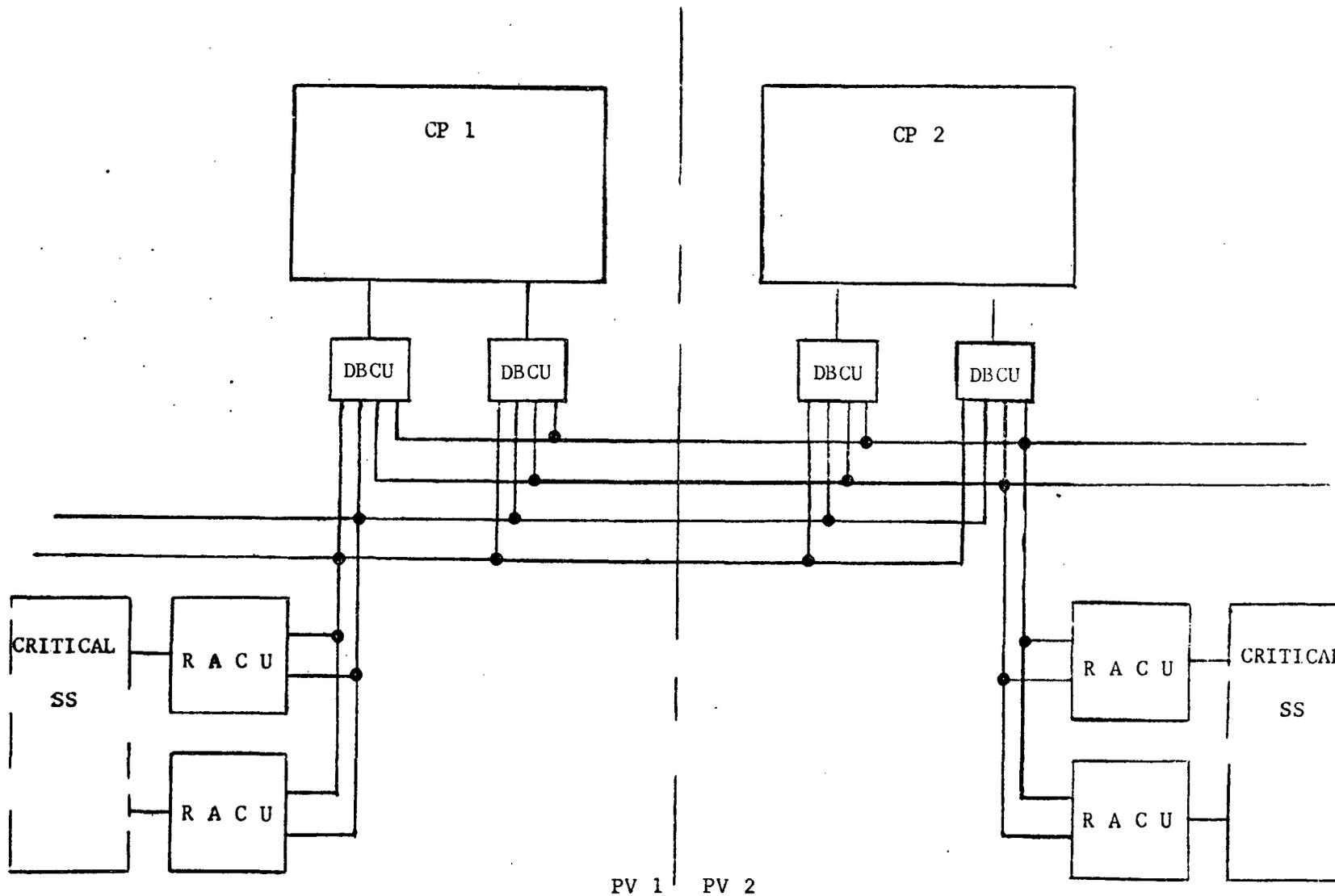


Figure 4-13. Recommended DACS Configuration

## 5. DACS BREADBOARD DESIGN REQUIREMENTS

### 5.1 GENERAL REQUIREMENTS

#### 5.1.1 Introduction

The DACS breadboard is an engineering model that is representative of the concepts for the data acquisition and control function for the Data Processing Assembly of the Modular Space Station. The NASA defines a breadboard as a unit which performs the same functions and according to the same characteristics as those defined by the hardware design.

A data acquisition and control subsystem is a semi-autonomous subsystem that provides controlled communication between a large number of remote locations and a control location. Insofar as possible, the DACS breadboard is a representation of such a subsystem. The DACS breadboard also provides a test bed for operation performance evaluation of numerous concepts for this type of subsystem oriented toward the specific needs and requirements imposed by the Modular Space Station.

The overall breadboard concept for the DACS is presented in this section. It is a highly flexible, configuration independent, building block breadboard approach. The DACS breadboard is composed of six basic types of units, as shown in 5.1.2. This approach allows a large number of different DACS configurations to be assembled as an operating data acquisition and control subsystem breadboard. Each configuration concept can then be operated, tested and evaluated for overall DACS concept and performance evaluation.

#### 5.1.2 DACS Breadboard

Six basic types of units, plus additional test equipment comprise the DACS breadboard. These are the following:

- a. Breadboard Modem Unit(s)
- b. Core Bus Interface Unit(s)
- c. Equipment Bus Interface Unit(s)
- d. Interconnecting Cable(s)
- e. Breadboard RACU(s)
- f. Breadboard DBCU
- g. Special Breadboard Test Equipment

The basic requirements for these units are operational compatibility between the various units, interconnection capability and flexibility, operational flexibility, and the required circuitry to demonstrate a wide variety of data acquisition and control concepts and configurations. Each

breadboard unit can be a single "universal" design to satisfy the various conceptual configurations, or it may require multiple units per type to satisfy different configurations. Internal changes by external methods, such as plugs, jumpers, software programs, switches, etc., can be used to provide the required unit flexibility and adaptability.

Provision for additional, as yet undefined, units is also basic to the breadboard concept philosophy. In this manner the DACS breadboard can provide a means for continuing evaluation of data acquisition and control concepts to satisfy the on-going space program requirements definition.

## 5.2 BREADBOARD DESIGN REQUIREMENTS

### 5.2.1 Breadboard Design Criteria

#### 5.2.1.1 Purpose

The purpose of the DACS breadboard is to provide a flexible vehicle for test and evaluation of DACS concepts. Each of the various breadboard units provide this flexibility, in their operation and in their configuration possibilities. The implementation of the breadboard units were designed to accommodate most of the design criteria delineated in the following paragraphs.

#### 5.2.1.2 System Constraints

##### 5.2.1.2.1 General

The DACS breadboard shall be designed within the limits of the following constraints:

- a. The DACS breadboard shall operate as a self-contained entity without the need for external intervention but under external command, control and influence.
- b. Operation of the DACS breadboard will be internally controlled by the Data Bus Control Unit (DBCUC).
- c. All control interaction with the DACS breadboard from external sources will be through the DBCUC.
- d. All breadboard performance evaluation will be provided external to the RACUC's, DBCUC's, and the data bus breadboard units.
- e. The breadboard shall operate with a fixed word length of 8 bits.
- f. The breadboard shall operate with a variable size message structure in all modes, of from zero to 124 data words.
- g. Hardware error detection and encoding shall be provided.
- h. The breadboard units will contain provisions for both fixed and variable coding and decoding of internal commands and data.

- i. All hardware breadboard units shall have a standard disconnect/interconnect scheme for ease of assembly into various configurations.
- j. A method (or methods) shall be provided for simulating faults within the breadboard units.
- k. Provision for test equipment and/or panels, for determining breadboard performance, shall be made.
- l. All breadboards will be non-redundant; redundant breadboard operation will be achieved through the use of multiple breadboard units in redundant configurations, and DBCU/Test Processor operational control programs and interaction.
- m. RACU and DBCU breadboards shall have internal power supplies operating from the primary power source.

#### 5.2.1.2.2 Error Protective Coding Concept

The DACS breadboard shall be designed with the following considerations for hardware error protection.

- a. Hardware shall be provided in both RACU and DBCU breadboard units to perform error protective encoding and detecting on a word or message basis. Correction is not necessary for the DACS breadboard since this can be evaluated off-line if desired.
- b. The encoding or non-encoding of the data shall be selectable by external control.
- c. Provisions shall be made to allow other coding schemes generated and detected external to RACU or DBCU to be passed through the RACU or DBCU.
- d. The data so encoded shall be passed through to external devices in either the encoded or decoded form, or both, for other evaluation, and vice versa.

#### 5.2.1.3 Functional Constraints

##### 5.2.1.3.1 Breadboard Data Bus

###### 5.2.1.3.1.1 General

The data bus breadboard shall be designed within the limits of the following constraints:

- a. The data bus breadboard shall be a self-contained time-division multiplexed communication link utilizing pulse code modulation over a hardwired transmission path.
- b. Bi-phase level (Manchester) data encoding will be utilized by the breadboard for transmitting data and control bits.

- c. The nominal operating frequency is 10 megabits per second.
- d. The longest data source to sink distance is 400 feet.
- e. The longest non-interrupted line segment is 125 feet.
- f. The number of equipments utilizing the data bus assembly in the operating system total less than 150.

#### 5.2.1.3.1.2 Breadboard Modem Units

The breadboard modem units shall be designed within the limits of the following constraints:

- a. Breadboard modem units include all circuitry necessary for bi-phase level modulation and demodulation, clock recovery from the bi-phase level modulated signal, bit timing, preamble decoding, and bus usage by a RACU or DBCU.
- b. Breadboard modem units shall operate in a half of full duplex mode.
- c. Breadboard modem units shall have the capability whereby the output power delivered to the line can be externally adjusted through a limited range including the minimum for data bus operation.
- d. The breadboard modem unit interface with other DACS breadboard elements consists of serial digital NRZ data, serial digital clock signals and various DC control signals.
- e. Equipments utilizing the breadboard modem units will be assumed in close physical proximity, i.e., less than five feet from the modem.

#### 5.2.1.3.2 Breadboard Remote Acquisition and Control Units

The RACU breadboard shall be designed with the following constraints:

- a. The RACU breadboard shall provide the standard interface between the digital data bus modem breadboard and a simulated subsystem functional loop.
- b. The RACU breadboard shall contain the circuitry necessary to accept standard format serial digital data and clock signals from the breadboard modem, to convert the data to signals which are compatible with the subsystem requirements, to generate the required control signals for both the subsystem and for data bus usage, to provide buffer storage for subsystem data, and to provide subsystem data to the bus on command converted to a standard serial digital format.
- c. The RACU breadboard shall be mechanized to respond to two types of messages as listed below:

1. Message A - Transmit Data to DBCU
2. Message B - Receive Data/Commands from DBCU

Responses shall be of three types; no response, acknowledge response, and acknowledge and accept DBCU verification.

- d. Each RACU breadboard shall have two input (receive) interfaces and two output (transmit) interfaces with the data bus breadboard modem(s).
- e. The RACU/modem interfaces shall have independent address recognition circuitry but are not required to have independent control.
- f. A provision shall be made for an external test interface in serial digital form consisting of the data received via the data bus (output) and accepting data for transmission via the data bus (input).
- g. The external serial digital test interface shall have provision for parity generation of data at its input or not, and similarly parity checking the output data or not, externally selectable.
- h. Internal data word buffer storage of 128 words minimum shall be provided for input and output messages.
- i. Data transfer to and from the breadboard modem shall operate at a nominal frequency of ten megabits per second.
- j. The RACU breadboard shall operate at appropriate time from three clock sources; either the receive clock from the modem, its own internal clock, or an external clock source via the test interface.
- k. A subsystem functional loop simulator interface shall have provision for variable numbers of DC analog and discrete signals to be multiplexed at varying sample rates, converted to digital form, and formatted.
- l. Special provision for local indication of responses to non-data commands from the DBCU should be considered.
- m. Provision shall be made for an external preprocessor interface.
- n. RACU addresses shall be externally pre-set.
- o. All RACU operation will be under higher level control by the DBCU. Operation will be specified by the DBCU commands in each message. RACU operational sequences can be performed by hardware and/or software techniques.
- p. A provision for variable decoding of DBCU commands shall be included to allow changing RACU operational modes and command responses.

### 5.2.1.3.3 Breadboard Data Bus Control Unit

The DBCU breadboard shall be designed within the limits of the following constraints:

- a. The DBCU shall provide all command and control capabilities for fully exercising the DACS breadboard.
- b. The DBCU breadboard shall contain the circuitry necessary to originate and control all messages for the DACS breadboard, to communicate with RACU breadboards via the breadboard modem units and data bus and to operate the DACS breadboard with or without test processor interaction.
- c. Two types of messages shall be originated by the DBCU breadboard as listed below:
  1. Message A - Request Data from an RACU
  2. Message B - Transmit Data/Command to an RACU
- d. The breadboard DBCU shall have dual, switch selectable, interface capability for communicating with breadboard modem units.
- e. Provision shall be included for an external test I/O interface consisting of serial digital data for transmission to RACU's and data received from RACU's.
- f. The external serial digital test interface shall have provision for parity generation and detection of I/O data or not, externally selectable.
- g. Internal data word storage shall be provided for message data sequences and buffering (minimum size of 128 words).
- h. Data transfer to and from the breadboard modem units shall operate at a nominal frequency of ten megabits per second.
- i. The DBCU breadboard shall operate at appropriate times from any of three clock sources; the receive clock from the modem, an internal clock source, or an external clock source via the test interface.
- j. Message generation shall be under internal program control, including such message options and factors as message type, message size, RACU acknowledge, command verification, data retransmission, procedures for operation with errors detected and improper response conditions, and interaction with external equipments.
- k. The internal DBCU breadboard operational program shall be both externally selectable and changeable via the test processor, test panel and breadboard user.

- l. A test processor and test panel interface shall be provided for parallel and/or serial digital data transfer and DBCU (and therefore DACS breadboard) higher level control.
- m. The DBCU breadboard shall be capable of providing internal operation status and DACS breadboard operational status data to the test panel and/or test processor. This status data will include that obtained from the RACU's, data bus breadboard status, the DBCU status, the errors detected, the operational DACS control modes being utilized (see item j), and improper DACS breadboard operation.

#### 5.2.1.3.4 Special Breadboard Test Equipment

Special test equipment for the DACS breadboard shall be furnished as required. Its design and function shall be consistent with the following design criteria.

- a. Special breadboard test equipment shall be kept to a minimum consistent with the breadboard evaluation, test and maintenance requirements.
- b. Wherever possible standard laboratory test equipment should be considered.
- c. Special breadboard equipment necessary to provide simulation of breadboard data bus features (such as fault introduction) shall be provided.
- d. Test and display panels shall be provided where necessary to externally control and/or evaluate the DACS breadboard operation and performance.

#### 5.2.1.4 Breadboard Operating Characteristics

##### 5.2.1.4.1 General -

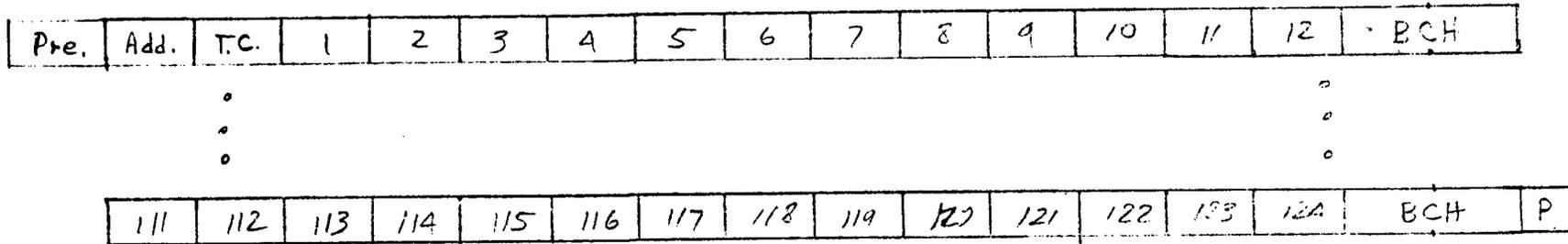
Digital data transmission between RACU's and the DBCU shall be in a word serial, bit serial time division multiplex (TDM) format over standard buses. The transmission of standard messages shall be accomplished using half duplex (two way transmission, but not simultaneously) operation controlled by the DBCU. One transmission cable set (containing both data and clock) shall constitute a multiplex data bus. Breadboard modem units will be connected to the buses in parallel (party line) fashion, therefore all units connected to a bus will see all data on the bus.

##### 5.2.1.4.2 Breadboard Standard Message and Word Format

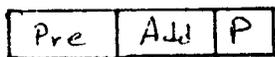
All data transmitted over multiplex buses interfacing with a DBCU shall be transmitted as standard messages. A standard message shall be composed of preamble, address, transmission code, data fields as applicable, error protective coding, and a postamble. Message formats are shown in Figure 5-1.

Data to a RACU, Normal Mode, BCH Mode

Command

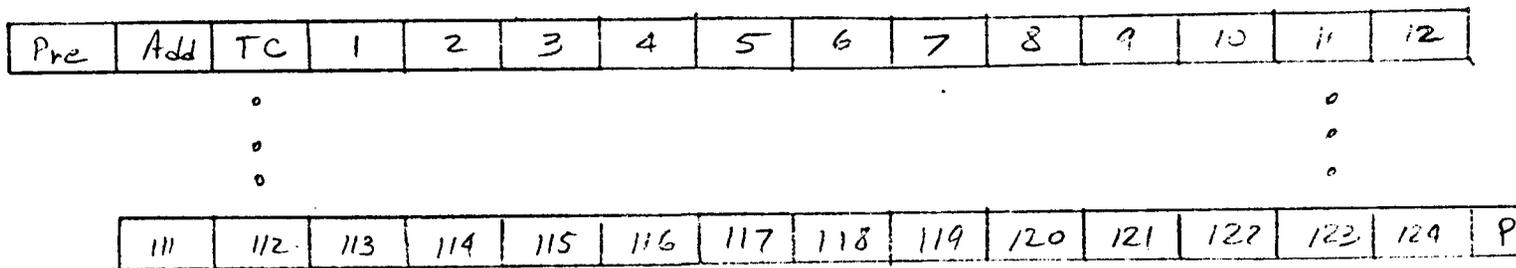


Response



Data to a RACU, Normal Mode, Non-BCH Mode

Command



Response

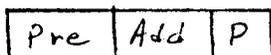


Figure 5-1. MESSAGE FORMATS (Page 1 of 2)

Data from a RACU, Normal Mode, BCH Mode

Command

|     |     |    |     |   |
|-----|-----|----|-----|---|
| Pre | Add | TC | BCH | P |
|-----|-----|----|-----|---|

Response

|     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| Pre | Add | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12  | 13  | BCH |
|     |     | 0   |     |     |     |     |     |     |     |     |     |     |     | 0   |     |
|     |     | 0   |     |     |     |     |     |     |     |     |     |     |     | 0   |     |
|     |     | 0   |     |     |     |     |     |     |     |     |     |     |     | 0   |     |
| 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 | 123 | 124 | 125 | BCH | P   |

Data from a RACU, Normal Mode, Non-BCH Mode

Command

|     |     |    |   |
|-----|-----|----|---|
| Pre | Add | TC | P |
|-----|-----|----|---|

Response

|     |     |     |     |     |     |     |     |     |     |     |     |     |     |    |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|----|
| Pre | Add | TC  | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12 |
|     |     | 0   |     |     |     |     |     |     |     |     |     |     |     | 0  |
|     |     | 0   |     |     |     |     |     |     |     |     |     |     |     | 0  |
|     |     | 0   |     |     |     |     |     |     |     |     |     |     |     | 0  |
| 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 | 123 | 124 | 125 | P  |

Figure 5.1. MESSAGE FORMATS (Page 2 of 2)

5-9

SD 72-SA-0114-3



#### 5.2.1.4.3 Preamble

Each message shall be preceded by a no-data period called a preamble. Recognition of the preamble shall be used to identify the word following as an address word, and the start of a message. The preamble shall be 9 bits in length.

#### 5.2.1.4.4 Control Fields

Control fields shall be used to initiate all data exchanges (messages) and shall originate only in the DBCU. A control field shall provide one of three functions: to request data transmission from an RACU; or to identify data transmitted from the DBCU to an RACU; and to command an RACU to take some action other than to transmit data. A control field shall be composed of two separate words: an eight-bit address word and an eight-bit transmission code word.

The address field shall contain a unique code identifying the unit on the bus to which the communication is being directed. The RACU acknowledges receipt of a command by transmitting its preset address word back to the DBCU following a preamble.

The transmission code word shall indirectly identify the required RACU operation for transmission or reception of data following the control word, or command the RACU to take some action other than transmit data. The transmission code word is used to define a starting location in the RACU program memory which in turn contains the sequence of instructions defining the command operation. These command operations in RACU memory will identify message length and content, and may define direct command functions such as turning on a portion of a subsystem functional loop, turning on a preprocessor, setting internal flip-flops, service requests, etc.

Since the RACU program memory needs to be programmable, special encoding techniques of the transmission code word may be evaluated. This extra protection of commands is a feature for breadboard evaluation, possible necessary since all RACU operation is a function of this command field structure. The message block type error protection option for the breadboard gives no indication of error location, nor does it tell the RACU how to operate when an error is detected. This secondary protection of commands allows RACU operation when errors are detected by the primary error control hardware.

By changing the command subroutines, various commands could be implemented by the DACS breadboard. This flexibility is very desirable for more complete breadboard utilization and adapting to changes in breadboard configurations. These breadboard changes will be desirable if and when the breadboard is expanded to redundant operations, or adjusted in light of laboratory evaluation to as yet unspecified configurations and operation. This provision also allows the DACS breadboard hardware to better meet the present and future technology goals without the need for new breadboard units as time progresses. Any and all methods for achieving this should be considered.

Command generation occurs at the DBCU breadboard, and/or in conjunction with the test processor. The DBCU shall have the capability of redefining command operation and the associated DBCU control subroutines for the different commands. The programmable control memory of the DBCU shall be provided just for this flexibility. The test processor can also be utilized directly for this function, either by reloading the DBCU control memory through the test processor interface or by utilizing the external test interface at the DBCU.

#### 5.2.1.4.5 Data Words

Data words originate either in the DBCU or a RACU and contain the data identified by the command field of the control word initiating the message. If the control word issued non-data command, the RACU can transmit a data word acknowledging receipt of the command. Data words shall be composed of 8-bit bytes. The total data field of the message may contain either packed data, i.e., data representing more than one parameter, or a numeric parameter and a group of discrettes, all discrettes, or acknowledge data. More than one data word can be used for data requiring greater than 8 bits.

#### 5.2.1.4.6 Breadboard Bus Control

5.2.1.4.6.1 General - The DBCU will select the data buses available for data transmission, and will initiate a data exchange by transmitting the appropriate address and transmission code over the selected bus(es). Only one of the two (redundant) buses to each RACU need be used for data transmission at any particular time. Data words transmitted by the DBCU shall be transmitted on the same bus(es) as the initiating control word. Peripheral units shall transmit the data defined by the transmission code over the bus(es) defined by DBCU commands.

5.2.1.4.6.2 Data Transmission from RACU's - A RACU shall transmit only after the receipt of a valid control field requiring data transmission and only if capable of initiating the data word(s) exactly when required. A valid control field requiring data transmission from a RACU shall meet the following criteria.

- a. A preamble shall have been detected on the data transmission line prior to receipt of the first bit of the control field.
- b. The code represented by the address field shall compare to the address code preset in the RACU.
- c. The code represented by the transmission code shall be recognized as a valid command by the RACU.
- d. The control field words pass all error detecting checks made by the RACU.
- e. No data dropouts shall have occurred during the 16 clock periods (plus parity when used) immediately following the start of the control word.

5.2.1.4.6.3 Data Acceptance by a RACU - A RACU shall accept data only after the receipt of a valid control field indicating data are to be transmitted by the DBCU to the RACU. The validating criteria for a control field directing a RACU to accept data shall be the same as that directing transmission. The number of words received must also agree with the number specified indirectly by the transmission code.

If an error or dropout is detected by the RACU during reception of any single data word or any block of data words, only that word or block need be invalidated.

5.2.1.4.6.4 Data Acceptance by the DBCU - If data words are not detected by the DBCU from a RACU after transmission of a control word requiring a response (i.e., a request for data or an acknowledge) the DBCU terminal shall flag a no-response condition. This no-response condition indicates that one of the following conditions exists:

- a. The RACU failed to recognize the control field as valid due to a signal dropout, a transmission error, or a transient malfunction.
- b. The peripheral was unable to reply within the required time.
- c. The communications link between the DBCU and the RACU has failed.
- d. The RACU has failed.

When a no-response condition is recognized by the DBCU, the DBCU may, at the option of its program, reinterrogate the RACU by retransmitting the same control field. If no response is received after the reinterrogation, the DBCU may, by program option: switch to another bus and repeat the interrogation process; ignore the no-response condition and go on; or, execute a test program to isolate the malfunction.

If a parity type error, dropout, invalid response, or RACU status error is detected by the DBCU during the reception of the data, the DBCU by program option, may request a repeat transmission by retransmitting the control field as a new message sequence.

## 5.2.2 Breadboard Performance Requirements

### 5.2.2.1 Data Rate

The DACS breadboard should operate at a nominal data rate of ten megabits per second. This data bit rate includes all overhead. The average data rate and message data rate will be correspondingly less than the peak bit rate depending on housekeeping overhead and bus utilization controlled by the DBCU breadboard.

### 5.2.2.2 Switching Capability and Delays

The DACS breadboard shall be capable of switching from a condition where the DBCU is transmitting data to a RACU breadboard, to a condition where the addressed RACU is transmitting back to the breadboard DBCU. This

switching should be done with a minimum amount of 'turn-around' time between the DBCU transmission and the RACU response.

This total sequence defines a request-acknowledge message sequence (non-acknowledged command messages are defined the same). This "turn-around" interval shall be distinguishable from the interval of "dead time" (or no transmission) between message sequences. All switching and propagation of data signals, enables and service signals shall be as fast as possible without excessive effect on error rates, cost, reliability, etc.

#### 5.2.2.3 Synchronization

The DACS breadboard shall be capable of being used by asynchronous sources. The probability of loss of data or synchronism at start-up time shall be similar to the probability of loss of data or synchronism in the middle of a message after the system has been synchronized.

#### 5.2.2.4 Resolution

The RACU breadboard analog signal conversion is specified with a resolution of 8 binary bits.

#### 5.2.2.5 Accuracy

The conversion accuracy for analog signal inputs to the breadboard RACU is specified as one percent, worst case.

#### 5.2.2.6 Bit Error Rate

The bit error rate of the DACS breadboard shall not exceed  $1 \times 10^{-8}$  under normal operation (benign external noise environment).

#### 5.2.2.7 Clock Frequency and Stability

Internal clock sources are required in all RACU and DBCU breadboard units. The clock frequency shall be 10 MHz or a multiple thereof. The clock frequency stability shall be consistent with normal design practices with no temperature compensation design requirements.

### 5.2.3 Interface Definition

#### 5.2.3.1 General

This section describes the various breadboard interfaces for all DACS breadboard units. Four main interfaces are provided between the various breadboard units. The major interface is between the breadboard modem units and the breadboard RACU and DBCU. The RACU shall also have interface provisions for a subsystem functional loop simulator and a pre-processor. The DBCU shall also have an interface provided for a test processor and a test panel. Both the RACU and DBCU breadboards shall have an external test interface for communications evaluation.

## 5.2.3.2 Data Bus/RACU/DBCUC Interface Definition

### 5.2.3.2.1 General

The data bus breadboard has an external interface at each breadboard modem unit. This interface is used for all access to the bus subsystem by the other DACS elements, be they RACU's or DBCU's. It can also be used for monitoring and test purposes in the breadboard evaluation of the DACS.

The interface is described functionally in the next section. The physical details are included in paragraph 5.2.3.2.3. Allowance for breadboard test utilization of any breadboard modem unit interface requires some flexibility in the provisions for this interface. Quick and easy disconnect and connection shall be provided by all breadboard units to allow configuration changes and laboratory evaluation of the numerous options available.

### 5.2.3.2.2 Functional Definition

5.2.3.2.2.1 General - The breadboard modem unit RACU/DBCUC interface is divided into three parts as listed below:

- a. Receive
- b. Transmit
- c. Power

5.2.3.2.2.2 Receive - The receive interface provides data output from the bus assembly to the RACU or DBCUC. This data is demodulated directly from the data link and has no physical or electrical interaction with the transmit interface.

The diagrammatic representation of this interface is shown in Figure 5-2. Four signals are shown crossing the interface between the breadboard modem unit and the equipment utilizing the receive interface. These four signals are the following:

R1) Serial NRZ Data (DR)

This is the digital data that was transmitted on the bus, in serial form. This line is active whenever the bus is active. The signal uses a non-return to zero format.

R2) Serial Clock (CR)

The clock signal is a series of pulses derived directly from the information on the bus. This line is active whenever the bus is active. The clock signal is a square wave pulse train, which is the timing source for the serial NRZ data.

