MAGICard
MULTI-PROTOCOL ADVANCED GATEWAY INTERFACE CARD

FEATURES

* UP TO 10 COMMUNICATION CHANNELS

* PROGRAMMABLE TRANSMISSION FEATURES
  - SYNC TIME (BETWEEN WORDS)
  - INTER-BLOCK TIME (PER BLOCK)
  - TX DATA BLOCK SIZE (PER CHANNEL)
  - VARIABLE TX AMPLITUDE (PER CHANNEL)
  - BIT RATES (HI/LO SPEED, VARIABLE)

* TRANSMISSION MODES
  - ONE-SHOT
  - LOOP
  - 'N' TIMES

* PARITY OPTIONS
  - ON/OFF
  - ODD/EVEN

* ERROR INJECTION PER BLOCK
  - BIT COUNT HI/LO
  - SYNC TIME
  - STRETCH BIT
  - BIT RATE (FREQUENCY)
  - PARITY

* RS-232/422/485/423 CHANNELS
  - PER CHANNEL BAUD RATE UP TO 256K
  - FROM 5 TO 8 DATA BITS
  - EVEN/ODD/NO/STICK PARITY
  - SELECTABLE CTS/DTR CONTROL (RS-232)
  - SELECTABLE LOOPBACK (RS-485)

* TWO RECEIVE/MONITOR MODES
  - SEQUENTIAL
  - LOOK-UP TABLE

* 32-BIT TIME TAGGING PER WORD

* WORD STATUS TAGGING

* LABEL FILTERING

* START TRIGGERS

* RCV ERROR COUNTER PER CHANNEL

* ERROR DETECTION PER WORD
  - BIT COUNT
  - SYNC TIME
  - PARITY
  - BIT CODING ERROR

* MERGE MODE STORES ALL DATA FROM ALL RECEIVE CHANNELS IN ONE BUFFER AREA

* RCV COUNT INTERVAL TRIGGER

* INTERRUPT AND POLLING MODES

* PROGRAMMABLE HARDWARE TRIGGER - SELECTABLE CTS/DTR CONTROL (RS-232)

* 32Kx8 DUAL PORT RAM

* EASY TO INSTALL AND OPERATE

The EXC-3000VME-VXI (MAGICard) is a multi-protocol test and simulation card for VME and VXI systems. One card can contain up to 10 ARINC and RS-232/422/485/423 channels in any combination of transmitters and receivers. In addition, one protocol [channel] plug-in adapter board can be added. These include MIL-STD-1553, IEEE-488 and SDLC. In addition, unique customer interfaces can usually be implemented either on the main board or in the form of plug-in adapter boards. The transmitters for all standard protocols operate via a transmitter "instruction" stack which allows for the scheduling of data transmission and reduce the need for host computer intervention. The receivers allow filtering and multi-storage modes of data words. The card is easy to use and custom application

programs can be written in all standard languages such as: Pascal, C, Basic, Assembler, etc.

TABLE OF CONTENTS

INTRODUCTION

This document refers to the EXC-3000VME-VXI card as the MAGICard. The MAGICard is a memory-mapped multi-channel test, simulation, and monitor card that operates within VME and VXI systems. The card comes in both "B" and "C" sizes. The card can be populated with up to 10 ARINC (429/561/568/575/582) and RS-232/422/485/423 channels - in any combination of transmitters and receivers. In addition, the card has error injection and error detection/ reporting capability in the ARINC modes on a channel by channel basis. The user can select from three different ARINC bit rates; Lo speed, Hi speed and programmable, and can select individual bit rates for each "RS" (232/422/ 485/423) channel.

The MAGICard is the perfect solution for developing, simulating, testing and monitoring avionics communication interfaces such as ARINC, RS-232/422/485/423, MIL-STD-1553, IEEE-488 and SDLC.

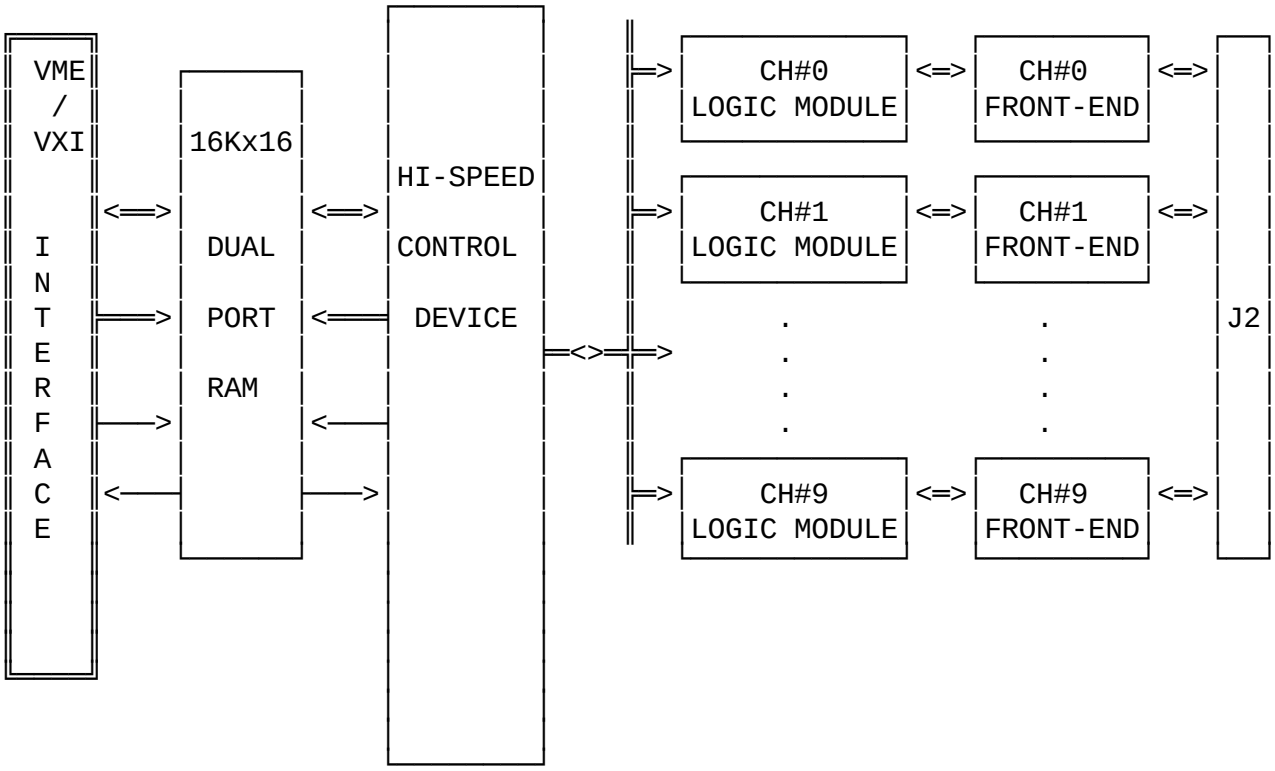


FIGURE 1. MAGICard BLOCK DIAGRAM

GENERAL

The MAGICard operation makes extensive use of pointers for setting up the size and location of both receiver and transmitter data blocks, transmitter instruction stacks, and receiver look-up tables. Each channel has its own pointer registers so that unique memory areas may be allocated for each channel. It is also possible, for multiple channels to share memory areas (more than one receiver channel, for example, may point to and use the same Label Look-Up Table which controls which labels will be stored by the card). The transmitter and receiver operations necessary to operate the card are described below in general terms. See the sections on the Transmitter and Receive Operations for details.

(after power-on)

1. POWER-ON OPERATION

- board clears the memory and executes initialization procedure
- user waits for BOARD READY Register to be valid (see register definitions)

2. SETUP/VERIFY THE "GLOBAL REGISTERS"

- user can check the results of the power-on self-test by reading the Board Status Register
- user can verify the configuration of each channel (TX or Rcv) by reading the Configuration Status Registers.
- update the Programmable Bit Rate Register (only if it is used).
- update the IRQSEL field in the CONTROL configuration register.

3. SETUP THE TRANSMITTER-RELATED "CHANNEL CONTROL REGISTERS"

- program the Channel Configuration registers (parity, bit rate,etc.)
- update the Transmit Instruction Stack Pointer for each channel
- update the Transmit Instruction Counter
- update the Transmit Amplitude Register for each channel
- for RS channels, set the Channel Start Transmit Register to 1

4. SETUP THE "TX INSTRUCTION BLOCKS"

- update the Instruction Blocks with information relating to each ARINC TX data block (i.e. error injection, pointer to the TX data blocks, delay between data blocks). See the section on Transmitter Operation for details)

5. WRITE THE "TX DATA BLOCKS"

- write the ARINC words (or bytes for RS channels) into the on-board memory at locations pointed to by the instruction stack's TX Data Pointers.

6. SETUP THE RECEIVER-RELATED "CHANNEL CONTROL REGISTERS"

- program the Channel Configuration registers (parity, bit rate,etc.)
- update the Receive Start and End Pointers
- update the Look-Up Table Start Address Register (if using this mode)
- update the Filter Table Start Address Register (if using this mode)
- update the Label Trigger Register (if using a Label to start storage)
- update Counter Trigger Registers (not required)
- for RS channels, set the Channel Start Receive Register to 1

7. START

- write to the Global Start Register (setting the appropriate channel(s) "start" bits. Each channel can be "started" individually - at different times. (see Global Registers and definitions).

8. READ THE RECEIVE STATUS REGISTERS (i.e. Word Counter, Error Counter)

- read the Receiver status registers to know how many words have been received and how many invalid words (if any) were detected.

9. READ THE "RCV DATA BLOCK"

- read the ARINC words or RS bytes (and RCV Status and Time Tag Words) from the on-board memory.

Note: for Williamsburg channels the Channel Control Register Block associated with the receive module is used for both receive and transmit control (see the section on Channel Control Register Blocks). Thus for a Williamsburg channel whose receive module is located in channel socket 4, transmit module is in channel socket 5, and

CRC module is in channel socket 6, only the channel 4 Control Register Block is used. The Control Register Blocks for channels 5 and 6 are not used.

Data Storage Area and Control Register Access

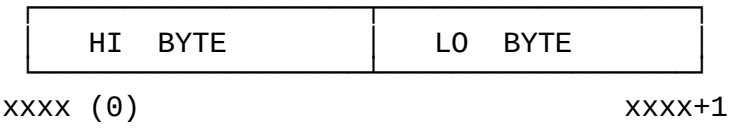
The Data Storage Area and Control Registers reside in 16Kx16 of true dual-port RAM. It is within this on-board, dual-port RAM that the user controls the operation of the board. The user is advised to use 16-bit addressing when operating with 16-bit control registers or data words. When accessing 8-bit control registers or data bytes, it is recommended that the user use byte addressing since there is a possibility of inadvertantly overwriting a byte-wide location (which resides next to the desired location) when using 16-bit word addressing.

