ANSI/VITA 6.1-199x
Draft 5.1
October 25, 1996

Extensions to ANSI/VITA 6-1994 Signal Computing System Architecture

A Specification Compatible with VME64x bus

Secretariat:
VME International Trade Association(VITA)

Abstract

This document contains a set of feature extensions to ANSI/VITA 6-1994 Signal Computing System Architecture(SCSA) which expand the switching capacity of the TDM subbus called the SCbus and provide for redundant buses or bus elements. This specification is built upon the VITA 1.1-199x VME64 Extensions document.

DO NOT CLAIM CONFORMANCE TO THIS DRAFT DOCUMENT

Forward (This forward is not part of the ANSI/VITA Standard)

This standard was processed and submitted to the VME-SCSA Technical Review Committee("the committee") of the VITA Standards Organization(VSO). The VSO is the technical standards body of the VMEbus International Trade Association(VITA) of Scottsdale, AZ, USA. The committee unanimously approved this standard by formal ballot. At the time of the approval of this standard the committee had the following members:

Ray Alderman, acting Chairman of the VSO and Executive Director of the VMEbus International Trade Association(VITA)

John Rynearson, Technical Director of VITA.

Louis Francz, Moderator, Dialogic Corp, Parsippany, NJ

Organizations Represented:

AT&T, Bill MacKillop

Airnet Communications, Terry Bauer

Ariel Corp, John Lynch

Dawn VME, Tad Kubic

Electronic Solutions, Frank Hom

FORCE COMPUTERS, Wayne Fischer, Istvan Vadasz

Heurikon Corp, Dennis Terry

MITRE Corp, Robert McKee

Nanotek, Joe George

Nortel Canada, Bruce Wallace.

Naval Surface Warfare Center Crane, Steve Cecil

Object Technology, Inc, Kim Clohessy

Schroff, Inc, Michael Thompson

Vero Corp, Martin Blake

VITA, John Rynearson

The SCSA architecture was originally developed and proposed in 1992 by Dialogic Corporation, Parsippany, NJ. to provide an industry standard by which to promote general industry involvement in the IBM PC compatible(ISA platform) call and voice processing environment.

The emerging VME market for high functionality voice, data and media processing products for connection to the public telephone network provided the ideal environment into which to introduce SCSA. ANSI/VITA 6-1994 was the result of 2 years' effort of a VITA workgroup culminating in two ANSI canvass ballots. The standard was registered by ANSI on July 24, 1995.

The availability of additional P2 pins due to the 160 pin, 5 row backplane connectors referenced in *VITA 1.1-199x VME64 Extensions* draft standard made this SCSA Extensions document possible. This specification covers additional hardware specific elements of the VME-SCSA TDM P2 subbus.

CONTENTS

List of Tables

List of Figures

CHAPTER 1

1. Introduction to Signal Computing System Architecture(SCSA)

1.1. Definitions

1.1.1. ASIC: Application Specific Integrated Circuit. In this context, an SCbus ASIC refers to a gate array or standard cell device that provides SCbus access.

1.1.2. Active Control Set: The SCbus uses several control signals to perform it's functions.(see section 1.5.4) For the purposes of this document the term "active control set" refers to either the "primary" set of SCbus signals: **SCLK**, **SCLKx2***, **FSYNC***, **CLKFAIL**, **MC** and, optionally, **SREF8K**, or the "alternate" control set of signals: **SCLKA**, **SCLKx2*A**, **FSYNCA***, **CLKFAILA**, **MCA** and, optionally, **SREF8KA**.

1.1.3. Bit Integrity: The ability of a digital transport mechanism to provide a bit for bit replica at it's receiving end the information presented to it's transmitting end.

1.1.4. Board: In this context, a VME hardware module that controls it's own physical and logical interface to the SCbus. It may also refer to a module or card.

1.1.5. Bus agent: In this context, a VME board or daughterboard which has a presence on the SCbus. A VME module that contains more than one SCbus agent or ASIC needs to observe module loading rules specified in ref. #1, sec 5.

1.1.6. Channel: A half or full duplex transmission path on the SCbus data bus or message bus that transmits data between internal system ports. It may also be used to reference to a telephony talk circuit within a digital public switched network trunk.

1.1.7. Clock irregularity: A condition of the SCbus clocking system deemed unacceptable by the designers of the SCbus clock master modules or the VME application system. Typical conditions are loss of **SCLK**, gross clock frequency error, loss of or incorrect framing by **FSYNC*** signal, excessive "slips" at the network interface(loss of network synchronization), etc.

1.1.8. Clock master: A system module that is currently driving the system clocks and sync pulse to the SCbus.

1.1.9. Clock slave: A system module that can receive and synchronize to the SCbus timing signals and exchange data with the SCbus.

1.1.10. Clock source: A system module that is capable of deriving the required system clocks either from an external source or internal generator.

1.1.11. Frame, data: Timeslots are grouped together into frames for synchronization purposes. The number of timeslots in each frame depends on the SCbus clock rate. However each frame has a fixed period of 125 ms. Frames are delineated by the timing signal **FSYNC***.

1.1.12. Frame, message: Control and signaling data transmitted on the SCbus or SCmessage bus is encapsulated in a data link layer frame. The form of a SCmessage bus frame is fully compliant with ISO-HDLC UI (Unnumbered Information) Frame specifications.

1.1.13. Frame "slip": A system interface condition whereby a difference in clock rate across a network interface that causes either an excess or deficiency of bits to accumulate in an "elastic" buffer until an entire frame must be deleted or added("controlled slip") in order to maintain synchronization.

1.1.14. Functional Module: A collection of electronic circuitry that resides on the VME bus and or SCSA bus and works with other system elements to perform a task.

1.1.15. HDLC: High-level Data Link Control. ISO standard #3309, bit oriented, data link layer protocol used for the transmission of data and messages on the message bus.

1.1.16. Host CPU: The VMEbus CPU or monarch that contains the SCSA system wide switching and control software and drivers.

1.1.17. Hyperchannel: A data path on the SCbus made up of more than one timeslot(N x 64 kbps). By bundling timeslots into a hyperchannel data path, a time-coherent channel bandwidth greater than 64 Kbps can be realized.

1.1.18. ITU-T: The International Telecommunications Union, telephony sub group. The organization assumed responsibility for establishing guidelines for international telecommunications technical standards from the previous CCITT(International Telegraph and Telephone Consultative Committee).

1.1.19. Idle: A state of the SCbus or message bus where no information is being transmitted and the bus line is passively pulled high.

1.1.20. MVIP-90: TDM bus standard developed by the Global Organization for Multi-Vendor Integration Protocol(GO-MVIP).

1.1.21. Message Bus(SCmessage bus): A collision-sense, multiple access and HDLC framed inter-module communication link implemented on a single wire bus and referenced to the SCbus clock signal **SCLK** or **SCLKA**.

1.1.22. Module: In the context of this document, an SCbus functional component. It may correspond with a "VME board" or a daughter assembly such as a PCI mezzanine module (PMC) defined by IEEE 1386.

1.1.23. Network: In the context of this document, the public switched telephone network providing switched analog or digital access with dial-up station to station addressing and audio, data or video bearer channel services.

1.1.24. Network module: A system component carrying electronics for interfacing a voice or signal processing system to the public telephone network. It fulfills the electrical and regulatory requirements for that connection and provides for the routing of SCbus timeslots to and from the PSTN. It is capable of providing network synchronized timing (clocks) to the system.

1.1.25. Node: An independent SCSA system unit or chassis in a distributed processing SCSA network, consisting of one or more resource and/or network boards.

1.1.26. Octet: Per ITU-T, a digital, binary, eight-bit data item or quantity. In this context, an octet is equivalent to a byte of data.

1.1.27. PLL: Phase Locked Loop. An electronic circuit that locks the frequency of a locally generated clock oscillator to an external reference clock signal. The PLL also attenuates phase jitter from the reference signal.

1.1.28. PCM: Pulse Code Modulation. In the context of this document, digitally encoded and volume companded speech or audio signal samples.

1.1.29. PSTN: Public Switched Telephone Network.

1.1.30. Port: In the context of this document, a port is an interface point where an external communication channel is terminated and information is exchanged.

1.1.31. Resource module: A system component carrying electronics for signal processing, for example voice recognition or text-to-speech processing. It provides a function or service to the system. It is also referenced as a server or function server.

1.1.32. SAPI: Service Access Point Identifier. A network node address that allows the routing of messages between physical SCSA chassis' in one network.

1.1.33. SC2000, SC4000: ASICs of VLSI Technology, Inc. that provide access to the SCSA TDM signal bus which also buffer signals to the SCbus intrasystem messaging bus.

1.1.34. SCSA: Signal Computing System Architecture. A generalized, open-standard architecture describing the components and standard interfaces for a generalized, computer telephony signal processing system. Embodied in standard ANSI/VITA 6-1994

1.1.35. SCSA compatible product: A generic term applied to hardware and/or software products that comply with the appropriate parts of the SCSA model, functionality and interface definitions.

1.1.36. SCbus: The standard bus for intra-module communication(board-to-board inside a box). The SCbus features a hybrid bus architecture consisting of a serial message bus for control and signaling, and a 16 or 32 wire, bi-directional, bit-serial, TDM data bus.