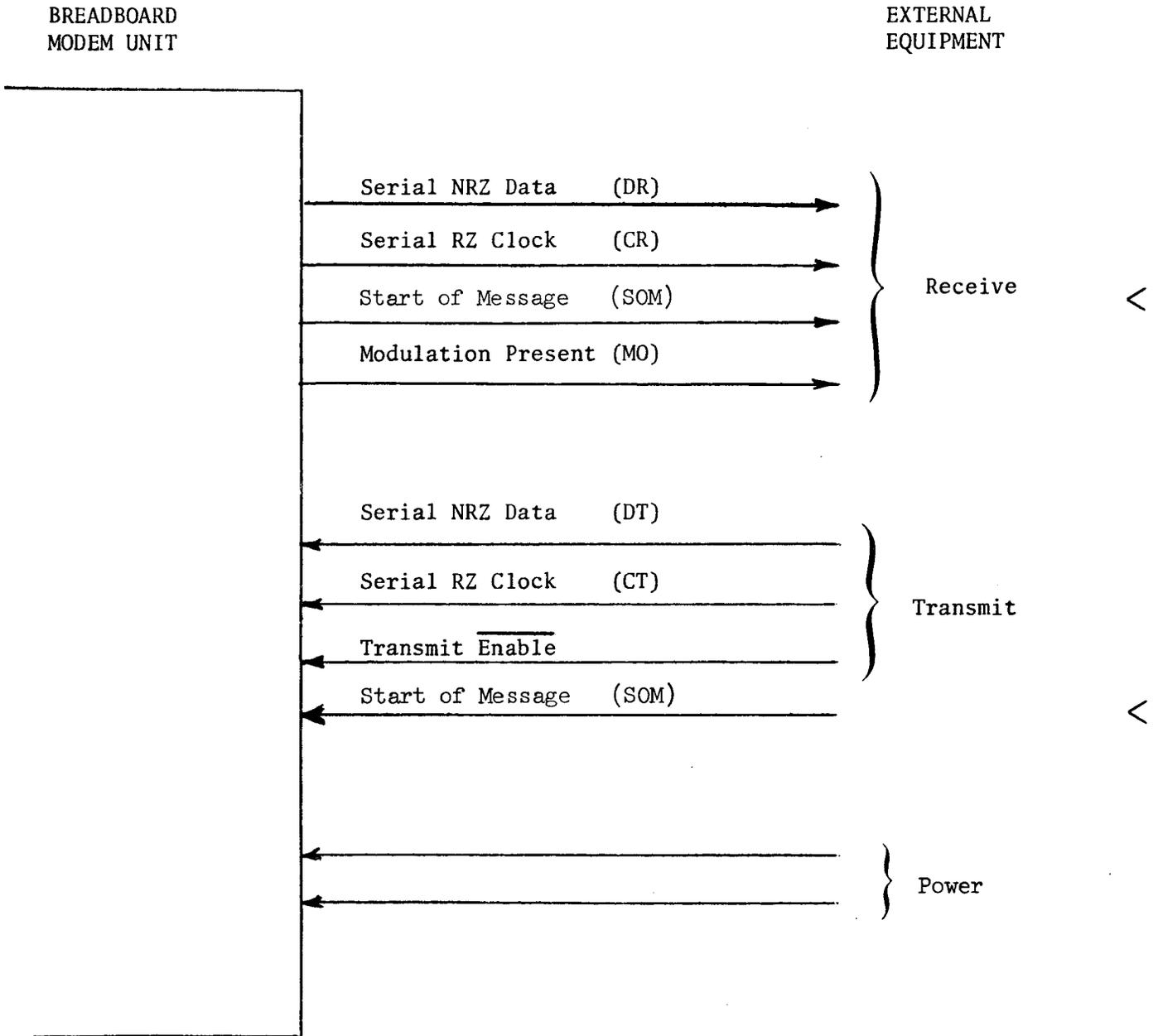


Figure 5-2. Breadboard Modem Unit Functional Interface

R3) Start of Message (SOM)

A pulse is used to indicate the start of all messages on the bus. This signal goes "false" two bits prior to the start of the first bit of the address word of each message and stays "false" until the first bit of the address word of the message. This signal is "true" during all other intervals.

R4) Modular Present (MO)

A DC level is used to signal the operation of the bus to the interfacing equipment. Any bi-phase level signal on the bus will cause this signal to go to a "true" state, including the preamble at a message.

The serial clock signal is used to strobe the serial data signal and to represent the timing of the data. It must be used to distribute any time-division multiplexed digital data. The modulation present signal "frames" the data of interest to the external user and is provided so that external equipment can monitor bus activity and prepare for bus usage itself.

5.2.3.2.2.3 Transmit - The transmit interface provides a point for insertion of data onto the breadboard data bus. It can be utilized by any external breadboard equipment, usually a RACU or DBCU breadboard unit. The data presented to the breadboard modem unit via this interface, with the appropriate control signals and conditions, is modulated and transmitted on the breadboard data bus to which the modem is connected.

Figure 5-2 also shows the diagrammatic representation of the transmit interface. Four signals cross the transmit interface between the breadboard modem unit and the external equipment. These signals are described below:

T1) Serial NRZ Data (DT)

This is the digital data to be transmitted on the data bus, in serial form. The presence of a signal is controlled by the external device. The signal is in a non-return to zero format.

T2) Serial Clock (CT)

The clock signal is provided to the breadboard modem unit as a square wave pulse train. This signal is continuously present when equipment is connected to this interface. This signal is used by the breadboard modem unit to produce the bi-phase level encoding of the serial NRZ data.

T3) Transmit Enable

A DC level is used to signal the external equipment transmit interval to the breadboard modem unit. A complementary form of the signal is used so that an unused interface is not signalling a transmit interval. A "false" level indicates a transmit enable interval.

#### T4) Start of Message (SOM)

A pulse is used to indicate the start of all messages on the bus. This signal goes "false" two bits prior to the start of the first bit of the address word of each message and stays "false" until the first bit of the address word of the message. This signal is "true" during all other intervals.

The serial clock signal is used by the breadboard modem unit modulator and represents the timing of the serial digital data. This serial clock signal is not the same signal as that provided at the receive interface.

5.2.3.2.2.4 Power - This interface is used to supply power to the breadboard modem unit from an external source. A maximum of three secondary DC voltage levels plus return are required.

#### 5.2.3.2.3 Physical Interface Definition

5.2.3.2.3.1 General - The physical interface between the breadboard modem unit and the RACU's or DBCU shall be provided by three separate connectors for the interfaces illustrated in Figure 5-2.

#### 5.2.3.3 RACU Only Interfaces Definition

##### 5.2.3.3.1 General

The breadboard RACU only has a requirement to interface with a subsystem functional loop simulator unit and a preprocessor. The detailed specification for this interface will be left to the RACU contractor. The functional design constraints on the interface requirements were enumerated in Section 5.2.1.3.2 as Items a, b, j, k, l, m, n, and o. The design goal for this interface is one which can be used to evaluate the feasibility of a standard RACU interface for subsystem usage. To this end, a modular, building block concept is deemed appropriate.

#### 5.2.3.4 DBCU Only Interfaces Definition

##### 5.2.3.4.1 General

The DBCU breadboard shall interface with both a test processor and a test panel. The detailed specification of test panel interface will be left to the DBCU contractor. Section 5.2.1.3.3 discusses the various functional constraints on the interface design requirements. They are listed in that section as items a, b, i, j, k, l, and m. The design of this interface shall provide the required flexibility for fully exercising and evaluating the DACS breadboard via the DBCU.

##### 5.2.3.4.2 DBCU/Test Processor Interface Definition

The DBCU/Test Processor interfaces will be used for both on-line and off-line DACS breadboard control. It will also be used for data collection for

DACS breadboard performance evaluation. The test processor is to be determined by MSC. The detail specification and required uses of this interface have been documented by the DBCU contractor in ICD #AN 26465 (latest revision dated January 21, 1972).

#### 5.2.3.4.3 DBCU/Test Panel Interface Definition

The DBCU breadboard shall interface with a test panel for operational control and mode selection. This interface shall be parallel digital as required. The detail specification and requirements for this test panel/ DBCU interface will be left to the DBCU contractor.

#### 5.2.3.5 External Test Interface Definition

##### 5.2.3.5.1 General

An external test interface shall be provided by both the RACU breadboard units and the DBCU breadboard unit. The design criteria and constraints for this interface are discussed in Section 5.2.1.3.2, Items f and g and Section 5.2.1.3.3, Items e and f for the RACU and DBCU breadboard units, respectively. This interface shall be provided for breadboard purposes only, to allow monitoring the serial digital communication data stream at all breadboard units. It also provides an alternate means of generating message data, either for special test sequence or for normal operations, bypassing the RACU or DBCU internal message storage.

##### 5.2.3.5.2 Functional Definition

The external test interface is composed of eight signals. These signals are similar in many respects to the breadboard modem/RACU and DBCU interface. They provide both input and output serial digital data and clock signals plus control signals. The external clock source also uses this interface.

A diagrammatic representation of this interface is shown in Figure 5-3. The signals are defined as follows:

##### X1) Serial NRZ Data (DRE)

This signal contains serial digital data in a non-return-to-zero format. This is the received message data words at this breadboard. This line is active when the breadboard is receiving data from the data bus breadboard modem.

##### X2) Serial Clock (CRE)

The clock signal is a train of square wave pulses. This is the same clock signal as received from the breadboard modem unit and is the timing source for X1. This line is active when the breadboard is receiving data.

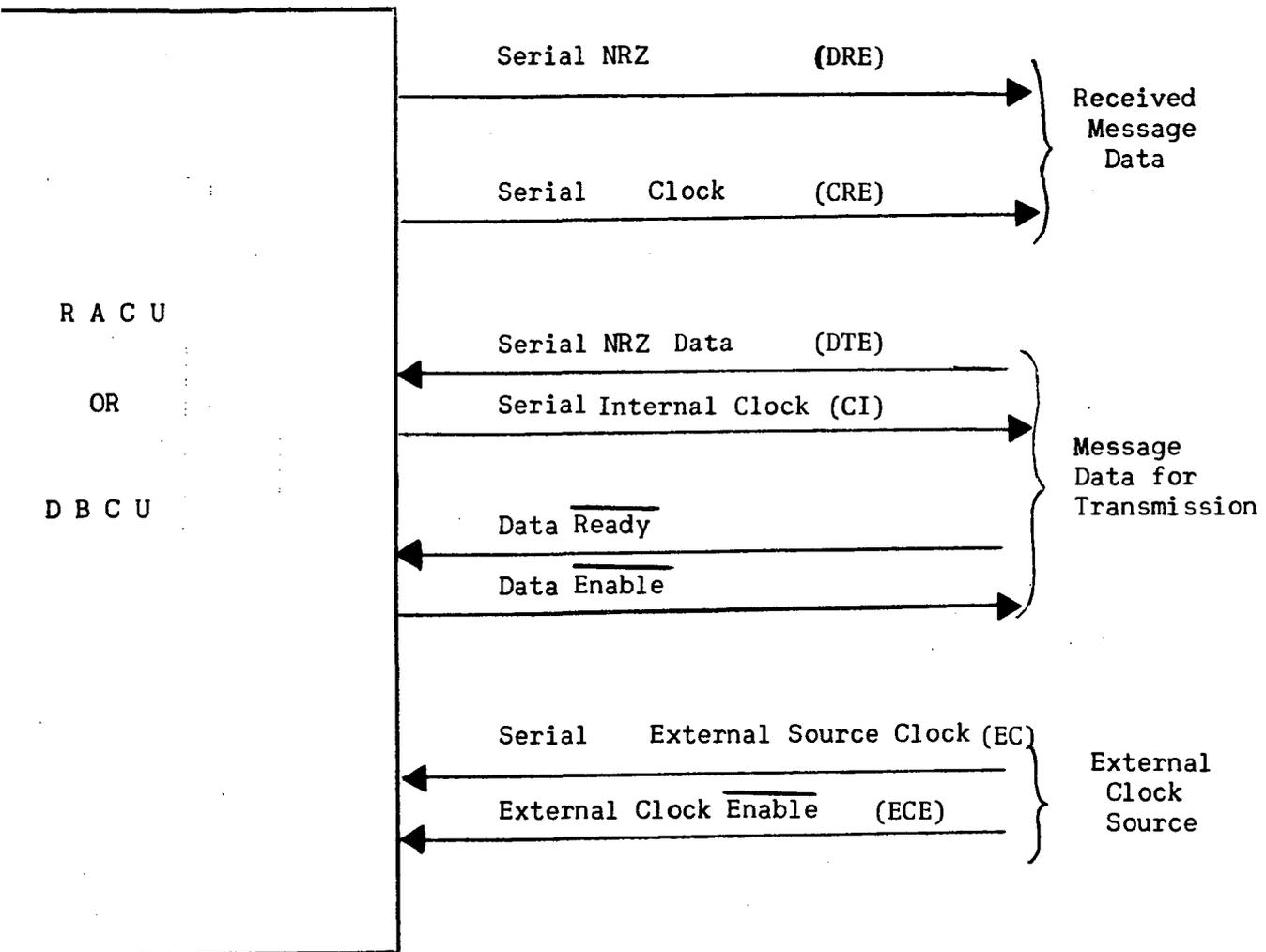


Figure 5-3. Breadboard External Test Function Interface

X3) Serial NRZ Data (DTE)

This signal contains serial digital data in a non-return-to-zero format. The data content is under external control, but the data timing is provided by the breadboard unit (signal X4). This data is used by the breadboard unit as its message for transmission on the data bus breadboard. The activity of this signal is controlled by the breadboard and the external device.

X4) Serial Clock (CI)

The clock from the breadboard unit to the external device is used to control the timing of the data signal X3. This clock is the one which the breadboard is using for data transmission to the breadboard modem unit. This signal is a continuously present square wave pulse train.

X5) Data Ready

This DC level indicates whether the external device is ready to provide data signal X3 to the breadboard unit. A "false" level indicates it is ready. An unused input (open input) is defined as a "true" level indicating no data X3 is available. When the external device is ready (signal is "false") the breadboard unit shall use data signal X3 for all message data, as many bits as are required by the message control word.

X6) Data Enable

The breadboard unit notifies the external device that it is accepting data signal X3 by this DC level. This "gating" signal is "false" when data is being accepted by the breadboard unit.

X7) Serial External Source Clock (EC)

This clock signal is provided by an external source for breadboard unit operation. For example, when this clock source is being used, clock signal X4 will be the same frequency. The selection and use of the external clock signal for breadboard unit operation is under external control by signal X8.

X8) External Clock Enable

This external DC signal selects the clock source, either internal to the breadboard or external, that shall be used at any time. A "false" level selects the external clock source, a "true" level the internal (to the breadboard unit) source. An unused input is defined as a "true" level.

It is anticipated that various delays will exist between the external signals and the appearance of those same signals on the data bus. This is

dependent on the internal mechanization of the RACU and the DBCU breadboard units and is allowable. Any adjustments for these by external monitoring devices will be the responsibility of the external device. All delays, if required, shall be fixed and definable for the various signals.

The use of the signals X3, X4, and X6 is specified externally by signal X5. This external selection of message data is therefore under DACS breadboard user control. The message structure, number of words, etc., must be co-ordinated with the DBCU program in progress for proper operation. This function is controlled by the user through the test processor/test panel interface.

External source clock selection is also available to the user. This is one means whereby the communication data rates can be controlled. It also allows different data rates for transmit and receive at a single location (and therefore on the data bus).

#### 5.2.3.5.3 Physical Interface Definition

5.2.3.5.3.1 General - The physical interface with the breadboard units shall be provided by three separate connectors. These three groups are shown in Figure 5-3 and are listed below:

- a. Received Message Data
- b. Message Data for Transmission
- c. External clock Source

This allows the three groups of signals to be used by different external equipment at different times and for different purposes if desired, or not at all.

#### 5.2.4 Breadboard Configuration Requirements

##### 5.2.4.1 General

The breadboard concept for the Data Acquisition and Control Subsystem is a highly flexible, building block approach. All breadboard units are available for assembly into a large number of different configurations for laboratory test and evaluation. All breadboard hardware units must be compatible and capable of the interconnections necessary to demonstrate a wide variety of DACS breadboard configuration concepts.

##### 5.2.4.2 DACS Breadboard Hardware Units

The DACS Breadboard Units included in this requirements document divide into nine areas. Each unit within the areas may require more than one hardware implementation for breadboard usage. The nine areas are the following:

- (1) Breadboard Modem Unit(s)
- (2) Core Bus Interface Unit(s)
- (3) Equipment Bus Interface Unit(s)
- (4) Breadboard Cable Sections
- (5) Breadboard Termination Unit(s)
- (6) Breadboard RACU(s)
- (7) Breadboard DBCU
- (8) Special Breadboard Test Equipment

#### 5.2.5 Electrical Requirements

##### 5.2.5.1 Electrical Power Supply

The electrical power supplied to all breadboard units is 115 VAC, 60 Hz, single-phase, two-wire plus ground. The data bus breadboard units require external secondary DC voltage levels. These will be obtained from laboratory type power supplies provided for this purpose.

##### 5.2.5.2 Power Consumption

Power consumption of all DACS breadboard units shall be minimized to the extent possible consistent with the requirements and good design practice. The number of secondary DC voltage levels supplying power to the data bus breadboard units shall also be minimized.

##### 5.2.5.3 Power Failure and Transients

The DACS breadboard units shall not be designed to operate through power failures or abnormal transients.

##### 5.2.5.4 Overvoltage Protection

The secondary DC voltage level inputs to the data bus breadboard units shall have some form of overvoltage protection against inadvertent application of incorrect DC levels. Such protection may take the form of zener diode clamps to protect the electronic circuitry.

#### 5.2.6 Packaging and Construction Requirements

##### 5.2.6.1 General

A breadboard unit is defined as one which performs the same functions and according to the same characteristics as those defined by the hardware design. It should make maximum use of pre-developed and off-the-shelf devices and functional elements, although the component technology should be the same as that defined for the actual operational design. Breadboard units do not have to be reduced to production techniques or qualified, but their quality must be such that they can be expected to survive the handling normally associated with extensive laboratory testing and operation.

Where practical, all breadboard equipment shall be mountable in standard 19-inch equipment racks. To the greatest extent possible, breadboard electronic hardware shall be contained in equipment drawers and mounted in high-density packaging panels in a manner permitting easy access to all components for test or repair.

#### 5.2.6.2 Weight and Size

Breadboard units shall, to the extent possible, be contained in standard 19-inch equipment rack drawers. Weight and size should be minimized.

#### 5.2.6.3 Modular Construction

Each breadboard unit shall be fabricated using modular construction wherever possible. The units shall be designed with a logical and functional breakdown of replaceable electro-mechanical and electronic modules.

#### 5.2.6.4 Reliability

Breadboard assemblies and subassemblies shall be repairable. The designer shall consider ease of repair and rework of the unit as an important part of the engineering. Maximum accessibility for wiring modification and repair shall be provided. Circuit component shall be legibly and permanently identified. Connectors shall be permanently identified.

#### 5.2.6.5 Standardization and Interchangeability

Standardization is defined here as the effort to select, design, and manufacture parts, assemblies, units, etc., so that they are identical and interchangeable without special considerations or adjustments. Standardization for interchangeability of breadboard units shall be a primary requirement.

#### 5.2.6.6 Accessibility

All breadboard unit subassemblies and individual components shall be arranged to provide ready access for removal and replacement of parts. As a general rule, it will be unacceptable to remove wires, subassemblies or parts in order to gain access to terminals, connections, mounting screws, etc. If this is unavoidable, those parts which must be displaced shall be so wired and mounted that they can be moved without being disconnected from the circuit or cause circuit detuning or instability.

#### 5.2.6.7 Test and Adjustment

When access for potentiometer adjustments, test connections, etc., is necessary, this access shall be free of interference from other components, wiring, subassemblies, etc., and clearly marked. Test points shall be easily accessible and clearly identified to facilitate rapid troubleshooting and repair.

#### 5.2.6.8 Special Tools

The design of the breadboard units shall be such that the need for special tools for tuning, adjustment and servicing shall be kept to an absolute minimum. Those required for operational adjustments shall be supplied securely affixed to the unit and readily accessible.

#### 5.2.6.9 Cooling Requirements

All breadboard units will be cooled by free convection with no cooling provided external to the unit. If necessary, a fan(s) shall be mounted integral to the breadboard unit inside the rack mounted drawer.

#### 5.2.7 Environmental

##### 5.2.7.1 Operative

The DACS breadboard shall be designed for normal performance and operation under the following conditions associated with exclusive laboratory testing, handling and operation.

##### 5.2.7.1.1 Temperature

The breadboard DACS units shall be designed to operate over the standard commercial component temperature range of 0°C to 70°C.

##### 5.2.7.1.2 Pressure

The DACS breadboard units shall be designed to operate over a pressure range corresponding to sea level down to 5,000 feet above sea level.

##### 5.2.7.1.3 Vibration and Shock

The DACS breadboard units shall be designed to operate through the chock and vibration due to normal laboratory environment handling requirements.

#### 5.2.8 Reliability

##### 5.2.8.1 Modularity

The DACS breadboard shall be composed of a minimum number of different units. Each identical unit shall be physically and electronically interchangeable. A replacement of a breadboard unit shall not affect the DACS operation.

##### 5.2.8.2 Propagation of Failure

As a design goal, the failure of one component shall not cause other components to fail. The breadboard data bus shall be insulated and isolated so that extraordinary effort is needed to produce a high voltage on these

lines and disable the whole bus. The data bus breadboard shall be designed so that the application of voltages up to 100 volts on an interface junction shall not cause components on other junctions or at other bus interface connections to fail.

#### 5.2.8.3 Transmit Security

As a design goal, no single failure in a breadboard modem or repeater unit shall cause that unit to inadvertently transmit over the breadboard data bus. As a design goal, no single component failure in a DBCU or RACU breadboard unit shall cause that unit to continuously signal a desire to transmit (signal T3 - Transmit Enable) to the breadboard modem unit.

## 6.0 DATA BUS BREADBOARD PERFORMANCE AND INTERFACE REQUIREMENTS

### 6.1 GENERAL REQUIREMENTS

#### 6.1.1 Introduction

The digital bus breadboard is an integral part of the DACS breadboard. The DACS breadboard is an engineering model but is representative of the concepts for the Data Acquisition and Control Subassembly for the Data Processing Assembly of the Modular Space Station. A breadboard unit is defined by the NASA as a unit which performs the same functions and according to the same characteristics as those defined by the hardware design.

The communication spine for the DACS breadboard is provided by the data bus breadboard. All communication between the breadboard remote acquisition and control units (RACU's) and the data bus control unit (DBCUC) utilize the data bus breadboard. This communication is also controlled by these other DACS breadboard unit.

The breadboard concept for the digital data bus is presented in this section as a highly flexible, configuration independent, building block approach. The data bus breadboard is composed of four basic types of units. This approach allows a large number of different data bus configurations to be assembled as part of the laboratory breadboard. Each configuration concept can then be evaluated for data bus performance, as well as performance of the other breadboard hardware units using the data bus, for overall DACS concept evaluation.

#### 6.1.2 Data Bus Breadboard

Four basic types of units plus all additional equipments and interconnecting cables, comprise the data bus breadboard. These are the following:

1. Breadboard Modem Unit(s)
2. Core Bus Interface Unit(s)
3. Equipment Bus Interface Units
4. Breadboard Termination Unit(s)
5. Breadboard Cable Sections
6. Special Breadboard Test Equipment

The basic requirements for these units are operational compatibility between the various blocks, interconnection capability and flexibility, and the required circuitry to demonstrate a wide variety of data bus concepts and configurations. Each breadboard unit can be a single "universal" design to satisfy the various conceptual configurations, or it may require multiple units to satisfy different configurations. Internal changes by external methods, such as switches, etc., can be used to provide the required unit flexibility and adaptability.

Provisions for additional, as yet undefined, units is also basic to the breadboard concept. In this manner the DACS breadboard can provide a test bed for continuing evaluation of concepts to satisfy the changing space station requirements.

Section 6.2.1 of this breadboard requirements document establishes the system design criteria and the unit-by-unit functional design criteria. The performance requirements are included in Section 6.2.2. The breadboard interface definition is included in Section 6.2.3, and the various breadboard conceptual configurations are documented in Section 6.2.4. Also included in Sections 6.2.5 through 6.2.8 are the electrical requirements, the packing and construction requirements, environmental considerations, and reliability design goals.

## 6.2 BREADBOARD DESIGN REQUIREMENTS

### 6.2.1 Breadboard Design Criteria

#### 6.2.1.1 Purpose

The purpose of the data bus breadboard is to provide a reliable and flexible building block type communications network between the Data Bus Control Unit (DBCUC) breadboard and a number of Remote Acquisition and Control Unit (RACUC) breadboards. This breadboard will be used for evaluation of data bus concepts and performance, as well as the evaluation of the remaining DACS hardware elements.

#### 6.2.1.2 System Constraints

The data bus breadboard shall be designed within the limits of the following constraints:

1. The data bus breadboard will be a self-contained, time-division multiplexed communication link utilizing pulse code modulation over a hardwired transmission path.
2. Bi-phase level (Manchester) data encoding will be utilized by the breadboard for transmitting data and control bus.
3. The operating frequency range is nominally 10 megabits per second.
4. All data and control bits transmitted on the data link, regardless of their logic states, shall contain equal amounts of energy. Synchronizing bits may have higher energy logic states.
5. The longest data source to sink distance is 400 feet.
6. The longest non-interrupted line segment is 125 feet.
7. The number of equipments utilizing the data bus assembly in the operating system total less than 150.

8. All hardware elements of the data bus breadboard must have a standard disconnect/interconnect scheme for ease of assembly into various configurations.
9. A method (or methods) shall be provided for simulating shorts and/or opens of various points and hardware elements within the data bus breadboard.
10. Provisions for test equipment sources and sinks, for determining data bus breadboard performance, shall be made.

#### 6.2.1.3 Functional Constraints

6.2.1.3.1 Breadboard Modem Units. The breadboard modem units shall be designed within the limits of the following constraints:

1. Breadboard modem units include all circuitry necessary for bi-phase level modulation and demodulation, clock recovery from the bi-phase level modulated signal, bit timing, preamble decoding, and bus usage by a RACU or DBCU.
2. Breadboard modem units shall operate in a half-duplex or full-duplex mode.
3. Breadboard modem units shall have the capability whereby the output power delivered to the line can be externally adjusted through a limited range including the minimum for data bus operation.
4. The breadboard modem unit interface with other DACS breadboard elements consists of serial digital NRZ data, serial digital clock signals and various dc signals.
5. Each breadboard modem unit can interface with two separate equipments, either RACU's or DBCU's, via dual identical interface connectors.
6. Equipments utilizing the breadboard modem units will be assumed in close proximity; i.e., less than five feet from the modem.

6.2.1.3.2 Core Bus Interface Units. Core bus interface units, used to join separate data bus subassemblies, shall be designed within the limits of the following constraints:

1. Core bus interface units shall include all circuitry necessary to interface two uni-directional equipment bus subassemblies with a bi-directional core bus subassembly.
2. Core bus interface units shall provide for half or full-duplex operation under external control.

3. The input and output of the core bus interface units will be bi-phase level encoded data.

6.2.1.3.3 Equipment Bus Interface Units. The equipment bus interface units used to connect modem units to the equipment buses shall be designed within the limits of the following constraints:

1. Provide dc isolation between cable sections and modem units of at least one megohm.
2. If both series and shunt type couplings are to be utilized, at least two types of coupling units will be required.

6.2.1.3.4 Breadboard Cable Sections. The cable sections for the breadboard data bus shall have the following provisions and considerations:

1. Cable sections shall be provided for the breadboard in a range of lengths.
2. More than one cable section of a given length should be provided.
3. Cable sections shall connect to each other as well as to breadboard coupling units and breadboard terminator units in a simple, quick fashion.
4. In lieu of very long cable sections a line simulator may be provided to simulate the cable section.
5. Provision for a degraded cable section for test purposes should be considered. This might be a cable section with a broken shield, etc.

6.2.1.3.5 Breadboard Termination Units. Breadboard termination units shall be designed with the following considerations:

1. Each breadboard termination unit should be capable of terminating at least one cable section.
2. The design of the breadboard termination units should consider the ability to multiplex dc and/or low frequency signals (audio tones) onto the data link at these points.

6.2.1.3.6 Special Breadboard Test Equipment. Special test equipment for the breadboard shall be considered within the limits of the following:

1. Special breadboard test equipment shall be kept to a minimum consistent with the breadboard evaluation, test and maintenance requirements.
2. Wherever possible, standard laboratory test equipment should be considered.

3. Special breadboard equipment necessary to provide simulation of breadboard data bus features (such as fault introduction) shall be provided.

#### 6.2.1.4 Operation

Normal operational control of the data bus breadboard shall be accomplished by control of secondary power to the modem and repeater units. Each of these units shall remain in a non-transmitting mode until application of both secondary voltage levels and the proper control signals from the DBCU and the RACU's.

#### 6.2.1.5 Flexibility

The data bus breadboard design concept is a highly flexible, building block approach. Thus, maximum flexibility is afforded the breadboard utilization of these building blocks. The breadboard hardware units can be single "universal" designs to satisfy the various configuration concepts, or they can be families of units to satisfy different configurations or requirements. Internal changes by external means, such as plugs or jumpers, can be utilized to provide breadboard unit flexibility.

### 6.2.2 Performance Requirements

#### 6.2.2.1 Data Rate

The data bus breadboard should operate at a nominal data rate of 10 megabits per second. This data bit rate includes all overhead. The average data rate and message data rate will be corresponding less than the peak bit rate depending on housekeeping overhead and bus utilization. The data bus will be evaluated at extreme data rates both above and below the nominal 10 megabits per second.

#### 6.2.2.2 Modulation

The modulation method for the data bus breadboard modulators and demodulators (modems) shall be bi-phase level (Manchester) encoding. This method of data encoding is illustrated in Figure 6-1.

The modulators will be provided with a serial NRZ data signal and serial clock signals at their input interface. They will provide a bi-phase level encoded serial data stream to the data link. The demodulator portion of the modem will extract the bi-phase level encoded serial data signal from the data link and reconvert it to a serial NRZ data signal and a serial clock signal at its output terminals.

#### 6.2.2.3 Switching Capability and Delays

The data bus breadboard shall be capable of switching from a condition where one modem (at a DBCU) is transmitting (and listening) and all other modems are listening, to a condition where a second modem is transmitting and the first is now only listening. This switching should be done with a minimum amount of "turn-around" time or dead time between the first transmission and the responding transmission. This total sequence defines a request-acknowledge message sequence.