All 16-bit words (data and control registers) contained within the board's dual-port RAM are stored in the following manner:

* The HI byte is accessed at EVEN addresses, while the LO byte is accessed at ODD addresses. *

Example

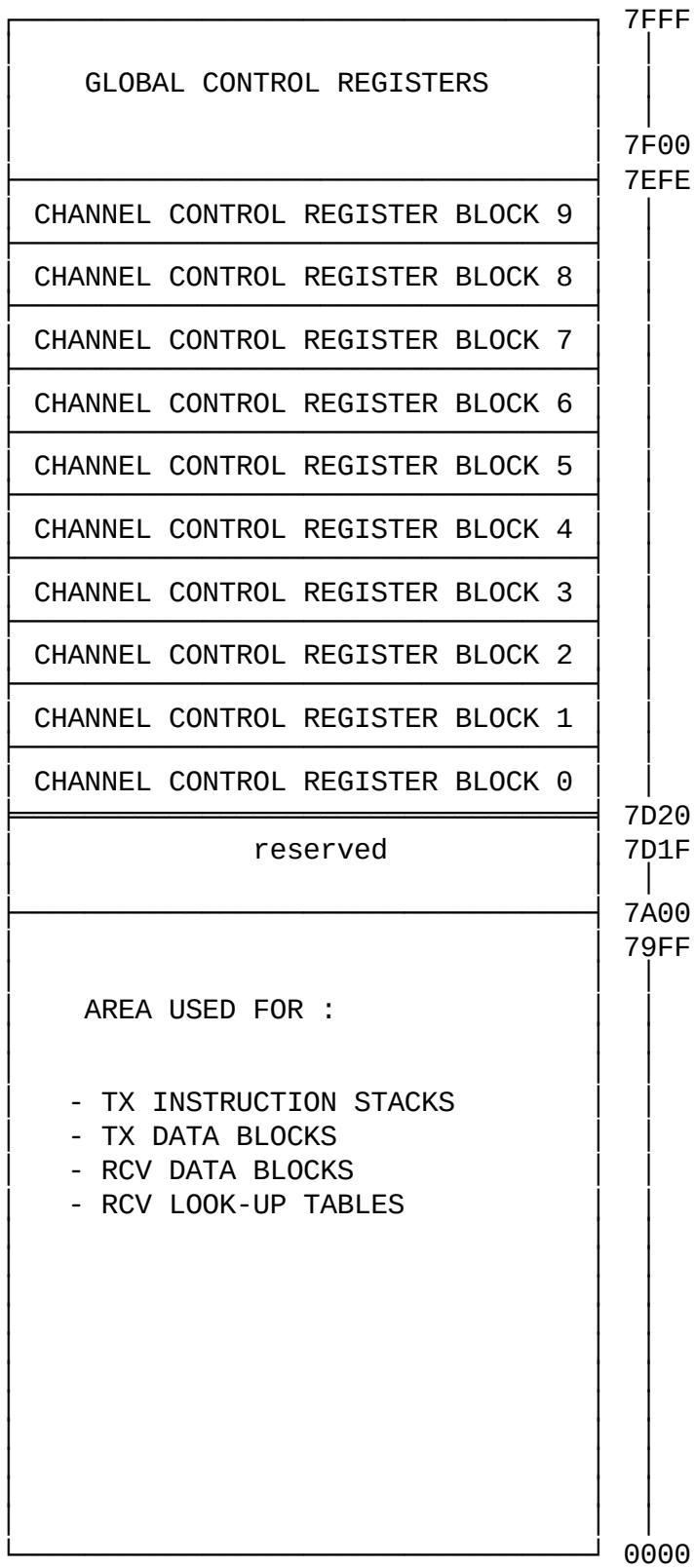
The Stack Pointer is located at address : xxxx



Note that all addresses listed in the remainder of this document are given relative to the Base Address written by the user into Offset Register as described in "Configuration Registers".

OPERATION MANUAL FOR THE ARINC-429/561/568/575/582 AND WILLIAMSBURG CHANNELS

MAGICard
GENERAL MEMORY MAP



ARINC Channels

GLOBAL CONTROL REGISTERS

reserved	7FFF | | 7F86
SOFTWARE RESET REGISTER	7F84
reserved	7F82
reserved	7F80
reserved	7F7E
RESET TIME TAG REGISTER	7F7C
START REGISTER	7F7A
reserved	7F78
BOARD READY REGISTER	7F76
BOARD STATUS REGISTER	7F74
FIRMWARE REVISION REGISTER	7F72
INTERRUPT STATUS REGISTER	7F70
PROGRAMMABLE BIT RATE REGISTER	7F6E
CHANNEL 9 CONFIGURATION STATUS REG.	7F6C
CHANNEL 8 CONFIGURATION STATUS REG.	7F6A
CHANNEL 7 CONFIGURATION STATUS REG.	7F68
CHANNEL 6 CONFIGURATION STATUS REG.	7F66
CHANNEL 5 CONFIGURATION STATUS REG.	7F64
CHANNEL 4 CONFIGURATION STATUS REG.	7F62
CHANNEL 3 CONFIGURATION STATUS REG.	7F60
CHANNEL 2 CONFIGURATION STATUS REG.	7F5E
CHANNEL 1 CONFIGURATION STATUS REG.	7F5C
CHANNEL 0 CONFIGURATION STATUS REG.	7F5A

ARINC Channels

RECEIVER DATA STORAGE MODE REGISTER	7F58
RECEIVER MERGE START POINTER	7F56
RECEIVER MERGE END POINTER	7F54
RECEIVER MERGE CURRENT POINTER	7F52
RCVER MERGE FILTER TABLE START ADDR.	7F50
RECEIVER MERGE WORD COUNTER	7F4E
RECEIVER MERGE WORD COUNT TRIGGER REG	7F4C
RECEIVER MERGE BUFFER WRAPAROUND REG	7F4A
RECEIVER MERGE INTERVAL COUNT TRIGGER	7F48
RECEIVER MERGE ERROR COUNT REGISTER	7F46
RECEIVER MERGE LABEL TRIGGER REGISTER	7F44
RECEIVER MERGE CONFIGURATION REGISTER	7F42
RCV MERGE INTR/TRIGGER CONDITION REG	7F40
RECEIVER MERGE STATUS REGISTER	7F3E
INTERRUPT STATUS BUSY REGISTER	7F3C
reserved	7F3A 7F00

SOFTWARE RESET REGISTER	7F84 (H)	(WRITE ONLY)

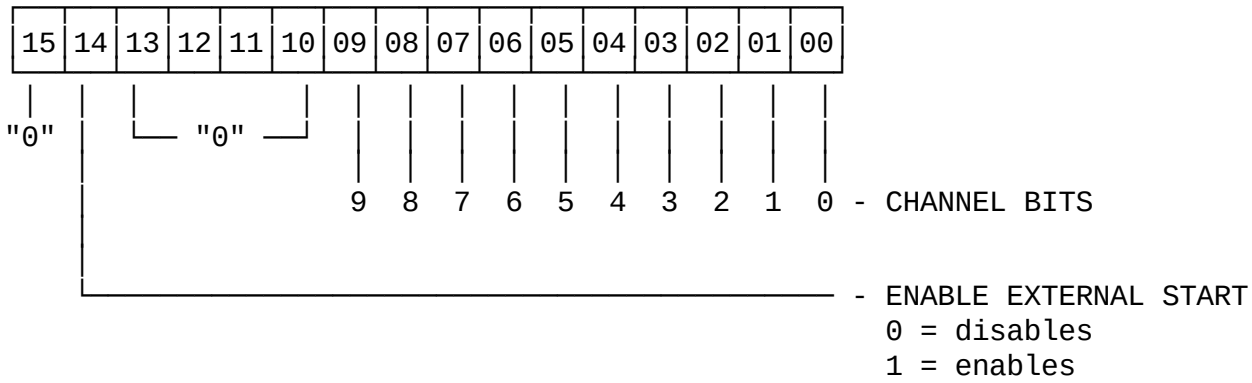
Writing a 0 to this register resets the board. Following a reset, the board will execute a self-test (both memory and channels), clear all the on-board memory and then update the Board Status Register. The board indicates that it is ready by writing a value of 3000(H) to the Board Ready Register.

RESET TIME TAG REGISTER	7F7C (H)

Writing any non-zero value to this register will cause the time tag to be reset to 0. Upon completion of the Time Tag Reset operation, this register will be cleared by the board.

START/STOP REGISTER	7F7A (H)

The user can start one or more channels at the same time. Writing a "1" to bit "00" starts channel "0" operation, writing a "1" to the next location starts channel "1", etc. Writing a "0" to a bit location will stop that channel's operation. To start or stop a Williamsburg channel it is necessary to write to the bit corresponding to the channel socket of the receive module. For example, for a Williamsburg channel whose receive module resides in channel socket 7, transmit module in channel socket 8, and CRC module in channel socket 9 it is necessary to write a "1" to bit 07 to turn the channel on, or a "0" to turn it off. The user should wait a minimum of 500 µsec between writes to the START/STOP register.
The Enable External Start bit is used in conjunction with on-board jumpers to start the board's operation from an external event (see section on VME/VXI Jumpers and External Triggers for details).



Note: a change in a channel's TX Amplitude Register or Configuration Register, in the Programmable Bit Rate Register, or in the Receiver Data Storage Mode Register is acted upon by the firmware only after the Start/Stop Register contains a "0" for at least 1 msec.

ARINC Channels

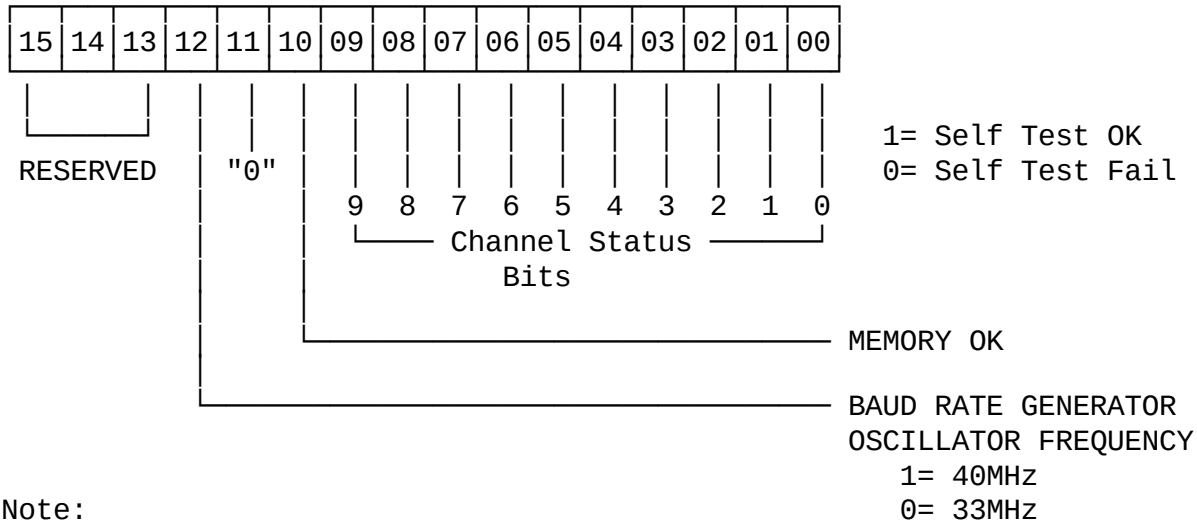
BOARD READY REGISTER 7F76 (H)

This register indicates that the board has finished its Power-on sequence and that the board is ready to be accessed by the Host. The board will write the value: 3000 (H) into this register when ready.