1.1.37. SCbus ASIC: A custom IC, such as the VLSI Technology SC4000, that is specially designed for to provide SCbus access.

1.1.38. SCSA product: A product that complies with the defined SCSA standards for data and signaling transfer. Device specific software will complete integration into the VME-SCSA platform.

1.1.39. SD bus: The generic term applied to the 16 or 32 isochronous, TDM SCbus, bit-serial data signals or streams.

1.1.40. Standby clock module: A module that is prepared to be SCbus clock master when the current clock module stops driving the bus clocks and **FSYNC*** and then releases the **CLKFAIL** signal.

1.1.41. Stream: For the purpose of this document one of the 16 or 32 serial data lines making up the SCbus TDM data bus.

1.1.42. System administrator: A term used in this document to represent a system functional component that manages events, and resources in a physical VME system.

1.1.43. TDM: Time Division Multiplexed signal stream, in this context, comprised of from 1 to *n* physical SCbus 8 bit serial timeslots forming a frame and having a frame rate of 8 kHz.

1.1.44. Timeslot: The smallest switchable data unit on the SCbus. A timeslot consists of 8 consecutive serial bits of data. One timeslot is equivalent to a PCM voice or data circuit with 64 Kbps bandwidth.

1.1.45. VME64 compliant: Modules that conform to all mandatory aspects of ANSI/VITA 1-1994 VME64 standard.

1.1.46. **VME64x**: The abbreviated form of VITA 1.1 VME64 Extensions.

1.2. Application System Objectives

The systems that are integrated using the functional elements described herein and the parent specification along with other VME system components are used for, but not limited to the input, output, switching and processing of speech, data, video and other signals commonly found in public and private telephone networks. The Processing and switching capability provided by these systems is incorporated by system developers into various services that are either made available or offered for a charge to the public at large or to subscribing groups of users as part of value-added communication facilities. In the main, these SCSA based systems and services mediate between pairs of human callers, between client computers or between humans and computers.

1.3. SCSA Extensions Objectives

The extensions contained in this document provide various forward and backward compatible options that provide both added bearer capacity and higher availability. Added data signals provide a 100% increase in switching and media handling capacity which will become important as video services are implemented on SCSA based systems. The addition of redundant control and clocking signals provides a very high degree of system resilience against faults occurring on the SCbus.

The software interface definition allows SCSA compliant products to provide common application programming commands.

1.4. Signal Computing System Architectural Model Extensions

The SCSA Extensions contained in this document do not add any additional architectural features. Rather, these are extensions are implementation options that increase the size range and type of media processing systems that can be built using SCSA compliant modules and systems.

1.5. Introduction to the VME-SCSA Extensions Specification

1.5.1. Specification Objectives

This specification is intended to provide the electrical, logical and mechanical information needed to implement VME-SCSA compatible functional modules with expanded system capability. Specifically, the number of SCbus data(SD) lines has been increased, duplicate control signals have been defined and P2 pin definitions have been added. Extensive fault control procedures are included to provide predictable system response to failures. These optional features all serve to improve a system's tolerance to faults occurring in modules on the SCbus. The additional P2 pin assignments for the alternate SCbus signals are defined in ref. #7 as user I/O in pin rows *z* and *d*. See Table 5 for additional details.

1.5.2. Precedence of Documents

This document applies only to modules designed for the extended capabilities described herein. This document does not redefine the requirements stated in ANSI/VITA 6-1994. However, where evident, and only for modules compliant with this standard the requirements contained herein supersede conflicting requirements of the parent document.

1.5.3. Interface Element Description

These VME-SCSA extensions must be used with the primary VME-SCSA physical layer buses described in ref. #1.

The functional portions of this standard electrically and logically describe: (a)the additions to the SCSA subbuses, (b) the permissible modes for operating the resultant extended buses. (c) how the additional signals are to be used. This standard also defines additional roles for SCbus agents in order to obtain more operating flexibility at the system level. The rules which govern additional bus operations are detailed in text and diagrams.

1.5.4. SCbus Operating Modes

In this document compliance is permitted in one of two broad categories. Modules that comply with ref. #1 and use just the additional 16 SD lines and P2 pinout described herein without any of the other enhancements operate in **S32** mode and may be called Extended SCSA compliant. Modules compliant with ref. #1 and with all rules of any or all of the **R16**, **R32** or **DI16** modes described herein may be called Redundant SCSA compliant. Since the latter term does not fully describe an operating mode, the applicable mode identifier(s), **R16**, **R32** or **DI16** should also be included in the product marketing specification.

The additional SCbus extension signals must be used with the **ANSI/VITA 6-1994 SCSA** bus signals in one of the following modes.: Except as noted, these are discrete operating modes and need not include other operating modes.

S16: Simplex 16 bit SCbus mode. This mode uses only the SCbus signals specified in **ANSI/VITA-6-1994**. This mode reference is included as a means to identify ANSI standard SCbus mode.

S32: Simplex 32 bit SCbus mode. Uses only the primary SCbus clocking and framing signals, **SCLK, FSYNC*, SCLKx2*** with the addition of 16 additional SD lines: **SD_[0-15]A**. The alternate bus control signals **SCLKA, SCLKx2A*, FSYNCA*, SREF8KA, PRIBUS** or **MCA** are not used in this mode. By definition, **S32** compliant modules are also **S16** mode compatible.

R16: Redundant 16 bit SCbus mode. This mode does not use the additional 16 data lines, **SD_[0-15]A**. This mode uses the alternate clocking and framing signals to allow SCbus bearer transport to continue on the primary data signals, **SD_[0:15]**, after a non-clearable fault occurs on one of the primary bus control or synchronization signals. If so equipped, the alternate message bus signal, MCA and synchronization signal, **SREF8KA**, is used. By definition, **R16** mode compliant modules are also **S16** mode compatible.

R32: Redundant 32 bit SCbus mode. This mode uses the additional 16 data lines, **SD_[0-15]A**, renamed as higher order data lines(i.e: **SD_[16:31]**, respectively). This mode uses the alternate clocking and framing signals to allow 32 bit SCbus operation to continue after a non-clearable fault occurs on one of the primary bus control or synchronization signals. By definition, **R32** mode modules are also **S16** and **R16** mode compatible.

DI16: Dual, independent 16 bit mode: This mode defines two completely separate 16 bit data buses, each with it's own clocking, framing, message bus and network synchronization bus. This mode provides a complete and isolated SCbus using **SCLKA, FSYNCA*, SCLKx2A*** and **SREF8KA** for control and the alternate data lines, **SD_[0-15]A**. The alternate **MCA** and **SREF8KA** signals are optional. By definition, **DI16** mode modules are also **S16** compatible on the primary SCbus signals. Signal **PRIBUS** is not used and is undefined in this mode.

1.5.5. Signal definitions

The following is a description of the additional "alternate" SCbus signals added by this standard:
Note: Signal names with an A suffix(see table x) indicate the alternate signals.

SCLKA *System clock, alternate*, driven by the current clock master. The clock frequency is selectable but must be the same as **SCLK**. The positive going edge of **SCLKA** indicates the beginning of the bit cell. This signal must coincide closely with the primary clock signal **SCLK**.

SCLKx2A* *System clock times two, alternate*, driven by the current clock master. This clock frequency is exactly twice that of **SCLKA**. It's edges define the transitions of the **SCLKA** and **FSYNCA*** signals on the clock master card. Note: this signal is only used with 2.048 MHz SCbus speed.

FSYNCA* *Frame sync, alternate*, driven by the current clock master. It is used to synchronize all SCSA system board timeslot counters. **FSYNCA*** occurs just prior to the first rising edge of **SCLK** at the start of a frame.

SREF8KA *Secondary clock reference, alternate*. This optional 8 kHz, 1544 kHz or 2048 kHz digital signal is driven by the SCbus synchronization source which provides backup synchronization to the SCbus. The only format restriction on the **SREF8KA** signal is a minimum high or low time of 100 nanoseconds. The expanded reference signal frequencies also apply to signal **SREF8K** when used under this specification.

This signal is intended to be used by the current SCbus clock master to provide an alternate network synchronization source for SCbus timing in the event of a failure of the primary **SREF8K** signal.

SD_[0:15]A *Serial Data, alternate*. These lines can be driven by any Extended SCbus module in the physical VME system. The timing references specified in ref. #1 that apply between **SCLK**(and **SCLKx2***) and **SD_[0:15]** also apply between **SCLKA**(and **SCLKx2A***) and these signals. In **S32** and **R32** bus modes these signals are renamed **SD_[16:31]**, resp. In **DI16** bus mode these lines must only be referenced to the **SCLKA**(and **SCLKx2A***) clock signals.

CLKFAILA *Clock failure, alternate*, driven by the Extended SCbus clock master module. During normal operation, it is driven low by the current clock master module to indicate that the alternate clock and frame signals are in normal operation. If all standby synchronization sources fail or if any of **SCLKA**, **FSYNCA***, **SCLKx2A***(in 2.048 MHz bus mode) drivers fail, the current clock master module detects this condition and releases the **CLKFAILA** signal which is then passively pulled to the high state. This indicates that a failure of the current clock master in the alternate SCbus clocking or framing system has occurred.