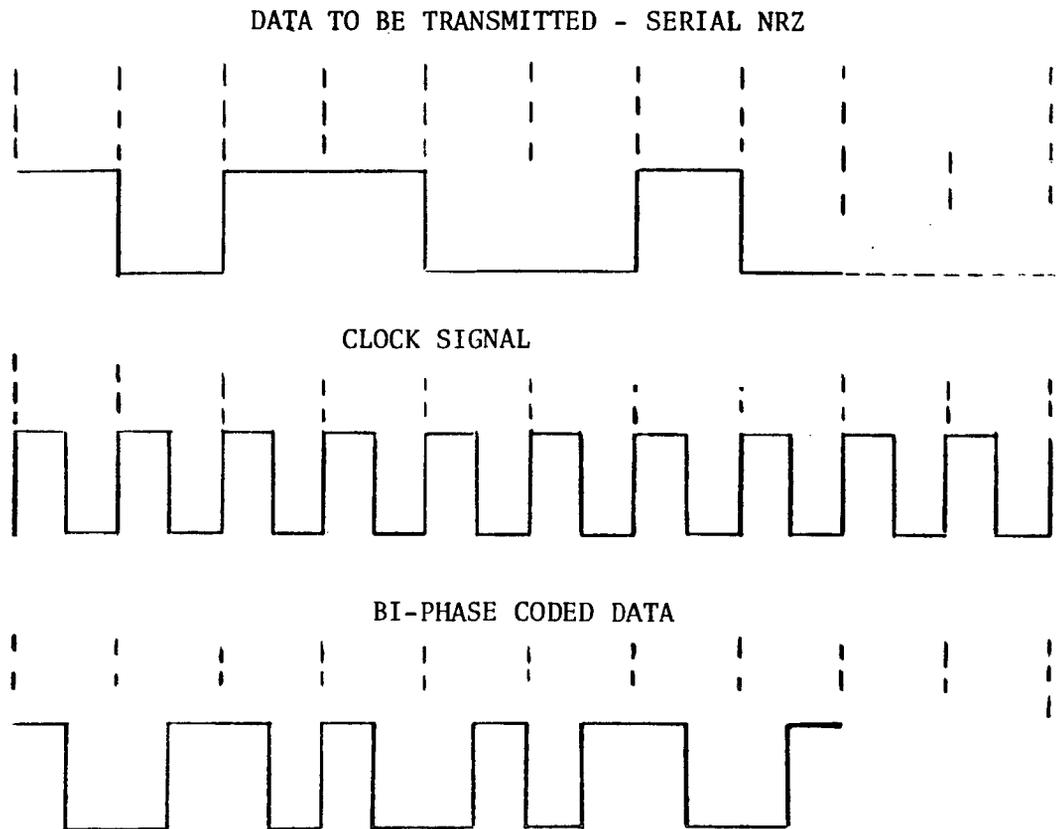


Figure 6-1. Bi-Phase Level Data Encoding

This "turn-around" interval should be distinguishable from the time interval between two first transmissions from a modem associated with a DECU. This second interval is the dead time between request-acknowledge message sequences.

All switching and propagation of data signals, enable signals, and service signals shall be as fast as possible without excessive effect on error rate, cost, reliability, etc. The following is a set of reasonable speed goals:

1. The propagation velocity in cable or wires including intermediate active circuits shall be more than one-third the speed of light.
2. The delay of data signals in the equipment bus interface shall be less than nine bits each way. The delay of data signals in the core bus interface shall be less than nine bits each way. The delay of data signals in the core bus interface shall be less than three bits each way.

Item 2 includes the combined effects of circuits and cables. The decisions when to transmit and what data to receive and all error correction and detection external to the data bus.

#### 6.2.2.4 Synchronization

The data cable will contain both data and timing so that the signal is self-timing. The modulation system shall be capable of being used by asynchronous sources which are switched onto the bus with a probability of loss of data or synchronism at start-up time that is similar to the probability of loss of data or synchronism in the middle of a message after the system has been synchronized.

A bus modem should be capable of recognizing four types of line intervals:

1. An interval ("turn-around") between a request message and an acknowledge message
2. An interval when no one is transmitting message sequences
3. A preamble interval during which bit timing synchronization and message synchronization is performed. The preamble interval shall be as short as possible within the requirement of low error rate, the input-output specification and a high probability of synchronization and low cost. The bus modem interface shall recover bit timing and should recover message synchronization with a low probability of error and low cost.
4. A data transmittal interval. The interval when no one is transmitting may be omitted between messages sent by the same source. The bus transmission interface and all bus-to-bus interfaces shall be consistent with the generation and transmission of the preamble and data transmission.

#### 6.2.2.5 Noise Environment

Noise within the transmission system inside the spacecraft constitutes interference in the form of undesirable electrical signals induced from sources which may be either external or internal to the system. Random noise and impulsive noise are two categories.

The design margin for interference shall assume a random noise power density of  $-120$  dbw/Hz. An additional design margin of 16 dbw shall be assumed for impulsive noise interference. The data bus breadboard shall meet the expected bit error rate requirement specified below when operating in this design environment.

#### 6.2.2.6 Bit Error Rate

The bit error rate will be calculated for the worst case path with the longest line lengths, worst location of transmitter and receiver, etc. The design of the data bus breadboard modems shall be for a raw bit error rate of less than  $10^{-8}$  in a benign external environment. In the environment of 6.2.2.5 and utilizing error encoding enhancement techniques (i.e., parity), the undetected bit error rate shall be less than  $10^{-10}$ .

#### 6.2.3 Interface Definition

##### 6.2.3.1 General

The data bus breadboard has an external interface at each breadboard modem unit. This interface is used for all access to the bus subsystem by the other DACS elements, be they RACU's or DBCU's. It can also be used for monitoring and test purposes in the breadboard evaluation of the DACS.

The interface is described functionally below. The physical details are included in 6.2.3.3. Allowance for breadboard test utilization of any breadboard modem unit interface requires some flexibility in the provisions for this interface. Quick and easy disconnect and connection shall be provided by all breadboard units to allow configuration changes and laboratory evaluation of the numerous options available.

##### 6.2.3.2 Functional Definition

6.2.3.2.1 General. The breadboard modem unit interface is divided into three parts as listed below.

1. Receive
2. Transmit
3. Power

6.2.3.2.2 Receive. The receive interface provides data output from the bus assembly to the RACU or DBCU. This data is demodulated directly from the data link and has no physical electrical interaction with the transmit interface.

The diagrammatic representation of this interface is shown in Figure 5-2. Four signals are shown crossing the interface between the breadboard modem unit and the equipment utilizing the receive interface. These four signals are the following:

R1 - Serial NRZ Data (DR)

This is the digital data that was transmitted on the bus, in serial form. This line is active whenever the bus is active. The signal uses a non-return-to-zero format.

R2 - Serial Clock (CR)

The clock signal is a series of pulses derived directly from the information on the bus. This line is active whenever the bus is active. The clock signal is a square wave pulse train, which is the timing source for the serial NRZ data.

R3 - Start of Message (SOM)

A pulse is used to indicate the start of all messages on the bus. This signal goes "false" two bits prior to the start of the first bit of the address word of each message and stays "false" until the first bit of the address word of the message. This signal is "true" during all other intervals.

R4 - Modulation Present (MO)

A dc level is used to signal the operation of the bus to the interfacing equipment. Any bi-phase level signal on the bus will cause this signal to go to a "true" state, including the preamble of the message.

The serial clock signal is used to strobe the serial data signal and to represent the timing of the data. It must be used to distribute any time-division multiplexed digital data. The modulation present signal "frames" the data of interest to the external user and is provided so that external equipment can monitor bus activity and prepare for bus usage itself.

6.2.3.2.3 Transmit. The transmit interface provides a point for insertion of data into the breadboard data bus. It can be utilized by any external breadboard equipment, usually a RACU or DBCU breadboard unit. The data presented to the breadboard modem unit via this interface, with the appropriate control signals and conditions, is modulated and transmitted on the breadboard data bus to which the modem is connected.

Figure 5-2 also shows the diagrammatic representation of the transmit interface. Four signals cross the transmit interface between the breadboard modem unit and the external equipment. These signals are described below.

#### T1 - Serial NRZ Data (DT)

This is the digital data to be transmitted on the data bus, in serial form. The presence of a signal is controlled by the external device. The signal is in a non-return to zero format.

#### T2 - Serial Clock (CT)

The clock signal is provided to the breadboard modem unit as a square wave pulse train. The signal is continuously present when equipment is connected to this interface. This signal is used by the breadboard modem unit to produce the bi-phase level encoding of the serial NRZ data.

#### T3 - Transmit Enable

A dc level is used to signal the external equipment transmit interval to the breadboard modem unit. A complementary form of the signal is used so that an unused interface is not signaling a transmit interval. A "false" level indicates a transmit enable interval.

#### T4 - Start of Message (SOM)

A pulse is used to indicate the start of all messages on the bus. This signal goes "false" two bits prior to the start of the first bit of the address word of each message and stays "false" until the first bit of the address word of the message. This signal is "true" during all other intervals.

The serial clock signal is used by the breadboard modem unit modulator and represents the timing of the serial digital data. This serial clock signal is not the same signal as that provided at the receive interface.

The transmit enable signal must be present during the preamble, a fixed interval of time before the actual data to be transmitted is transferred across the interface to allow for a transmit preamble and synchronization.

6.2.3.2.4 Power. This interface is used to supply power to the breadboard modem unit from an external source.

#### 6.2.3.3 Physical Interface Definition

6.2.3.3.1 General. The physical interface with the breadboard modem unit shall be provided by a minimum of two separate connectors for the interfaces illustrated in Figure 5-2. This will allow flexibility in the use of the breadboard unit for various test purposes and configurations.

#### 6.2.4 Breadboard Configuration Requirements

##### 6.2.4.1 General

The breadboard concept for the digital data bus is a highly flexible, building block element approach. This approach allows a number of different data bus configurations to be assembled as part of the laboratory breadboard. Each configuration concept can then be evaluated for data bus performance, as well as performance of the other breadboard hardware elements using the data bus, for overall DACS evaluation.

##### 6.2.4.2 Breadboard Data Bus Units

The building block hardware divides into six areas. Each block area may require more than one hardware implementation for breadboard usage. The hardware must be compatible between the various blocks, and capable of the interconnections necessary to demonstrate the various data bus configuration concepts. The six areas are the following:

1. Breadboard Modem Unit(s)
2. Core Bus Interface Unit(s)
3. Equipment Bus Interface Unit(s)
4. Breadboard Cable Sections
5. Breadboard Termination Unit(s)
6. Special Breadboard Test Equipment

Figure 6-2 shows diagrammatically one of many possible conceptual configurations for the breadboard DACS utilizing the above data bus assembly breadboard units. The actual numbers of units and hardware for each concept configuration have yet to be determined. This figure illustrates the concepts at a functional work level and shows possible configuration for the data bus breadboard units and signal flow.

Many configurations may be evaluated with the DACS breadboard in light of the various NASA technology goals discussed in 3.2.1.

##### 6.2.4.3 Redundant Configurations

Redundant configurations utilizing the data bus breadboard units can be achieved very simply. The baseline data bus concept utilizes multiple independent data buses for redundancy. Each bus is complete within itself, physically and electrically independent of all other subassemblies.

For the breadboard data bus, the breadboard units can be configured functionally as shown in Figure 6-2, for example. This configuration can be replicated to allow redundancy to be demonstrated by the breadboard. In essence, two separate data bus breadboards would be configured in the laboratory. The exact same building block units would be utilized.

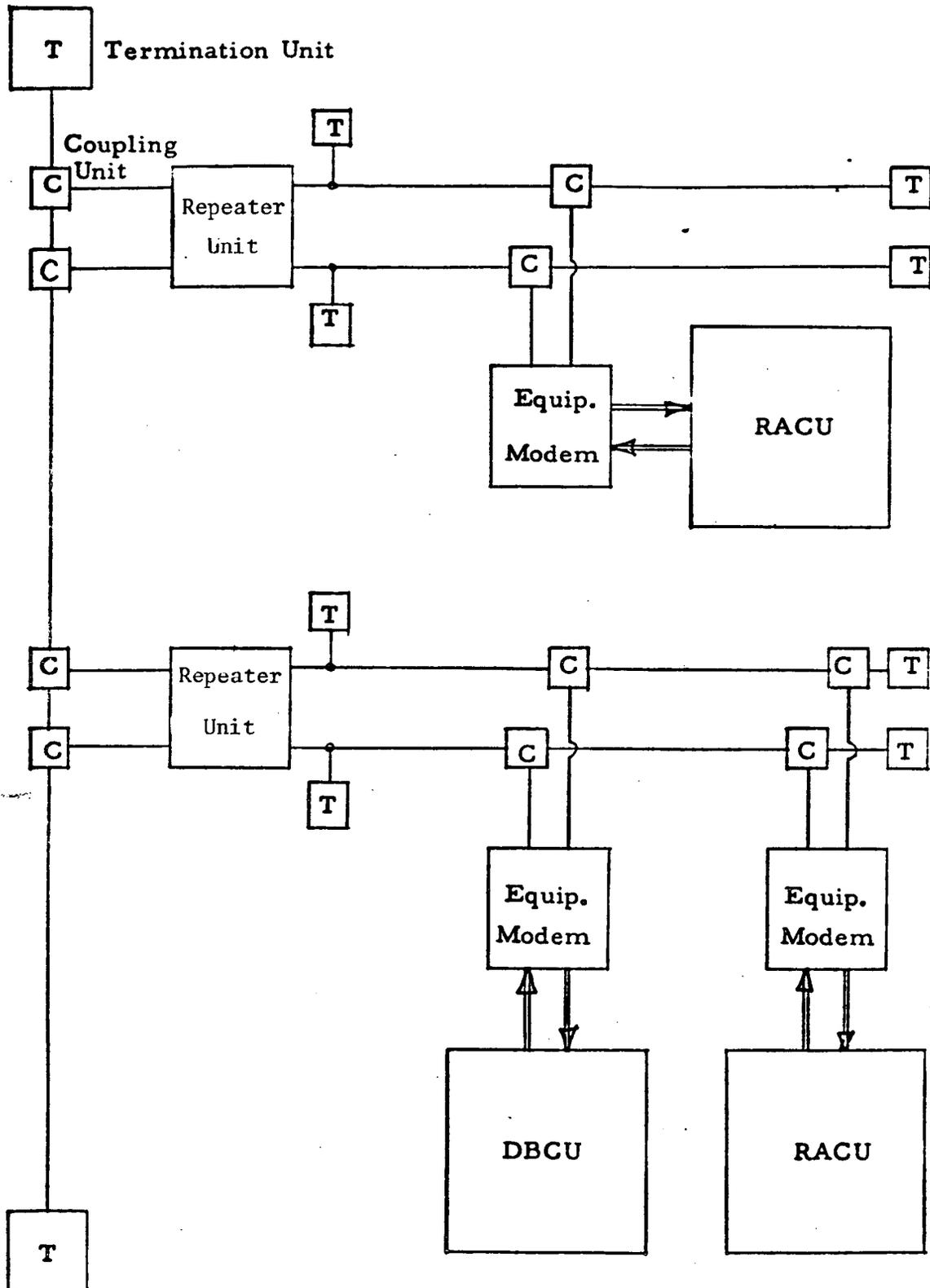


Figure 6-2. Uni-directional Equipment/Bi-directional Core

This replication method of providing redundancy also allows some leeway. Exact replication is not necessary for redundancy simulation. Only a portion of the breadboard may need to be replicated, or the redundant breadboard can utilize different configurations of breadboard units for the different independent buses. This flexibility is left to the user and can be easily performed in the laboratory environment with the data bus breadboard units provided.

The utilization of two different configurations for redundant implementation of the data bus breadboard allows real-time performance comparison and evaluation between configurations with a properly defined test setup. This may have some distinct advantages in the DACS performance evaluation. Full simultaneous redundant utilization of the bus breadboard can require considerable additional breadboard hardware for meaningful evaluation, similar (in the worst case to having two complete DACS breadboards.

### 6.2.5 Electrical Requirements

#### 6.2.5.1 Electrical Power Supply

The electrical power supplied to all breadboard units 115 vac, 60 Hz, single-phase, two-wire plus ground.

#### 6.2.5.2 Power Consumption

Power consumption of the data bus breadboard units shall be minimized to the extent possible consistent with the requirements and good design practice. The number of secondary dc voltage levels supplying power to the data bus breadboard units shall also be minimized.

#### 6.2.5.3 Power Failure and Transients

The data bus breadboard units shall not be designed to operate through power failures or abnormal transients.

#### 6.2.5.4 Overvoltage Protection

The secondary dc voltage level inputs to the data bus breadboard units shall have some form of overvoltage protection against inadvertent application of incorrect dc levels. Such protection may take the form of zener diode clamps to protect the electronic circuitry.

### 6.2.6 Packaging and Construction Requirements

#### 6.2.6.1 General

A breadboard unit is defined as one which performs the same functions and according to the same characteristics as those defined by the hardware design. It should make maximum use of predeveloped and off-the-shelf devices and functional elements, although the component technology should be the same as that defined for the actual operational design. Breadboard units do not have to be reduced to production techniques or qualified, but their quality must be such that they can be expected to survive the handling normally associated with extensive laboratory testing and operation.

Where practical, all breadboard equipment shall be mountable in standard 19-inch equipment racks. To the greatest extent possible, breadboard electronic hardware shall be contained in equipment drawers and mounted in high density packing panels in a manner permitting easy access to all components for test or repair.

#### 6.2.6.2 Weight and Size

Breadboard units shall, to the extent possible, be contained in standard 19-inch equipment rack drawers. Weight and size should be maintained.

#### 6.2.6.3 Modular Construction

Each breadboard unit shall be fabricated using modular construction wherever possible. The units shall be designed with a logical and functional breakdown of replaceable electro-mechanical and electronic modules.

#### 6.2.6.4 Repairability

Breadboard assemblies and subassemblies shall be repairable. The designer shall consider ease of repair and rework of the unit as an important part of the engineering. Maximum accessibility for wiring modification and repair shall be provided. Circuit components shall be legibly and permanently identified. Connectors shall be permanently identified.

#### 6.2.6.5 Standardization and Interchangeability

Standardization is defined here as the effort to select, design, and manufacture parts, assemblies, units, etc., so that they are identical and interchangeable without special considerations or adjustments. Standardization for interchangeability of breadboard units shall be a primary requirement.

#### 6.2.6.6 Accessibility

All breadboard unit subassemblies and individual components shall be arranged to provide ready access for removal and replacement of parts. As a general rule, it will be unacceptable to remove wires, subassemblies or parts in order to gain access to terminals, connections, mounting screws, etc. If this is unavoidable, those parts which must be displaced shall be so wired and mounted that they can be moved without being disconnected from the circuit or cause circuit detuning or instability.

#### 6.2.6.7 Test and Adjustment

When access for potentiometer adjustments, test connections, etc., is necessary, this access shall be free of interference from other components, wiring, subassemblies, etc. and clearly marked. Test points shall be easily accessible and clearly identified to facilitate rapid trouble shooting and repair.



#### 6.2.6.8 Special Tools

The design of the breadboard units shall be such that the need for special tools for tuning, adjustment, and servicing shall be kept to an absolute minimum. Those required for operational adjustments shall be supplied securely affixed to the unit and readily accessible.

#### 6.2.6.9 Cooling Requirements

All breadboard units will be cooled by free convection with no cooling provided external to the unit. If necessary, a fan(s) shall be mounted integral to the breadboard unit inside the rack-mounted drawer.

#### 6.2.7 Environmental

##### 6.2.7.1 General

The data bus assembly breadboard units shall be exposed to the following operative environments.

6.2.7.1.1 Operative. The data bus breadboard shall be designed for normal performance and operation under the following conditions associated with extensive laboratory testing, handling and operation.

6.2.7.1.1.1 Temperature. The breadboard data bus units shall be designed to operate over the standard commercial component temperature range of 0 C to 70 C.

6.2.7.1.1.2 Pressure. The data bus breadboard units shall be designed to operate over a pressure range corresponding to sea level down to 5000 feet above sea level.

6.2.7.1.1.3 Vibration and Shock. The data bus breadboard units shall be designed to operate through the shock and vibration due to normal laboratory environment handling requirements.

#### 6.2.8 Reliability

##### 6.2.8.1 Modularity

The data bus breadboard shall be composed of a minimum number of different units. Each identical unit shall be physically and electrically interchangeable. A replacement of a breadboard unit shall not affect the bus operation.

##### 6.2.8.2 Propagation of Failure

As a design goal, the failure of one component shall not cause other components to fail. The breadboard data bus shall be insulated and isolated so that extraordinary effort is needed to produce a high voltage on these lines and disable the whole bus. The data bus breadboard shall be designed so that the application of voltages up to 100 volts on an interface junction shall not cause components on other junctions or at other bus interface connections to fail.



### 6.2.8.3 Transmit Security

As a design goal, no single failure in a breadboard modem or repeater unit shall cause that unit to inadvertently transmit over the breadboard data bus.



## 7. DBCU BREADBOARD PERFORMANCE AND INTERFACE REQUIREMENTS

### 7.1 ITEM DEFINITION

The Data Bus Controller Unit (DBCUC) is one item of the major elements which make up the modular space station (MSS) data processing assembly. The functional relationships of the various elements of the DACS breadboard equipments are shown in Figure 7-1. Direct interfacing elements with the breadboard DBCUC include the test processor, test panel and special test equipment, MODEM, and prime power source.

The major breadboard units are the data bus, consisting of cables, cable adapters, couplers, amplifiers, receivers, and modems; the data bus controller unit; and the remote acquisition and control units (RACUC's). These units are used in conjunction with a test processor and preprocessors to provide a flexible test bed to evaluate MSS data processing concepts and configurations.

### 7.2 FUNCTIONAL DESCRIPTION

The DBCUC breadboard performs as an input/output device for the test processor, controls the information flow on the data bus, and performs the following functions:

- a. Provides all command and control capabilities to fully exercise the DACS breadboard, to communicate with the RACUC breadboards via the breadboard modem units and data bus, and to operate the breadboard with or without the use of the test processor.
- b. Provides the capabilities to initiate all read/write actions with the test processor or test panel interfaces.
- c. Provides the buffering and formatting of all input/output data to the test processor, test panel, or data bus interfaces.
- d. Provides the capability for error protective encoding and checking of all data to and from the test processor or data bus as selected by the test panel.

### 7.3 INTERFACE DEFINITION

#### 7.3.1 DBCUC/Data Bus Modem Interface

The DBCUC/modem interface shall be as shown in Figure 7-2. The DBCUC interfaces with two modems on separate buses identical to that shown in Figure 7-2.

##### 7.3.1.1 DBCUC to Modem Signals

The DBCUC provides each modem with the following signals.

**7.3.1.1.1 Transmit Enable.** The transmit enable signal is low during preamble, message transmission, and post-amble, and high otherwise. The preamble precedes the message by nine bit times and the post-amble is present for five bit times following the message. The transitions of this signal occur a minimum of 0 ns and a maximum of 25 ns after the transmit clock negative transition. While the transmit enable signal is in the low state, the modem is transmitting a message provided by the DBCUC.

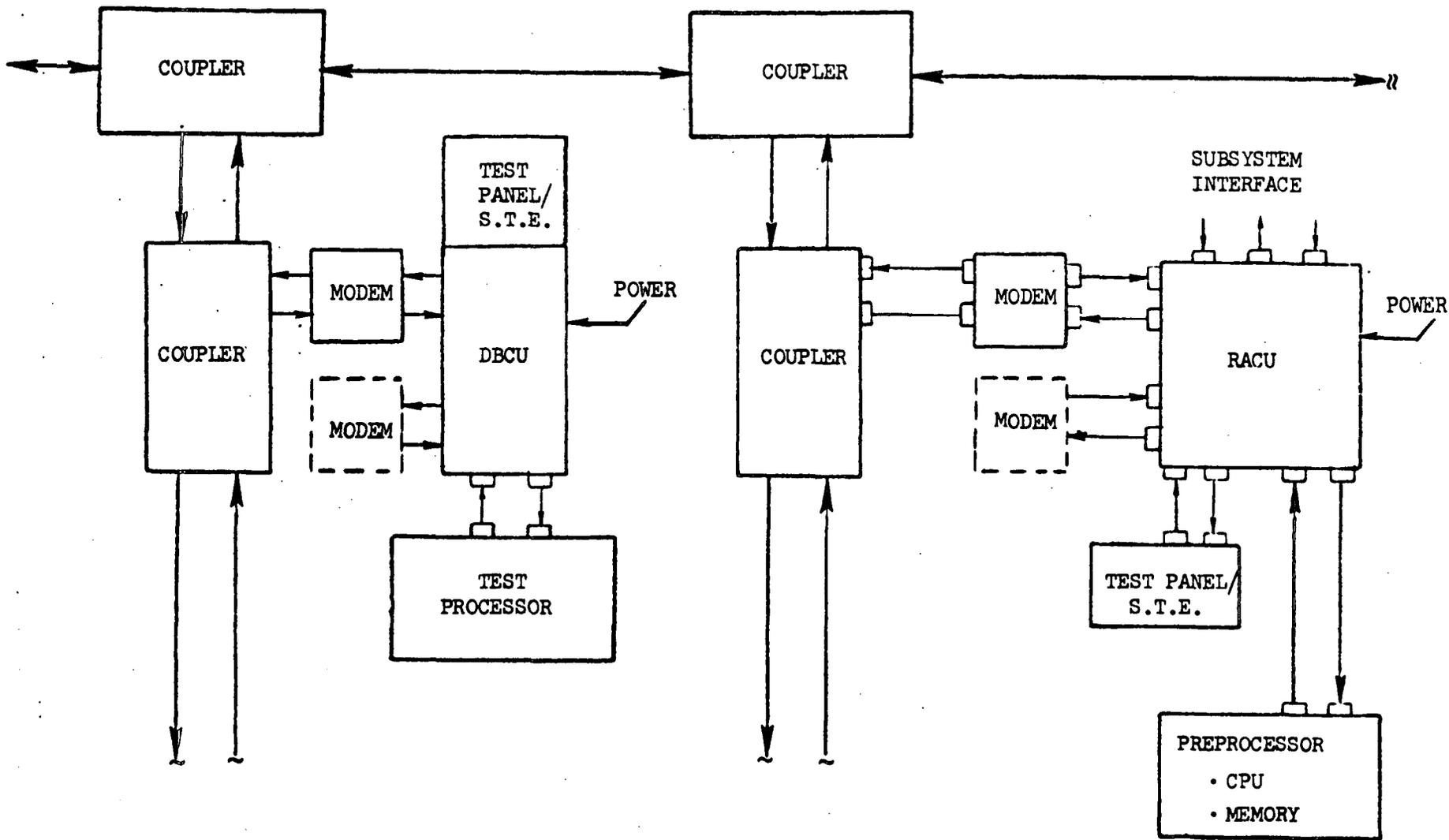
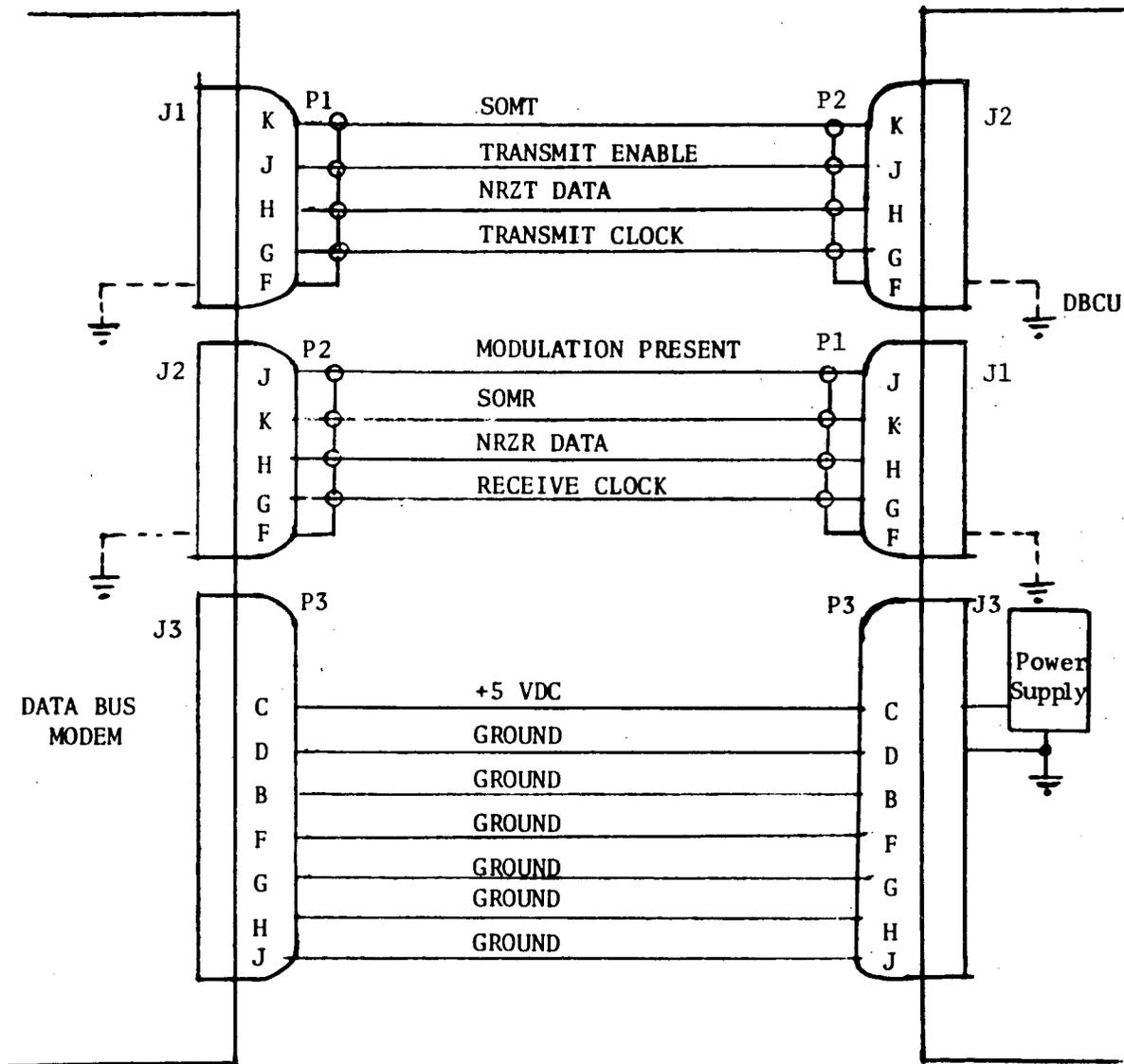


Figure 7-1. Functional Relationship of Breadboard Elements



|    |          | <u>MODEM</u> | <u>DBCU</u> |         |
|----|----------|--------------|-------------|---------|
| J1 | MS3112A  | 12-10 P      | MS3112A     | 12-10 P |
| J2 | MS3112A  | 12-10 S      | MS3112A     | 12-10 S |
| J3 | MS3112A  | 18-11 P      | MS3112A     | 18-11 S |
| P1 | MS3116AC | 12-10 S      | MS3116AC    | 12-10 S |
| P2 | MS3116AC | 12-10 P      | MS3116AC    | 12-10 P |
| P3 | MS3116AC | 18-11 S      | MS3116AC    | 18-11 P |

ALL SIGNALS, POWER, GROUND NO. 20 AWG

Figure 7-2. DBCU/MODEM Interface

3

7.3.1.1.2 Start of Message Transmit (SOMT). The SOMT signal is a negative going pulse of two bit times duration which indicates that the data portion of the message is about to begin. The SOMT signal is low during the last two bit times of the preamble and high otherwise. The negative transition of the SOMT signal occurs a minimum of 9 ns and a maximum of 25 ns after the negative transition of the seventh transmit clock pulse during transmit enable. The SOMT positive transition occurs within the same delay tolerance after the negative transition of the ninth transmit clock pulse during transmit enable.