3 0	0 0
HI BYTE	LO BYTE

BOARD STATUS REGISTER 7F74 (H)

Indicates the result of the Power-on, self-test of the board. Williamsburg receive, transmit, and CRC modules are treated independently.



Note:

1. The "Self Test Fail" is set when the channel self-test fails or when the channel is not present on the board.

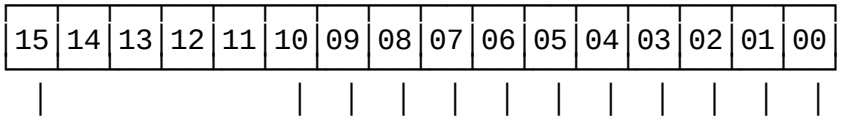
2. The board will continue to operate on condition of Channel Self-Test Failures BUT will not continue to operate on condition of a Memory failure.

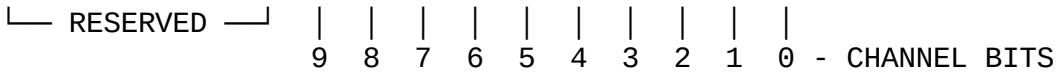
FIRMWARE REVISION REGISTER 7F72 (H)

Indicates the revision level of the firmware (ie: 0114 (H) = Rev 1.14)

INTERRUPT STATUS REGISTER 7F70 (H)

Indicates which channel issued the interrupt (`1' = Active). The status bit(s) are only reset by the user. All interrupts from a Williamsburg channel set the bit corresponding to the position of the receive module.

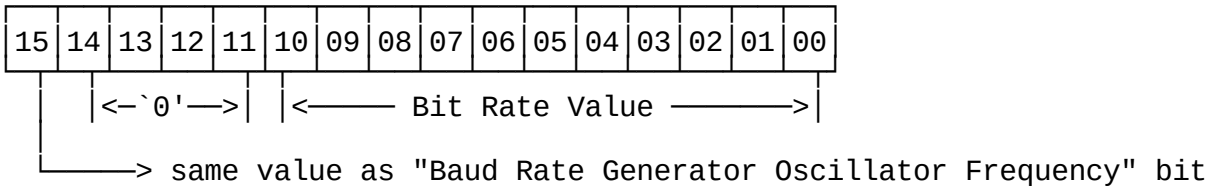




ARINC Channels

PROGRAMMABLE BIT RATE REGISTER	7F6E (H)

Selects the programmable bit rate value for the ARINC channels. This register is only read by the firmware when the Start Register (7F7A) contains a value of 0 (i.e. all channels are inactive) for at least 1 msec.



If the Baud Rate Generator oscillator frequency is 33MHz (as indicated by a value of 0 in the "Baud Rate Generator Oscillator Frequency" bit of the Board Status Register) bit 15 of the Programmable Bit Rate Register must be 0. The formula for calculating the Bit Rate Value (BRV) is:

$$BRV = \frac{4125}{\text{freq (Khz)}} - 1$$

Example Desired frequency is 100 Khz

$$BRV = \frac{4125}{100} - 1 = 41 - 1 = 40 \text{ Dec. or } 0028 \text{ (Hex)}$$

NOTE THAT THE NUMBER HAS BEEN ROUNDED OFF (41 - NOT 41.25)

Write the word "0028(H)" to this register.

If the Baud Rate Generator oscillator frequency is 40MHz (as indicated by a value of 1 in the "Baud Rate Generator Oscillator Frequency" bit of the Board Status Register) bit 15 of the Programmable Bit Rate Register must be 1. The formula for calculating the Bit Rate Value (BRV) is:

$$BRV = \frac{5000}{\text{freq (Khz)}} - 1$$

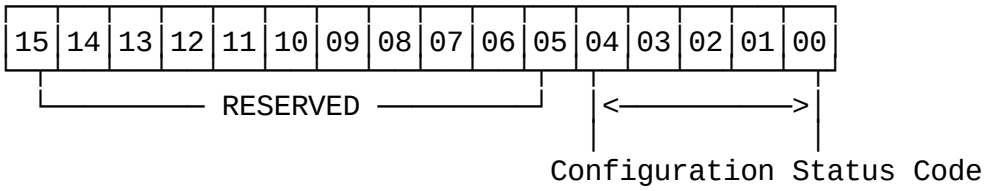
Example Desired frequency is 12.5 Khz

$$BRV = \frac{5000}{12.5} - 1 = 400 - 1 = 399 \text{ Dec. or } 018F \text{ (Hex)}$$

Write the word "818F(H)" to this register.

CHANNEL x CONFIGURATION STATUS REGISTER (see Global Memory Map)

This register indicates to the host the type of channel configured in each channel socket on the board.



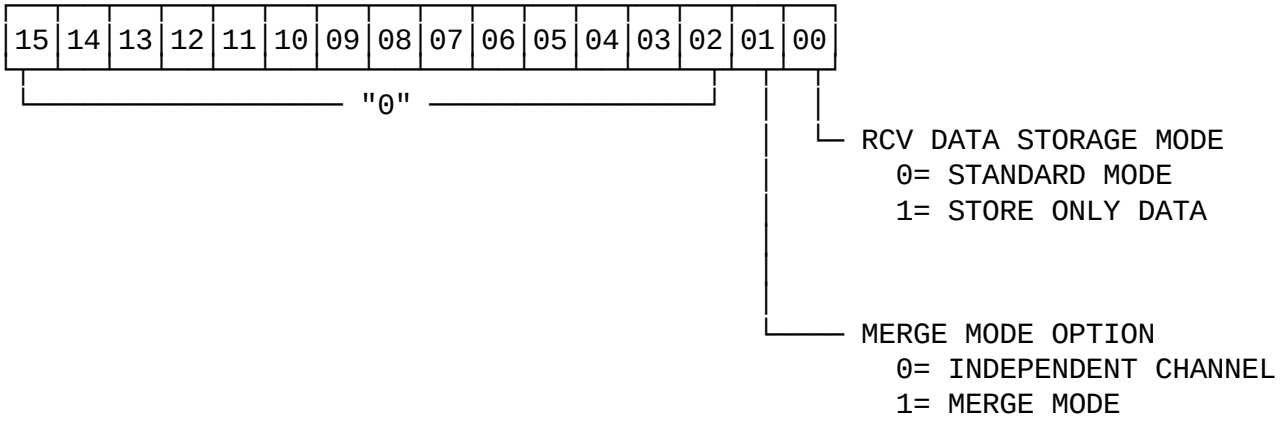
4 3 2 1 0	Configuration Status Code
0 0 0 0 0	Undefined Channel
0 0 0 0 1	ARINC-429 Receive Channel
0 0 0 1 0	ARINC-429 Transmit Channel
0 0 0 1 1	ARINC-561 Receive Channel
0 0 1 0 0	ARINC-561 Transmit Channel
0 0 1 0 1	ARINC-568 Receive Channel
0 0 1 1 0	ARINC-568 Transmit Channel
0 0 1 1 1	ARINC-575 Receive Channel
0 1 0 0 0	ARINC-575 Transmit Channel
0 1 0 0 1	ARINC-582 2-Wire Receive Channel
0 1 0 1 0	ARINC-582 2-Wire Transmit Channel
0 1 0 1 1	ARINC-582 6-Wire Receive Channel
0 1 1 0 0	ARINC-582 6-Wire Transmit Channel
0 1 1 0 1	RESERVED
0 1 1 1 0	RESERVED
0 1 1 1 1	*
1 0 0 0 0	RS-232 Channel
1 0 0 0 1	RS-422 Channel
1 0 0 1 0	RS-485 Channel
1 0 0 1 1	RS-423 Channel (Note 2)
1 0 1 0 0	*
1 0 1 0 1	*
1 0 1 1 0	*
1 0 1 1 1	*
1 1 0 0 0	*
1 1 0 0 1	*
1 1 0 1 0	*
1 1 0 1 1	*
1 1 1 0 0	*
1 1 1 0 1	ARINC-429 Williamsburg Rx Channel
1 1 1 1 0	ARINC-429 Williamsburg Tx Channel
1 1 1 1 1	ARINC-429 Williamsburg CRC Channel

ARINC Channels

Notes: 1) * - Free Codes for future channels
2) The RS-423 channel requires firmware Revision 1.31 and up.

RECEIVER DATA STORAGE MODE REGISTER	7F58 (H)

This register is used to select the Receiver Data Storage Mode and the Merge Mode option. ARINC data words can be stored with Time Tag and Status words appended to the data block or without these additional words. Set bit "00" to a logic 0 in order to select the standard mode which appends both Time Tag and Status Words to each ARINC word stored in memory. Set register bit "00" to a logic 1 to select Data Only mode. Bit "01" controls the Receiver Merge Mode selection. A logic "0", selects the standard independent mode which utilizes different receive buffer areas for each receive channel. A logic 1 selects the Merge Mode which utilizes a single receiver buffer for all channels. Each Receive Status Word, in this case, is tagged with Channel ID information.



Notes:

1) If Data Only Storage Mode is selected (bit 00 set to 1), storage will be per independent channel regardless of the state of bit 01.

2) Data Only Storage Mode is not available in Lookup Table Mode.

3) Merge Mode is not available for Williamsburg channels, or for RS-232, 422 or 485 channels. Even if Merge Mode is chosen, these channels will store received data in the channel receive buffer.

4) A change in this register is only noted by the firmware after the Start/Stop Register contains a value of "0" for at least 1 msec.

RECEIVER MERGE START POINTER	7F56 (H)

Set the start address of the Receive Data Buffer. The address must be even.

RECEIVER MERGE END POINTER	7F54 (H)

Set the End Address of the Receive Data Buffer. The data will wrap around or stop when the buffer is full (End address is reached) - depending upon the contents of the Receiver Merge Configuration Register Wrap-around bit.