PRIBUS *Primary bus control signals active*. This open collector signal is driven by the primary or standby clock master. This signal, when true, indicates that extended SCbus slave cards are using the primary bus control signals **SCLK**, **SCLKx2***, **FSYNC***, **CLKFAIL**, and , optionally **SREF8K**. The current clock master or standby master card drives this signal line low to cause all redundant SCbus slave cards to synchronously stop using the primary bus control signals and to begin using the alternate bus control signals.

MCA (optional) *Message bus, alternate*. This open collector, bit serial bus is shared by all equipped SCbus modules for alternate intermodule communications. It is edge synchronous with the alternate SCbus **SCLKA** but runs at a 2.048 MHz rate regardless of the **SCLKA** rate. This signal is terminated on each SCSA compatible module in the system whether a user of the message bus or not.

1.5.6. Interface covered by this standard.

This standard adds additional functionality to the TDM data bus and serial message bus at the system interface illustrated by the dotted line in the generalized system block diagram of Figure 1. This figure depicts typical SCSA use in two typical and compliant system components. The VMEbus is not shown and may not exist between modules in all systems if control takes place over the SCSA message bus.

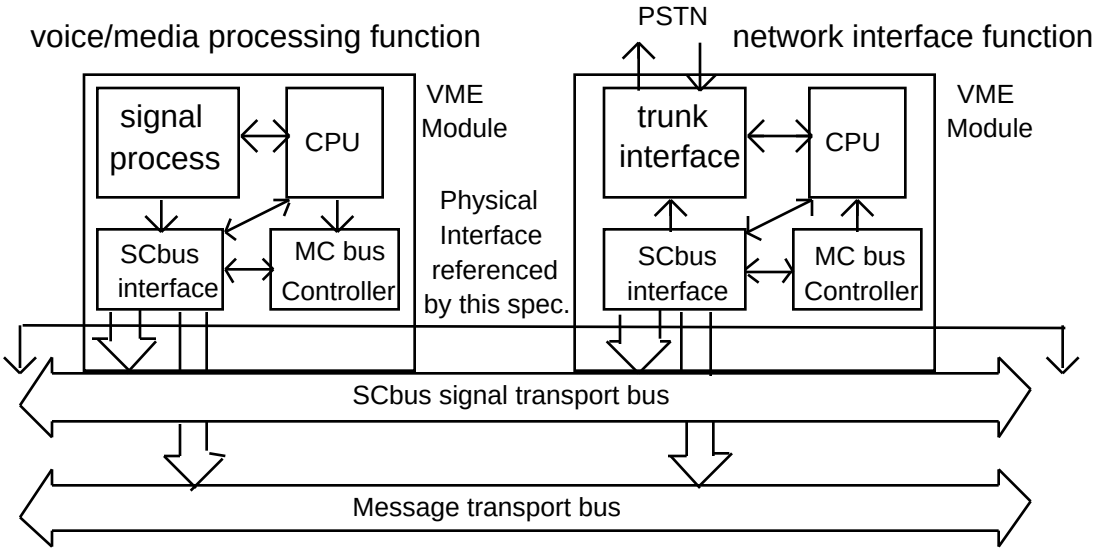


Figure 1 - Interfaces Extended by this Standard

1.6. Specification Terminology

To avoid confusion and to make very clear what the requirements for compliance are, many of the paragraphs in this document are labeled with keywords that indicate the type of information they contain. The keywords are listed below.

RULE:

RECOMMENDATION:

SUGGESTION:

PERMISSION:

OBSERVATION:

Any text not labeled with one of these keywords describes the VME-SCSA structure or operation. It is written in either a descriptive or narrative style. These keywords are used as follows:

RULE chapter.number:

Rules form the basic framework of the VME-SCSA specification. They are sometimes expressed in text form and sometimes in the form of figures, tables or drawings. All SCbus or SCmessage bus rules SHALL be followed to ensure compatibility between compliant designs. Rules are characterized by an imperative style. The upper case words SHALL and SHALL NOT are reserved exclusively for stating rules in this document and are not used for any other purpose.

RECOMMENDATION chapter.number:

Wherever a recommendation appears designers would be wise to take the advice given. Doing otherwise may result in some awkward problems or poor performance. While SCSA has been designed to support high performance systems it is possible to design a system that complies with all the all the rules but has performance or reliability problems.

SUGGESTION chapter.number:

In this specification a suggestion contains advice which is helpful but not vital. The reader is urged to consider the advice given before discarding it. Some suggestions have to do with designing boards that can be easily configured or that make the job of system debugging easier.

PERMISSION chapter.number:

In some cases an SCSA rule does not specifically prohibit a certain design approach but the reader might be left wondering whether that approach might violate the spirit of the rule, or whether it might lead to some subtle problem. Permissions reassure the reader that a certain approach is acceptable and will cause no problems. The upper case word MAY is reserved exclusively for stating permissions in this document and is not used for any other purpose.

OBSERVATION chapter.number

Observations do not offer any specific advice. They usually follow naturally from what has just been discussed. They spell out the implications of certain SCSA rules and bring attention to things that might otherwise be overlooked. They also give the rationale behind certain rules so that the reader understands why the rule must be followed.

1.7. Protocol Overview

1.7.1. Data bus

The primary and alternate SD data buses use a physical layer protocol with bit cells defined by a single edge of **SCLK** or **SCLKA**, resp. There is no source or destination addressing beyond the data line(SD) number and timeslot location which are not module specific. The alternate SD bus framing and clocking is identical to that specified in ref. #1, section 2.1.

For the purpose of this document, no changes are made to the framing format or clock rate options specified in ref. #1, section 2.1.

1.7.2. Message bus

The primary and alternate message buses use an identical, multi-layer packetized protocol whose data link layer is compatible with ISO #3309 (HDLC) (see ref. #1, chapter 3). See section 3 for the network layer definition.

CHAPTER 2

2. Data Transport

2.1. Introduction

SCSA data transport is provided by a synchronous, bit-serial, TDM bus operating at either 2.048 MHz, 4.096 MHz, or 8.192 MHz per ref. #1, section 2. It is defined by one or two clocks and one frame sync pulse. It uses 16 bit-serial data signals **SD_[0:15]**, and 1 clock master control signal, **CLKFAIL**. See ref. #1, section 2. for additional details).

This standard adds an additional 16 data lines, **SD_[0:15]A**, which use the same clock and frame signals as in ref. #1. The net result is an optional doubling of bearer capacity at any bus speed.

2.2. Extended SCbus modes

The basic SCbus plus the extended SC data bus signals, under this standard, can be operated in four compatible modes in addition to the original SCbus mode. For reference, the original SCbus is described as *simplex controls* with 16 SD lines or **S16** mode):

2.2.1. Simplex 32 bit (S32) mode

This mode uses all 32 SD lines to provide up to 4,096 SCbus timeslots as defined in Table 10 but only uses the primary SCbus control signals: **SCLK**, **SCLKx2***, **FSYNC***, **CLKFAIL**, **SREF8K** and **MC**. The alternate SCbus control signals(with **A** suffix) and signal **PRIBUS** are not used in this mode.

RULE 2.1: In both 32 bit SCbus modes, **S32** or **R32**, the alternate SD lines **SD_[0:15]A** SHALL be redesignated as **SD_[16:31]**, respectively.

2.2.2. Redundant 16 bit (R16) mode

This mode uses only the primary **SD_[0:15]** data signals to provide the same 2,048 timeslots as the basic SCbus in **S16** mode. However, this mode uses the primary and alternate SCbus control signals to provide clock signal redundancy. The alternate SCbus control signals use the same signal names as the primary signals with the addition of an A suffix. The alternate set of SCbus control signals provide backup bus control in the event of an unrecoverable failure of the primary control signals. SCbus signal **PRIBUS**, when high, indicates that the primary bus control signals are active.

2.2.3. Redundant 32 bit (R32) mode

This mode uses the **SD_[0:15]** data signals plus the **SD_[0:15]A** lines which are redesignated as **SD_[16:31],** respectively, to provide up to 4,096 SCbus timeslots as defined in Table 1. This mode also uses the primary and alternate SCbus control signals to provide clocking redundancy as in **R16** mode, above. SCbus signal **PRIBUS**, when high, indicates that the primary bus control signals are active.

2.2.4. Dual, independent 16 bit (DI16) mode

This mode allows the primary and alternate SCbus signal sets to operate as two independent **S16** buses. using **SD_[0:15]** and the primary SCbus control signals as bus #1 and using **SD_[0:15]A** and the alternate SCbus control signals as bus #2.

The switching and recovery philosophy when using dual independent buses is application specific and beyond the scope of this standard. SCbus signal **PRIBUS** is not defined for this SCbus mode.

2.2.5. Additional SCbus capacity

Table 1 shows the specified SCbus bus speeds in Megabits/second and the resultant SD data line format with the number of time-slots per frame defined on each data wire times the number of data wires. Total SCbus bearer capacity is identical for all extended operating modes(**S32**, **R32** and **DI16**) except for **R16** mode which provides no additional capacity over the basic SCbus **S16** mode.

Table 1 - SCbus Speed and Capacity

Bus speed	2.048M Hz	4.096M Hz	8.192M Hz
# of Timeslots per SD	32	64	128
# of Timeslots per SCbus frame	32x32 = 1024 timeslots	64x32 = 2048 timeslots	128x32 = 4096 timeslots
Total bandwidth	**65M bps**	**131M bps**	**262M bps**

A timeslot is the smallest switching unit on the bus and accounts for 64K bps of bandwidth on any wire of the SD bus.