7.3.1.1.3 NRZ Transmit Data (NRZT). These data are to be transmitted synchronous with the transmit clock during the transmit ready signal. The first nine bits shall be the message preamble (000010111) followed by the data portion of the message, and finally the five bit postamble (11111). The NRZT data transitions shall occur a minimum of 0 ns and a maximum of 25 ns after the negative transitions of the transmit clock. The NRZT data signal shall be low whenever the transmit enable signal is high.

7.3.1.1.4 Transmit Clock. This signal provides timing for the transmit enable, SOMT, and NRZT data signals. The phasing relation between the transmit clock and the other signals is such that signal transitions occur between 0 and 25 ns after the transmit clock negative transition. This ensures that the signals are stable during the positive clock transition and also allows for internal delays before the signals are strobed by the clock. The duty cycle of the transmit clock shall be between 48 percent and 52 percent. The transmit clock shall operate continuously. The clock period shall be 100 ns  $\pm$  1 ns from clock period to clock period with a long-term frequency stability of 10 MHz  $\pm$  0.01 percent.

7.3.1.1.5 Power. The DBCU shall supply each modem with +5 vdc  $\pm$  5 percent at 1 ampere.

Signal ground - To be isolated from chassis ground by a minimum of 10 megohms.

#### 7.3.1.2 Modem to DBCU Signals

Each modem shall provide the following signals to the DBCU.

7.3.1.2.1 Modulation Present. The modulation present signal indicates if the data bus is in use. The signal is high when the bus is idle and low when in use. The negative transition shall occur prior to the end of the sixth bit of the preamble. The positive transition shall occur within five bits after the postamble.

7.3.1.2.2 Start of Message Receive (SOMR). The SOMR signal is a pulse of two bit times duration which indicates that the data portion of the message is about to begin. The SOMR signal is low during the two bit times preceding the first bit of the address code and is high otherwise. The transitions occur a minimum of 0 ns and a maximum of 25 ns after the negative transition of the receive clock.

7.3.1.2.3 NRZ Receive Data (NRZR). Data received synchronous to the receive clock while the data bus is active. The NRZR data shall be in accordance with the message format except for the preamble and the postamble. During preamble, postamble, and when the data bus is idle, NRZR is unspecified. The NRZR data transitions shall occur a minimum of 0 ns and a maximum of 25 ns after the negative transition of the receive clock.

7.3.1.2.4 Receive Clock. Provides timing for the SOMR, modulation present, and NRZR data signals. The phasing relation between the receive clock and the other signals is such that transitions occur between 0 and 25 ns after the receive clock negative transition. This ensures that the signals are stable during the positive clock transition and also allows for internal delays before the signals are strobed by the clock. The duty cycle of the receive clock shall be between 40 percent and 60 percent, with  $\pm 10$  percent transition distortion. The receive clock is unspecified when the data bus is idle. The clock period shall be 100 ns  $\pm 10$  ns between periods.

#### 7.3.1.3 Timing

The basic timing for the signals interfacing between the DBCU and the modem are shown in Figure 7-3. The DBCU shall operate at a required data bus frequency of 10 MHz with  $\pm 0.01$  percent long-term stability.

#### 7.3.1.4 Electrical Characteristics

Outputs from the DBCU and the modem shall be single ended signals driven from Texas Instruments SN74H series circuits or the equivalent from another manufacturer. Output circuits shall be selected from the following: SN74H00, SN74H04, SN74H10, SN74H20, SN74H21, SN74H30, SN74H50, SN74H51, SN74H52, SN74H53, SN74H54, and SN74H55. Output circuits shall not drive any loads within the originating unit. Inputs shall be loaded by no more than one 74 series gate and a space for pull-down or pull-up resistors as required.

7.3.1.4.1 Rise and Fall Times. All rise and fall times shall be less than or equal to 10 ns at the modem/DBCU output connector.

7.3.1.4.2 Propagation Delays. All propagation delays and duty cycles are referenced to the 1.5 volt point of the voltage transitions at the modem/DBCU output connector.

#### 7.3.1.5 Mechanical Characteristics

The DBCU shall have three separate connectors on the rear of the unit to connect to each modem. A separate connector will be provided for each of the following: data bus to DBCU signals, DBCU to data bus signals, and DBCU to data bus power. The connector selections and pin assignments are shown in Figure 7-2. Shield wires shall be brought to the modem and provisions will be made for grounding the shield wires in the modem. Similar provisions shall be made in the DBCU. The shield ground configuration shall be determined during the breadboard integration.

#### 7.3.2 DBCU/BIO Interface

The DBCU/BIO interface shall be as shown in Figure 7-4.

##### 7.3.2.1 DBCU to BIO Signals

The DBCU provides the test processor with the following signals.

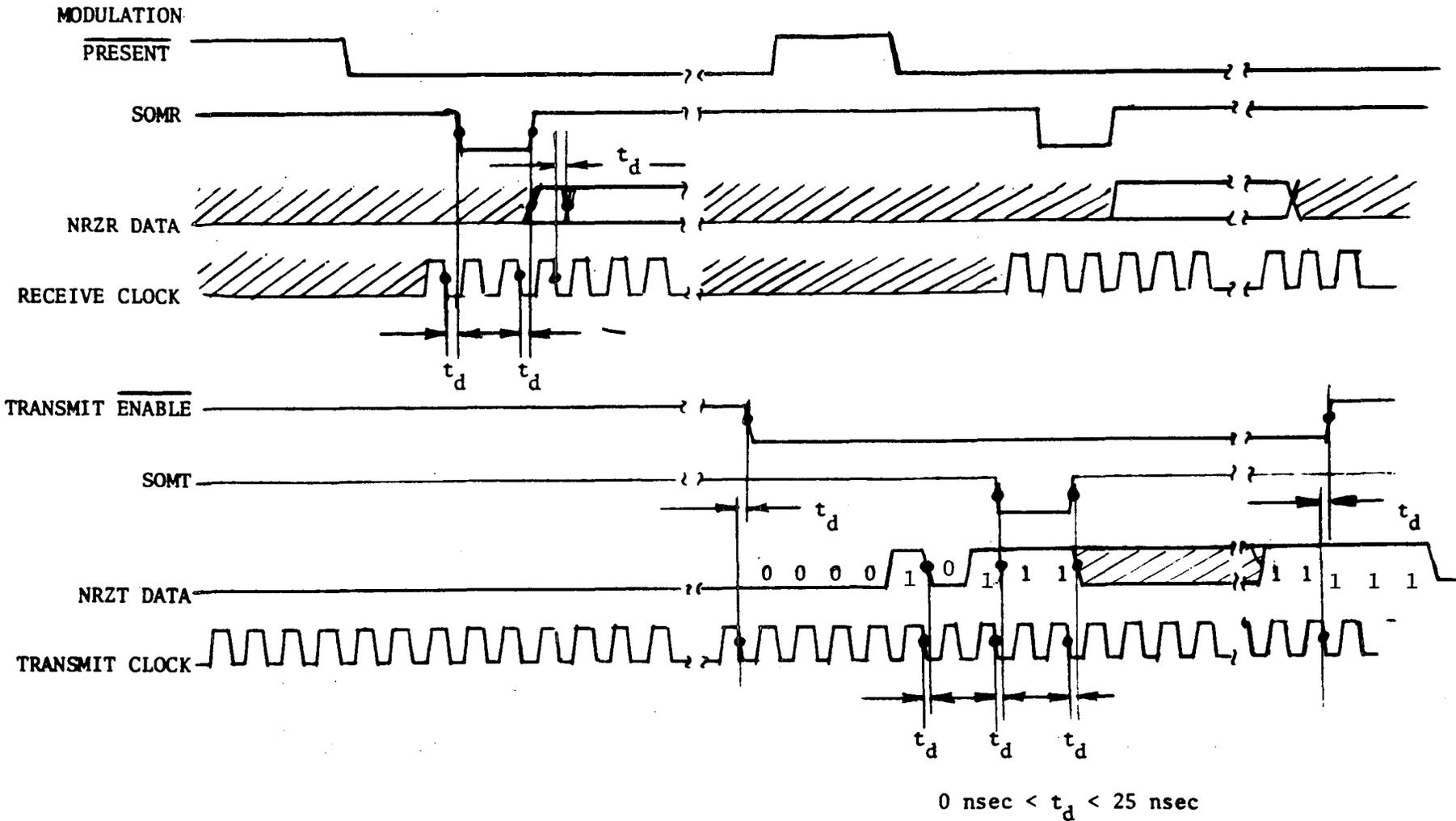
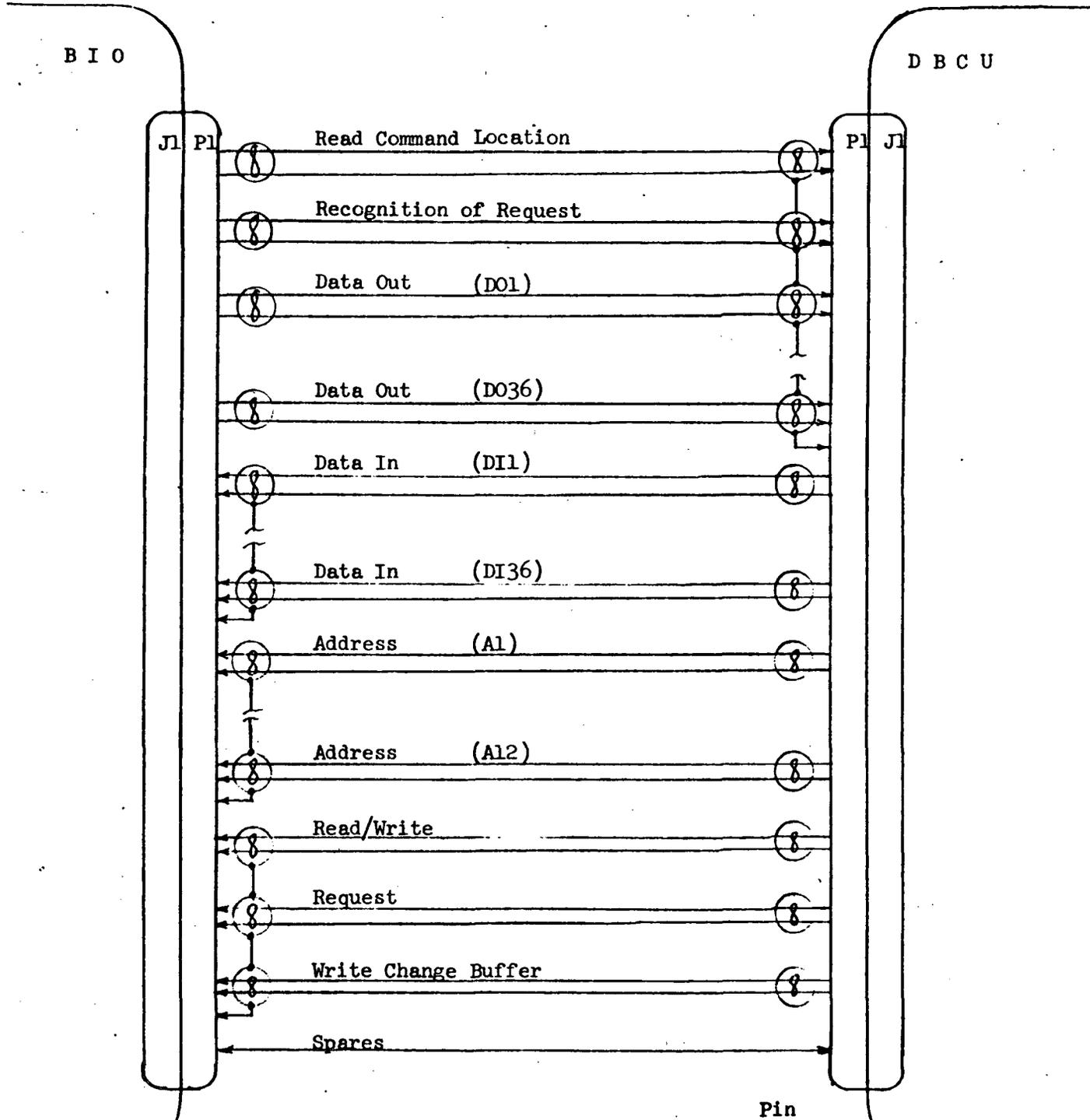


Figure 7-3. DBCU/MODEM Interface Timing Diagram



J1 Hughes WSS 0244S00CY000  
P1 Hughes WSS 0244P03CY541  
All signal wires 22 AWG twisted-shielded pairs.  
All spare wire 22 AWG single conductor. Spares  
to be used as signal ground as required.

WS22Y22C130  
WP22Y22C130

Figure 7-4. BIO/DCU Interface  
7-7



#### 7.3.2.1.1 Read/Write

Read active low, Write active high. This signal is used to indicate to the BIO whether the addressed data are to be read from memory or written into memory. This signal shall be in its proper state prior to activation of the Request line and remain in that state until after the active period of ROR. (see 7.3.2.2.1).

#### 7.3.2.1.2 Request

Active high. All data, address logic, and control signals to the BIO must be valid when the request is activated. The request line remains active before and throughout the active period of ROR (see 7.3.2.2.1).

#### 7.3.2.1.3 Write Change Buffer

Active low. This signal informs the BIO that the data words are to be written into the Write Change Buffer, addressed according to the state of the Change Buffer Control Counter.

#### 7.3.2.1.4 Address

Logic "1" active high, Logic "0" active low. The twelve (12) address lines contain the address of the data to be written into or read from the BIO memory.

#### 7.3.2.1.5 Data In

Logic "1" active high, Logic "0" active low. The thirty-six (36) data input lines contain the data read into the BIO memory during the active period of ROR.

### 7.3.2.2 BIO to DBCU Signals

The BIO provides the DBCU with the following signals.

#### 7.3.2.2.1 Recognition of Request (ROR)

Active low. This signal indicates to the DBCU that the request is being processed. DBCU data to the BIO must remain active during the entire active period of the ROR signal. Data from the BIO to the DBCU are valid for 100 nsec minimum starting from 0 to 40 nsec before the ROR signal goes high.

#### 7.3.2.2.2 Data Out

Logic "1" active high, Logic "0" active low. The thirty-six (36) data input lines contain the data readout of the BIO memory and are valid a minimum of 100 nsec.

#### 7.3.2.2.3 Read Command Location

Active low. This signal indicates to the DBCU that command data is ready to be read from location  $0010_8$ . The signal is normally in the high

state and goes low for 190 to 300 nsec when active. The DBCU need not respond to this signal until its message in process is completed.

#### 7.3.2.3 Timing

The basic timing for the signals interfacing between the DBCU and the BIO is shown in Figure 7-5.

#### 7.3.2.4 Electrical Characteristics

All outputs on the DBCU/BIO interface shall be single-ended signals driven from Texas Instruments SN74 or SN74H (open collector) driver circuits or the equivalent from another manufacturer. Inputs on the DBCU/BIO interface shall be loaded by no more than one SN74 series gate or equivalent. All terminating and/or pull-up resistors shall be on the receiver side of the interface. The signals shall be driven over twisted shielded pair lines with a characteristic impedance no greater than 120 ohms.

7.3.2.4.2 Rise and Fall Times. All rise and fall times shall be less than or equal to 20 nsec at the source or driver interface connector.

7.3.2.4.3 Propagation Delays. All propagation delays and duty cycles are referenced to the 1.5-volt point of the voltage transitions at the source or driver interface connector.

7.3.2.4.4 Shields and Grounds. Cable or wire shield ties shall be connected to signal ground at the receiving end only. Signal ground is to be isolated from chassis ground by a minimum of 10 mehoohms in both the DBCU and the BIO.

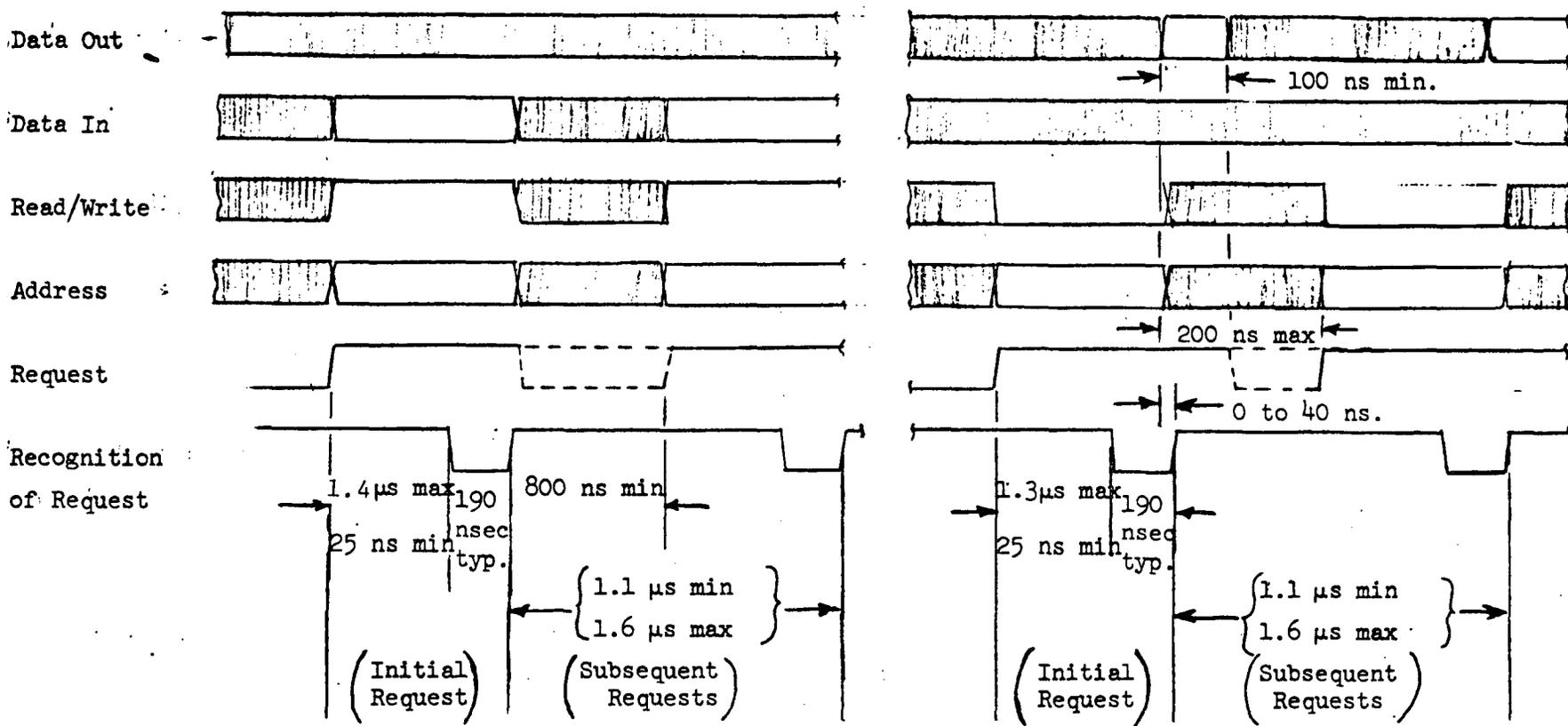
#### 7.3.2.5 Mechanical Characteristics

7.3.2.5.1 Cables. Separate cables shall be provided for interconnection between the DBCU and the BIO, as shown in Figure 7-4. The cable lengths shall be no greater than 8 feet.

7.3.2.5.2 Connectors. Plug-type connectors shall be provided on the interconnecting cables and mating receptacles shall be mounted on the rear of the DBCU and on the BIO. Connector types and pin types are shown in Figure 7-4.

DBCU to BIO (Write)

BIO to DBCU (Read)



 "Don't Care" Logic State  
 Active Logic State

Figure 7-5. BIO/DBCU Timing Diagram

### 7.3.3 DBC/Power Interface

The DBCU shall operate properly when supplied with 115  $\pm 10$  percent volts, 60  $\pm 2$  Hz, single phase input power. The power cord shall have a minimum of six feet extending from the rear of the panel or drawer and be of the standard three-prong grounded variety.

### 7.3.4 DBC/Special Test Interface

The interface between the DBCU and special test equipment (STE) shall, as a minimum, be as shown in Figure 7-6.

#### 7.3.4.1 DBCU to STE Signals

The DBCU shall provide the following signals for external STE usage.

7.3.4.1.1 NRZ Receive Data External (NRZRE). Data received synchronous to the receive clock while the data bus is active. The NRZRE data shall be in accordance with the transmitted message format except for the preamble and the postamble. During preamble, postamble, and when the data bus is idle, the NRZRE data signal is unspecified. The NRZRE data transitions shall occur a minimum of 0 ns and a maximum of 25 ns after the negative transition of the receive clock external.

7.3.4.1.2 Receive Clock External (RCE). Provides timing for the data enable and the NRZRE data signals. The phasing relation between RCE and the other signals is set up such that signal transitions occur between 0 and 25 ns after the receive clock external negative transition. This ensures that the signals are stable during the positive clock transition and also allows for internal delays before the signals are strobed by the clock. The duty cycle of RCE shall be between 40 percent and 60 percent with  $\pm 10$  percent transition distortion. RCE is unspecified when the data bus is idle. The clock period shall be 100 ns  $\pm 10$  ns.

7.3.4.1.3 Data Enable (DE). The data enable signal indicates that the data bus is in use. The signal is high when the bus is idle and low when in use. The negative transition shall occur prior to the end of the sixth bit of the preamble. The positive transition shall occur within 11 bits after the last message bit.

7.3.4.1.4 Internal Clock (IC). The DBCU internal clock is used at this interface to provide timing synchronization for external test equipment. This clock is the same one used for data transmission to the breadboard modem unit. The duty cycle of the clock shall be between 48 percent and 52 percent. The clock shall operate continuously and the period shall be 100  $\pm 5$  ns from clock period to clock period with a long-term frequency stability of 10 MHz  $\pm 0.01$  percent.

#### 7.3.4.2 STE to DBCU Signals

The DBCU shall have the capability to accept the following signals from external test equipment.

7.3.4.2.1 Data Ready (DR'). The data ready signal is low during preamble, message transmission, and postamble, high otherwise. (The preamble precedes the message by nine bit times and the postamble is present for five bit times following the message.) While the data ready signal is in the low state, the STE is providing data to the DBCU.

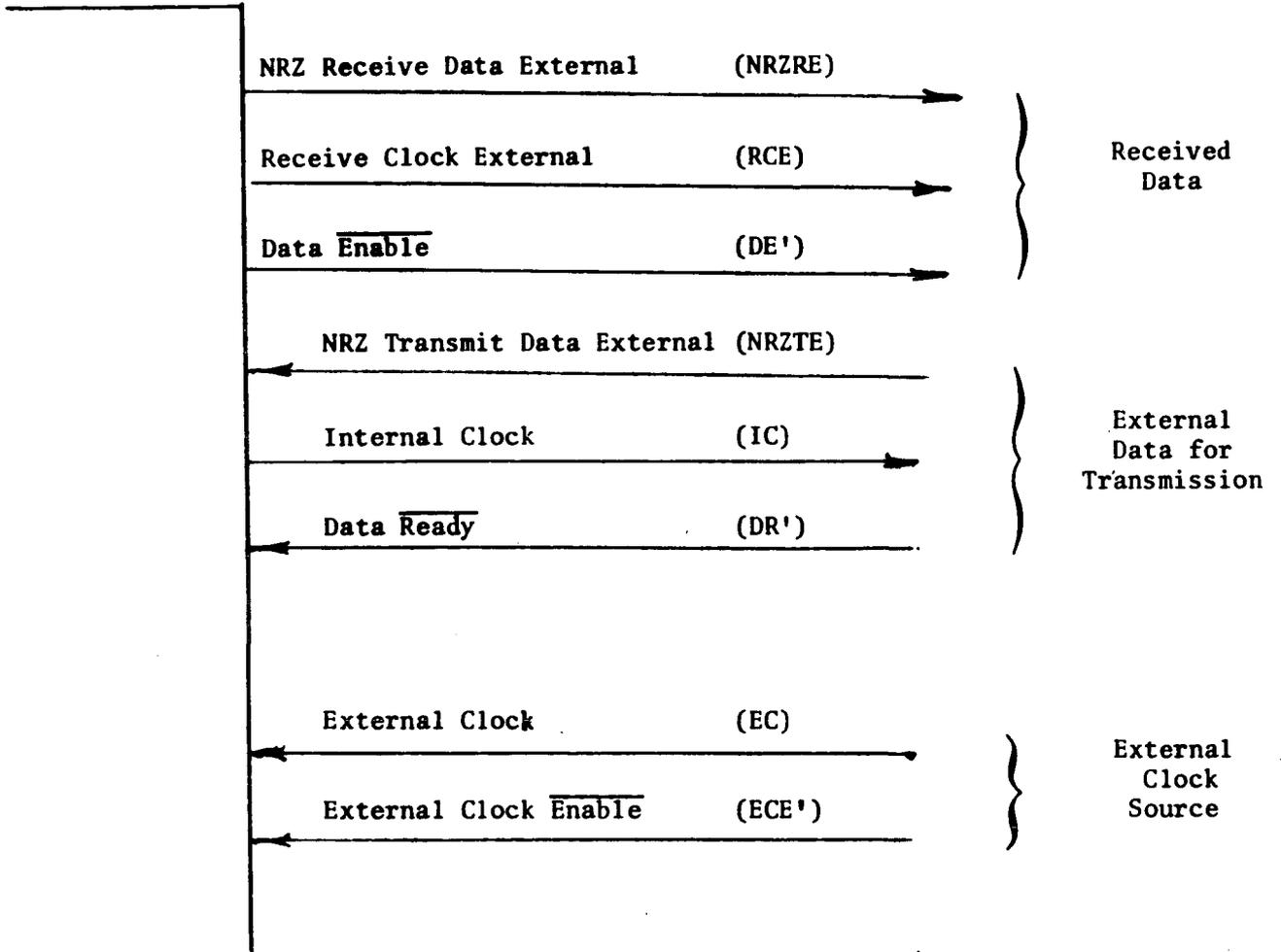


Figure 7-6. DBCU/STE Interface

7.3.4.2.2 NRZ Transmit Data External (NRZTE). These data are to be transmitted synchronous with the IC clock during the data ready signal. The first nine bits shall be the message preamble (000010111), followed by the data portion of the message, and finally the five bit postamble (11111). The NRZTE data transitions shall occur a minimum of 0 ns and a maximum of 25 ns after the negative transitions of the transmit clock.

7.3.4.2.3 External Clock (EC). This interface provides a variable clock source interface to the DBCU. As a design goal, the DBCU shall be capable of operation over a clock frequency range of 3 to 13 MHz. The duty cycle of the external clock shall be between 48 percent to 52 percent.

7.3.4.2.4 External Clock Enable (ECE'). This signal is used to gate the external clock source to the DBCU. The signal low selects the external clock source and a high signal selects usage of the internal clock for DBCU operations. An unused input is defined as a high level.

#### 7.3.4.3 Timing

The basic timing for the signals interfacing between the DBCU and the STE are similar to those shown in Figure 7-3. The DBCU shall operate at the selected external data bus frequency with  $\pm 0.1$  percent long-term stability. As a goal, it shall operate over a range of from 3 to 13 MHz with no component changes.

#### 7.3.4.4 Electrical Characteristics

Outputs from the DBCU and the STE shall be single-ended signals driven from Texas Instruments SN74H series circuits or the equivalent from another manufacturer. Output circuits shall be selected from the following: SN74H00, SN74H04, SN74H10, SN74H11, SN74H20, SN74H21, SN74H30, SN74H50, SN74H51, SN74H52, SN74H53, SN74H54, and SN74H55. Output circuits shall not drive any loads within the originating unit. Inputs shall be loaded by no more than one 74 series gate.

7.3.4.4.1 Rise and Fall Times. All rise and fall times shall be less than or equal to 10 ns at the STE/DBCU output connector.

7.3.4.4.2 Propagation Delays. All propagation delays and duty cycles are referenced to the 1.5 volt point of the voltage transitions at the STE/DBCU output connector.

#### 7.3.4.5 Mechanical Characteristics

The connector types and locations, and wire size and types, are to be determined. The signals may be made available at the test panel or a rear panel connector as appropriate.

## 7.4 PERFORMANCE

The DBCU shall provide satisfactory performance of all functions specified herein when subjected to the environments specified.

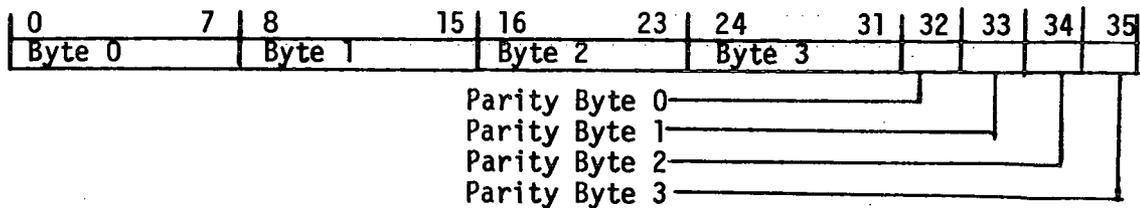
### 7.4.1 Organization

The DBCU shall be organized as shown in Figure 7-7. The functional blocks shall provide the following capabilities.

#### 7.4.1.1 Test Processor Input/Output Section

The test processor I/O section shall be used to buffer and format all data and controls between the DBCU and the test processor. It provides the capability of controlling the execution of input/output operations between the test processor and the DBCU. The DBCU interfaces with the test processor through one of eight ports which are sequentially scanned for memory access requests by the test processor scanner. When a request by the DBCU is recognized, a Recognition of Request (ROR) is returned by the scanner.