RECEIVER MERGE CURRENT POINTER	7F52 (H)

Indicates the current address within the Receiver buffer. This pointer value is incremented after the entire receiver block (ARINC word, time tag, and status) is written into memory.

RECEIVER MERGE FILTER TABLE START ADDRESS	7F50 (H)

Sets the start address of the (256x8) Label Filter Table as described in the Sequential storage mode. The address must be even.

RECEIVER MERGE WORD COUNTER	7F4E (H)

Indicates the number of words received (0-64k). This register wraps around, and may be reset by the user only after the channel is stopped.

RECEIVER MERGE WORD COUNT TRIGGER REG	7F4C (H)

Sets a trigger (used for polling or interrupts) which indicates when a specific number of words have been received (1-64k). The appropriate bit must also be set in the Receiver Merge Interrupt/Trigger Condition Register.

RECEIVER MERGE BUFFER WRAPAROUND REG	7F4A (H)

This register contains 2 bits for synchronization with the "C" drivers. If bit 14 is set to 1, the receive buffer has wrapped around once since the last data read. If bit 15 is set to 1, there have been multiple wraparounds.

RECEIVER MERGE INTERVAL COUNT TRIGGER	7F48 (H)

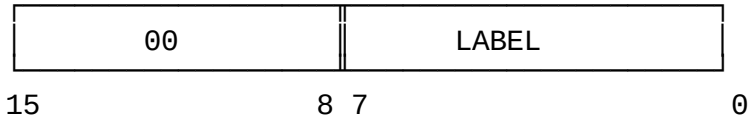
This 16-bit value allows the user to generate an interrupt (or pollable bit) every "N" number of words. The appropriate bit must also be set in the Receiver Merge Interrupt/Trigger Condition Register.

RECEIVER MERGE ERROR COUNT REGISTER	7F46 (H)

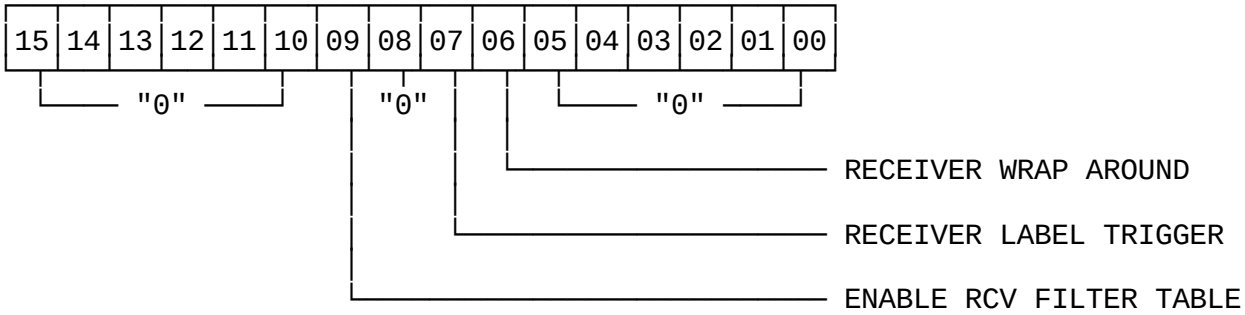
16-Bit counter. Indicates the number of error words received on the channel. This counter register wraps around and is only reset by the user.

RECEIVER MERGE LABEL TRIGGER REGISTER	7F44 (H)

Starts storage of data upon receipt of Label xx(H).



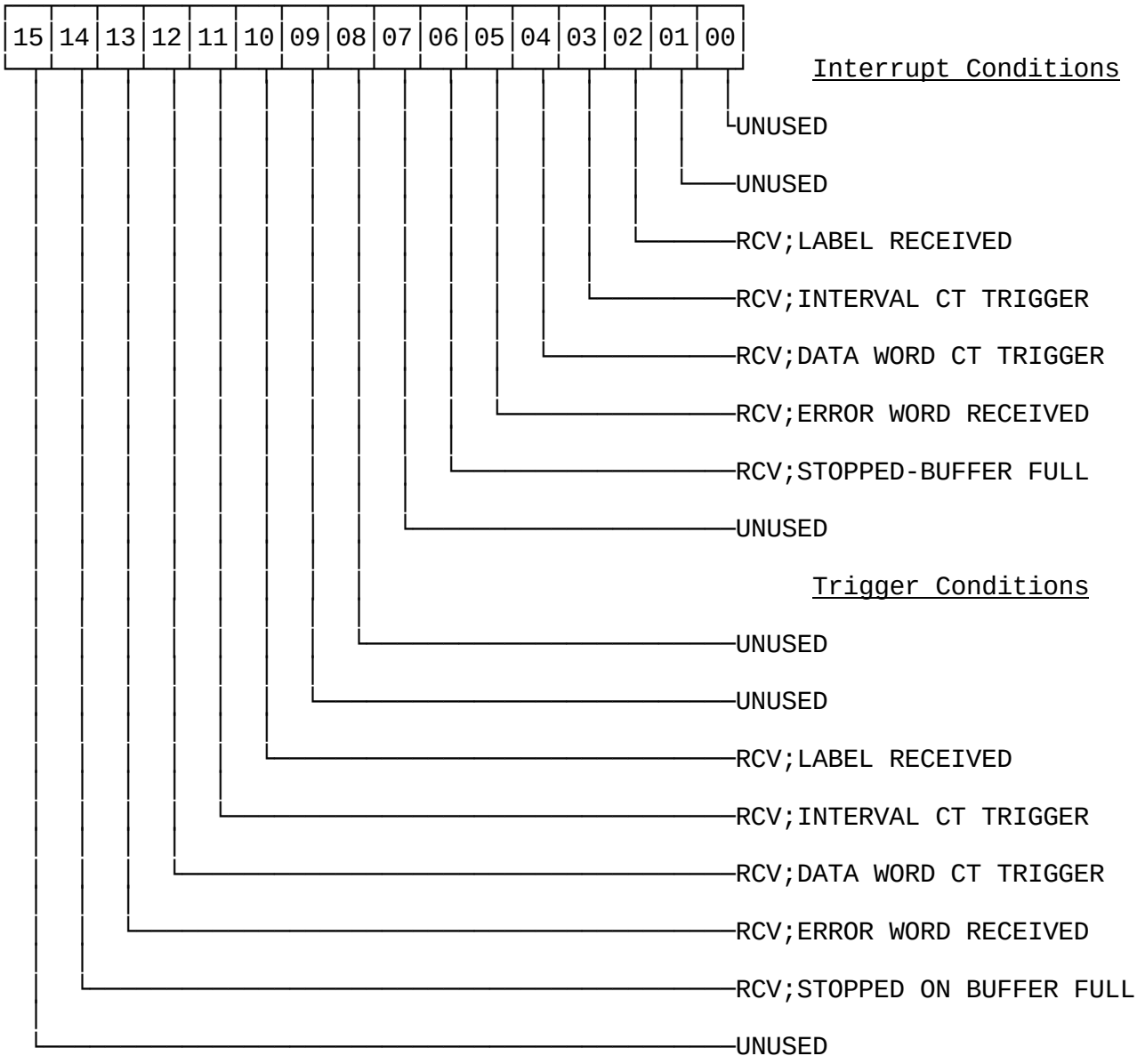
RECEIVER MERGE CONFIGURATION REGISTER	7F42 (H)



Receiver Wrap Around	`1' = Data storage is halted when the buffer is full `0' = Receiver wraps around the data within the block
Receiver Label Trigger	`1' = Start data storage upon receipt of Label xx. `0' = Receiver stores data without Start Label Trigger
Enable Rcv Filter Table	`1' = Enable filter table (stores Labels per table) `0' = Disables table. Stores all Labels.

RECEIVER MERGE INTERRUPT/TRIGGER CONDITION REGISTER	7F40 (H)

Sets the Interrupt and Trigger condition(s) for ARINC receive channels in Merge Mode. Bits 00-07 are the interrupt condition bits while bits 08-15 relate to the hardware trigger bits. The trigger conditions set a pulse on the trigger [subminiature BNC] J1 connector.



Notes:

- The RCV;LABEL RECEIVED interrupt or trigger only occurs upon reception of a label which has been marked for interrupt in the filter table.

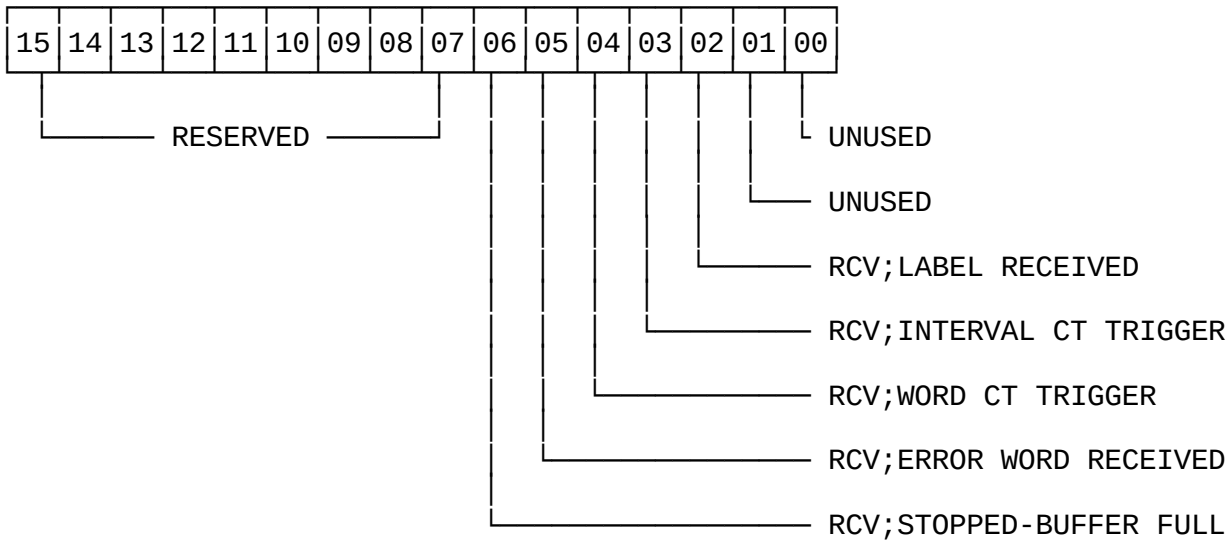
- In order to activate the RCV;INTERVAL CT TRIGGER interrupt or trigger, the Receiver Merge Interval Count Trigger Register must also be set.

ARINC Channels

- In order to activate the RCV;DATA WORD CT TRIGGER interrupt or trigger, the Receiver Merge Word Count Trigger Reg must also be set.