RULE 2.2:: In all modes supported by this standard the grouping of multiple 8 bit timeslots to form a wideband (hyper-) channel requires the handling of this composite wideband channel with the same delay. That is, all grouped timeslots SHALL maintain their original position in a frame of reference and SHALL NOT be re-sequenced or delayed to subsequent frames. This SD bus mode SHALL be allowed and supported by all modules compliant to this standard. Timeslot sequence SHALL be considered maintained when a set of timeslots within a frame retain their same relative positions within the frame from source module to destination module.

2.3. Data Transfer bus.

2.3.1. Bus format

Depending on the SCbus clock rate, each SD serial stream is divided into 256, 512 or 1024 bit groups called a frame(see ref. #1, section 2.2.1). Each frame is exactly 125 microseconds long. Each frame is then further divided, starting at the frame boundary, into a fixed number of 8 bit sub-frames called timeslots. Consequently, each frame consists of 32, 64, or 128 timeslots(per SD line) corresponding to the respective clock rates of 2.048, 4.096 or 8.192 MHz.

2.3.2. Frame alignment

The relationship between data, clock, and frame pulse is also illustrated in diagram of Figure 3. See ref. #1, section 2.4 for detailed SCbus frame alignment information.

2.3.3. Voice/data Transfers

Voice/data transfer on the SD bus is accomplished by assigning one or more timeslot IDs (an **SD_[*n*]** bus stream number plus a time-slot number) to the sender and receiver. At the selected time-slot, the sender drives the bus and the receiver clocks-in the data bits. There will be no handshake or confirmation taking place between the sender and the receiver(s). The operation is solely based on the assumption that all boards in the system use the **SCLK** and have their timeslot counters synchronized to the frame sync pulse, **FSYNC***.

2.4. Synchronization

There are three types of SCbus synchronization: bit, timeslot, and frame. Bit synchronization establishes the proper timing to send and/or receive data bits. Timeslot and frame synchronization ensure that each bus agent is synchronized with the other bus agents within the system. Since the operation of the bus does not require any handshake between a sender and the receiver(s), synchronization is fundamental to system operation (see ref. #1, section 2.3.2 and 2.4. for details).

To allow higher availability systems to be implemented with SCSA, this standard adds an optional redundant set of SCbus clocking and control signals. These redundant signals are designated with an A suffix on the signal name(see sect. 1.5.4 and Table 5).

The following applies to all compliant bus agents or modules used in an extended SCbus system:

RULE 2.4: The SCbus control signals SHALL be used together as a primary or an alternate set. I.E: **SCLKA**, **FSYNCA***, **SCLKx2A***, **CLKFAILA**, **SREF8KA** and **MCA**.

RULE 2.5: In systems using a redundant bus mode(**R16** or **R32**) when SCbus signal **PRIBUS** is TRUE(high) the primary SCbus control signal set SHALL be used to provide timing and sync for the active **SD_[*n*](A)** lines. When SCbus signal **PRIBUS** is FALSE(low) the alternate SCbus control signal set SHALL be used to provide timing and sync for the active **SD_[n](A)** lines.

RULE 2.6: All SCbus data SHALL be sent to the SCbus SD lines on the rising edge of the **SCLK** OR **SCLKA**.

RULE 2.7: All SCbus data SHALL be sampled on the SD lines at or after the falling edge of **SCLK** OR **SCLKA**.

RULE 2.8: Timeslot counters SHALL use the rising edge of **SCLK** OR **SCLKA**, to increment or reset the count.

RULE 2.9: Timeslot counters SHALL be reset to zero at the rising edge of **SCLK** OR **SCLKA** coincident with logic TRUE(TTL low) **FSYNC*** OR **FSYNCA*** frame sync pulse, respectively.

2.4.1. Redundant Bus signals

This standard defines an alternate set of data bus control and synchronization signals that are used, as a set, upon loss of any primary bus control or synchronization signals. These signals are **SCLKA**, **SCLKx2A***, **FSYNCA***, **MCA**, **CLKFAILA** and optionally, **SREF8KA**. These signals are used in **R16**, **R32** and **DI16** extended SCbus modes.

RULE 2.10: All extended SCbus master modules that support either **R16** or **R32** modes, when acting as SCbus clock master, SHALL simultaneously drive the primary and alternate SCbus control signal sets. These signals (and alternates) are: **SCLK(A)**, **SCLKx2(A)***, **FSYNC(A)***, **CLKFAIL(A)** and optionally, **SREF8K(A)** which is driven by the standby clocking or reference master.

RULE 2.11: The timing relationship and tolerances between the alternate SCbus control signals **SCLKA**, **SCLKx2A*** and **FSYNCA*** and all active SCbus **SD_[*n*]** signals on SCSA Extended and redundancy equipped modules SHALL also comply to ref. #1, section 2.6.

2.4.2. Redundant Bus Control signal

Open collector signal **PRIBUS** is essential to performing synchronized bus fallback and is considered part of the redundant bus control signal set. Signal **PRIBUS** provides the means to indicate to all redundancy equipped SCbus agents in the system that either the primary or the alternate bus control set is currently active.

RULE 2.13: Redundant SCbus agents that support **R16** or **R32** modes SHALL implement the **PRIBUS** signal as defined.

RULE 2.14: The **PRIBUS** signal SHALL be driven only by the primary or standby SCbus master agents using the same setup and hold timing as specified in ref. #1, sec. 4 for the **FSYNC*** signal.

RULE 2.15: The **PRIBUS** signal SHALL be clocked in and acted upon by slave SCbus agents during the rising edge of **SCLK** or **SCLKA** coincident with the **FSYNC*** or **FSYNCA*** signal, as appropriate to the currently active SCbus control signal set. However, **PRIBUS** SHALL only by driven FALSE(low) at the rising edge of **SCLKA** coincident with **FSYNCA*** TRUE(low) and SHALL only be driven TRUE(high impedance) at the rising edge of **SCLK** coincident with **FSYNC*** TRUE(low).

2.4.3. Extended SCbus Agents

In order to be considered compliant with this standard, SCbus agents or modules must support either 32 SD lines, redundant bus control (clocking) signals or both.

RULE 2.16: Extensions compliant SCbus modules SHALL assume one of the following identities with respect to SCbus clocking:

> **(a)** *Extended Clock Slave(**S32**)*: An SCbus agent or module which accesses the primary and alternate set of **SD** lines(32 total) but receives only the primary SCbus timing signals **SCLK**, **SCLKx2*** and **FSYNC*** for use in exchanging timeslot data with the **SD** lines.

(b) *Redundant Clock Slave(**R16, R32**)*: An SCbus agent or module which receives the primary SCbus timing signals **SCLK**, **SCLKx2*** and **FSYNC*** and the alternate bus timing signals, **SCLKA**, **SCLKx2A***, and **FSYNCA*** for use in exchanging timeslot data with the SD lines. A Redundant SCbus Clock slave uses signal **PRIBUS**, when TRUE(high), to select the primary bus control signal set and, when FALSE(low), to select the alternate bus control set.

(c) *Extended Clock Master(**S32**)*: An SCbus agent or module which is capable of assuming SCbus primary or standby mastership by properly driving the SCbus timing signals per ref. #1. Extended SCbus masters SHALL have the following capabilities:

1. Be capable of driving SCbus timing and control signals **SCLK**, **SCLKx2***, **FSYNC*** and **CLKFAIL**.

2. When acting as a primary or standby SCbus master, be able to detect when SCbus clocking or timing has failed and be able to notify the system host.

3. Be able to detect when it's synchronization or SCbus timing drive circuits have failed and be armed to release the **CLKFAIL** signal and to cease driving the SCbus clocks and frame sync.

4. Be able to be armed as an SCbus standby clock master which will assume SCbus mastership if and when the SCbus **CLKFAIL** signal becomes high(true).

5. Be able to supply an accurate internal clock reference source(recommend +/- 32 ppm or better) from which to derive SCbus timing for system boot-up or for non-network connected installations.

6. Be capable of also operating as an SCbus clock slave using the primary bus control set.

(d) *Redundant Clock Master(**R16, R32**)*: An SCbus agent or module which is capable of assuming SCbus primary or standby mastership by properly driving the primary and alternate SCbus timing signals. Redundant SCbus masters SHALL have the following capabilities:

1. When acting as a primary SCbus master be capable of simultaneously driving both sets of SCbus timing and control signals **SCLK**, **SCLKx2***, **FSYNC***, and **CLKFAIL** as well as **SCLKA**, **SCLKx2A***, **FSYNCA*,** and **CLKFAILA**.

2. Be able to detect when any of the primary clocking or timing signals have failed, AND drive bus signal **PRIBUS** low AND be able to notify the system host CPU.

3, When acting as a primary SCbus master AND using the primary bus control signals(**PRIBUS** = TRUE{high}), be able to detect and report to the host CPU when any of it's alternate bus control signals have failed or do not agree with the primary control signals.

4. When acting as a primary SCbus master be able to detect when it's bus control or network synchronization circuits have failed AND THEN release the appropriate **CLKFAIL** signal AND cease driving the SCbus clocks and frame sync AND assume redundant slave agent status AND notify the system host CPU.