The basic word format shall be as shown below. The word is composed of 32 bits of data (four 8 bit bytes) and 4 bits of parity (one odd parity bit for each data byte).



The four data bytes shall be gated one at a time to internal registers for instruction and control of the message formatting, to the data bus I/O section for further transmittal over the data bus, or to fill the transmit memory section for temporary storage.

An odd parity bit shall be checked on every data byte coming from the processor and generated for every data byte going to the processor.

The control word, instruction word, and data word formats shall be in accordance with ICD No. 26465.

#### 7.4.1.2 Data Bus Input/Output Section

The data bus I/O section shall be used to buffer and format all data between the DBCU and RACU's via the data bus. This section shall provide parallel-to-serial and serial-to-parallel conversion, BCH coding generation and detection, generation of the preamble, storage of the RACU address and transmission code, message formatting capabilities, address comparison logic, MODEM interface control logic, and controls.

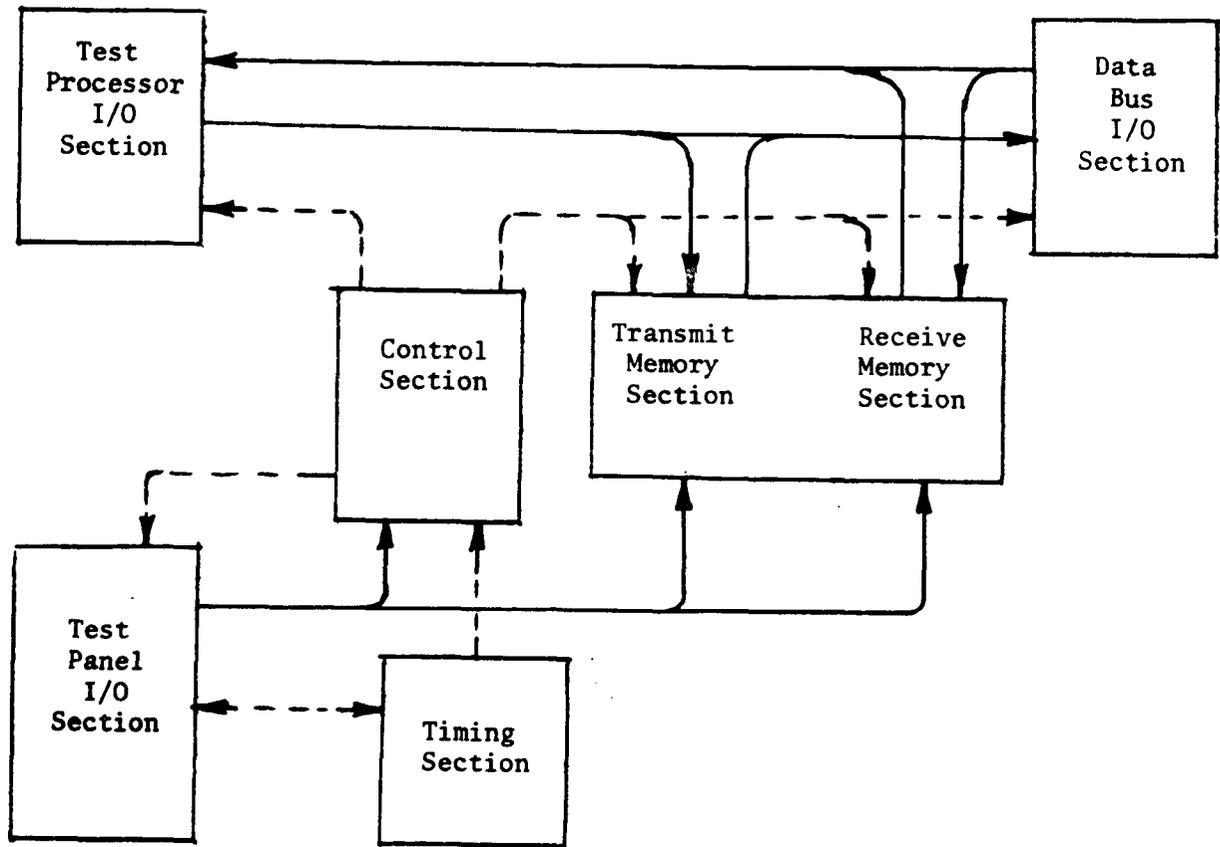


Figure 7-7. DBCU Functional Organization

The DBCU shall be capable of interfacing with two modems under control of either the test panel or test processor.

#### 7.4.1.3 Test Panel Input/Output Section

The test panel I/O section shall be used to buffer and format all data and controls between the other DBCU sections and the test panel. This section provides a means to fill the transmit memory section, a means to read out the receive memory section, control logic to load the various buffer registers in the DBCU and the interface logic and controls for test panel display of selected internal registers.

#### 7.4.1.4 Control Section

The control section contains all logic and decoding associated with the modes of operation. It provides control signals to all other sections as a function of the test panel switch settings and/or test processor instructions as gated by the timing section.

#### 7.4.1.5 Timing Section

The timing section contains the basic clock oscillator and bit, word, and frame counters as required by the control section. The basic timing for this section may come from an external clock source upon command from the test panel.

#### 7.4.1.6 Transmit Memory Section

This section shall be a (size TBD) read-write memory used to contain a sequence of operations or messages for further transmittal on the data bus. It shall be capable of being filled from either the test processor or test panel interface.

#### 7.4.1.7 Receive Memory Section

This section shall be a 512-word, 8 bit read-write memory used to contain a sequence of data or messages received on the data bus for further transmittal to the test processor or test panel. It shall be capable of being filled from the data bus I/O section.

#### 7.4.1.8 Power Converter Section

The power converter section shall be capable of converting a primary input of 115 volts, 60 Hertz, single phase power to the dc levels required by all internal circuitry. It shall also be capable of supplying secondary power to the interfacing modems as specified in paragraph 7.3.1.1.5.

#### 7.4.2 Operation Modes

The DBCU shall be capable of operating in several modes of operation concurrently. The major control modes of operation are the test panel mode and the test processor mode. In both modes of control the DBCU may be required to process messages continuously, one at a time, or in bursts, depending on panel

switch setting or processor software. The DBCU shall also be capable of performing the following minor modes of operation.

#### 7.4.2.1 Error Protective Encoding

Hardware shall be provided to perform error protective encoding and checking on the data over the data bus interface and the test processor interface.

7.4.2.1.1 Data Bus Interface. The data bus interface shall use the BCH (127-112) code for error protective encoding. The use of error encoding or not and the capability of inserting false check bits shall be selectable from the test panel controls. The status of the error code checks shall also be displayed at the test panel.

7.4.2.1.2 Test Processor Interface. The test processor interface shall employ a simple odd parity bit on each of the four data bytes in every data word.

#### 7.4.2.2 Channel Selection

The DBCU shall have selectable transmit/receive capability on both modem channels A and B. The channel mode options shall be:

1. Transmit/receive on Channel A
2. Transmit/receive on Channel B
3. Transmit Channel A, receive Channel B
4. Transmit Channel B, receive Channel A

The channel selection shall be made by test panel selection or by control word format from the test processor.

#### 7.4.2.3 Data Flow

Data flow through the DBCU shall be under control of the test panel interface. The data flow modes available shall be:

1. Transmit/receive direct between the test processor and the data bus
2. Transmit/receive between the DBCU internal memories and the data bus
3. Fill or dump data between the DBCU internal memories and the test processor or test panel

#### 7.4.2.4 Timing

Timing for the DBCU shall be derived internally or externally, selectable from the test panel. The basic timing for the DBCU shall be derived from an internal clock oscillator. The DBCU and the data bus shall operate at a required frequency of 10 MHz  $\pm 0.01$  percent. As a goal, it shall also operate over the range from 3 to 13 MHz from an external clock source when selected

by the test panel. The test panel/DBCUC shall also have the capability to advance the clock one step at a time as an aid in troubleshooting. The DBCUC shall contain the logic to provide proper bit timing, word counters, gating logic, etc., to format all messages on the associated interfaces.

#### 7.4.3 Message Format

The DBCUC shall communicate with the RACUC's via the digital data bus bread-board using three message formats which are the normal mode, the override mode, and the preprocessor memory access mode. The messages are transmitted on a command-reply basis with commands going from the DBCUC to the RACUC and subsequent replies coming from the RACUC to the DBCUC.

##### 7.4.3.1 Normal Mode

The normal mode message format is shown in Figure 7-8. The data bytes are 8 bits in length. Each message starts with the preamble which is 000010111. The address code follows and specifies the RACUC being interrogated. The next word in the message is the transmission code. This indicates if the message is in the normal mode or override mode. When in the normal mode it indicates the starting location in the RACUC PROM for message processing and formatting. Zero to 124 data words follow. After every 14 words in the message, excluding preamble, there are two encoding check words. The first 15 bits are formatted according to the BCH (127,112) code and the last bit is unspecified. The message always ends with the two encoding check words. Following the data portion of the message, there is a postamble 5 bits in length which is 11111.

If the command message does not have valid check words, a valid address code, and a valid transmission code, the RACUC does not respond. If a proper message format is verified the RACUC transmits a reply. The reply begins with the preamble and then the RACUC address. If there are no data words to be transmitted back to the DBCUC, the postamble is sent after the address. If there are data, the 1 to 124 words follow the address code and check words are sent in a manner similar to that in the command message. Again, the message ends with the postamble.

Once the command message has been sent, the DBCUC anticipates an RACUC response (SOMR) within 16 bit times plus 1 microsecond. If the RACUC does not respond or if the address in the response message does not compare with the address of the command message, an error condition exists which is flagged to the test processor or test panel.

##### 7.4.3.2 Override Mode

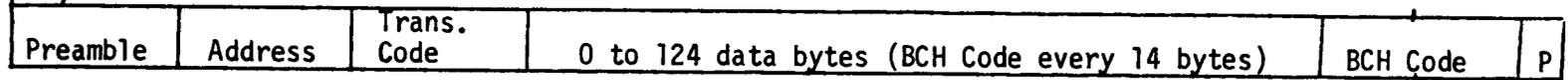
The override mode message format is similar to the normal mode except that the transmission code can only specify that one word be sent to the RACUC or that one word be sent from the RACUC to the DBCUC.

##### 7.4.3.3 Preprocessor Memory Access Mode

The preprocessor memory access mode format is shown in Figure 7-8. As in the other modes the message begins with preamble, address code, and transmission

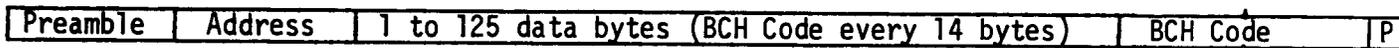
0 Normal Mode

(Command)



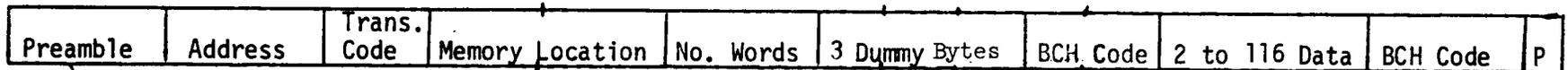
Postamble

(Response)

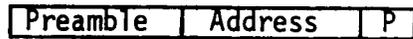


0 Preprocessor Memory Access Mode To Memory

(Command)



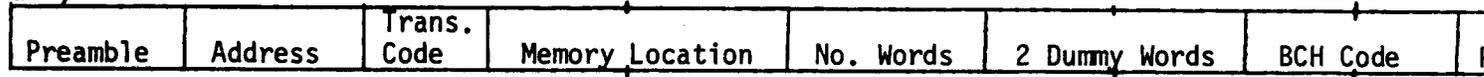
(Response)



(with encoding)

From Memory

(Command)



(Response)

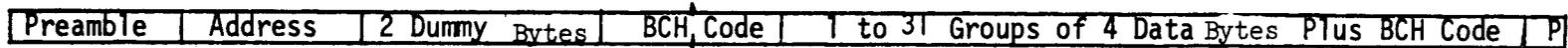


Figure 7-8. DBCU/RACU Message Formats

7-19

SD 72-SA-0114-3

code. The transmission code indicates that the message is in the preprocessor memory access mode and indicates the direction of data transfer. The next two bytes indicate the starting memory location for access. The next byte indicates the number of 16 bit memory words to be transferred up to 58. Four dummy bytes or BCH code bytes are then required to allow for the latency in acquiring the direct memory access channel.

If data are being written into the preprocessor memory, three dummy words and BCH code are used and the data bytes follow. Data bus bytes are combined in two's to form single preprocessor memory words. Encoding check bytes are used every 14 bytes as in the normal mode format. The number of data bus bytes to be written into memory is limited to 116 per message and must be in even multiples. Proper receipt of memory load information is acknowledged by the RACU with preamble and address code. The same encoding and timing checks are made as in the normal mode.

If data are to be read out of the preprocessor, two memory check bytes and postamble follow two dummy words. The RACU then responds with preamble, address code, two dummy bytes, and encoding check bytes if the previous check bytes have been verified. Memory data are then transmitted. Each two memory words transmitted require six data bus bytes. The first four contain the actual 16 bit memory words and the fifth and sixth contain encoding check bytes for data transmission enhancement and time difference between the preprocessor memory access output channel rate and the data bus rate. The number of data bus information words is limited to 1 to 31 groups of 4 data bytes plus 2 BCH code bytes for a total of 124 information data bytes maximum.

#### 7.4.4 Word Formats

The data messages are structured such that most or all messages contain identical or similar word formats such as the preamble, address, transmission code, and postamble. Other word formats are unique to special modes of operation such as the preprocessor memory access mode and status word mode, etc. The word formats shall be as described herein.

##### 7.4.4.1 Preamble and Postamble

Each message, command or reply, put on the data bus, starts with a 9 bit preamble which has a 000010111 bit pattern. This pattern is put on the line by the data source (either a DBCU or RACU), but it is stripped off by the modem and not seen on the data line to the data sink(s) (either a RACU or DBCU).

The postamble is a five bit pattern of 11111 which ends every message on the data bus. This pattern is also put on the line by the data source and may or may not be seen by the data sink.

##### 7.4.4.2 Address

The preamble is always followed by an 8 bit address word. The address word contains a unique code identifying the unit on the bus to which communication is being directed. The RACU acknowledges receipt of a command by transmitting its preset address word back to the DBCU following a preamble.

The 8 bit address word allows a total of 256 unique address codes; however, it is doubtful that even 200 unique RACU addresses are required. Certain codes can then be used to command all RACU's or groups of RACU's simultaneously.

Universal (all RACU's) addresses may be used to provide timing pulses, initiate simultaneous discrettes for operations or visual displays, etc. Group addresses are used for the same purposes but are restricted to certain families of RACU's such as at the module or equipment bay level. For example, a group address might be used to instantaneously and simultaneously shut down all subsystem equipment power in a particular damaged equipment module; or it may be used to turn on distributed fire control equipment in the same module.

Eight codes are reserved for universal addresses and 24 codes are reserved for group address assignment as shown in Table 7-1. For breadboard test purposes, only octal codes 370,374 and a switch selectable one, will be utilized. The remainder are reserved for future consideration and implementation.

#### 7.4.4.3 Transmission Code

The transmission code word indirectly identifies the required RACU operation for transmission or reception of data, or commands the RACU to take some action other than transmit data. The transmission code word is used to define a starting location in the RACU program memory which, in turn, contains the sequence of instructions defining the command operation. These command operations in RACU memory will indirectly identify message length, data location and content, and may define direct command functions such as turning on a portion of a subsystem functional loop, setting internal flip-flops, service requests, etc.

As in the address word, the 8-bit transmission code allows a total of 256 unique starting locations in the RACU instruction memory. Again, it is doubtful that any one RACU will require 256 unique message structures and thus many transmission codes may be used to have command usage in all RACU's addressed.

The transmission code indicates if the message is for the normal mode, the override mode, the preprocessor memory access mode, or other information, depending on the format. These are shown in Figure 7-9.

The normal mode is defined by a zero (0) in bit 1 of the transmission code. Bits 2 through 7 are decoded to give the initial address for the RACU read only memory to control further sequencing of the RACU. The seven bits allow a maximum of 128 unique starting locations.

The override mode is defined by a 10 pattern in bits 1 and 2 of the transmission code. Bit 3 defines the direction of transfer for the data sent or requested. A "1" in bit 3 indicates data to be sent to the DBCU and a "0" indicates data to be sent to the RACU. Bits 4 through 8 of this word and bits 1 through 8 of the following word define the channel address of the data to be sent or updated. Only one word may be sent or received in the override mode of operation.

Table 7-1. Universal and Group Address Codes

| Octal Code Word | Binary Bit* |   |   |       | Code Type | Function       |
|-----------------|-------------|---|---|-------|-----------|----------------|
|                 | 3           | 4 | 5 | 6 7 8 |           |                |
| 340             | 1           | 0 | 0 | 0 0 0 | Group     | To be assigned |
| 341             | 1           | 0 | 0 | 0 0 1 |           |                |
| 342             | 1           | 0 | 0 | 0 1 0 |           |                |
| 343             | 1           | 0 | 0 | 0 1 1 |           |                |
| 344             | 1           | 0 | 0 | 1 0 0 |           |                |
| 345             | 1           | 0 | 0 | 1 0 1 |           |                |
| 346             | 1           | 0 | 0 | 1 1 0 |           |                |
| 347             | 1           | 0 | 0 | 1 1 1 |           |                |
| 350             | 1           | 0 | 1 | 0 0 0 |           |                |
| 351             | 1           | 0 | 1 | 0 0 1 |           |                |
| 352             | 1           | 0 | 1 | 0 1 0 |           |                |
| 353             | 1           | 0 | 1 | 0 1 1 |           |                |
| 354             | 1           | 0 | 1 | 1 0 0 |           |                |
| 355             | 1           | 0 | 1 | 1 0 1 |           |                |
| 356             | 1           | 0 | 1 | 1 1 0 |           |                |
| 357             | 1           | 0 | 1 | 1 1 1 |           |                |
| 360             | 1           | 1 | 0 | 0 0 0 |           |                |
| 361             | 1           | 1 | 0 | 0 0 1 |           |                |
| 362             | 1           | 1 | 0 | 0 1 0 |           |                |
| 363             | 1           | 1 | 0 | 0 1 1 |           |                |
| 364             | 1           | 1 | 0 | 1 0 0 |           |                |
| 365             | 1           | 1 | 0 | 1 0 1 |           |                |
| 366             | 1           | 1 | 0 | 1 1 0 |           |                |
| 367             | 1           | 1 | 0 | 1 1 1 |           |                |
| 370             | 1           | 1 | 1 | 0 0 0 |           |                |
| 371             | 1           | 1 | 1 | 0 0 1 |           |                |
| 372             | 1           | 1 | 1 | 0 1 0 |           |                |
| 373             | 1           | 1 | 1 | 0 1 1 |           |                |
| 374             | 1           | 1 | 1 | 1 0 0 |           |                |
| 375             | 1           | 1 | 1 | 1 0 1 |           |                |
| 376             | 1           | 1 | 1 | 1 1 0 |           |                |
| 377             | 1           | 1 | 1 | 1 1 1 |           |                |

\*Bits 1, and 2 are "1 1"

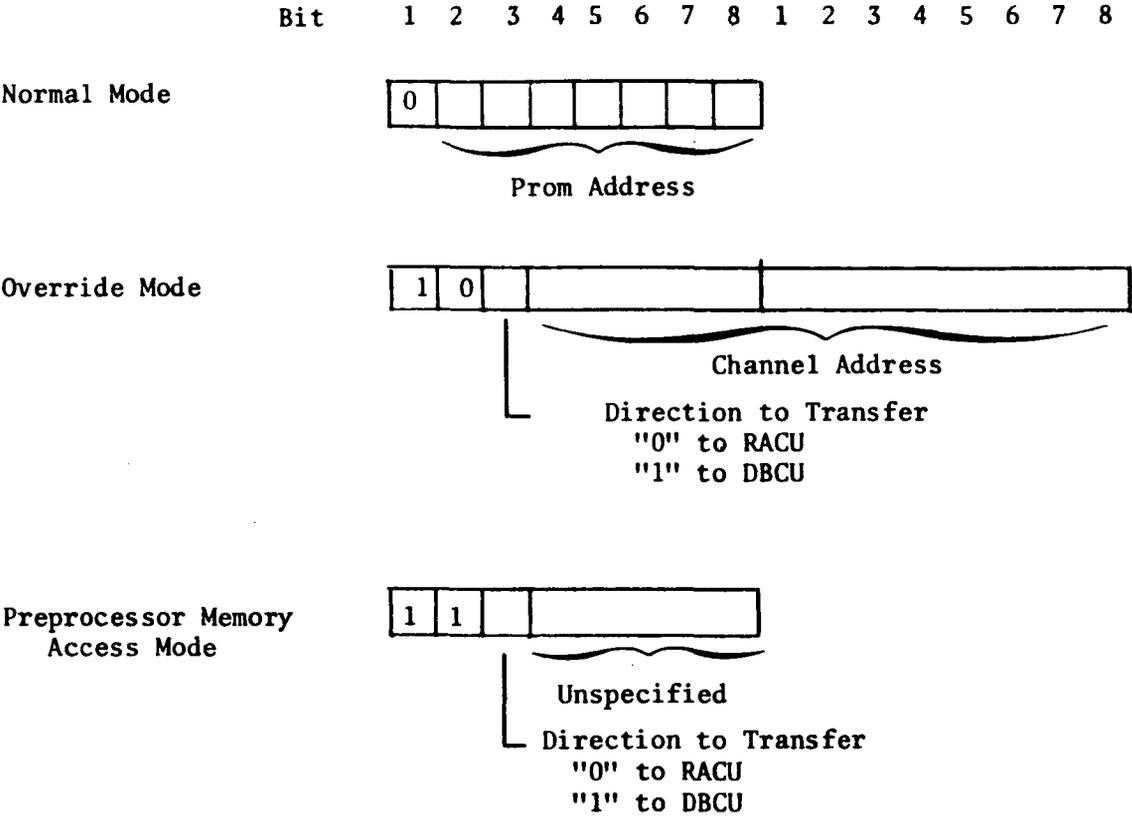


Figure 7-9. Transmission Code Formats

The preprocessor memory access mode is defined by an 11 pattern in bits 1 and 2 of the transmission code. Bit 3 defines the direction of the data transfer and bits 4 through 8 are unspecified.

Four common transmission codes associated with the normal mode of operation designate the same starting address in the read only memory of each RACU addressed, and also have the same meaning. Transmission code octal 000 is a "no operation" instruction which is used along with the universal or group address. No RACU response is expected with this transmission code. Transmission code octal 001 designates a "modify status register" command. The addressed RACU is to modify the status register with the single data word following if encoding is verified on the message. Transmission code octal 006 designates a "transmit status word" command and the addressed RACU is to transmit the contents of the status register if encoding is verified on the message. Transmission code octal 015 designates a "message turnaround" command. This command asks the RACU to transmit data back to the DBCU which were received in the last message and are contained in the output channel register(s) specified by the operand(s) of the previous message.

#### 7.4.4.4 Status Word

The status word shall indicate the condition of the status word register. A "true" (logic 1) in the indicated bit times shall have the following meanings:

- Bit 1 Receive data on Modem A
- 2 Receive data on Modem B
- 3 Transmit data on Modem A
- 4 Transmit data on Modem B
- 5 There was a parity error on the message prior to this request
- 6 An invalid address code was recognized on the message prior to this request
- 7 An invalid transmission code was recognized on the message prior to this request
- 8 Spare

Upon power turn-on of the RACU's, the status register will assume a 11100000 bit pattern.

## 7.5 PHYSICAL CHARACTERISTICS

### 7.5.1 General

The DBCU breadboard unit shall make maximum use of predeveloped and off-the-shelf devices and functional elements, although the component technology should be the same as that defined for the actual operational design. The breadboard unit does not have to be reduced to production techniques or qualified, but its quality must be such that it can be expected to survive handling normally associated with extensive laboratory testing and operation. The breadboard electronic hardware shall be contained in equipment drawers and mounted in high-density packing panels in a manner permitting easy access to all components for test or repair.

### 7.5.2 Size

The DBCU breadboard unit shall be contained in a standard 19-inch equipment rack drawer. MIL-STD-189, "Racks, Electrical Equipment, 19-inch and Associated Panels," shall be used as the guidelines and constraints for panel sizes and rack capacities. The 19-inch panel height shall not exceed  $8 \frac{23}{32}$  and the drawer depth no more than 23 inches.

### 7.5.3 Repairability

The DBCU subassembly shall be repairable. The designer shall consider ease of repair and rework of the unit as an important part of the design. Maximum accessibility for wiring modification and repair shall be provided. Circuit components shall be legibly and permanently identified. Connectors shall be permanently identified.

### 7.5.4 Standardization and Interchangeability

Standardization is defined here as the effort to select, design, and manufacture parts, assemblies, units, etc., so that they are identical and interchangeable without special considerations or adjustments. Standardization for interchangeability of breadboard units shall be a primary requirement.

### 7.5.5 Accessibility

The DBCU subassembly and individual components shall be arranged to provide ready access for removal and replacement of parts. As a general rule, it will be unacceptable to remove wires, subassemblies or parts in order to gain access to terminals, connections, mounting screws, etc. If this is unavoidable, those parts which must be displaced shall be so wired and mounted that they can be moved without being disconnected from the circuit or cause circuit detuning or instability.

### 7.5.6 Test and Adjustment

When access for potentiometer adjustments, test connections, etc., is necessary, this access shall be free of interference from other components, wiring, subassemblies, etc., and clearly marked. Test points shall be easily accessible and clearly identified to facilitate rapid troubleshooting and repair.

### 7.5.7 Special Tools

The design of the breadboard units shall be such that the need for special tools for tuning, adjustment and servicing shall be kept to an absolute minimum. Those required for operational adjustments shall be supplied securely affixed to the unit and readily accessible.

### 7.5.8 Cooling Requirements

The DBCU shall be cooled by free convection or, if necessary, a fan(s) mounted integral to the DBCU inside the rack-mounted drawer.

## 7.6 ENVIRONMENTAL

The DBCU breadboard unit shall perform properly after and during exposure to the following nonoperative and operative elements.

### 7.6.1 Nonoperative

The following conditions occurring separately or in combination may be encountered by the DBCU during transportation, handling, and storage. The breadboard shall operate and meet the performance requirements after it has been subjected to the following conditions.

#### 7.6.1.1 Temperature

-15 C to 70 C

#### 7.6.1.2 Pressure

From seal level to 30,000 feet

#### 7.6.1.3 Shock

Normal handling requirements

### 7.6.2 Operative

The DBCU shall be designed for normal performance and operation under the following conditions associated with laboratory testing, handling, and operation.

#### 7.6.2.1 Temperature

The breadboard DACS units shall be designed to operate over the standard laboratory ambient temperature range of 60 F to 90 F.

#### 7.6.2.2 Pressure

The DBCU breadboard shall be designed to operate over a pressure range corresponding to sea level down to 5000 feet above sea level.

#### 7.6.2.3 Vibration and Shock

The DBCU breadboard shall be designed to operate through the shock and vibration due to normal laboratory environment handling requirements.

## 8.0 DIGITAL DATA BUS DESIGN

This section contains the system and subsystem performance specifications that guided the design of the various subsystems or are as a result of the detail subsystem design of the digital data bus breadboard.

### 8.1 SYSTEM SPECIFICATION

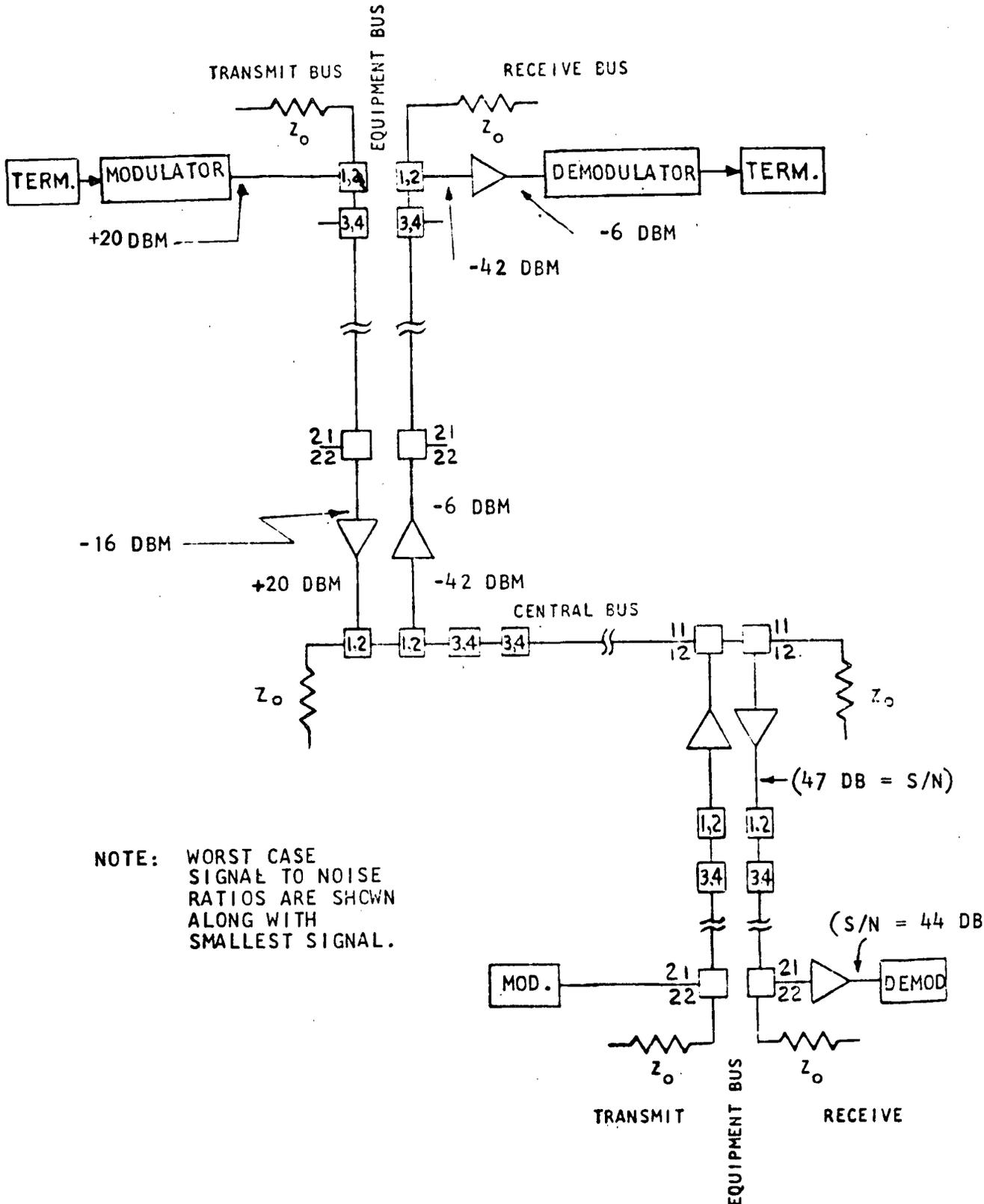
#### 8.1.1 Station Configuration (Figure 8-1)

A simplex data bus contains a central (or core) bus section and a number of equipment buses. All terminal equipments are connected to equipment buses. The central bus is connected only to equipment buses and serves as a hub to provide a transmission path between the equipment buses. Information flows bidirectionally on the central bus but each equipment bus has a separate path for information flow towards and away from the central bus. A data bus also contains modems which interface with the terminal equipments, couplers which provide for connecting modems to equipment buses and equipment buses to the central bus, and amplifiers to provide necessary gain. Information about the individual bus assemblies is contained below. The following is general system information.

|                               |   |
|-------------------------------|---|
| Bit Rate                      | 10 MBS $\pm$ 200 PPM  |
| Line Lengths                  | Central Bus 100 feet maximum. Equipment buses 125 feet maximum including inter-connecting section to central bus. Modem to Equipment bus section 25 feet maximum. Total transmission length 400 feet maximum. |
| Maximum Simplex Configuration | One Central Bus<br>Twelve Equipment Buses<br>22 Modems per equipment bus<br>264 Modems per data bus   |
| Error Rate                    | $P_e < 10^{-8}$ (Nominal Environment)   |
| System Delays                 | Modem delay approximately 4 bit times.<br>Cable propagation delay of 1.6 nanoseconds/foot.<br><br>Total delay for longest path 1200 ns maximum.   |

#### 8.1.2 Breadboard Configuration (Figure 8-2)

The breadboard will contain two bus systems, each with a central bus, two equipment buses, two modems, and the required couplers and amplifiers. In addition, test equipment will be supplied. Details on the equipment supplied are given later.