RECEIVER MERGE STATUS REGISTER	7F3E (H)

This register indicates the operational status of the Merge Mode receive buffer. This register can be used to poll the status of the channel or it can be used with interrupts. When used in conjunction with interrupts the register indicates the condition(s) which caused the interrupt. A logic 1 indicates an active bit. Status bits are only reset by the user.

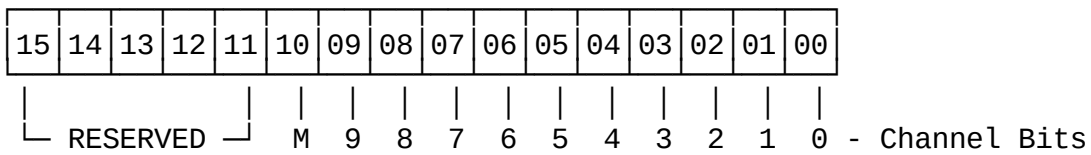


The RCV;LABEL RECEIVED Status bit is set upon receipt of any label for which an interrupt has been requested via the filter table.

INTERRUPT STATUS BUSY REGISTER	7F3C (H)

The Interrupt Status **Busy** Register indicates whether the Channel *x* Status Register, the Receiver Merge Status Register, and the Interrupt Status Register may be accessed by the user. A '1' in any channel bit position indicates that the corresponding Channel Status Register and the global Interrupt Status Register are busy and should not be accessed by the user. A '1' in bit 10 indicates that the Receiver Merge Status Register is busy.

Note: Before accessing the global Interrupt Status Register, the user should wait until the Interrupt Status **Busy** Register = 0. He will then have at least 15 microseconds to safely access the status registers.



ARINC Channels

CHANNEL CONTROL REGISTER BLOCKS

CHANNEL 0 CONTROL REGISTER BLOCK

CHANNEL 0 CONFIGURATION REGISTER	7D4E
reserved	7D4C
CHANNEL 0 RCV DATA START POINTER	7D4A
CHANNEL 0 RCV DATA END POINTER	7D48
CHANNEL 0 RCV DATA CURRENT POINTER	7D46
CHANNEL 0 RCV LOOK-UP TABLE START ADDRESS	7D44
CHANNEL 0 RCV FILTER TABLE START ADDRESS	7D42
CHANNEL 0 RCV DATA WORD COUNT REGISTER	7D40
CHANNEL 0 RCV BUFFER WRAPAROUND REGISTER	7D3E
CHANNEL 0 RCV WORD COUNTER TRIGGER RGSTER	7D3C
CHANNEL 0 RESTART WILLIAMSBURG TRANSMIT	7D3A
CHANNEL 0 RCV INTERVAL COUNTER TRIGGER	7D38
CHANNEL 0 RCV ERROR COUNT REGISTER	7D36
CHANNEL 0 RCV LABEL TRIGGER REGISTER	7D34
CHANNEL 0 TX INSTRUCTION STACK POINTER	7D32
CHANNEL 0 TX INSTRUCTION COUNTER	7D30
CHANNEL 0 TX LOOP COUNTER	7D2E
CHANNEL 0 TX AMPLITUDE REGISTER	7D2C
CHANNEL 0 INTERRUPT/TRIGGER CONDITION REG.	7D2A
CHANNEL 0 STATUS REGISTER	7D28
CHANNEL 0 DEST CODE BUSY / BUFFER START	7D26
CHANNEL 0 FULL-DUPLEX / BUFFER END	7D24
CHANNEL 0 TX SAL REGISTER	7D22
CHANNEL 0 RCV SAL REGISTER	7D20

CHANNEL 1 CONTROL REGISTER BLOCK

CHANNEL 1 CONFIGURATION REGISTER	7D7E
reserved	7D7C
CHANNEL 1 RCV DATA START POINTER	7D7A
CHANNEL 1 RCV DATA END POINTER	7D78
CHANNEL 1 RCV DATA CURRENT POINTER	7D76
CHANNEL 1 RCV LOOK-UP TABLE START ADDRESS	7D74
CHANNEL 1 RCV FILTER TABLE START ADDRESS	7D72
CHANNEL 1 RCV DATA WORD COUNT REGISTER	7D70
CHANNEL 1 RCV BUFFER WRAPAROUND REGISTER	7D6E
CHANNEL 1 RCV WORD COUNTER TRIGGER	7D6C
CHANNEL 1 RESTART WILLIAMSBURG TRANSMIT	7D6A
CHANNEL 1 RCV INTERVAL COUNTER TRIGGER	7D68
CHANNEL 1 RCV ERROR COUNT REGISTER	7D66
CHANNEL 1 RCV LABEL TRIGGER REGISTER	7D64
CHANNEL 1 TX INSTRUCTION STACK POINTER	7D62
CHANNEL 1 TX INSTRUCTION COUNTER	7D60
CHANNEL 1 TX LOOP COUNTER	7D5E
CHANNEL 1 TX AMPLITUDE REGISTER	7D5C
CHANNEL 1 INTERRUPT/TRIGGER CONDITION REG.	7D5A
CHANNEL 1 STATUS REGISTER	7D58
CHANNEL 1 DEST CODE BUSY / BUFFER START	7D56
CHANNEL 1 FULL-DUPLEX / BUFFER END	7D54
CHANNEL 1 TX SAL REGISTER	7D52
CHANNEL 1 RCV SAL REGISTER	7D50

CHANNEL 2 CONTROL REGISTER BLOCK

CHANNEL 2 CONFIGURATION REGISTER	7DAE
reserved	7DAC
CHANNEL 2 RCV DATA START POINTER	7DAA
CHANNEL 2 RCV DATA END POINTER	7DA8
CHANNEL 2 RCV DATA CURRENT POINTER	7DA6
CHANNEL 2 RCV LOOK-UP TABLE START ADDRESS	7DA4
CHANNEL 2 RCV FILTER TABLE START ADDRESS	7DA2
CHANNEL 2 RCV DATA WORD COUNT REGISTER	7DA0
CHANNEL 2 RCV BUFFER WRAPAROUND REGISTER	7D9E
CHANNEL 2 RCV WORD COUNTER TRIGGER	7D9C
CHANNEL 2 RESTART WILLIAMSBURG TRANSMIT	7D9A
CHANNEL 2 RCV INTERVAL COUNTER TRIGGER	7D98
CHANNEL 2 RCV ERROR COUNT REGISTER	7D96
CHANNEL 2 RCV LABEL TRIGGER REGISTER	7D94
CHANNEL 2 TX INSTRUCTION STACK POINTER	7D92
CHANNEL 2 TX INSTRUCTION COUNTER	7D90
CHANNEL 2 TX LOOP COUNTER	7D8E
CHANNEL 2 TX AMPLITUDE REGISTER	7D8C
CHANNEL 2 INTERRUPT/TRIGGER CONDITION REG.	7D8A
CHANNEL 2 STATUS REGISTER	7D88
CHANNEL 2 DEST CODE BUSY / BUFFER START	7D86
CHANNEL 2 FULL-DUPLEX / BUFFER END	7D84
CHANNEL 2 TX SAL REGISTER	7D82
CHANNEL 2 RCV SAL REGISTER	7D80

CHANNEL 3 CONTROL REGISTER BLOCK

CHANNEL 3 CONFIGURATION REGISTER	7DDE
reserved	7DDC
CHANNEL 3 RCV DATA START POINTER	7DDA
CHANNEL 3 RCV DATA END POINTER	7DD8
CHANNEL 3 RCV DATA CURRENT POINTER	7DD6
CHANNEL 3 RCV LOOK-UP TABLE START ADDRESS	7DD4
CHANNEL 3 RCV FILTER TABLE START ADDRESS	7DD2
CHANNEL 3 RCV DATA WORD COUNT REGISTER	7DD0
CHANNEL 3 RCV BUFFER WRAPAROUND REGISTER	7DCE
CHANNEL 3 RCV WORD COUNTER TRIGGER	7DCC
CHANNEL 3 RESTART WILLIAMSBURG TRANSMIT	7DCA
CHANNEL 3 RCV INTERVAL COUNTER TRIGGER	7DC8
CHANNEL 3 RCV ERROR COUNT REGISTER	7DC6
CHANNEL 3 RCV LABEL TRIGGER REGISTER	7DC4
CHANNEL 3 TX INSTRUCTION STACK POINTER	7DC2
CHANNEL 3 TX INSTRUCTION COUNTER	7DC0
CHANNEL 3 TX LOOP COUNTER	7DBE
CHANNEL 3 TX AMPLITUDE REGISTER	7DBC
CHANNEL 3 INTERRUPT/TRIGGER CONDITION REG.	7DBA
CHANNEL 3 STATUS REGISTER	7DB8
CHANNEL 3 DEST CODE BUSY / BUFFER START	7DB6
CHANNEL 3 FULL-DUPLEX / BUFFER END	7DB4
CHANNEL 3 TX SAL REGISTER	7DB2
CHANNEL 3 RCV SAL REGISTER	7DB0

CHANNEL 4 CONTROL REGISTER BLOCK

CHANNEL 4 CONFIGURATION REGISTER	7E0E
reserved	7E0C
CHANNEL 4 RCV DATA START POINTER	7E0A
CHANNEL 4 RCV DATA END POINTER	7E08
CHANNEL 4 RCV DATA CURRENT POINTER	7E06
CHANNEL 4 RCV LOOK-UP TABLE START ADDRESS	7E04
CHANNEL 4 RCV FILTER TABLE START ADDRESS	7E02
CHANNEL 4 RCV DATA WORD COUNT REGISTER	7E00
CHANNEL 4 RCV BUFFER WRAPAROUND REGISTER	7DFE
CHANNEL 4 RCV WORD COUNTER TRIGGER	7DFC
CHANNEL 4 RESTART WILLIAMSBURG TRANSMIT	7DFA
CHANNEL 4 RCV INTERVAL COUNTER TRIGGER	7DF8
CHANNEL 4 RCV ERROR COUNT REGISTER	7DF6
CHANNEL 4 RCV LABEL TRIGGER REGISTER	7DF4
CHANNEL 4 TX INSTRUCTION STACK POINTER	7DF2
CHANNEL 4 TX INSTRUCTION COUNTER	7DF0
CHANNEL 4 TX LOOP COUNTER	7DEE
CHANNEL 4 TX AMPLITUDE REGISTER	7DEC
CHANNEL 4 INTERRUPT/TRIGGER CONDITION REG.	7DEA
CHANNEL 4 STATUS REGISTER	7DE8
CHANNEL 4 DEST CODE BUSY / BUFFER START	7DE6
CHANNEL 4 FULL-DUPLEX / BUFFER END	7DE4
CHANNEL 4 TX SAL REGISTER	7DE2
CHANNEL 4 RCV SAL REGISTER	7DE0