5. When acting as a primary SCbus master, be able to detect when it's internal active bus control status indicator does not match the **PRIBUS** signal AND THEN release the appropriate **CLKFAIL** signal AND cease driving the SCbus clocks and frame sync AND assume redundant slave agent status.

6. Be able to be armed as an SCbus standby clock master which will assume primary SCbus mastership when the SCbus **CLKFAIL** or **CLKFAILA** signal becomes TRUE(high), depending on whether signal **PRIBUS** is TRUE(high) or FALSE(low), resp. The new primary master must initially select the primary bus control signals by NOT driving signal **PRIBUS** FALSE(low).

7. Be able to supply an accurate internal clock reference source(recommend +/- 32 ppm or better) from which to derive SCbus timing for system boot-up or for non-network connected installations.

8. Be capable of also operating as a redundant SCbus clock slave using **PRIBUS** to select the primary or alternate bus control set, AND, as in (4.) above, after detection of a failure of it's SCbus sync. circuits, release the **CLKFAIL** signal AND notify the host CPU.

9. When acting as a standby bus master, be able to detect that the primary bus control signal set has failed, AND THEN drive **PRIBUS** = FALSE(low) AND notify the host CPU.

RULE 2.17: If **SREF8K** mode is supported, Extended SCbus clock master modules SHALL provide the following:

1. *SREF8K* *Slave capability*: SCbus master agent modules must synchronize their **SCLK** reference circuit to 8 kHz, (optionally 1544 or 2048 kHz) **SREF8K** signal when enabled under host CPU or internal module control.

2. *SREF8K* *Master capability*: SCbus master agent modules must derive an 8 KHz (see ref. #1, sect. 2.6.1.4)(optionally, 1544 or 2048 kHz) reference signal from their network synchronization circuits or from an internal clock reference and then drive SCbus signal **SREF8K** when enabled under host or internal module control.

3. *SREF8K* *equipped modules:* SCbus agent modules must be able to detect when the **SREF8K** signal contains no clocking transitions and report this condition to the system host.

OBSERVATION 2.1: The clock reference PLL on **SREF8K** equipped SCbus master modules will not be locked if the PLL synchronization source has been switched to the SCbus **SREF8K** signal when it is off-frequency or not present.

RULE 2.18: If **SREF8K** mode is supported, redundant SCbus clock master modules SHALL provide the following:

1. *SREF8K* *Slave capability*: SCbus master modules must synchronize their **SCLK** and **SCLKA** reference circuit(PLL) to 8 kHz(optionally, 1544 or 2048 kHz) **SREF8K** or **SREF8KA** signal depending on the state of signal **PRIBUS**.

2. *SREF8K* *Master capability*: SCbus master modules must selectably derive an 8 KHz (see ref. #1, sect. 2.6.1.4)(optionally, 1544 or 2048 kHz) reference signal from their network synchronization circuits or from an accurate internal clock reference and then drive both SCbus signals **SREF8K** and **SREF8KA**.

3. *SREF8K equipped:* SCbus master modules must be able to detect when the **SREF8K** or **SREF8KA** signal, depending on the state of signal **PRIBUS,** contains no clocking transitions AND must report this condition to the host CPU.

OBSERVATION 2.2: The clock PLL on equipped SCbus master modules will not be locked if the PLL synchronization source has been switched to the SCbus **SREF8K** or **SREF8KA** signal when it is off-frequency or contains no clocking transitions.

RULE 2.19: If **SREF8K** mode is supported, all SCbus clock master modules SHALL comply with the SCbus electrical load(slave) and drive(master) requirements specified in ref #1, table 9 for both the **SREF8K** and **SREF8KA** interface circuits.

2.5. SCLK Jitter and Wander

Extended SCbus clock master modules need to meet some minimum requirements for the quality of the **SCLK** signal. Jitter can be visualized as short duration displacement of **SCLK** edges from their ideal position in a stream of "perfect" clock pulses (see ref. # 8). Wander is low frequency modulation (<10 Hz) of the **SCLK** frequency. A specific amount of jitter or wander on **SCLK** does not directly translate to amounts present at a slaved network interface transmit port due to the customary practice of following the received SCLK with a PLL and jitter attenuator buffer.

Note: The following limits apply when a clock master module is receiving a valid synchronizing signal that is without significant jitter or wander.

2.5.1. SCLK Jitter

RULE 2.20 Extended SCbus clock master modules **SHALL NOT** drive the **SCLK** (or **SCLKA**, if equipped) signals having jitter exceeding the following limits:

2.5.1.1. Allowable SCLK Phase Noise Jitter

Table 2 contains the maximum allowable Jitter components on SCLK per frequency band indicated.

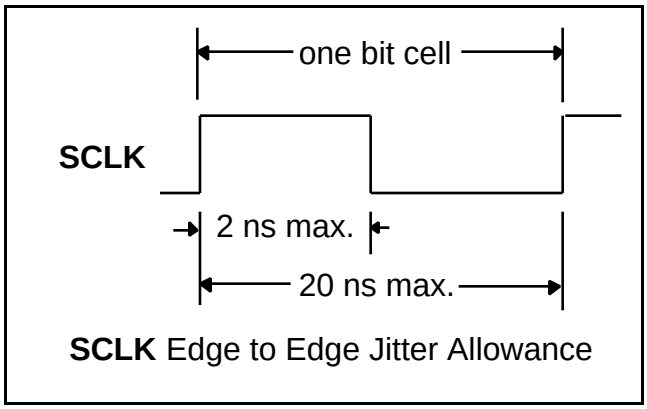
Table 2. Maximum Allowable SCLK Jitter

Jitter Spectrum	Jitter
400 Hz up to 8 kHz	244 ns(p-p)
8 kHz up to 40 kHz	35 ns(p-p)

2.5.1.2. Allowable SCLK Edge to Edge Jitter

Figure 2 illustrates the maximum allowable per bit-cell SCLK edge deviation from nominal.

Figure 2. Per-Bit-Cell Jitter Allowance



SCLK Edge to Edge Jitter Allowance

2.5.2 SCLK Wander

RULE 2.21: Extended SCbus clock master modules SHALL NOT drive the **SCLK**(and **SCLKA**, if equipped) signals with wander exceeding the limits in Table 2.

Table 3. Maximum Allowable SCLK Wander(p-p)

Observation period	SCLK Wander (max.)
Per 25 Hours	18 us(p-p)
Per 1 hour	15 us(p-p)
Per 15 Minutes	5.2 us(p-p)

2.5.2.1 SREF8K Master Timing

The signal format specified in ref. #1, section 1.5.3. for **SREF8K** also applies to extended SCbus compliant modules and signal **SREF8KA**.

2.5.2.2 Synchronizing to SREF8K

RULE 2.22: Extended SCbus clock master modules using **SREF8K** (or **SREF8KA**, if equipped) as network timing reference for the SCbus clock signals, when enabled, SHALL frequency lock to the exact integer multiple of reference frequencies of 8 or 2048 kHz as appropriate to the SCbus speed chosen. For clock master modules equipped to use the 1544 kHz **SREF8K** reference frequency is used, the SCbus clock signal SHALL be frame synchronized with the 1544 kHz reference, i.e. every 193 periods of the 1544 kHz reference signal(one DS1 frame) SHALL exactly correspond with one SCbus frame of 256, 512 or 1024 SCbus clock cycles for the 2.048, 4.096 or 8.192 MHz SCbus clock rates, respectively.

RULE 2.23: Extended SCbus modules that use **SREF8K** (or **SREF8KA**, If equipped) for network timing reference must rely only on the 8 or 2048 kHz signal period and must be duty cycle independent within the restrictions provided by ref. #1, section 1.5.3.

2.5.2.3 SREF8K Generation.

For master clock modules that generate **SREF8K** (and **SREF8KA**) the signal format specified in ref. #1, section 1.5.3 applies.

RULE 2.24: Extensions compliant clock master modules that generate **SREF8K**(and **SREF8KA**) signal SHALL comply with the maximum jitter and wander specified below and in Table 4.

(a). **Maximum Allowable SREF8K Frame to Frame Jitter**: 244 ns(p-p)

(b). SREF8K Wander

Table 4. Maximum Allowable SREF8K Wander

Observation period	SREF8K Wander (max.)
Per 25 Hours	18 us(p-p)
Per 1 hour	15 us(p-p)
Per 15 Minutes	5.2 us(p-p)

Signal **SREF8K** (and **SREF8KA**, if equipped) have no specified timing relation to each other or to any other SCbus signals for the purposes of this standard.

2.6. Bus Fault Recovery (Fallback) Modes

High availability systems need failure recovery processes for any single SCbus related failure. The following paragraphs describe the recommended SCbus fallback procedures for extended SCbus systems

2.6.1. SCbus Clocking fallback

Clock fallback is the controlled process by which a different physical source of SCbus clocking is provided for the SCSA equipped physical backplane. This process is described in ref. #1, sect 2.4. This procedure involves organizing all boards that can drive network synchronized SCbus backplane clocking signals into a clock fallback list. The first level of fallback is accomplished through the **CLKFAIL** hardware mechanism(see ref. #1, sect 2.4.2). Additional layers of fallback are to be implemented by the system software by assigning subsequent standby clock masters. For Extensions equipped SCbus systems operating in **R16** and **R32** modes the clock fallback processes are identical when using the primary or alternate SCbus control signal set.