NOTE: WORST CASE SIGNAL TO NOISE RATIOS ARE SHOWN ALONG WITH SMALLEST SIGNAL.

Figure 8-1. Skeleton Diagram of Data Bus System

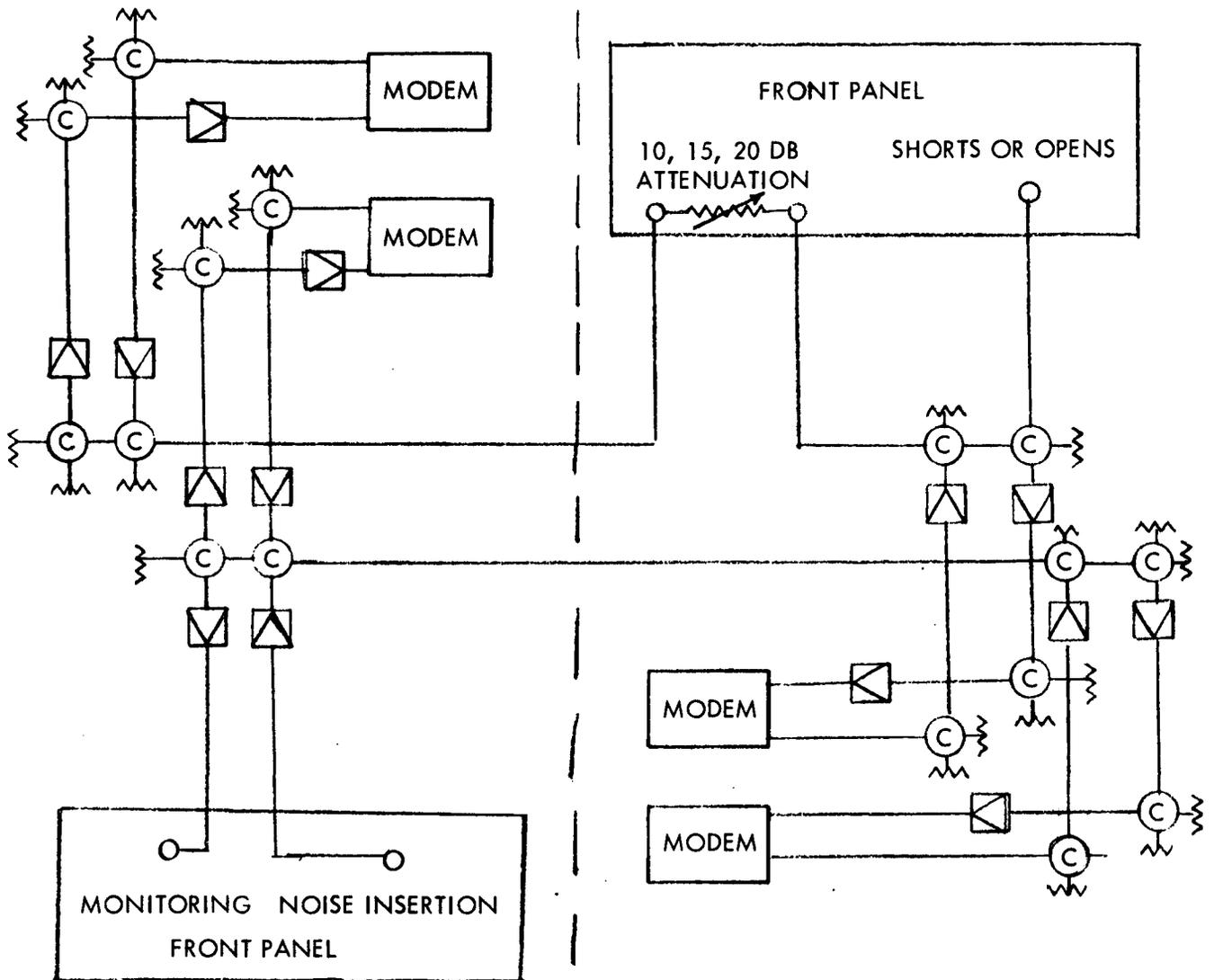


Figure 8-2. Breadboard Configuration, Block Diagram

## 8.2 COMPONENT PERFORMANCE SPECIFICATIONS

This section contains specifications for the major data bus assemblies.

### 8.2.1 Bus Interface Unit

The bus interface unit is used to couple the central bus to the equipment bus or the equipment bus to the modem. A fully configured unit contains:

Two couplers

Two Transmit Amplifiers (used only at central bus/equipment bus interface)

Two Receiver Amplifiers

Two sets of Power Supplies

Inputs/Outputs--Specified in the subassemblies attached to the bus interface unit.

#### Physical

Size: 19" W x 5-1/4" H x 8" D

Mounting: 19" rack mounted

Interconnections: All connections are made to the rear of the unit except for fault simulation connections, which have front panel access. Power on/off is also on front panel.

#### 8.2.1.1 Coupler

Introduction: The coupler is the device used for transmitting, receiving or interconnecting signals via the digital data buses.

Inputs/Outputs: Reference circuit diagram, Figure 8-3. For purposes of specifying performance of the coupler the ports are designated as shown.

The attenuation between port 11 and 22 shall be  $1\text{db} \pm 0.2\text{db}$  and the attenuation between port 11 and 33 or 22 and 33 shall be  $25.3\text{db} \pm 0.5\text{db}$ .

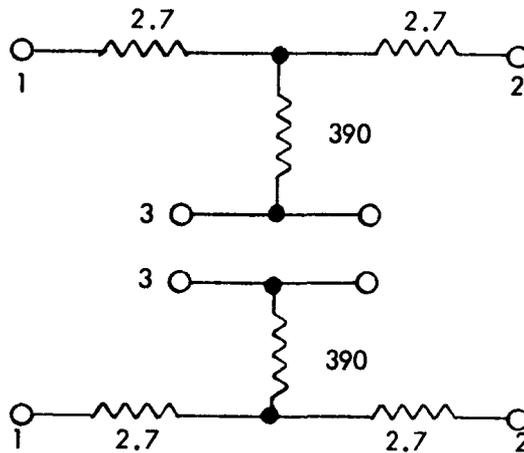
Z across any output terminals = 95 ohms.

DC Power In = zero.

Physical (Breadboard)

Enclosure Size: 4-1/4" L x 2-1/2" W x 1-1/2" H

Interconnections: All connections at the sides of enclosure.



RESISTORS IN OHMS  
1/4 WATT, 5% CARBON

Figure 8-3. Coupler, Schematic Diagram

#### 8.2.1.2 Transmit Amplifier

Introduction: The transmit amplifier is used to amplify and limit signals traveling between the equipment bus and central bus.

Inputs: Level = -6 to -16 dbm

Z = 95  $\Omega$  (Balanced)

Coupled = Transformer

Outputs: Level = +20 dbm

Z = 95  $\Omega$  (Balanced)

Coupled = Transformer

Rise Time = 12 ns maximum

Noise Figure = 8 db

Power: +6V @ 75 ma; -5V @ 75 ma

Physical (Breadboard): Enclosure Size = 4-1/4" L x 1-1/2" W x 1-1/2" H

Interconnections: All connections are at the sides of the enclosure.

#### 8.2.1.3 Receive Amplifier

Introduction: The receiver amplifier is used to amplify and limit signals traveling between the central bus and the equipment bus, and between the equipment bus and demodulator.

Inputs: Level = -32 to -42 dbm

Z = 95  $\Omega$  (Balanced)

Coupled = Transformer

Outputs: Level = -6 dbm minimum; -3 dbm maximum

Z = 95  $\Omega$  (Balanced)

Coupled = Transformer

Rise Time = 12 ns maximum

Noise Figure: 8 db

DC Power: +6V @ 36 ma; -6V @ 36 ma

Physical (Breadboard): Enclosure Size = 4-1/4" L x 2-1/2"W x 1-1/2" H

Interconnections = All connections are at the sides of the enclosure.

#### 8.2.1.4 AC-DC Power Supplies

Off-the-shelf commercial power supplied from Power Mate Corporation are used as follows.

| <u>Part No.</u> | <u>Volts</u> | <u>Current Rating</u> |
|-----------------|--------------|-----------------------|
| MM6B            | 6            | 200 ma                |

#### 8.2.2 Modem

Introduction: The modem consists of the modulator and the demodulator. The modulator receives individual clock, data, and start of message signals and combines them into a single waveshape. The modulator also receives a transmit enable signal which is used to determine whether the modulator output shall provide a signal or no signal. The output is brought to the proper power and impedance levels for interface with other data bus components. There is a delay of 1-1/2 bit times through the modulator. The demodulator receives an input signal similar to the modulator output signal and supplies individual clock, data, start of message, and modulation present signals to a data bus terminal equipment. There is a delay of from 2 to 2-1/2 bit times through the demodulator.

##### 8.2.2.1 Modulator

###### Inputs

Transmit Clock. Provides timing for the transmit enable, SOM, and NRZ data signals. The phasing relation between the transmit clock and the other signals is set up such that transitions of the other signals occur between 0 to 25 ns after the transmit clock negative transition. This ensures that the signals are stable during the negative clock transition and also allows for internal delays before the signals are strobed by the clock. The duty cycle of the transmit clock shall be between 48% and 52%. The transmit clock shall operate continuously. The clock period shall be between 99 ns to 101 ns from clock pos. trans. to clock pos. trans.

NRZ Data. Data to be transmitted synchronous with the transmit clock during the transmit enable signal. The first nine bits shall be the message preamble (000010111) followed by the data portion of the message and finally the five bit postamble (11111). The NRZ data transition shall occur a minimum of 0 ns and a maximum of 25 ns after the negative transitions of the transmit clock. The NRZ data signal shall be logic 0 whenever the transmit enable signal is logic 1.

Start of Message (SOM). The SOM signal is a negative going pulse of two bit times duration which indicates that the data portion of the message is about to begin. The SOM signal is logic 0 during the last two bit times of the preamble and logic 1 otherwise. The negative transition of the SOM signal occurs a minimum of 0 ns and a maximum of 25 ns after the negative transition of the seventh transmit clock pulse during transmit enable. The SOM positive transition occurs with the same delay limits after the negative transition of the ninth transmit clock pulse during transmit enable.

Transmit Enable. Logic 0 during preamble, message transmission, and postamble, logic 1 otherwise. The preamble precedes the message by nine bit times and the postamble is present for five bit times following the message. The transitions of this signal occur a minimum of 0 ns and a maximum of 25 ns after the transmit clock negative transition. While the transmit enable signal is in the logic 0 state, the modem is transmitting a message provided by the RACU.

Output: When not transmitting = No Signal  
95  $\Omega$  Balanced termination

When transmitting = -6 dbm signal  
95  $\Omega$  Balanced  
Waveshape contains clock, Data, and SOM

#### 8.2.2.2 Demodulator

Inputs: When no transmission is present on bus = No Signal  
95  $\Omega$  Balanced

When transmission is present on bus = -6 dbm signal  
95  $\Omega$  Balanced  
Waveshape contains  
Clock, Data,  
and SOM

#### Outputs

Receive Clock. Provides timing for the SOM and NRZ data signals. The phasing relation between the receive clock and the other signals is set up such that transitions of the other signals occur between 0 and 25 ns after the receive clock negative transition. This ensures that the signals are stable during the negative clock transition and also allows for internal delays before the signals are strobed by the clock. The receive clock shall be present from one bit time before the SOM until the last bit time of the message and is unspecified otherwise. The period shall be between 90 ns and 110 ns on a short term basis. The duty cycle shall be between 40% and 60%.

The combined effect of worst case variation in duty cycle and period is depicted in Figure 8-4. If the clock period is 90 ns, either positive or negative portion of the clock may vary between 36 ns and 54 ns. Similarly, if the period is 110 ns, either portion may vary between 44 ns and 66 ns.

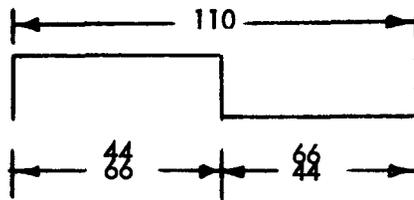
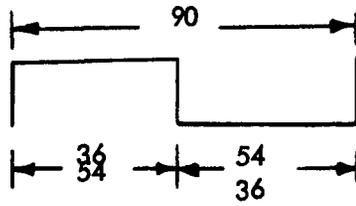
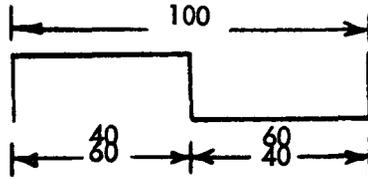


Figure 8-4. Receive Clock Tolerances

NRZ Data. Data received synchronous to the receive clock while the data bus is active. The NRZ data shall be in accordance with the message format except for the preamble and the postamble. During preamble, postamble and when the data bus is idle, the NRZ data signal is unspecified. The NRZ data transitions shall occur a minimum of 0 ns and a maximum of 25 ns after the negative transition of the receive clock.

Start of Message (SOM). The SOM signal is a pulse of two bit times duration which indicates that the data portion of the message is about to begin. The SOM signal is logic 0 during the two bit times preceding the first bit of the address code and is logic 1 during the message and postamble and is otherwise unspecified. The transitions occur a minimum of 0 ns and a maximum of 25 ns after the negative transition of the receive clock.

Modulation Present. The modulation present signal indicates if the data bus is in use. The signal is logic 1 when the bus is idle and logic 0 when in use. The negative transition shall occur prior to the end of the sixth bit of the preamble. The positive transition shall occur prior to two bits after the postamble.

#### 8.2.2.3 Modem Power

+5 VDC @ 955 ms (334 ma modulator, 621 ma demodulator)

120 VAC 60 cycles for DC power supplies

+10 @ 38 ma (10 ma modulator, 28 ma demodulator)

-10 @ 21 ma (1 ma modulator, 20 ma demodulator)

+6 @ 108 ma (48 ma modulator, 60 ma demodulator)

-6 @ 118 ma (58 ma modulator, 60 ma demodulator)

#### 8.2.2.4 Modem Physical

Chassis Size: 3" x 4" x 17"

Mounting: 3-1/2" x 19"

Interconnections: All connections at rear of panel

#### 8.2.3 Cable

The cable to be used for the data bus is RG22B/U. Its characteristics are as follows:

OD: 0.420 inches

Weight: 0.151 lb/ft

Dielectric: Polyethylene

Jacket: Black PVC non-contaminating

Impedance: 95 ohms

Figure 8-5 is a plot of attenuation versus frequency for the RG22B/U.

ATTENUATION (DB/FT)

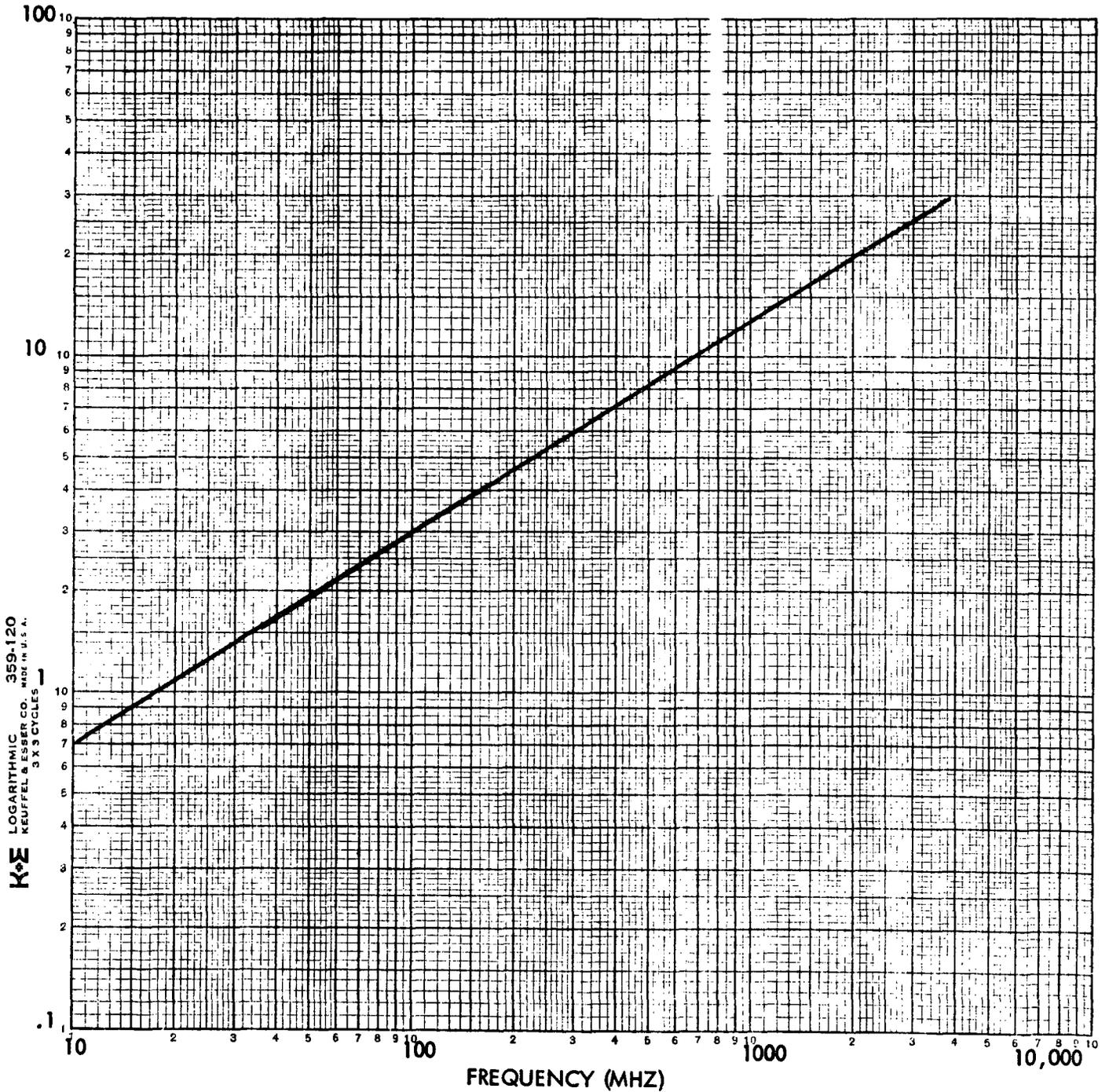


Figure 8-5. Attenuation vs Frequency for RG-22B/U Cable

## 8.2.4 Pattern Generator and Correlator

### Introduction

The pattern generator and correlator provides test messages to be sent through the data bus and means to check the resulting but output message for bit errors. A 20 MHz square wave input clock is required for operation and an output is available in the form of error pulses to be applied to a counter.

### Inputs

20 MHz square wave clock (TTL compatible).

Bus demodulator outputs.

### Outputs

Error Pulses (TTL signal)

Bus modulator inputs.

### Controls

MODE--Selects either a continuous stream of test messages or a single message.

START--The SINGLE MESSAGE position.

PATTERN--A SWITCH SELECTABLE bit pattern for the test messages.

BIT PATTERN--PATTERN switch is in the SWITCH SELECTABLE position.

MESSAGE LENGTH--Selects either LONG (993 bits) or SHORT (8 bits) test messages. (Lengths are in addition to preamble and postamble.)

INTERVAL BETWEEN MESSAGES--Selects either a SHORT (6 bits) or LONG (17 bits) interval between messages when in the continuous mode.

CLOCK SHIFT BETWEEN MESSAGES--Selects either NONE, 90°, or 180° phase shift between adjacent test messages.

TEST COUNTER--Provides means for artificially generating error pulses so that the error counter can be adjusted.

Power

120 VAC 60 cycles for +5 VDC, 3A power supply

Physical

Chassis Size: 3" x 8" x 12"

Mounting: 5-1/4" x 19"

Interconnections: Input clock and Errors output via front panel  
Connections to modems via rear panel umbilicals.

## 9. DBCU DESIGN

### 9.1 PHYSICAL DESCRIPTION

The DBCU functions are physically located in two standard 19 inch equipment rack drawers. The major electronics is in the DBCU drawer which also has all peripheral interface connectors and the display panel and controls. The power converter for the DBCU is located in the Tape Reader/Power Unit drawer. The two drawers are connected during operation by six foot pigtail connections from the Tape Reader/Power Unit.

#### 9.1.1 DBCU Logic

The DBCU logic is contained in a drawer with a standard 19 inch width front panel which is 8-23/32 inches high. Drawer slides provide 180° tilt and lock features. The drawer depth is 21 inches excluding rear panel connectors, and the height is 5 inches.

The logic ICs are contained on two wire wrapped planes within the enclosed drawer. The upper plane can also be tilted to a 90° position and locked in place. Two muffin fans are provided internally for forced air cooling of the ICs. The planes are positioned such that the wire wrap pins are in the center where maximum air flow is directed.

#### 9.1.2 Tape Reader/Power Unit

The front panel measures 19 by 5-1/2 inches and the drawer depth is 15 inches. The tape reader unit extends 2-3/4 inches in front of the front panel.

The tape reader is a unidirectional unit capable of reading 5, 6, 7, or 8 level tape. It is completely asynchronous and will operate at any speed up to 50 characters per second. The drive unit requires a 48 volt power converter which is mounted on a bracket behind the front panel. The associated drive electronics is located on a printed circuit board on the left side of the reader behind the front panel.

The DBCU power converter is mounted across the 17 inch drawer width behind the tape reader and electronics. It is capable of supplying +5 VDC  $\pm$  .05% at 25 amperes and has overload and overvoltage protection features.

The TR/PU makes all external connections through two six foot pigtail connections and two six foot power cords.

## 9.2 CONTROLS AND DISPLAYS

### 9.2.1 DBCUC Logic Drawer

The DBCUC control, display, and monitor functions are shown in the front panel photo of Figure 9-1. The functions and normal settings of the various switches and controls are described below.

#### 9.2.1.1 Switches

BIO/TEST PANEL (SPDT TOGGLE) - When the "Test Panel" position, it allows control of all DBCUC operation and data flow from the test panel switches and the BIO interface is disabled. When in the "BIO" position, it allows control of certain DBCUC operations from the BIO interface. In this mode, however, certain of test panel switches can override the BIO commands when not in the center off position. This switch is normally in the "BIO" position.

OPERATE/LOAD (SPDT TOGGLE) - Normally in the "Operate" position, it allows normal operation of the DBCUC under control of the BIO or test panel switches. When in the "load" position it provides access to the DBCUC internal memory for program loading from either a keyboard or tape reader.

KEYBOARD/READER (SPDT TOGGLE) - This switch is active only when the Operate/load switch is in the "Load" position. In the "Keyboard" position it allows data entry to the DBCUC memory from the panel mounted keyboard. In the "Reader" position, it allows data entry from an external tape reader connected to the DBCUC through the rear panel connectors.

KEYBOARD (11 SPST MOMENTARY PUSHBUTTONS) - The keyboard allows entry of data into either the DBCUC memory or the memory address register. The data are entered to interim storage in groups of three octal digits (equivalent to an eight bit binary byte). The data is transferred (NDRO) into the DBCUC memory address register when the "ADDR" key is depressed. It is transferred (NDRO) into the memory location indicated by the memory address register when the "DATA" key is depressed. The interim storage is cleared to zero when the "CLEAR" key is depressed.

As the digit keys are depressed, their value will be displayed in proper position if the "Display Control" switch is in position five. If a wrong key or too many keys are depressed, the "CLEAR" key should be depressed and the data re-entered.

To enter many data bytes into successive memory locations, first enter the starting memory address with the proper three digits and the "ADDR" key. Next, enter the desired data into this location with the proper three octal digits and the "DATA" key. When the "DATA" key is depressed, the data is loaded into the designated memory location and then the memory address register is automatically incremented to the next subsequent locations. The keyboard register is now ready for the next data byte. The same data may be entered again by depressing the "DATA" key or new data loaded by depressing the digit keys.

RESET (SPST MOMENTARY PUSHBUTTON) - When depressed, this button resets

# DATA BUS CONTROLLER UNIT

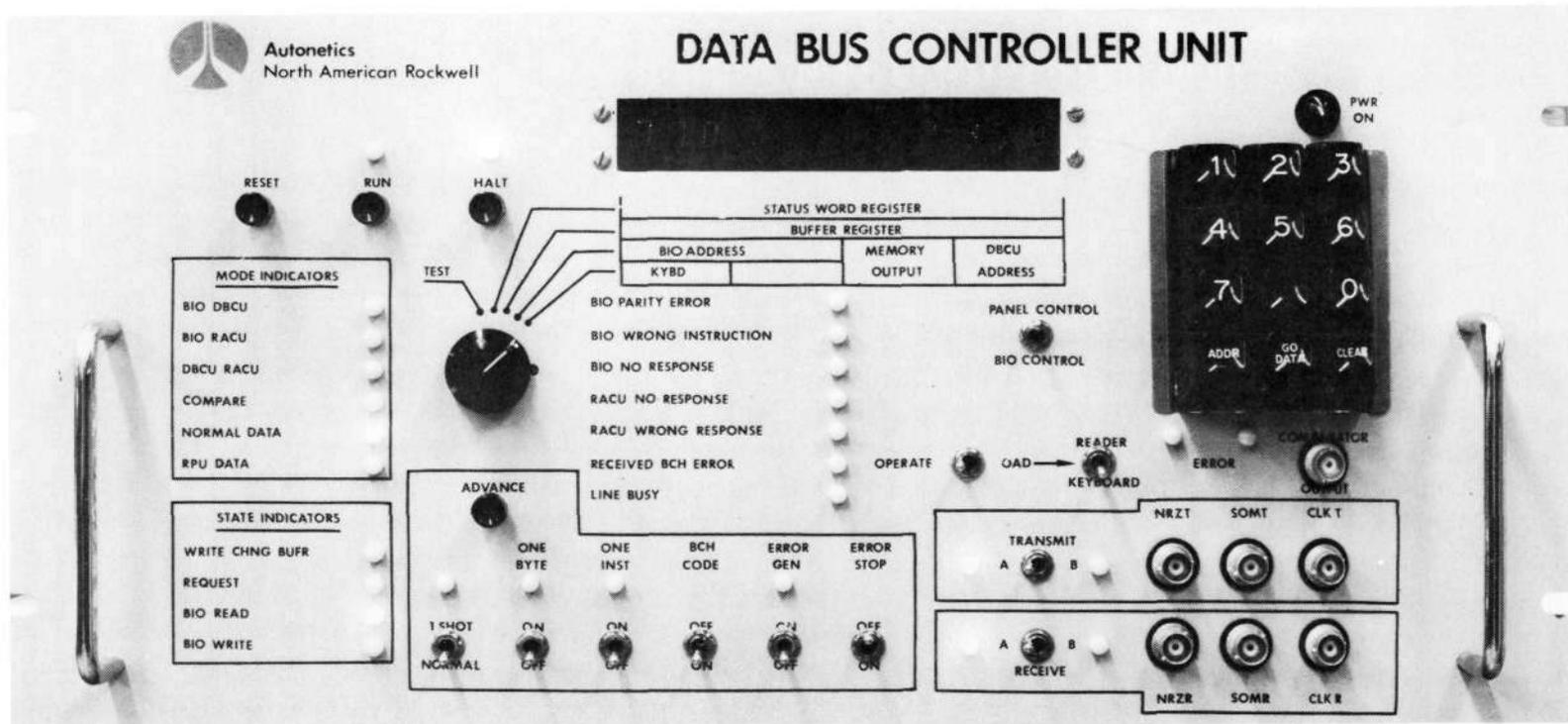


Figure 9-1. DBCU Front Panel Assembly

the DBCU logic to initial conditions. The DBCU registers and flip-flops are set to the following initial conditions:

|                                |         |
|--------------------------------|---------|
| Keyboard register to           | 0       |
| Tape input register to         | 0       |
| Run FF to                      | HALT    |
| Status Error FFs to            | 0       |
| Byte Select Counter to         | 00      |
| Tape Reader Stop to            | 0       |
| Tape Reader Parity Error FF to | 0       |
| Read/Write FF to               | READ    |
| Normal/RPU FF to               | NORMAL  |
| Chain Bit FF to                | CHAINED |
| Pre Write Change Buffer FF to  | PWCB    |

RUN (SPST MOMENTARY PUSHBUTTON) - When depressed, this switch starts the DBCU operation. The mode of operation is dependant upon internal logic set previously from the front panel switches and/or software program control instructions from the BIO or DBCU internal memory.