CHANNEL 5 CONTROL REGISTER BLOCK

CHANNEL 5 CONFIGURATION REGISTER	7E3E
reserved	7E3C
CHANNEL 5 RCV DATA START POINTER	7E3A
CHANNEL 5 RCV DATA END POINTER	7E38
CHANNEL 5 RCV DATA CURRENT POINTER	7E36
CHANNEL 5 RCV LOOK-UP TABLE START ADDRESS	7E34
CHANNEL 5 RCV FILTER TABLE START ADDRESS	7E32
CHANNEL 5 RCV DATA WORD COUNT REGISTER	7E30
CHANNEL 5 RCV BUFFER WRAPAROUND REGISTER	7E2E
CHANNEL 5 RCV WORD COUNTER TRIGGER	7E2C
CHANNEL 5 RESTART WILLIAMSBURG TRANSMIT	7E2A
CHANNEL 5 RCV INTERVAL COUNTER TRIGGER	7E28
CHANNEL 5 RCV ERROR COUNT REGISTER	7E26
CHANNEL 5 RCV LABEL TRIGGER REGISTER	7E24
CHANNEL 5 TX INSTRUCTION STACK POINTER	7E22
CHANNEL 5 TX INSTRUCTION COUNTER	7E20
CHANNEL 5 TX LOOP COUNTER	7E1E
CHANNEL 5 TX AMPLITUDE REGISTER	7E1C
CHANNEL 5 INTERRUPT/TRIGGER CONDITION REG.	7E1A
CHANNEL 5 STATUS REGISTER	7E18
CHANNEL 5 DEST CODE BUSY / BUFFER START	7E16
CHANNEL 5 FULL-DUPLEX / BUFFER END	7E14
CHANNEL 5 TX SAL REGISTER	7E12
CHANNEL 5 RCV SAL REGISTER	7E10

CHANNEL 6 CONTROL REGISTER BLOCK

CHANNEL 6 CONFIGURATION REGISTER	7E6E
reserved	7E6C
CHANNEL 6 RCV DATA START POINTER	7E6A
CHANNEL 6 RCV DATA END POINTER	7E68
CHANNEL 6 RCV DATA CURRENT POINTER	7E66
CHANNEL 6 RCV LOOK-UP TABLE START ADDRESS	7E64
CHANNEL 6 RCV FILTER TABLE START ADDRESS	7E62
CHANNEL 6 RCV DATA WORD COUNT REGISTER	7E60
CHANNEL 6 RCV BUFFER WRAPAROUND REGISTER	7E5E
CHANNEL 6 RCV WORD COUNTER TRIGGER	7E5C
CHANNEL 6 RESTART WILLIAMSBURG TRANSMIT	7E5A
CHANNEL 6 RCV INTERVAL COUNTER TRIGGER	7E58
CHANNEL 6 RCV ERROR COUNT REGISTER	7E56
CHANNEL 6 RCV LABEL TRIGGER REGISTER	7E54
CHANNEL 6 TX INSTRUCTION STACK POINTER	7E52
CHANNEL 6 TX INSTRUCTION COUNTER	7E50
CHANNEL 6 TX LOOP COUNTER	7E4E
CHANNEL 6 TX AMPLITUDE REGISTER	7E4C
CHANNEL 6 INTERRUPT/TRIGGER CONDITION REG.	7E4A
CHANNEL 6 STATUS REGISTER	7E48
CHANNEL 6 DEST CODE BUSY / BUFFER START	7E46
CHANNEL 6 FULL-DUPLEX / BUFFER END	7E44
CHANNEL 6 TX SAL REGISTER	7E42
CHANNEL 6 RCV SAL REGISTER	7E40

CHANNEL 7 CONTROL REGISTER BLOCK

CHANNEL 7 CONFIGURATION REGISTER	7E9E
reserved	7E9C
CHANNEL 7 RCV DATA START POINTER	7E9A
CHANNEL 7 RCV DATA END POINTER	7E98
CHANNEL 7 RCV DATA CURRENT POINTER	7E96
CHANNEL 7 RCV LOOK-UP TABLE START ADDRESS	7E94
CHANNEL 7 RCV FILTER TABLE START ADDRESS	7E92
CHANNEL 7 RCV DATA WORD COUNT REGISTER	7E90
CHANNEL 7 RCV BUFFER WRAPAROUND REGISTER	7E8E
CHANNEL 7 RCV WORD COUNTER TRIGGER	7E8C
CHANNEL 7 RESTART WILLIAMSBURG TRANSMIT	7E8A
CHANNEL 7 RCV INTERVAL COUNTER TRIGGER	7E88
CHANNEL 7 RCV ERROR COUNT REGISTER	7E86
CHANNEL 7 RCV LABEL TRIGGER REGISTER	7E84
CHANNEL 7 TX INSTRUCTION STACK POINTER	7E82
CHANNEL 7 TX INSTRUCTION COUNTER	7E80
CHANNEL 7 TX LOOP COUNTER	7E7E
CHANNEL 7 TX AMPLITUDE REGISTER	7E7C
CHANNEL 7 INTERRUPT/TRIGGER CONDITION REG.	7E7A
CHANNEL 7 STATUS REGISTER	7E78
CHANNEL 7 DEST CODE BUSY / BUFFER START	7E76
CHANNEL 7 FULL-DUPLEX / BUFFER END	7E74
CHANNEL 7 TX SAL REGISTER	7E72
CHANNEL 7 RCV SAL REGISTER	7E70

CHANNEL 8 CONTROL REGISTER BLOCK

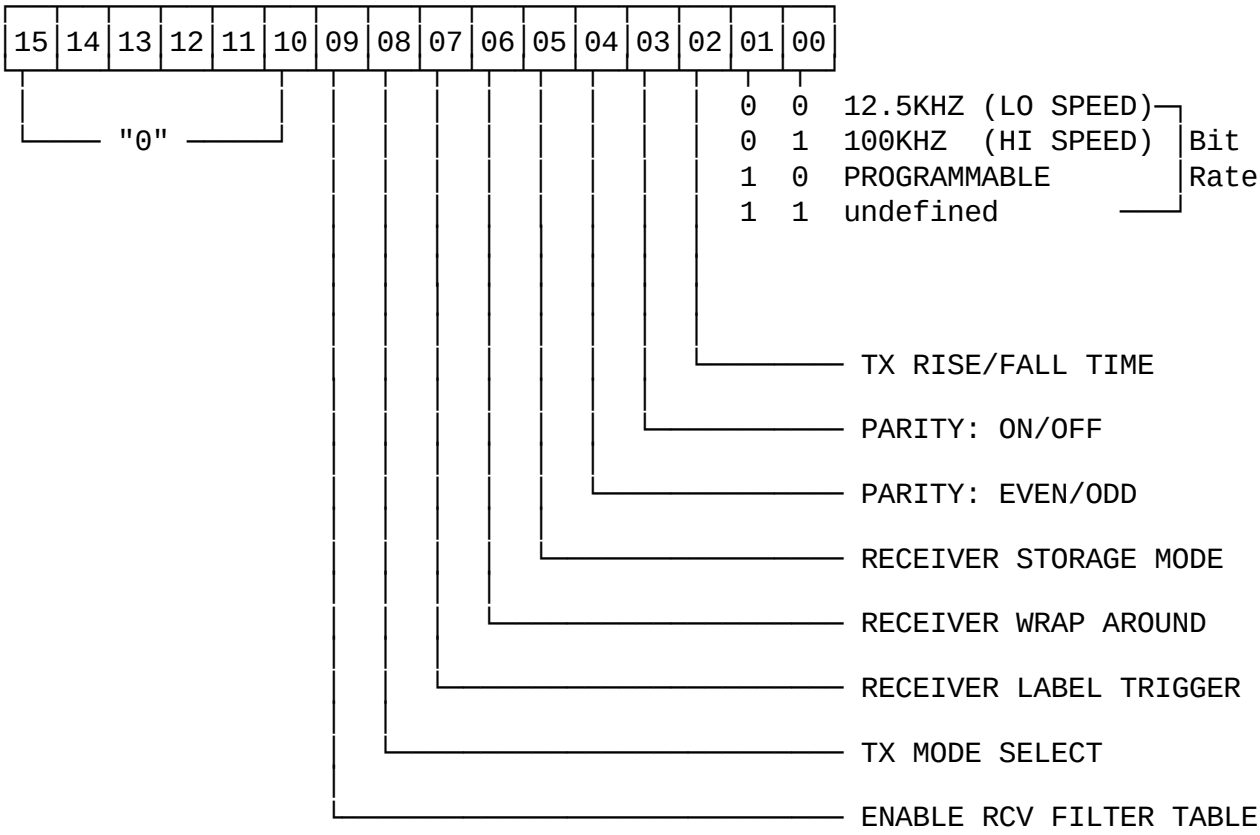
CHANNEL 8 CONFIGURATION REGISTER	7ECE
reserved	7ECC
CHANNEL 8 RCV DATA START POINTER	7ECA
CHANNEL 8 RCV DATA END POINTER	7EC8
CHANNEL 8 RCV DATA CURRENT POINTER	7EC6
CHANNEL 8 RCV LOOK-UP TABLE START ADDRESS	7EC4
CHANNEL 8 RCV FILTER TABLE START ADDRESS	7EC2
CHANNEL 8 RCV DATA WORD COUNT REGISTER	7EC0
CHANNEL 8 RCV BUFFER WRAPAROUND REGISTER	7EBE
CHANNEL 8 RCV WORD COUNTER TRIGGER	7EBC
CHANNEL 8 RESTART WILLIAMSBURG TRANSMIT	7EBA
CHANNEL 8 RCV INTERVAL COUNTER TRIGGER	7EB8
CHANNEL 8 RCV ERROR COUNT REGISTER	7EB6
CHANNEL 8 RCV LABEL TRIGGER REGISTER	7EB4
CHANNEL 8 TX INSTRUCTION STACK POINTER	7EB2
CHANNEL 8 TX INSTRUCTION COUNTER	7EB0
CHANNEL 8 TX LOOP COUNTER	7EAE
CHANNEL 8 TX AMPLITUDE REGISTER	7EAC
CHANNEL 8 INTERRUPT/TRIGGER CONDITION REG.	7EAA
CHANNEL 8 STATUS REGISTER	7EA8
CHANNEL 8 DEST CODE BUSY / BUFFER START	7EA6
CHANNEL 8 FULL-DUPLEX / BUFFER END	7EA4
CHANNEL 8 TX SAL REGISTER	7EA2
CHANNEL 8 RCV SAL REGISTER	7EA0