RULE 2.24: All Extensions compliant, SCbus master modules SHALL provide clock fallback as per ref. #1, section 2.4.2 for both the primary and, if equipped, the alternate bus control set.

The referenced fallback process is invoked by the standby clock master automatically becoming the active SCbus clock master upon the previous clock master's release of the **CLKFAIL** or **CLKFAILA**(if redundant bus equipped) signal, depending on the state of signal **PRIBUS**.

2.6.2. Data Stream fallback

Stream fallback is the process by which, following the identification of the failure of an individual SD line, all currently assigned timeslots on the affected SD line are reassigned to another SD line. System software is responsible for this reassignment. All SCbus compliant systems can use stream fallback for fault recovery.

RULE 2.25: All compliant SCbus modules SHALL implement stream fallback on failure of a timeslot or SD line.

RECOMMENDATION 2.1: SCbus interface designs should support collision detection on the SD lines to allow rapid detection of hardware or software faults.

2.6.3. Bus fallback

Bus fallback is the process by which the standby clock master module and all bus slave modules transition, when either the primary or standby clock master module drives the **PRIBUS** signal FALSE(low), to the alternate bus control signal set: **SCLKA, SCLKx2A***, **FSYNCA***, **CLKFAILA (**and optionally **SREF8KA).** Since the bus clocking and framing signals represent a single point of failure for an SCbus system the alternate bus control set allows recovery from this condition. The transition to the alternate bus control signal set is applicable to SCbus agents or modules operating in **R16** and **R32** modes.

OBSERVATION 2.3: In **DI16** mode the **PRIBUS** signal is not defined. Therefore, transition of SCbus agents to the alternate bus control signals and **SD_[0-15]A** lines must be controlled by software.

2.6.4. Synchronization fallback

Synchronization fallback is the process by which a failed source of network synchronization for the SCbus is replaced. Sync. fallback can be accomplished in two ways. The preferred method of sync. fallback requires separation of the function of deriving network synchronization from that of SCbus mastership and requires the use of SCbus signal **SREF8K**(optional alternate, **SREF8KA**). Section 2.7 details synchronization fallback methods.

Alternatively, synchronization fallback, per ref. #1, section 2.4.2, can be triggered by the current primary SCbus and sync. master module by releasing **CLKFAIL (**and **CLKFAILA)** and allowing the standby SCbus master module to provide both network sync. and SCbus mastership.

RECOMMENDATION 2.2: Extended SCbus systems should provide at least one additional module which can derive network synchronization and drive the **SREF8K (**and **SREF8KA)** signals. The SCbus primary master module should be capable of using the **SREF8K** (or **SREF8KA** signal, depending on the state of signal **PRIBUS**, for redundant modules)to develop network synchronized SCbus clocks and frame sync.

2.6.5. Bus Fallback States

Synchronization and data stream fallback are totally under the control of VME host system software so that the operating state transitions are not defined for the purposes of this standard. The bus clocking and bus master fallback processes are hardware processes with host CPU input. The recommended bus and clock control fallback processes for redundant bus modes can be visualized with a state diagram shown as figure 2. Extended mode(**S32**) clock master modules use the standard SCSA clock fallback process described in ref. #1, section 2.4.2.

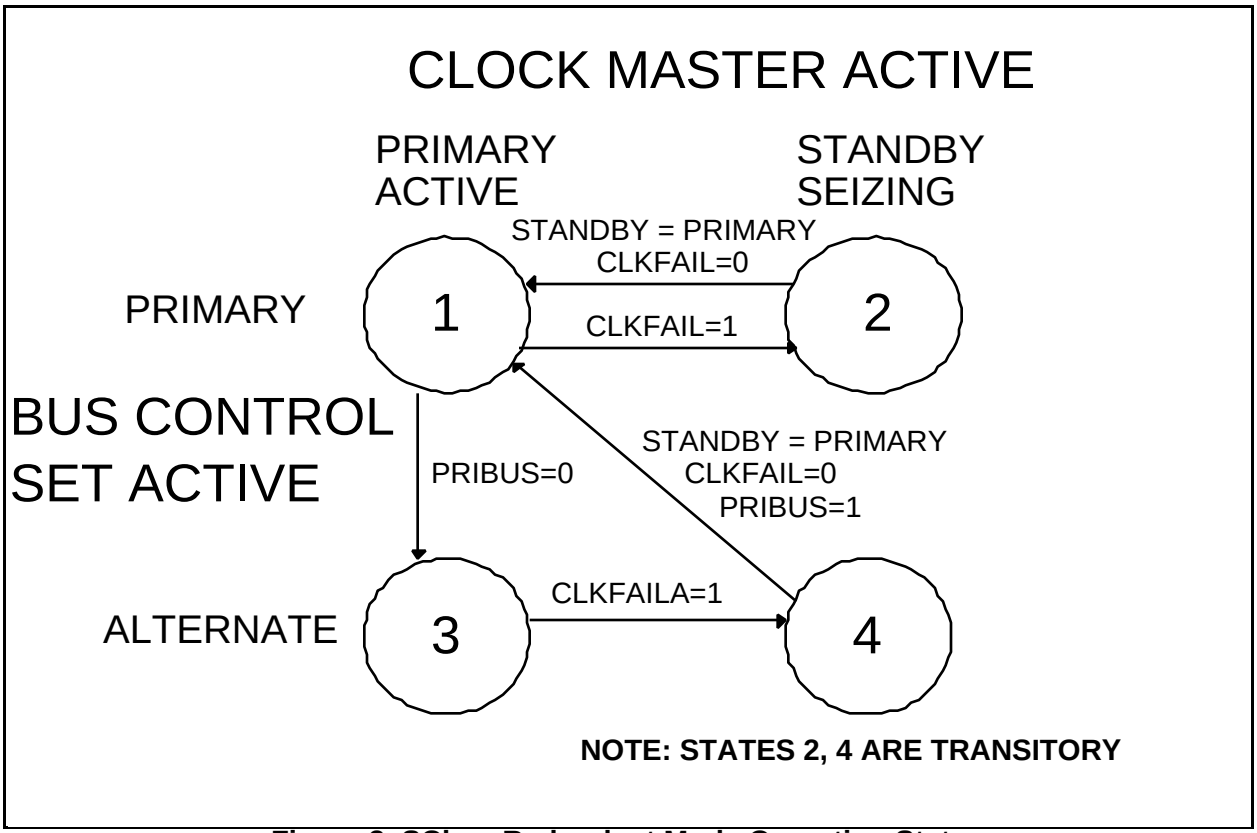


Figure 3. SCbus Redundant Mode Operating States

Figure 3 describes the recommended bus state transitions in recovering from either bus master or bus clocking signal failure. The following text section describes the conditions under which a change in clock master or in active bus control signal set occurs.
Note: The host CPU must be notified of all SCbus state transitions to apply system-wide diagnostic logic that prevents recovery oscillations and also to interface with site maintenance processes.

Initial Condition:

State 1: Initial SCbus operating state. Primary clock master drives the SCbus and the primary bus control set is active(**PRIBUS** = TRUE(high), **CLKFAIL(A)** = FALSE(low)).

Allowable Failure transitions:

State 1 to state 2: This SCbus state transition in figure 2 shows the **CLKFAIL** initiated clock fallback process described in ref. #1, section 2.4.2. The primary clock master recognizes a clocking failure and stops driving the bus controls, releasing **CLKFAIL**, the standby clock master recognizes signal state **CLKFAIL**=TRUE(high) and prepares to become the primary clock master.

State 1 to state 3(master initiated): Since redundant primary clock masters drives both primary and alternate bus control signals this state transition in figure 2 actually occurs at each SCbus redundant slave module. A failure is detected by the primary clock master which initiates recovery by driving **PRIBUS**=FALSE(low). All redundant slaves respond to signal **PRIBUS** = FALSE by switching to the alternate SCbus control signals.

State 1 to state 3(standby initiated): A failure is detected by the standby clock master which initiates recovery by driving **PRIBUS**=FALSE(low). All redundant slaves respond to signal **PRIBUS** = FALSE by switching to the alternate SCbus control signals.

State 3 to state 4: This state transition in figure 2 shows the **CLKFAILA** initiated clock master fallback process on the alternate control signals. If state 1 to state 3 transition was initiated by the standby master, the primary clock master detects a difference between the state of SCbus signal **PRIBUS** and it's internal bus control state status and automatically assumes SCbus slave status, releasing both **CLKFAIL** and **CLKFAILA**. The standby clock master module recognizes signal state **CLKFAILA**=TRUE and prepares to become the primary clock master.

Recovery Transitions:

State 2 to state 1: As per ref. #1, section 2.4.2, this state transition in figure 2 is initiated by the standby master's internal frame sync. signal. The standby master assumes primary mastership by driving **CLKFAIL** FALSE(low). Upon notification, the host CPU assigns a new standby clock master module.

State 4 to State 1: This transition occurs as the abdicated primary clock master, now in slave mode, is replaced by the standby clock master using the alternate bus control set. The standby clock master assumes primary clock mastership by simultaneously driving **CLKFAIL** and **CLKFAILA**=FALSE(low) and also causes all slave modules to resume operation on the primary bus control signals by driving bus signal **PRIBUS** TRUE(high). Upon notification, the host CPU assigns a new standby clock master module.