HALT (SPST MOMENTARY PUSHBUTTON) - When depressed, this switch stops the DBCU operation at the end of the operation or instruction set presently being performed. When the "RUN" switch is depressed the operation will begin with the next subsequent instruction.

DISPLAY CONTROL (5 POSITION ROTARY) - This switch selects the data to be displayed at the front panel numeric display readouts. The display contains twelve solid state numeric indicators which are spaced in groups of three characters per byte. The readouts are used as octal displays and have a right hand decimal point on the right most character of each byte for certain read-out functions.

Position 1: This position is used to test the displays. Each character should have all 24 LED's lit to form a figure eight as shown.



Position 2: This position displays the contents of the status register as four eight bit bytes in octal (i.e; a maximum number readout would be displayed as 377. 377. 377. 377. ). Upon initial turn-on in the reset mode this display should read XXX 000.000.XXX.

Position 3: This position displays the contents of the accumulator register as four eight bit bytes in octal. In addition, the odd parity bit of each byte is displayed in the right hand decimal point of the right most character of each byte. A logic true parity bit will be indicated by a lit decimal point. Upon power turn-on or in the reset mode, this display should read 000. 000. 000. 000. with the decimal points lit.

Position 4: This position displays the contents of the BIO address register in the first four readouts, blanks in readouts five and six, the DBCU memory data output in readouts seven through nine, and DBCU memory address register #1 in readouts ten through twelve. A maximum count readout would be displayed as 777 7BB 377 777, 1 Upon power turn-on or in the reset mode, this display should read XXX XBB XXX XXX.

Position 5: This position displays the contents of the keyboard input register in readouts one through three, blanks in the next byte (readouts four through six) the memory data output in readouts seven through nine, and the memory address register in readouts ten through twelve.

The keyboard register is displayed only when the front panel switches are in the "LOAD" and "KEYBOARD" positions. In other modes this portion of the display is blanked out.

A maximum count readout in the "LOAD-KEYBOARD" mode would be displayed as 777 BBB 377 777. Upon power turn-on or in the reset mode, this display should read XXX BBB XXX XXX. In the non-Keyboard mode a maximum count would read as BBB BBB 377 777 and a reset mode readout should read EBB BBB XXX XXX.

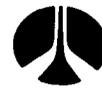
CLOCK NORMAL/ONE-SHOT (SPDT - CO) - When in the "NORMAL" position, this switch forces DBCU timing from a continuous internal clock source of 10 MHz. When in the "One-Shot" position, the switch gates the internal timing one clock pulse at a time each time the "ADVANCE" pushbutton is depressed. These positions are overriding to any program control of these functions. This switch is normally in the center position which allows software program control of the clock function.

ONE BYTE ON/OFF (SPDT - CO) - When in the "ON" position this switch gates the internal timing in bursts of eight clock pulses at a time each time the "ADVANCE" pushbutton is depressed. In the "OFF" position it locks out the byte at a time operation. The switch is normally in the center position which allows software program control of this operation.

ONE INST ON/OFF (SPDT-CO) - When in the "ON" position this switch gates the internal timing such that one instruction is executed each time the "ADVANCE" pushbutton is depressed. In the "OFF" position it locks out the instruction-at-a-time operation. The switch is normally in the center position which allows software program control of this operation.

NOTE: Use of the above three options is for trouble shooting only - clock pulses (synchronization) will eventually be lost in the logic due to the use of asynchronous signals.

NOTE (1) "B" signifies a blank or no LED's lit and "X" signifies an indeterminate state.



ADVANCE (SPDT PUSHBUTTON) - This pushbutton is used to sequence or "advance" the functions of clock-at-a-time, byte-at-a-time, or instruction-at-a-time as described above.

BCH CODE ON/OFF (SPDT TOGGLE) - When in the "ON" position, this switch enables the operation of the BCH code generator and detector for the NRZ serial data at the modem interfaces. In this mode, a BCH (127,112) code is generated after every 14 data bytes (112 bits) and the 15 bit BCH code is inserted in the bit stream with a trailing zero bit. Likewise, this BCH code is detected for errors on all incoming data.

When this switch is in the "OFF" position, the BCH code generation and detection capability is inhibited. The switch is normally in the "ON" position.

ERROR GENERATOR OFF/ON (SPDT - CO) - When in the "ON" position, this switch causes an error bit to be placed in the last BCH code of an outgoing message and is active only when the BCH code switch is "ON". In the "OFF" position it holds the BCH operation in a normal non-error mode. This switch is normally in the center position which allows software program control of the error generation function.

ERROR STOP ON/OFF (SPDT TOGGLE) - When in the "ON" position this switch places the DBCU in a mode which halts operation when a detected error occurs and informs the BIO by transfer of a status word. When in the "OFF" position, this function is inhibited and the DBCU does not halt operation upon occurrence of these errors. The detected errors involved include

1. BCH error
2. No RACU response
3. RACU address error
4. Line busy error
5. No BIO response to a write action
6. Instruction error
7. Comparison error (when override is off)

This switch is normally in the "ON" position. BIO parity errors always cause a stop regardless of the position of this switch.

TRANSMIT MODEM SELECT A/B (SPDT - CO) - This switch selects the modem line "A" or "B" which is to receive the NRZT data for transmittal. It also gates the SOMT and transmit enable lines to the appropriate modem. The switch is normally in the center position which allows software program control of this function.

RECEIVE MODEM SELECT A/B (SPDT - CO) - This switch selects the modem "A" or "B" lines which will be gated into the DBCU to receive the NRZR data, received clock, SOMR pulse, and the modulation present signal. The switch is normally in the center position which allows software program control of this function.

### 9.2.1.2 Monitor Jacks

NRZT - Monitors the NRZ data output to modem A or B as selected by the Transmit A/B switch.

SOMT - Monitors the start of message transmit pulse to modem A or B as selected by the Transmit A/B switch.

CLKT - Monitors the continuous clock line which is connected to both modems A and B.

NRZR - Monitors the NRZ data input received from modem A or B as selected by the Receive A/B switch.

SOMR - Monitors the start of message receive pulse from modem A or B as selected by the Receive A/B switch.

CLKR - Monitors the received clock signal from modem A or B as selected by the Receive A/B switch.

COMPARATOR OUTPUT - Monitors the output of the internal comparator circuit. The output is active during the compare mode of operation when memory transmitted data is compared to memory received data on a byte by byte basis. The output goes true (high) for 1 bit time each time a comparison error occurs. At all other times the output is false (low). Continuous errors would be seen as a pulse train which is high for 1 bit time and low for approximately 8 bit times.

### 9.2.1.3 Panel Lights

BIO/DBCUC - Indicates that the DBCU is operating in a BIO to DBCU or DBCU to BIO data transfer mode.

BIO/RACUC - Indicates that the DBCU is operating in a BIO to RACU or RACU to BIO data transfer mode.

DBCUC/RACUC - Indicates that the DBCU is operating in a DBCU to RACU or RACU to DBCU data transfer mode.

COMPARE - Indicates that the DBCU is in an internal comparison mode in which transmitted data is compared to received data.

NORMAL DATA - Indicates that data in the "normal" format is being transmitted to/from a RACU.

RPU DATA - Indicates that data in the "RPU" format is being transmitted to/from an RPU through a RACU.

REQUEST - Indicates that the request line to the BIO is on when lit. A request for BIO access is being made.

WRITE CHANGE BUFFER - Indicates that the write change buffer flip-flop is on when lit. The request is to write data into the BIO write change buffer.

BIO READ - Indicates that the request is to read data from the BIO when lit.

BIO WRITE - Indicates that the request is to write data into the bio when lit.

ONE SHOT - Indicates that the DBCU timing is in a bit at a time mode of operation (see note in paragraph 9.2.1.1).

ONE BYTE - Indicates that the DBCU is in the byte at a time mode of operation when lit (see note in paragraph 9.2.1.1).

ONE INSTRUCTION - Indicates that the DBCU is in the instruction at a time mode of operation when lit (see note in paragraph 9.2.1.1).

ERROR GENERATOR - Indicates that the DBCU is generating a BCH encoding error when lit.

TRANSMIT A - Indicates that the DBCU is transmitting data over the Modem B lines.

TRANSMIT B - Indicates that the DBCU is transmitting data over the Modem B lines.

RECEIVE A - Indicates that the DBCU is receiving data over the Modem A lines.

RECEIVE B - Indicates that the DBCU is receiving data over the Modem B lines.

RUN - Indicates that the DBCU is in the run mode of operation.

HALT - Indicates that the DBCU is in the halt mode of operation.

COMPARATOR ERROR - Indicates that the DBCU is in an error mode caused by a comparison error. This output is also available at the comparator output BNC jack.

TAPE READER ERROR - Indicates that a parity error was detected during memory fill from the tape reader. The DBCU is put in an error stop mode.

BIO NO RESPONSE - Indicates that the DBCU is in an error mode caused by lack of a BIO response to a DBCU write request.

WRONG INSTRUCTION - Indicates that the DBCU detected an invalid OP code is now in an error mode.

BIO PARITY ERROR - Indicates that a parity error has been detected on the BIO to DBCU data transfer. The DBCU halts upon error detection.

RACU NO RESPONSE - Indicates that a RACU did not respond to its message in a given time period and the DBCU is in an error stop mode.

RACU WRONG RESPONSE - Indicates that an invalid address mode was detected from a responding RACU and the DBCU is now in an error stop mode.

RECEIVED BCH ERROR - Indicates that an error was detected in the received BCH encoded message and DBCU is in an error stop mode.

POWER ON - Indicates the 5VDC prime power is being supplied to the DBCU.

### 9.2.2 Tape Reader/Power Unit Drawer

The TR/PU control and monitor lights are shown in the sketch of Figure 9-2.

#### 9.2.2.1 Switches

Tape Reader - This switch controls the prime power to the tape reader power converter for the motor and drive electronics. The tape reader requires + 48 VDC at 250 ma peak.

DBCU Power - This switch controls the prime power to the DC power converter for the DBCU. The DBCU requires + 5 VDC  $\pm$  5% at 20 amperes maximum. This converter is fused at 20 amperes at the DBCU drawer rear panel.

#### 9.2.2.2 Monitor Lights

DBCU Power - This light indicates when 110 VAC prime power is being supplied to the DBCU power converter in this drawer and to the cooling fans in the DBCU drawer.

### 9.3 BASIC OPERATION

The Data Bus Control Unit (DBCU) software instructions are executed depending upon the basic mode of operation of the DBCU. The basic mode determines where from instructions are executed such as

- I. The DBCU (internal memory).
- II. The Buffer Input/Output unit (BIO)

Instructions obeyed in Mode I are

- 001 Compare (data internal to DBCU memory).
- 011 Set Mode of Operation.
- 010 DBCU/RACU data transfer.

9-10

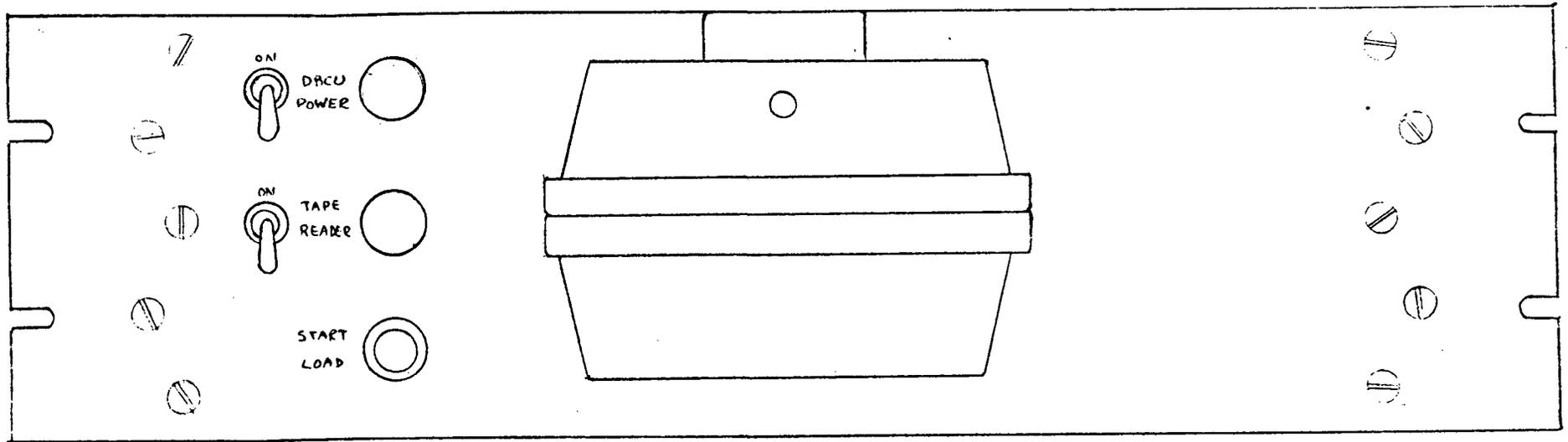


Figure 9-2. Tape Reader/Power Unit, Front Panel

Instructions obeyed in Mode II are

- 111 BIO/RACU data transfer
- 101 BIO/DBCUC data transfer
- 011 Set Mode of Operation

Figures 9-3 through 9-7 show in detail the structure of these commands.

Mode I is further subdivided into two modes which are the Operate and Load Mode. The operate mode retrieves instructions and processes data to/from internal memory. The load mode loads internal memory from a keyboard or tape reader.

Error and status conditions are reported via a status register to a visual display and to the BIO. The format of the status word is shown in Figure 9-8.

Data from the BIO or the DBCUC memory to be sent to the RACUC has already been described. The format of the word to/from the BIO is discussed in ICD AN26465.

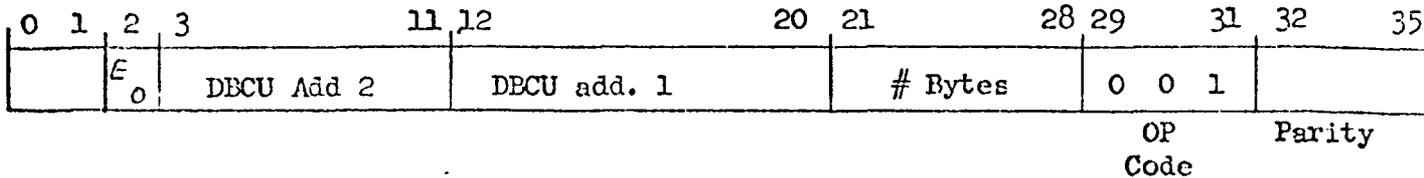
The discussion of the operation of the DBCUC is presented here in reference to data flow paths designated by Figure 9-9.

#### 9.3.1 Load Mode

The load mode is selected by placing the OPERATE/LOAD switch in the LOAD position and the BIO/PANEL switch in the PANEL position. Loading can then occur from the device selected by the TAPE/KBRD switch. Under tape reader control the tape is read, ignoring all blanks, until the first address frame (identified by a particular tape code) is read. The first address frame contains the most significant three bits of the address, which are loaded into a register in the tape reader interface. After a total of three frames have been read, the nine bit address is stored in the register. After the last address frame is stored, the contents of the tape reader interface register are copied into the memory address register.

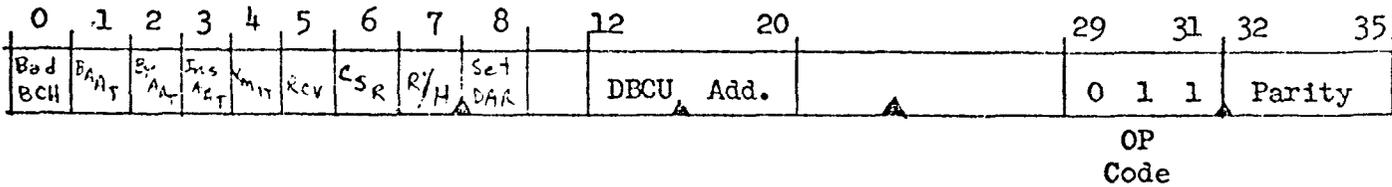
The next frame of tape contains the first four bits of the first data byte. This frame is copied into four bits of the tape reader interface register. A second data frame follows and the remaining four bits are copied. The tape reader interface register contents are then made available to the memory data input and a memory write cycle is initiated copying the byte into the location designated by the address register. At the end of the write cycle the address register is incremented. The next data byte from the tape is then read, copied into the tape reader interface and into memory. This process ends when the tape reader interface detects an end of tape code or a parity error.

The tape is coded using an odd parity bit in channel eight. Whenever incorrect parity is detected during a tape reading operation the tape reader is stopped and the front panel READER ERROR indicator is lit.



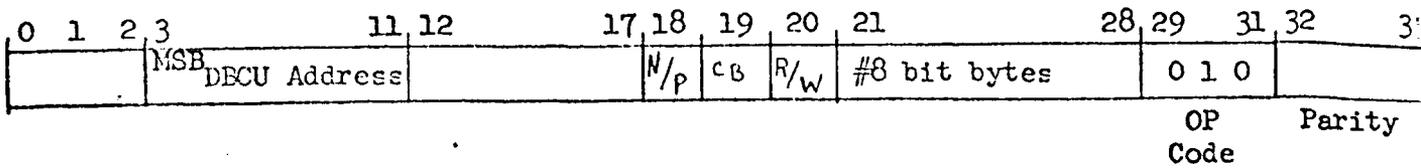
- I (2) =1= Error override. Do not halt on comparison error.
- I (3:11) = Starting DECU memory address #2. MSB in I (3). Compare (internal) data words starting with address I (12:20) to data words starting with address I (3:11) for the number of 8 bit bytes defined in I (21:28).
- I (12:20) = Starting DECU memory address #1. MSB in I (12).
- I (21:28) = Number of 8 bit bytes to be compared. MSB at I (21).
- I (29:31) = OP code 001.
- I (32:35) = Parity bits.

Figure 9-3. Compare



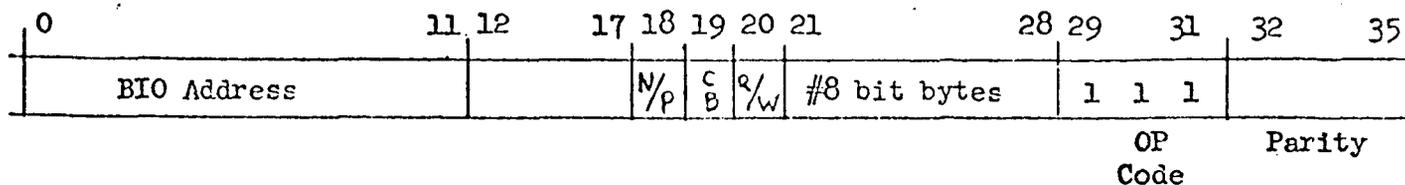
- I (0)=1= Set bad parity on BCH code generator
- I (1)=1= Bit at a time operation
- I (2)=1= Byte at a time operation
- I (3)=1= Instruction set (message) at a time operation
- I (4)=0= Transmit on Channel A  
=1= Transmit on Channel B
- I (5)=0= Receive on Channel A  
=1= Receive on Channel B
- I (6)=1= Clear Status Register
- I (7)=0= Halt mode  
=1= Run mode
- I (8)=1= Set DBCU address register to value indicated in I (12:21). (ie; JUMP to location I (12:20)).
- I (12:20)= Starting DBCU memory address. MSB at I (12).
- I (29:31)= OP code 011.
- I (32:35)= Parity bits.

Figure 9-4. Set Mode of Operation



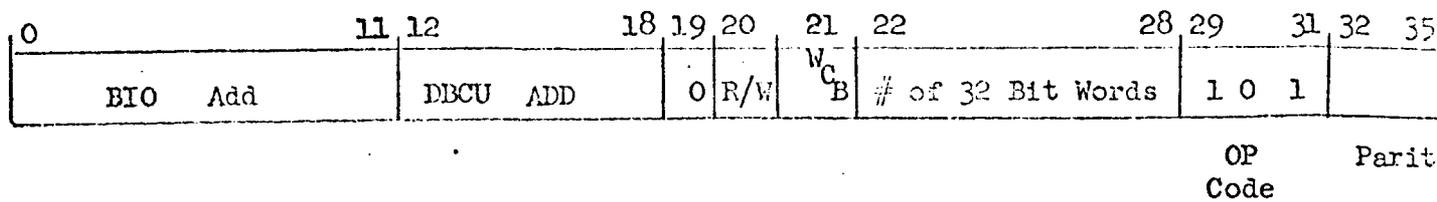
- I (0:2) = Spare
- I (3:11) = DECU memory address. MSB = I (3).
- I (12:17) = Spare
- I (18) =0= Normal data format to/from the RACU  
 =1= Preprocessor (RPU) data format to/from the RACU.
- I (19) = Chain bit. 1 = Next word is an instruction word.
- I (20) =0= Read from DECU memory starting at address I (3:11).  
 =1= Write into DECU memory starting at address I (3:11).
- I (21:28) = Number of 8 bit bytes to be transferred into or out of memory. - MSB at I (21).
- I (29:31) = OP Code. 010
- I (32:35) Parity bits.

Figure 9-5. DECU/RACU Data Transfer



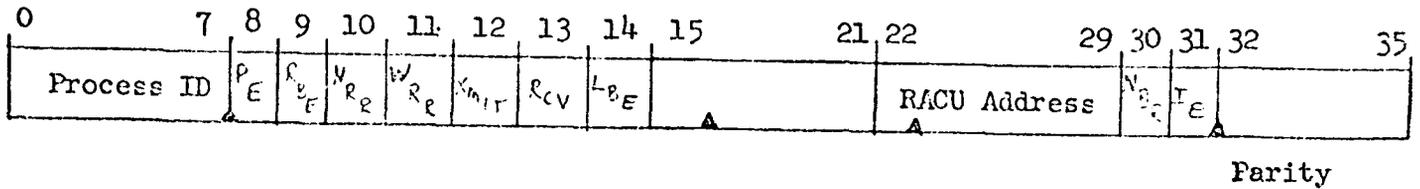
- I (0:11) = BIO address. MSB = I (0).
- I (12:17) = Spare.
- I (18) =0= Normal data format to/from the RACU.  
=1= Preprocessor data format to/from the RACU.
- I (19) = Chain bit. 1 = Next word is an instruction word.
- I (20) =0= Read from BIO memory starting at address I (0:11).  
=1= Write into BIO memory starting at address I (0:11).
- I (21:28) = Number of 8 bit bytes to be transferred. MSB at I (21).
- I (29:31) = OP Code. 111
- I (32:35) = Parity bits.

Figure 9-6. BIO/RACU Data Transfer



- I (0:11) = BIO memory address. MSB = I (0)
- I (12:18) = DBCU memory address. MSB = I (12) = Binary 256.
- I (19) = 0
- I (20) =0= Read from BIO memory address starting at I (0:11) into DBCU memory address starting at I (12:18).  
=1= Write into BIO memory address starting at I (0:11) from DBCU memory address starting at I (12:18).
- I (21) =1= Write into Write Change Buffer. (one word).
- I (22:28) = Number of 32 bit words to be transferred. MSB at I (22).
- I (29:31) = OP Code 101.
- I (32:35) = Parity bits

Figure 9-7. BIO/DBCUC Data Transfer



- I (0:7) = Process ID. ( 08 ) Hex
- I (8) =1= Parity error detected on BIO data transfer.
- I (9) =1= BCH error detected in data from RACU I (22:29).
- I (10) =1= RACU I (22:29) did not respond.
- I (11) =1= Wrong RACU response. Should be from RACU I (22:29).
- I (12) =0= Transmitting on channel A  
=1= Transmitting on channel B
- I (13) =0= Receiving on channel A  
=1= Receiving on channel B
- I (14) =1= Linebusy error. Modulation present on line indicated when attempting to transmit.
- I (15:21) = Spare
- I (22:29) = State of RACU address register. MSB at I (22)
- I (30) = No BIO response to request for write operation.
- I (31) = Instruction error. Improper OP Code detected.
- I (32:35) = Parity bits.

Figure 9-8. Status Word



Keyboard operation is analogous to the tape reader mode. Address or data bytes are entered into the keyboard interface register by depressing three successive digits on the keyboard. Depressing the ADDRESS key copies the contents of the keyboard interface register into the memory address register. When the DATA key is depressed the contents of the keyboard interface register are made available to the memory data input and a write cycle is initiated. At the end of the write cycle the memory address register is incremented.

### 9.3.2 Operate Mode

Placing the OPERATE/LOAD switch in the OPERATE position selects the operate mode. In this mode the BIO/PANEL switch selects the source of control instructions and data.

#### 9.3.2.1 BIO Control Modes

In these modes 36 bit control and data words are received from the BIO and 36 bit data words are transferred to the BIO. A hardwired DBCU operation reads a command from location  $0010_8$  of the BIO memory by setting BIO Address Register #1 to  $0010_8$  and requesting a BIO read operation. This command is loaded into the DBCU Buffer (B) register. The outputs of the B register are monitored for correct parity. If a parity error exists on any byte the BIO PARITY ERROR indicator is lit, and the error mode is entered. (See 9.3.2.3).

Assuming that no parity error exists, the command in B is copied into the Accumulator (A) register. The operation code bits of the command work are then decoded from A to identify one of the following modes: BIO/RACU, BIO/DBCU or SET MODE. If any other code is detected the INSTRUCTION ERROR indicator is lit and the error mode is entered.

##### 9.3.2.1.1 BIO/RACU Mode

The DBCU operates as a formatter in this mode, translating between a parallel BIO interface and a serial modem interface. 36 bit words are copied into B and then into A. From A, eight bit bytes are alternately copied into Shift Registers #1 and #2 (SR1, SR2). The first byte, which contains the RACU address, is also copied into the RACU Address Register. Data is serially shifted from SR1 and SR2 alternately, through the BCH Code Register to add proper encoding and through the Modem Interface for eventual transmission to the selected RACU. The Modem Interface logic adds the preamble, postamble and transmission control signals.

When data is received from the RACU, the Modem Interface logic strips the preamble and initiates serial transfer of data through the BCH Code Register alternately to SR1 and SR2. In this configuration the BCH Code Register operates as a BCH code checker, checking incoming data for proper encoding. If a BCH code error is detected the error mode is entered.

The first byte received by SR1 contains the address of the transmitting RACU. This address is checked against the contents of the RACU Address Register. Incorrect RACU address will cause an exit to the Error Mode. The

data shifted into SR1 and SR2 is copied into B one byte at a time. When a total of four bytes has been loaded into B or when the end of transmission from the RACU is indicated, whichever occurs first, B is copied into A, thereby freeing B to accept the next byte from SR1. Simultaneously with the transfer from B to A, four parity bits are generated and stored by the Parity Generator and Checker. The contents of A plus the four parity bits are then transmitted to the BIO.

If it is desired to complete a data exchange with a RACU (transmit a string of data to a RACU and then receive a string of data from the RACU) without waiting for a separate instruction word access from the BIO, the chained mode is used. This mode allows the DBCU to store an instruction, load another instruction from the BIO, operate on the instruction just loaded, and then operate on the previously stored instruction.

The first instruction from the BIO (address  $0010_8$ ) would typically be a Write to BIO (receive from RACU) with a chain bit of one. This instruction is stored by copying the BIO address bits of A into BIO Address Register #2, the byte count bits into Byte/Word Count Register #2 and the Read/Write and Normal/Preprocessor bits into the secondary control flip-flops.

The next instruction from the BIO (address  $0011_8$ ) would be a Read from BIO (transmit to RACU) with a chain bit of zero. This instruction would be copied into BIO Address Register #1, Byte/Word Counter #1 and the primary control flip-flops and then executed. Upon termination of this execution BIO Address Register #2 is copied into BIO Address Register #1, Byte/Word Count Register #2 is copied into Byte/Word Counter #1 and the secondary control flip-flops are copied into the primary control flip-flops. This sets up the Write to BIO instruction which is then executed.

BIO Address Register #1 always contains the next BIO address to be read from or written into. This register is incremented after each work transfer between the BIO and DBCU.

Byte/Word Counter #1 holds the current number of data bytes remaining in the message. This counter is decremented once for every byte transferred between the DBCU and RACU. Outputs of this counter also control incrementing of BIO Address Register #1.

#### 9.3.2.1.2 BIO/DBCU Mode

Memory transfers between the BIO and DBCU are accomplished using this mode. Also, by commanding any BIO/DBCU operation with bit 21 of the command word set to one overrides the designated operation and reads the DBCU status word into the BIO.

When the instruction from the BIO is a Read from BIO (load DBCU memory) the BIO address bits of A are loaded into BIO Address Register #1 and the BIO word count bits are loaded into Byte/Word Counter #1. The DBCU address bits are loaded into Address Register #1 and then copied into the Memory Address Register (MAR). Another word is then received from the BIO and stored in A. Byte 1 of A is then selected and written into memory. The contents

of MAR are incremented and then byte two of A is written into memory. This continues until all four bytes of A are loaded into successive locations in the memory. Another word is then transferred from BIO to A and the process repeats until Byte/Word Counter #1, which is decremented after each access to the BIO, goes to zero.

To read the DBCU memory a Write to BIO instruction is used. The registers are loaded as in the previous instruction. In this case four byte words are assembled in B from successive locations in memory, transferred to A and then transmitted to the BIO.

#### 9.3.2.1.3 Set Mode

The set mode instruction provides the means to set certain DBCU control flip-flops to particular states and to set MAR to a defined location. Bits 0 through 7 of A are copied into the appropriate mode control flip-flops. The DBCU address bits are copied into MAR if bit eight of the instruction is a one.

#### 9.3.2.2 DBCU Control Modes

DBCU control modes operate on data and instructions stored in DBCU memory. The initial location in memory for the first instruction is normally set from the keyboard. Four successive bytes are read from memory to B and then transferred to A where the instruction is decoded. The following modes are available; DBCU/RACU, COMPARE and SET MODE.