CHANNEL 9 CONTROL REGISTER BLOCK

CHANNEL 9 CONFIGURATION REGISTER	7EFE
reserved	7EFC
CHANNEL 9 RCV DATA START POINTER	7EFA
CHANNEL 9 RCV DATA END POINTER	7EF8
CHANNEL 9 RCV DATA CURRENT POINTER	7EF6
CHANNEL 9 RCV LOOK-UP TABLE START ADDRESS	7EF4
CHANNEL 9 RCV FILTER TABLE START ADDRESS	7EF2
CHANNEL 9 RCV DATA WORD COUNT REGISTER	7EF0
CHANNEL 9 RCV BUFFER WRAPAROUND REGISTER	7EEE
CHANNEL 9 RCV WORD COUNTER TRIGGER	7EEC
CHANNEL 9 RESTART WILLIAMSBURG TRANSMIT	7EEA
CHANNEL 9 RCV INTERVAL COUNTER TRIGGER	7EE8
CHANNEL 9 RCV ERROR COUNT REGISTER	7EE6
CHANNEL 9 RCV LABEL TRIGGER REGISTER	7EE4
CHANNEL 9 TX INSTRUCTION STACK POINTER	7EE2
CHANNEL 9 TX INSTRUCTION COUNTER	7EE0
CHANNEL 9 TX LOOP COUNTER	7EDE
CHANNEL 9 TX AMPLITUDE REGISTER	7EDC
CHANNEL 9 INTERRUPT/TRIGGER CONDITION REG.	7EDA
CHANNEL 9 STATUS REGISTER	7ED8
CHANNEL 9 DEST CODE BUSY / BUFFER START	7ED6
CHANNEL 9 FULL-DUPLEX / BUFFER END	7ED4
CHANNEL 9 TX SAL REGISTER	7ED2
CHANNEL 9 RCV SAL REGISTER	7ED0

CHANNEL x CONFIGURATION REGISTER

This register sets up various run parameters for both the receive and transmit channels. Bits which are unused (ie receiver-related bits while operating as a transmitter) are ignored by the board.



Notes:

1. This register can only be written to when ALL channels are turned OFF (via the Start Register).

2. The board should be started (via the Start Register) only after a minimum of 1 msec from the time that the contents of this register have been modified.

3. It is recommended that ALL active channel Configuration Registers be set up immediately following the Board Ready before programming any other parameters.

4. ARINC-561/568/575/582 specifications specify Lo-Speed (Bit Rate) operation only.

CHANNEL x CONFIGURATION REGISTER; BIT DEFINITIONS

Bit Rate	Select Bit Rate of channel. If the Programmable Bit Rate is selected then the bit rate is defined by the Global Programmable Bit Rate Register.
Tx Rise/Fall Time	`0' = Hi Speed (1.5 +/- 0.5usec) `1' = Lo Speed (10 +/- 5 usec)
Parity On/Off	`1'= Off `0' = On
Parity Even/Odd	`1'= Even `0' = Odd (standard ARINC mode)
Rcv Storage Mode	`1'=Sequential Storage Mode `0'= Look-up Table Mode
Receiver Wrap Around	`1' = Data storage is halted when the buffer is full `0' = Receiver wraps around the data within the block (This bit is not used in Lookup Table Storage Mode)
Receiver Label Trigger	`1' = Start data storage upon receipt of Label xx. `0' = Receiver stores data without Start Label Trigger
TX Mode Select (see note)	`1' = Data Rate mode (per data block). See text. `0' = Interblock Gap Time mode
Enable Rcv Filter Table	`1' = Enable filter table (stores Labels per table) `0' = Disables table. Stores all Labels.

Notes:

1. TX Mode Select: This bit allows the user to select the transmission mode. A logic `1' instructs the transmitter to use the Interblock Time/Data Rate Word as a Data Rate value and to send data blocks on a scheduled data rate basis (ie Data Block 1; every 50 ms and Data Block 2; every 25 ms, etc.). In this mode, a scratch buffer must be allocated via the Channel Buffer Start and Channel Buffer End Registers. This mode is not supported by Williamsburg channels. A logic `0' instructs the transmitter to use the Interblock Time/Data Rate Word as an Inter-block Gap Time.

2. Receiver Label Trigger: See description of the RCV Start Label Trigger Register.

3. Tx Rise/Fall Time bit: Not applicable for ARINC-561/568/582-6wire channels.

CHANNEL x RECEIVE DATA START POINTER (WR)

This register is used to set the start address of the receive data buffer. The address must be even. This register is used in the Sequential Mode of operation.

CHANNEL x RECEIVE DATA END POINTER (WR)

Sets the end address of the receiver data buffer. This is used in the Sequential Mode of operation. The data will wrap around or stop - depending upon the Receiver Wrap Around control bit within the Configuration Register.

CHANNEL x RECEIVE DATA CURRENT POINTER (RD)

In Sequential Mode this register indicates the address where the next ARINC receive word is to be placed within the buffer. This pointer value is incremented after the entire receiver block is written into memory.

In Lookup Table Mode this register contains the address of the last ARINC receive word written to the receive data area.

CHANNEL x RECEIVE LOOK-UP TABLE START ADDRESS (WR) LOOK-UP TABLE MODE

Sets the Start address (must be even) of the 256x16 byte Receiver Look-Up Table. This address points to the FIRST location of the look-up table. The board will store ONE ARINC data block for each Label received. The data block contains:32-bit ARINC word, 32-bit Time Tag, Error Count Word, and the Receive Status Word. The data block will be overwritten by the subsequent reception and storage of the same ARINC Label.

CHANNEL x RECEIVE FILTER TABLE START ADDRESS (WR) SEQUENTIAL MODE

Sets the start address of the (256x8) Label Filter Table as described in the Sequential storage mode. The address must be even. It is valid for several channels to use the same filter table.

CHANNEL x RCV DATA WORD COUNT REGISTER (RD) SEQUENTIAL MODE

Indicates the number of words received (0-64k). This register wraps around, and may be reset to zero by the user only when the channel is stopped.

CHANNEL x RCV BUFFER WRAPAROUND REGISTER (WR)

This register contains 2 bits for synchronization with the "C" drivers. If bit 14 is set to 1, the receive buffer has wrapped around once since the last data read. If bit 15 is set to 1, there have been multiple wraparounds.

CHANNEL x RCV DATA WORD COUNTER TRIGGER REGISTER (WR) SEQUENTIAL MODE

This register allows the user to generate an interrupt and set a flag when a certain number of words have been received (1-64k). The appropriate bit must also be set in the Channel x Interrupt/Trigger Condition Register.

CHANNEL x RESTART WILLIAMSBURG TRANSMISSION (WR)

Writing any non-zero value to this register will reinitiate transmission in a Williamsburg channel. It allows the user to reload new messages after the termination of transmission and to transmit them without interfering with the receive function. The register is automatically cleared to zero.

CHANNEL x RCV INTERVAL COUNTER TRIGGER REGISTER (WR) SEQUENTIAL MODE

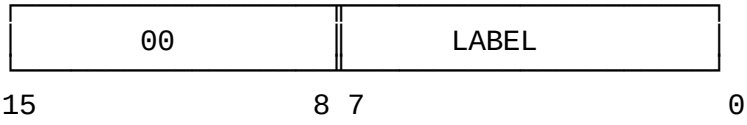
This register allows the user to generate an interrupt and set a flag upon reception of every "N" number of words. The appropriate bit must also be set in the Channel x Interrupt/Trigger Condition Register.

CHANNEL x RCV ERROR COUNT REGISTER (RD) SEQUENTIAL MODE

This 16-bit counter indicates the number of errors received on a particular channel. This register wraps around. The user can reset this register by writing `0000' to it.

CHANNEL x RCV LABEL TRIGGER REGISTER (WR) SEQUENTIAL MODE

This register is used in conjunction with the Receiver Label Trigger bit within the Configuration Register to enable the reception and storage of data upon receipt of a specific ARINC label; xxxx xxxx.



CHANNEL x TX INSTRUCTION STACK POINTER (WR)

Sets the starting address of the TX Instruction Stack. This address must be even. For a Williamsburg channel which is being started that has no data to transmit, the stack pointer should be given a value of FFFF (H) (all ones).

CHANNEL x TX INSTRUCTION COUNTER (WR)

Sets the number of TX Instruction blocks to process.

CHANNEL x TX LOOP COUNTER (WR)

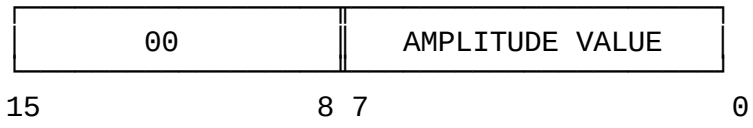
Sets the number of times to execute the TX instruction blocks; `N' Times or Continuous Loop. If the continuous value is selected, the channel's operation can be terminated by setting the related channel bit within the Global Start Register to a "0".

Value:

0000 = CONTINUOUS
0001 = One Time
0002 = Two Times
.
.
FFFF = 65535 Times

CHANNEL x TX AMPLITUDE REGISTER (WR)