Note: If the SCbus fault has not been cleared by the removal of the previous primary clock master, the new primary clock master will again detect clocking failure and drive **PRIBUS** FALSE(low), causing all slaves to again transition to the alternate bus control set (SCbus returns to state 3).

2.6.6. Clock fallback List.

As described in section 2.5.5 and in ref. #1, sect. 2.4.2 the SCbus provides the hardware mechanism by which the current SCbus master clock module relinquishes clocking responsibility to the designated standby SCbus master. The preferred method of managing the hierarchy of potential SCbus clock master modules is with a fallback list. Maintained by the VME bus host CPU, the list supplies the next module in order of preference to become the current standby clock master(see ref. #1, section 2.4.2.1.).

2.6.7. System Recovery Performance Requirements

To provide standardized response to network carrier or clock module failure the following are to be implemented at system level:

RULE 2.26: The designated clock master and standby modules SHALL be able to determine the quality of the SCbus clocks, and report to the host when any clocking irregularity persists for longer than 2 frames.

OBSERVATION 2.5: Clocking irregularity is not defined for the purposes of this specification. Sample clock irregularity criteria follow:

-Too few or too many timeslots between frame sync pulses.

-Gross **SCLK** or **SREF8K** frequency error(>50%)

-More than two SCbus-slaved network interface transmitted frame “slips” in a predetermined period(suggest 5 minutes).

RECOMMENDATION 2.4: To minimize disruptions to SCbus slaves transmitting to the network, the designated clock standby module should be able to seize control of the SCbus at the next frame boundary of the SCbus master’s release of the CLKFAIL signal.

2.7. Network Synchronization Fallback

Bus based synchronization is an additional and optional SCbus network synchronization method that uses the **SREF8K** signal, or it’s alternate **SREF8KA,** to facilitate the seamless transfer of network sync. sources(see ref. #1, chapter. 2, RULE 2.7). The **SREF8K** signal can be used in two ways to achieve smooth transfer of network synchronization. Throughout this subsection references to **SREF8K** signal also apply to the **SREF8KA** signal for SCbus modes **R16** and **R32**.

2.7.1. SREF8K based synchronization backup.

RECOMMENDATION 2.5: This is the recommended method of using **SREF8K** to provide sync. backup.

Under bused synchronization backup, the current SCbus clock master uses it’s own network interface, if equipped, to provide primary network sync. to it’s SCbus clock PLL. Another network-connected module is assigned by the host CPU to derive sync. from it’s network interface and to drive the SCbus **SREF8K** signal line. The SCbus standby clock master is a good candidate module for this duty. Upon failure of the network link to the primary clock master module it should automatically switch it’s PLL’s source of network sync. to the **SREF8K** signal. This method results in the shortest sync. recovery time. This method also allows seamless restoration of network synchronization via the original link, when returned to operation.

2.7.2. SREF8K based synchronization

Under bused synchronization, primary network sync. is distributed to the system modules via an 8, 1544 or 2048 kHz periodic signal carried by the SCbus signal line **SREF8K**. The SCbus clock master module uses **SREF8K** to lock a PLL which is used to generate SCbus clocking. (This is in contrast to deriving sync. only from network links resident on SCbus master modules.) Thus, failure of the network sync. source results in replacement of the network sync. module that is driving **SREF8K**, not the SCbus clock master. Due to it’s loop filter, the PLL on the SCbus clock master remains very near network lock for several milliseconds until a new **SREF8K** source module is assigned.

This sync. method allows SCbus clock master modules to be built without any network interface at all (similar in concept to a standalone VME64 central services module (CSM)).

RECOMMENDATION 2.6 Under **SREF8K** based synchronization, the current **SREF8K** drive module should detect loss of network sync and report the loss to the system host which must then assign a new network sync. module to drive **SREF8K**.

OBSERVATION 2.6: To minimize the **SREF8K** source switchover time this source reassignment should be initiated by an interrupt to the host CPU. Excessive switchover time may result in unnecessary network slips.

OBSERVATION 2.7: Switching between non-phase-synchronized **SREF8K** sources may cause the current clock master module to generate transitional SCbus clock timing which exceeds the maximum time interval error (MTIE) allowed for certain classes of network switching equipment. Refer to ANSI T1.101-1994. The clock master modules may need special design considerations to permit this **SREF8K** source switching.

2.8. Initialization

2.8.1. SD bus

RULE 2.27: When the power is initially turned on, all extended SCbus boards SHALL assume SCbus slave mode and disable all their SCbus drivers and redundant master modules SHALL prepare to use the primary bus control signals.

2.8.2. Clocks and Frame Sync Pulse.

When the power is initially applied, all signal lines float "high" due to the bus terminators until a system board is designated as the system clock master, either by a software command or the clock-fallback mechanism. The recommended system clock initialization sequence is listed in ref. #1, Table 3.

OBSERVATION 2.8: The system-wide clock failure detection and fallback functions for primary and alternate buses, if so equipped, may need to initially be turned off to allow the system to be downloaded and stabilized.

2.9. Data Transfer Modules - Basic Description

VME bus modules that connect to the Extended SCbus are expected to maintain synchronization with the appropriate bus control signals and utilize the SD signals and timeslots to receive as well as drive byte oriented data onto the bus.

2.9.1. Typical Operation.

The Extended SCbus is designed to operate in one of three environments.

-In pure VME SCSA systems with terminated PCB backplanes the bus is clocked at 8.192 MHz providing the maximum number of SCbus timeslots.

-In mixed VME-ISA bus chassis or when ribbon cable is used to connect the SCbus across the VME cards, SCLK may be reduced to 4.096 MHz and the ribbon cable is not externally terminated.

-In systems containing MVIP-90 bus compatible modules, SCLK is reduced to 2,048 MHz, SCLK*2 is required and the ribbon cable, If used, should be terminated as per the MVIP-90 bus requirements.

The Extended SCbus timing for these rates is identical to those in ref. #1, section 4.

2.9.2. Timing Rules and Observations

All Extended SCbus data transfer timing within either the SD bus or the SD_A bus is identical to that specified in reference #1, section 4.

RECOMMENDATION 2.7: Extensions equipped SCbus modules should provide collision detection when driving the SD lines or SD_A lines to detect collisions between modules or to detect a data line stuck-at-high or stuck-at-low.

RULE 2.28: For extended SCbus modules operating in **S32** mode, modules SHALL drive all 32 SD and SD_A lines in a manner compliant with the timing requirements of ref. #1, section 4.

RULE 2.29: For redundant SCbus modules operating in the **R16** mode, modules SHALL drive all 16 SD lines in a manner compliant with ref. #1, section 4, with respect to the bus control signal set active at a given time.

RULE 2.30: For redundant SCbus modules operating in the **R32** mode, modules SHALL drive all 32 SD_*n* and SD_*n*A lines in a manner compliant with ref. #1, section 4, with respect to the bus control signal set active at a given time.

RULE 2.31: For extended SCbus modules operating in **DI16** mode, modules SHALL drive the primary bus and secondary bus, individually, in a manner compliant with ref. #1, section 4.

For systems using **DI16** mode there are no other timing restrictions between **SD_[0-15]** and **SD_[0-15]A** lines or between the primary and alternate bus control signal set for the purposes of this standard.

CHAPTER 3

3. Message Bus

3.1. Introduction

SCSA also provides a secondary, intrasystem communication medium. It is provided to support low latency and high priority inter-board and inter-chassis messages. This channel uses a single wire, peer-to-peer serial bus called the message bus. The message bus operates in a collision sense, multiple access with collision detection(CSMA/CD) access mode. The information transported on the bus is in packet format using the ISO #3309 (HDLC) structure(see ref. #6).

RULE 3.1: All physical layer and data link layer characteristics of the message bus used with extended SCSA modules shall comply with ref. #1, section 3.

RULE 3.2: Extended SCSA modules that operate in **R16** or **R32** mode and provide a message bus SHALL provide an alternate message bus using signal **MCA**.

RULE 3.3: The alternate message bus electrical and frame format provided by extended SCSA modules SHALL comply with ref. #1, section 3.

No logical or timing relationship is implied between the primary and alternate message bus for the purpose of this standard.

3.2. Message bus fallback

Redundancy equipped modules also need to provide backup for the message bus if the primary SCbus control signal set is declared inoperable by the SCbus clock master or standby master module by driving signal **PRIBUS** low.

RULE 3.7: For extended SCbus modules that support **R16** or **R32** mode and message bus, fallback operation SHALL be provided as follows:

-MC clock must be derived from the **SCLK** signal when **PRIBUS** is TRUE(high) and from **SCLKA** when **PRIBUS** is FALSE(low). (see ref. #1, section 3.2.1.2 for MC clock generation method).

-When the **PRIBUS** signal changes state, equipped modules SHALL abort any ongoing message transmission or reception and release the previous message bus signal line.

-When MC clock is synchronized to the new **SCLK** or **SCLKA**, as appropriate, modules may initiate retransmission or reception of interrupted messages.

3.3. Frame Format

The message frame structure described in ref. #1, sections 3.2.2 through 3.3 also applies to extended SCbus modules equipped with message bus.