##### 9.3.2.2.1 DBCU/RACU Mode

Operation in this mode is analogous to the BIO/RACU mode discussed in 3.3.2.1.1 except that the source of data for transmission and the sink for received data is the DBCU memory rather than the BIO. Data to be transmitted to a RACU is transmitted from memory to SR1 and SR2 alternately. Data received from a RACU is copied into B and then A as in the BIO/RACU mode. From A, the 32 bit word is transferred into four successive memory locations as in the BIO/DBCU memory load operation.

Chaining is similar to the BIO/DBCU chained operation. The DBCU memory address of the stored instruction is stored in Address Register #2. No BIO address is used. All other registers and control flip-flops are identical to the BIO/RACU mode.

##### 9.3.2.2.2 Compare Mode

Comparison of byte streams from two sections of the DBCU memory is provided in this mode. Address #1 and address #2 of the instruction are loaded into Address Register #1 and #2 respectively. The byte count bits are loaded into Byte/Word Counter #1. The error override bit is copied into a flip-flop.

The first memory address is copied from Address Register #1 to MAR. The first byte from memory is copied into the RACU Address Register. The contents of Address Register #2 are then copied into MAR. The memory output is copied into SR1, and then the Address Comparator is strobed. If the error override

flip-flop is set to zero and a comparison error exists, the error mode is entered. Whether or not a comparison error exists, if the override flip-flop is set to one the process continues.

To access the next two bytes for comparison, Address Registers #1 and #2 are simultaneously incremented as Byte/Word Counter #1 is decremented. The comparison process continues by alternately loading MAR from the two registers and performing the comparisons until Byte/Word Counter #1 reaches zero. The output of the Address Comparator is available at the front panel during this mode so that an error count may be tabulated.

#### 9.3.2.2.3 Set Mode

This DBCU control mode is identical to the BIO control mode discussed in 9.3.2.1.3.

#### 9.3.2.3 Status Word and Error Mode

A status word, comprised of significant DBCU internal flip-flop states plus the contents of the RACU Address Register is available to the BIO through the Write Change Buffer (WCB) interface. The BIO has three means to access this word. One method is to command a BIO/DBCUC operation with bit 21 of the instruction set to one. A second method is an internal DBCUC hardwired instruction which reads the status word to the BIO after completion of any single instruction or chained pair. The third method is triggered whenever the error mode is entered.

In all three cases, action is initiated by setting the Pre WCB flip-flop. Whenever this flip-flop is set, the status word is switched to the BIO/DBCUC data lines and then the WCB signal to the BIO is energized. This section reads the status word into the BIO change buffer.

All errors with the exception of the BIO parity error can be disabled from setting the Pre WCB flip-flop by turning off the ERROR STOP switch. In the DBCUC control modes the Pre WCB flip-flop stops execution after the instruction during which the error occurred has terminated. This state is indicated by lighting of the WRITE CHANGE BUFFER lamp.

## APPENDIX A. PRELIMINARY DACS BREADBOARD LAYOUT

This study was performed to achieve an integrated layout of the DACS breadboard units in test configuration to assure physical compatibility between units.

### Requirements

Physical compatibility includes the physical interface of signals (cabling) and physical interface of hardware (packaging). Both of these aspects have been discussed in DP107 "DACS Breadboard Design Requirements" dated 16 September 1971. Significant points in this document are repeated below:

Page 3-36

"A breadboard unit ... should make maximum use of pre-developed and off-the-shelf devices and functional elements, although the component technology should be the same as that defined for the actual operational design. Breadboard units do not have to be reduced to production techniques or qualified, but their quality must be such that they can be expected to survive the handling normally associated with extensive laboratory testing and operation."

"Where practical, all breadboard equipment shall be mountable in standard 19" equipment racks. To the greatest extent possible, breadboard electronic hardware shall be contained in equipment drawers and mounted in high-density packaging panels in a manner permitting easy access to all components for test or repair."

Page 3-38

"All breadboard units will be cooled by free convection with no cooling provided external to the unit. If necessary, a fan(s) shall be mounted integral to the breadboard unit inside the rack mounted drawer."

### Racks and Panels

MIL-STD-189, "Racks, Electrical Equipment, 19 inch and Associated Panels" should be used as the guidelines and constraints for panel sizes and rack capacities. The 19 inch panel widths are specified in steps of 1-3/4" from 1-23/32" up to 31-15/32". Front panel space on racks range from 17-21/32" to 80-21/32" in 7" steps.

### Basic Breadboard Units

Figure A-1 is a block diagram of a dual redundant test setup of the DACS breadboard. Sufficient units will be available for full integration of the units shown except for the pre-processor, the test processor, and the sub-system simulator. Various lengths of cables, cable adaptors, and line

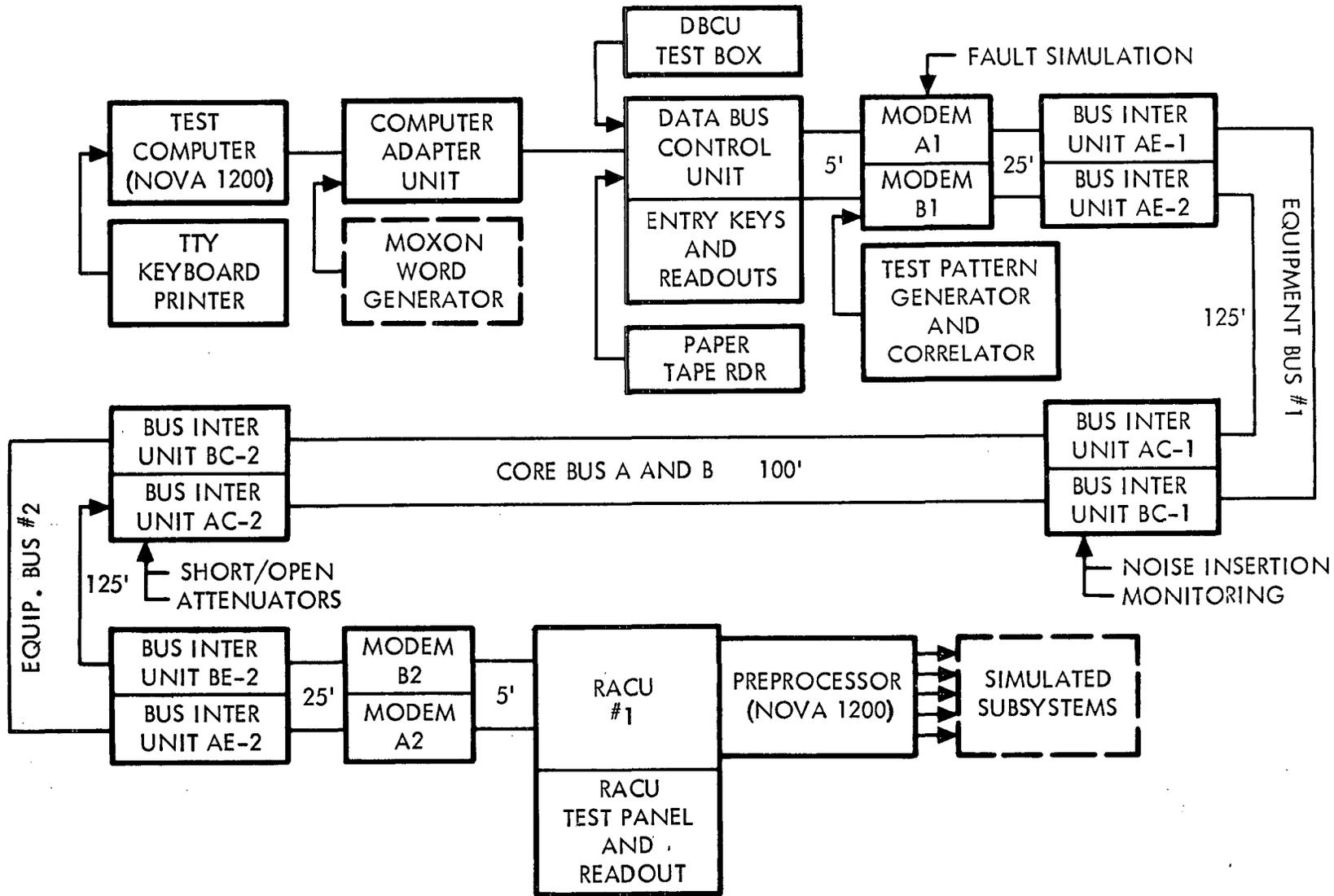


Figure A-1. Data Acquisition and Control Subassembly Breadboard Configuration

A-2

SD 72-SA-0114-3

terminators are also provided to accommodate various test configurations.

The deliverable item list is as follows:

A. Data Bus Control Assembly, consisting of:

- 1 - Data Bus Control Unit
- 1 - Tape Reader/Power Supply Unit
- 1 - Support Test Equipment/Load Box Unit
- 1 - Connecting Cable Assembly
- Assorted Spare IC's

B. Digital Data Bus Assembly, consisting of:

- 3 - MODEM, Type G1
- 1 - MODEM, Type G2  
(with provisions for fault simulation)
- 2 - Bus Interface Unit, Type G1
- 1 - Bus Interface Unit, Type G2  
(with provisions for noise insertion)
- 1 - Bus Interface Unit, Type G3  
(with provisions for inserting attenuators)
- 4 - Bus Interface Unit, Type G4
- 27 - Terminations
- 3 - 100' Sections of Cable
- 8 - 125' Sections of Cable
- 9 - 25' Sections of Cable
- 4 - Jumper Cable Sections
- 1 - Test Pattern Generator and Correlator
- 3 - Line Attenuators

C. Support Documentation, consisting of:

- 2 - Test Program Tapes
- 1 - Tape Reader Manual
- 6 - DP-109/110 DBCU Final Design Reports  
(operating instructions, maintenance data,  
acceptance procedures and test results)
- 6 - DB-104 DDB Final Design Reports  
(operating instructions, maintenance data,  
acceptance procedures and test results)

### Equipment Racks

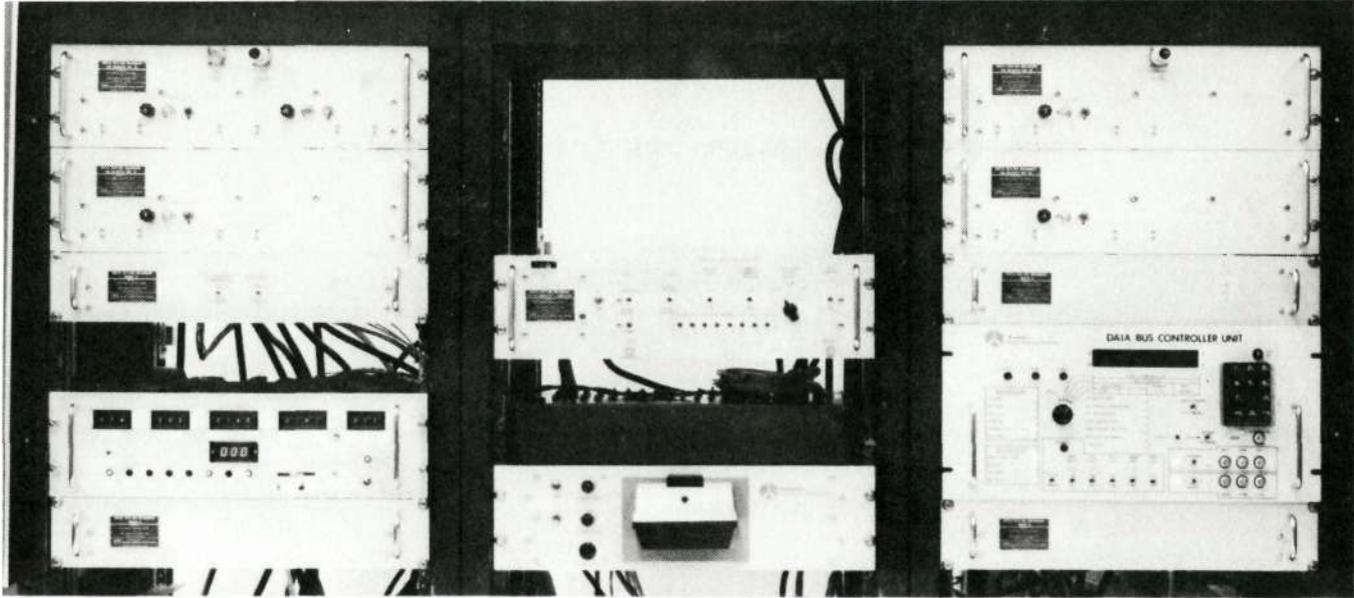
The basic units are shown mounted in a rack in Figure A-6. The center unit is either a DBCU or a RACU which interfaces with up to two modems which in turn each interface with a bus interface unit (BIU). If an RACU is mounted in this area, a spacer panel(s) may be used. The spaces between the BIU's may be used to hold special test equipment such as attenuators to simulate line lengths and loads, power adjustments for BIU fault simulation, methods for inserting noise onto the data bus, or bit error rate detection equipment.

The RACU to Modem pigtails have been specified by ICD to be 36" long. All other interconnecting cables shall be consistent with the equipment rack layout of Figure A-6 so that a panel or drawer may be accessed or pulled out from the front of the rack without disconnecting the cables in the rear. All ac power cords shall have a minimum of 4-1/2 feet extending from the rear of the panel or drawer and be of the 3 prong grounded variety.

The volume at the bottom of each rack may be used for storage of data bus cables, adaptors, and terminators. When in use, the terminators are connected directly to the BIU's or BIU front panel interface.

A bench rack test setup is shown in the photograph of Figure A-2. All cabling is at the rear of the racks including the data bus cables. The BIU panel layouts allow bus connection at the rear or front of the racks to accommodate various test setups for initial integration or future evaluation testing. The test setup shown allows integration tests of a full-up dual redundant DACS breadboard.

Figures A-3 through A-6 are photographs of the Data Bus Controller Unit. Figures A-7 and A-8 are photographs of the RACU.



A-5

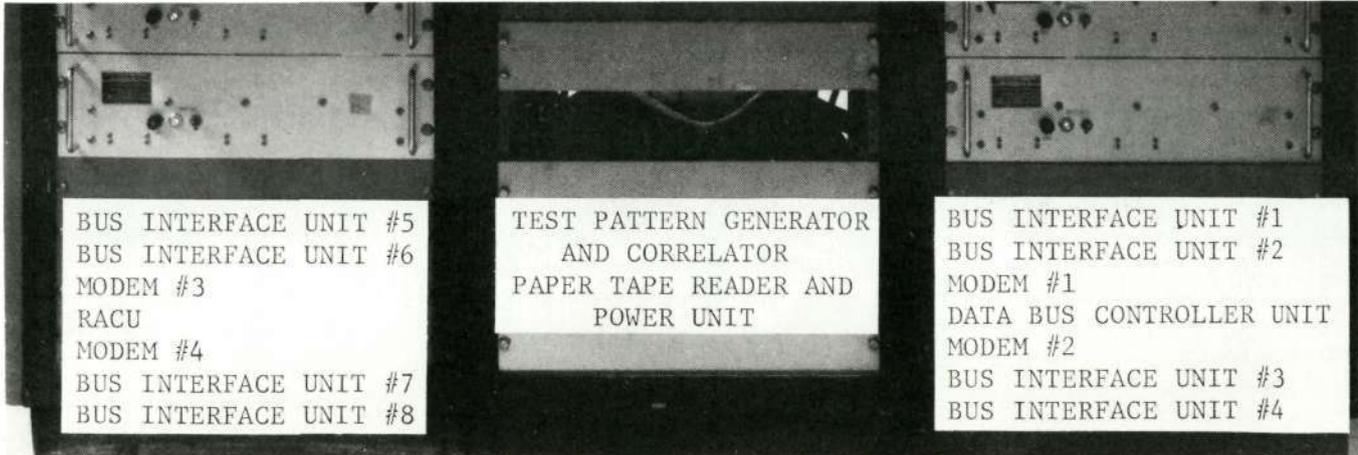


Figure A-2. DACS Breadboard Configuration

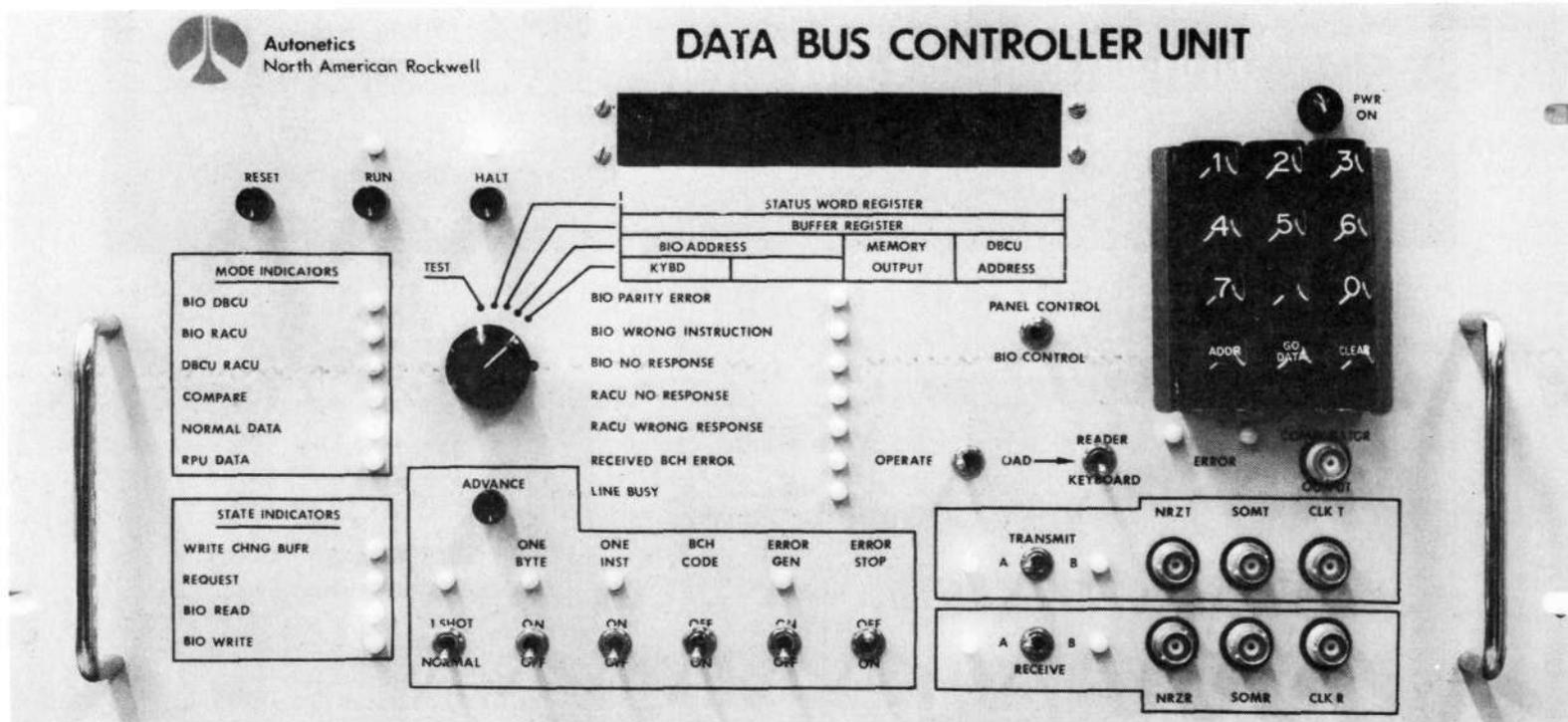


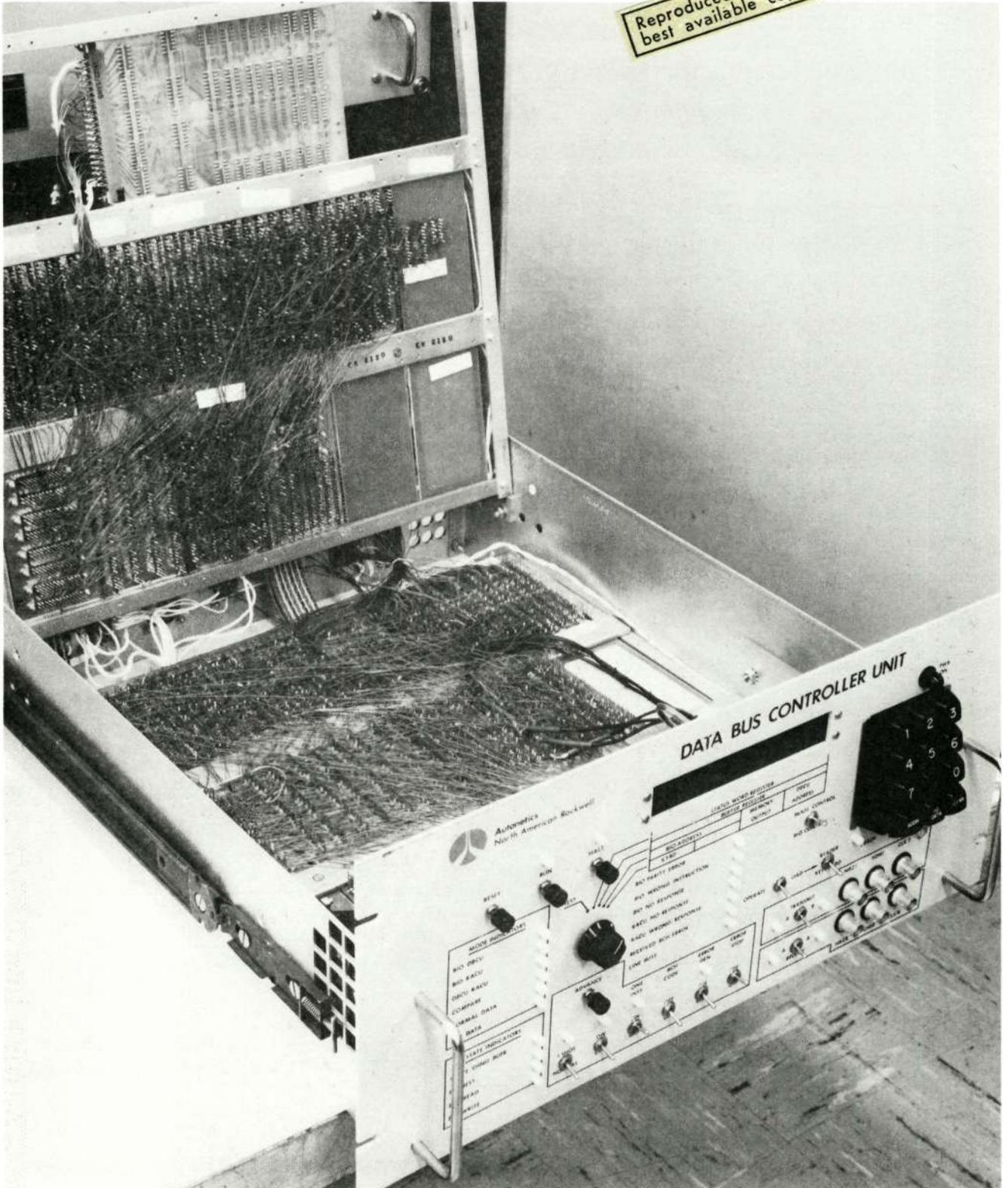
Figure A-3. Data Bus Control Unit Front Panel

A-6

SD 72-SA-0114-3



Reproduced from  
best available copy.



700-83-840 E

Figure A-5. Data Bus Control Unit Top Wiring Plane Lifted

700-83-840 B

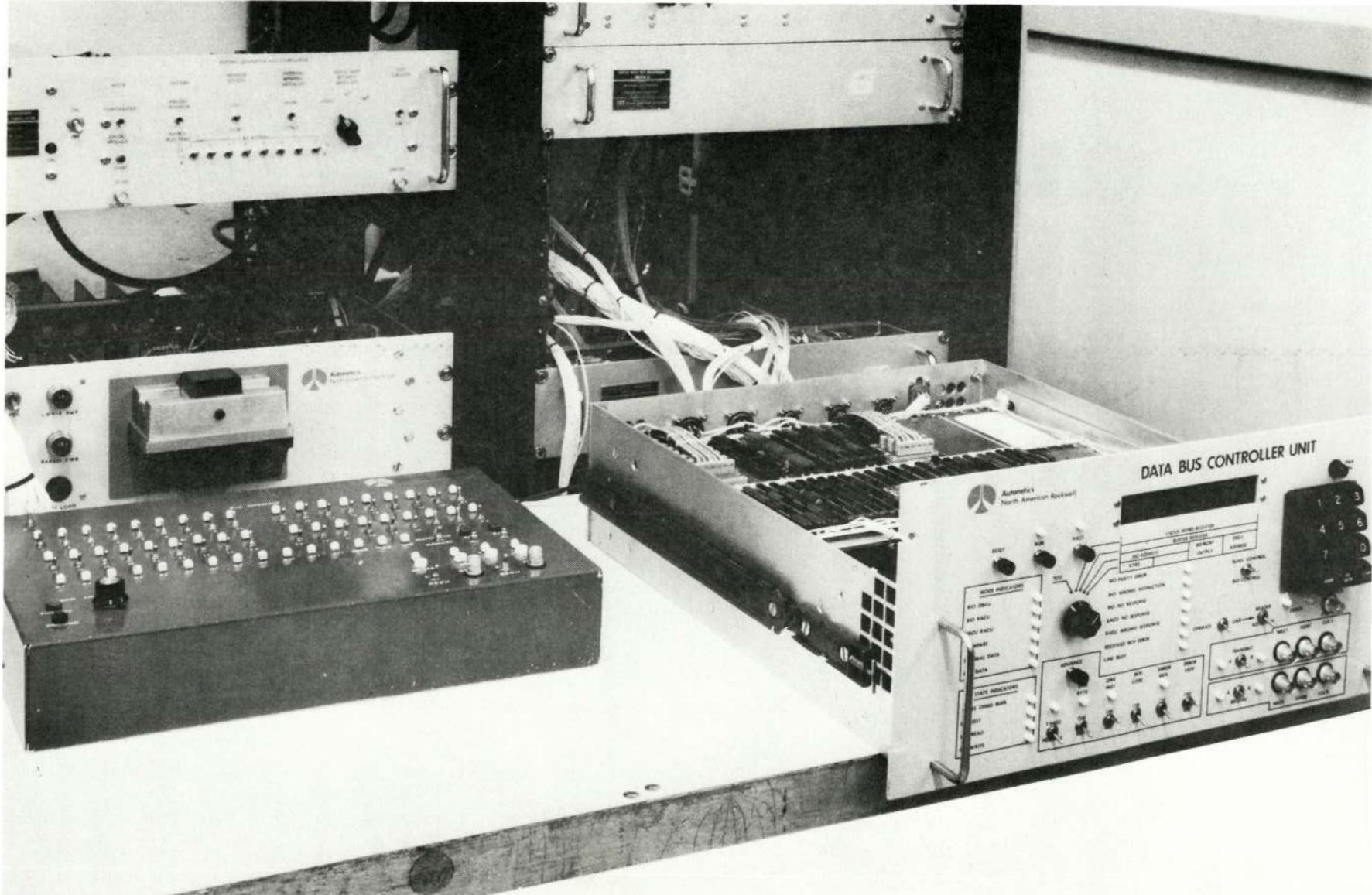


Figure A-6. Support Test Equipment/Load Box

A-9

SD 72-SA-0114-3

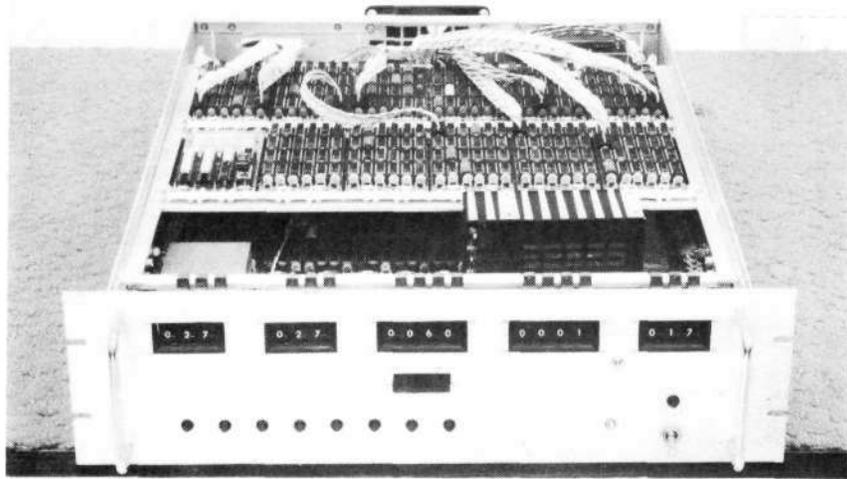


Figure A-7. Remote Acquisition/Control Unit

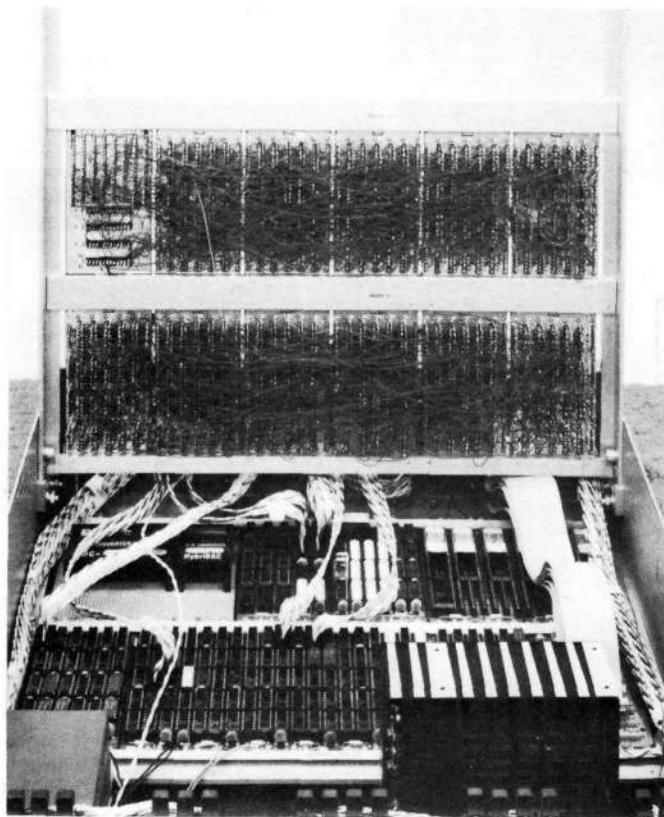


Figure A-8. RACU Interior