3.4. Typical Message Bus Operation

3.4.1. Initialization

RULE 3.8: Message bus equipped modules that support redundant operation SHALL initialize so as to use the primary message bus signal MC along with the primary bus control signal set.

3.4.2. Message Bus I Field Structure

The data link layer definitions contained in ref. #1, section 3 provide non-interfering operation between differing network layers.

CHAPTER 4

4. SCbus Electrical Specifications

4.1. Bus Driving and Receiving Requirements

The general SCbus driving and receiving requirements of ref. #1, sec. 5.3 also apply to extended SCbus modules.

RECOMMENDATION 4.1: Extended and redundant clock master modules may need to provide for series passive drive components in the **SCLK** and **SCLKx2*** signal lines to permit individual card designs to be tailored to expected SCbus load conditions.

RECOMMENDATION 4.2: Depending on the clock drivers used Extended SCbus clock master modules may need to be designed to either be driven from the electrical center of the SCbus or from the ends. Product documentation should also reflect that fact.

4.2. Signal Line Terminations

The general SCbus signal line termination requirements of ref. #1, sec. 5.5 also apply to extended SCbus modules.

4.2.1. Clocks and FSYNC*

RULE 4.1: Each Extended SCbus module, either clock master or slave, SHALL provide a 33 KOhm pull up to +5 Volts for the **SCLK**, **SCLK*2**, and **FSYNC*** signal lines and, for modules that support **R16**, **R32** and **DI16** modes, also the **SCLKA**, **SCLKx2A***, **FSYNCA*** and signal lines.

RULE 4.2: For SCbus operation at 8 MHz the **SCLK(A)**, **SCLKx2(A)*** and **FSYNC(A)*** signals should be terminated by a 470/680 Ohm pull-up/pull-down network at the electrical ends of the backplane.

If the PC backplane is 4.8 inches or shorter ref. #1, PERMISSION 5.3 also applies to these signal lines.

4.2.2. CLKFAIL SREF8K and PRIBUS

RULE 4.3: The SCbus **CLKFAIL**, **CLKFAILA, SREF8K**, **SREF8KA** and **PRIBUS** signals SHALL each be pulled-up to +5 Volts by a 4.7K pull-up resistor on each extended SCbus clock master module.

RECOMMENDATION 4.3: Since **SREF8K** and **SREF8KA** are asynchronous (to the SCbus signals), these signal rise and fall times should be at least 15 ns. but no longer than 25% of the minority period, to minimize crosstalk.

For example: if an 8 kHz **SREF8K** signal has an active low period of 122 ns. out of 125 us. then the rise and fall times should be greater than 15 ns but not exceed 30.5 ns.

RECOMMENDATION 4.4: To avoid floating signal lines when boards are plugged into non-extended SCbus backplanes, extended SCbus modules should provide signals **SCLKA**, **SCLKx2A***, **FSYNCA***, **CLKFAILA**, **SREF8KA** and **PRIBUS** each with a 100 KOhm nominal pull-up resistor to +5 Volts.

CHAPTER 5

5. Mechanical Specifications

5.1. Introduction

The Extended SCbus signals are assigned to rows (z) and (d) of the VME64x 160 pin J2/P2 connector. The basic mechanical parameters are contained in ref's #3 and #4. There are no special mechanical considerations for implementing SCbus Extensions on VME64x compliant boards or daughterboards.

The Extended SCbus signals coexist with all VME64x bus extensions. Products compliant with this standard will also be mechanically compatible with ANSI/VITA 1 VME64 bus backplanes but will provide no Extended SCbus features that are assigned to the 5 row P2 connector rows (z) and (d) of VME64x.

5.2. VME Bus Backplane Connectors and VME Board Connectors

5.2.1. SCbus Connector

The Extended SCbus uses 62 total pin locations of rows (z), (a), (c) and (d) of the 5 row, 160 pin VME64x J2/P2 connector as defined in ref. #7.

RULE 5.1: The Extended SCbus signals SHALL be assigned as per table 5.

5.2.2. VME64 Extensions Compatible SCbus J2/P2 PIN ASSIGNMENT

Table 5 - J2/P2 Pin Assignments

Pin#	Row (z)	Row (a)	Row (b)	Row (c)	Row (d)
1	*SD_15A*	SD_15	+5V	MC	User I/O
2	GND	SD_13	GND	SD_14	User I/O
3	*SD_13A*	GND	RETRY	SD_12	*SD_14A*
4	GND	SD_10	A24	SD_11	*SD_12A*
5	*SD_11A*	SD_8	A25	SD_9	*SD_10A*
6	GND	GND	A26	SD_7	*SD_8A*
7	*SD_9A*	SD_5	A27	SD_6	*SD_6A*
8	GND	SD_3	A28	SD_4	*SD_4A*
9	*SD_7A*	SD_1	A29	SD_2	*SD_3A*
10	GND	SD_0	A30	GND	*SD_2A*
11	*SD_5A*	FSYNC*	A31	CLKFAIL	*SD_1A*
12	GND	SCLK	GND	SREF8K	*SD_0A*
13	*SCLKA*	SCLKx2*	+5 V	GND	*MCA*
14	GND	*CLKFAILA*	D16	SL_4*	*SREF8KA*
15	*SCLKx2A**	SL_3*	D17	SL_2*	*FSYNCA**
16	GND	SL_1*	D18	SL_0*	*PRIBUS*
17	User I/O	User I/O	D19	User I/O	User I/O
18	GND	User I/O	D20	User I/O	User I/O
19	User I/O	User I/O	D21	User I/O	User I/O
20	GND	User I/O	D22	User I/O	User I/O
21	User I/O	User I/O	D23	User I/O	User I/O
22	GND	User I/O	GND	User I/O	User I/O
23	User I/O	User I/O	D24	User I/O	User I/O
24	GND	User I/O	D25	User I/O	User I/O
25	User I/O	User I/O	D26	User I/O	User I/O
26	GND	User I/O	D27	User I/O	User I/O
27	User I/O	User I/O	D28	User I/O	User I/O
28	GND	User I/O	D29	User I/O	User I/O
29	User I/O	User I/O	D30	User I/O	User I/O
30	GND	User I/O	D31	User I/O	User I/O
31	User I/O	User I/O	GND	User I/O	GND
32	GND	User I/O	+5 V	User I/O	Vpc

Note: The VME64 bus signals in row (b), above, are for reference only. See ref. #3, section 7.6.2, Table 54 for J2/P2 pin assignments.

CHAPTER 6 (this chapter is informative only)

6. Inter-chassis Link Facilities

Please refer to reference #1, section 7 for the basic recommendations.

6.1. Introduction

The intent of this section is to establish broad guidelines for the development of interchassis links for the purpose of resource sharing, call routing, and overall redundancy as though provided by one large system.

Link cards are needed to exchange some or all of the SCbus timeslots from one electrical SCbus to another. The mapping of the SCbus backplane timeslots through the link card will require a low level software driver that should be integrated into the SCSA switching and routing model to permit full flexibility inter-system sharing. The identity of timeslots should be preserved though the link.

6.2. SD bus link Requirements:

Interchassis link cards that transport data bus timeslot content between Extended SCbuses in physically independent SCSA systems need to adhere to just a few additional guidelines beyond those in reference #1, section 7. These are:

-Preserve frame structure. This means that the exact relationship between all timeslots in a given frame of 32 SD signals should be reconstructed faithfully at the remote chassis. This is necessary not only to ensure reliable transport but also to permit the use of hyperchannel data paths for the purpose of transporting N x 64 kbps wideband data and media streams.

-Permit remote synchronization. All connected SCbus systems may not have access to network synchronization. It would then be necessary for some remote SCbus systems to obtain network-locked clocking through their link products and some will need to become an SCbus primary or standby clock master and/or to drive signal **SREF8K** and **SREF8KA**, if so equipped. For **R16** or **R32** mode link products, redundant synchronization facilities should be provided for all SCSA multi-chassis systems.

6.3. Message Bus

The message bus protocols above OSI layer 2 are still in definition at this time but are not dependent on data link layer parameters.

-Interface with local message bus(es). Interchassis link cards need to be aware of the extended SCbus mode being used to provide gateway connectivity to the appropriate message bus **MC** or **MCA**.

Appendix A
(Normative and Informative References)

References

1. ANSI/VITA 6-1994 Signal Computing System Architecture, VITA, Scottsdale, AZ
2. VMEbus Specification Revision B and C.1, Motorola Corp; Tempe, AZ.
3. ANSI/IEEE 1014-1987 VMEbus Specification; IEEE, Piscataway, NJ
4. ANSI/VITA 1-1994, VME64 Specification, VITA, ibid
5. SC2000 or SC4000 data sheet; VLSI Techology, San Jose, CA
6. ISO/IEC #3309, High Level Data Link Protocol, Nov. 1993; ANSI, New York, NY.
7. VITA 1.1-199x, VME64x draft standard rev. 1.4, Sept. 20, 1996, VITA, ibid
8. ANSI T1.403-1089 Carrier to Customer - DS1 Metallic Installation, ANSI, ibid
9. ANSI T1.408-1990 Integrated Services Digital Network-Primary Rate, ANSI, ibid
10. AT&T PUB 62411, ACCUNET T1.5 Service Description and Interface Specification, AT&T, Piscataway, NJ.