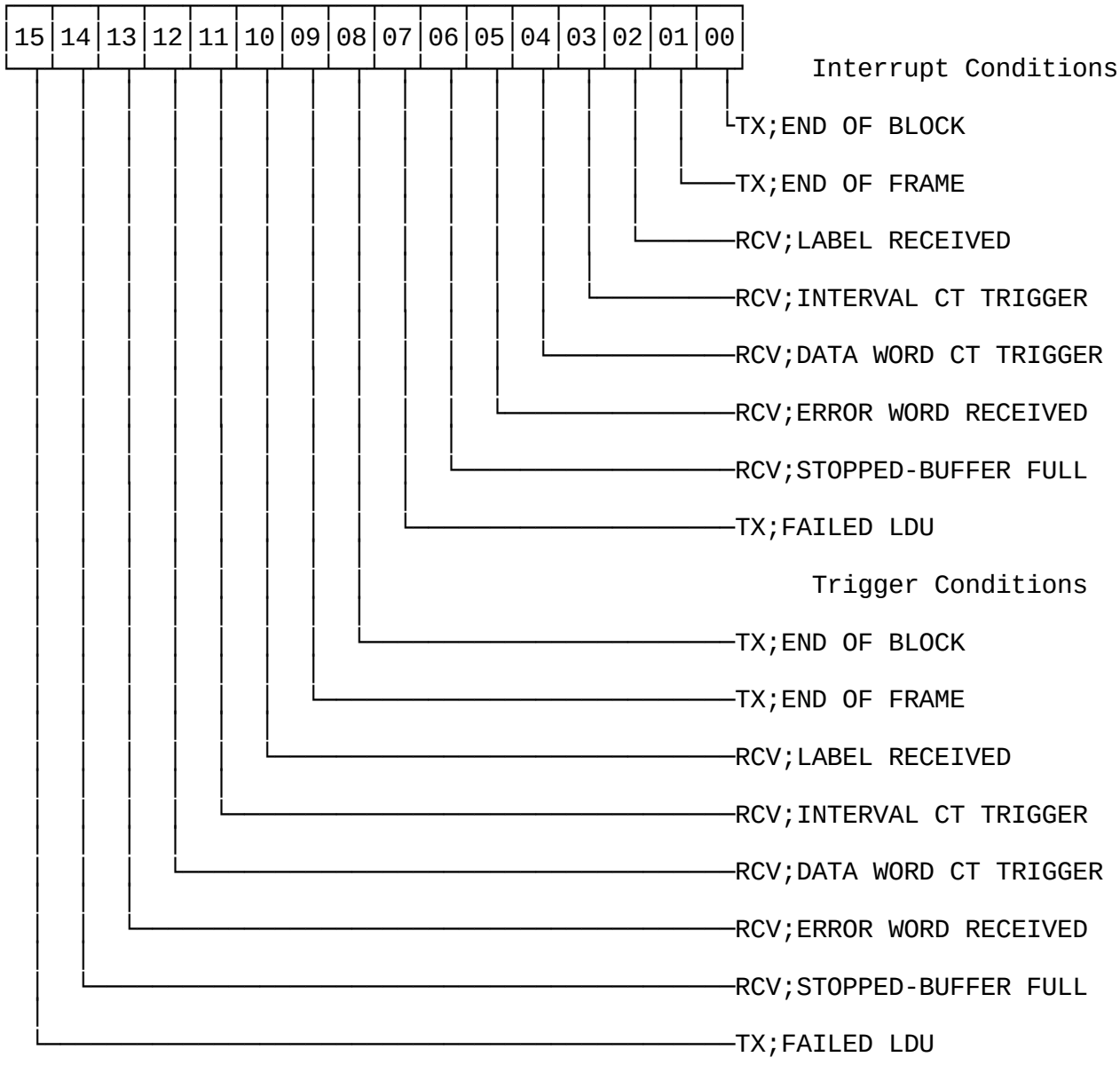
Sets the TX amplitude level of the TX channel. This register has a resolution of 39mv/bit. The maximum register value gives a voltage of 10V(peak) measured across the ARINC bus differentially. At startup and after a software reset this register is initialized to FF (H) which is the maximum value.



Note: This register is not applicable for ARINC 561/568/582-6wire channels.

CHANNEL x INTERRUPT/TRIGGER CONDITION REGISTER (WR)

Sets the Interrupt and Trigger condition(s) of the board. Bits 00-07 are the interrupt condition bits while bits 08-15 relate to the hardware trigger bits. The trigger conditions set a pulse on the trigger [subminiature BNC] J1 connector or the VXI "TRIG0$" TTL Trigger Output signal.



Notes:

- The RCV;LABEL RECEIVED interrupt or trigger only occurs upon reception of a label which has been marked for interrupt in a filter table or lookup table.

- To activate the RCV;INTERVAL CT TRIGGER interrupt or trigger, the Channel x Rcv Interval Counter Trigger Register must also be set.

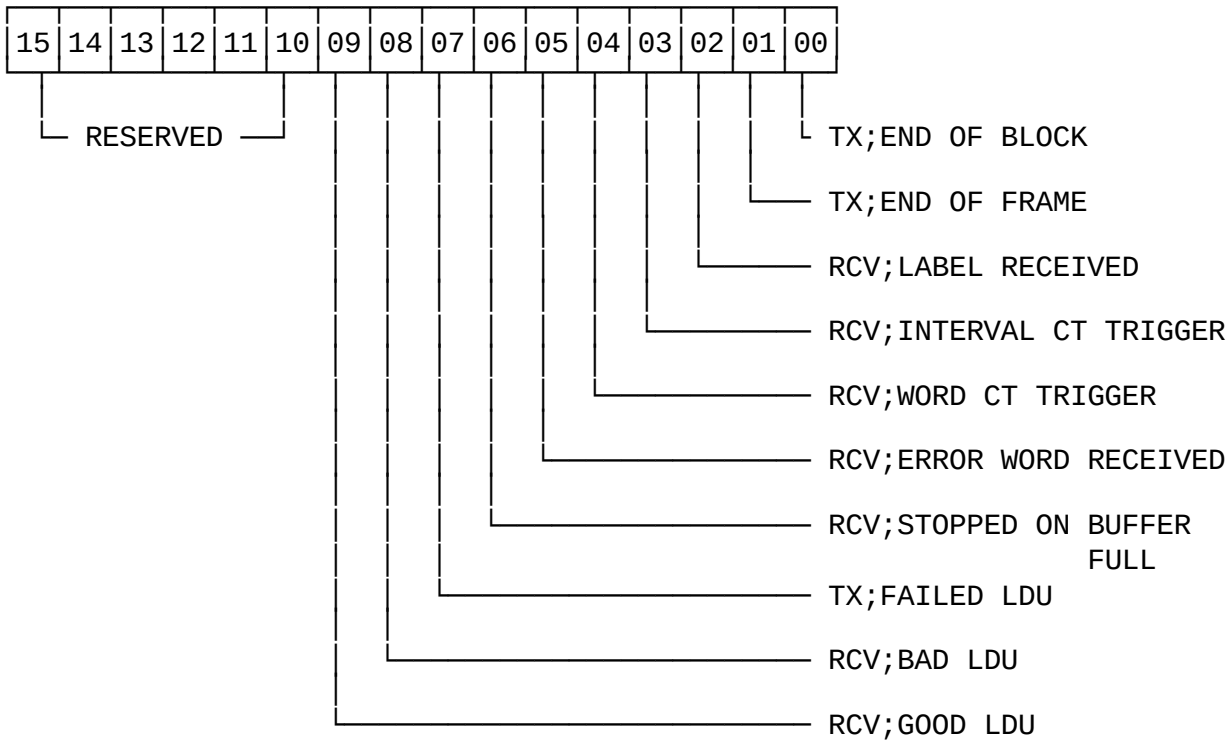
- To activate the RCV;DATA WORD CT TRIGGER interrupt or trigger, the

Channel x Rcv Word Counter Trigger Register must also be set.

- The TX;FAILED LDU bit is relevant to Williamsburg channels only.

CHANNEL x STATUS REGISTER (RD)

This register indicates the operational status of the channel. This register can be used to poll the status of the channel or it can be used with interrupts. When used in conjunction with interrupts the register indicates the condition(s) which caused the interrupt. A logic "1" indicates an active bit. Status bits must be reset by the user.



In Look Up Mode, the RCV;LABEL RECEIVED Status bit is set upon receipt of any label for which an interrupt has been requested via the label's Control Byte. In Sequential Mode, it is set upon receipt of any label for which an interrupt has been requested via the filter table.

The TX;LDU FAILED Status bit is relevant only to Williamsburg channels. It is set when a block was not successfully transmitted due to timeouts, or lack of the desired response. It is relevant to blocks containing a Williamsburg TEST word, an ALO word, or an LDU (blocks requiring responses).

The RCV;BAD LDU and RCV;GOOD LDU bits are relevant only to Williamsburg channels. They are set according to the quality of received LDU's and are not associated with an interrupt. Conditions which cause the BAD LDU bit to be set are bad word

count, bad CRC, missing SOT, timeout, sent response of BUSY, and LDU reception aborted due to reception of another RTS or ALO.

CHANNEL x DEST CODE BUSY / BUFFER START (WR)

For Williamsburg channels, it is possible to force a response of BUSY to an RTS for test purposes by using this register. If the (non-zero) value of this register matches the destination code of the received RTS, the channel will respond with BUSY instead of the normal CTS.

For transmission in Data Rate Mode, the user must allocate a scratch buffer for use by the firmware. Its length must be at least (TX INSTRUCTION COUNTER x 10 + 4) bytes. If the buffer is not long enough, the channel will turn itself off without transmitting. This register contains the start address of the buffer. The address must be a word boundary.

CHANNEL x FULL-DUPLEX / BUFFER END (WR)

For Williamsburg channels, it is possible to operate in full-duplex or half-duplex mode, as described below. Writing a 1 to this register configures the channel for full-duplex operation. Writing a 0 results in half-duplex operation.

For transmission in Data Rate Mode, this register contains the end address of the buffer which must be assigned by the user.

CHANNEL x TX SAL REGISTER (WR)

This register is relevant to Williamsburg channels only. This register contains the SAL (system address label) of the destination for transmitted Williamsburg blocks. Every word contained in a block of Williamsburg format data to be transmitted must contain this SAL in its label field. This value is used for constructing responses (CTS, NCTS, BUSY, ACK, NAK, SYN, ALR, and LOOP).

CHANNEL x RCV SAL REGISTER (WR)

This register is relevant to Williamsburg channels only. This register contains the SAL of the label. Only received words which contain this SAL in their label field are considered to be in the Williamsburg format. Words not containing this SAL are taken to be ARINC-429 words with labels.

RECEIVER / MONITOR OPERATION

GENERAL INFORMATION:

The user sets up each receiver channel's mode of operation by writing to the various Channel Control Registers (one set per channel). Each receiver channel has three basic modes of operation; Sequential mode which stores data in sequential locations within the receive data area, the Look-up table mode which allows the user to store words in specific locations of memory according to the Label, and the Merge Mode which syphons ALL receiver channel data into ONE receiver buffer area. In all these modes, the data words are stored with a 16-bit Receiver Status Word and a 32-bit time tag value. In data only mode the status and time tag is not stored. In the Merge Mode, the channel ID information (indicating on which channel the data was received) is contained within the Receiver Status Word.

OPERATION:

The board is initialized in a wait loop - looking for a START command from the computer. This command, issued by writing to the Global Start Register, instructs the board to begin operation on the active channel(s).

SEQUENTIAL MODE:

The sequential mode has a software-selectable feature which filters the storage of specific, user-selected Labels or stores all Labels within a buffer. The data buffer's size and location within the memory is programmed via a Start and End pointer. Each received ARINC data word is tagged with a Status word indicating the status of the receive word and a 32-bit Time Tag value.

These five, 16-bit words make up a single receive data block. In addition, the sequential mode offers the user the capability of storing just the ARINC data without the Time Tag and Status Words. This is global selection which affects all receiver channels (see: Receiver Data Storage Mode Register in the Global Register section of this manual). A 16-bit Error Count register indicates the number of invalid words received. Interrupts and pollable status registers allow for numerous event recognition and are described in the Channel Register section of this manual.

The Sequential Storage Mode is the only storage mode supported by Williamsburg channels. Received Williamsburg words which are responses to transmissions are stored sequentially with received data, in order of arrival. If a received LDU must be repeated due to an error the defective LDU will not be overwritten, but rather the retransmitted LDU will be stored after the defective version.

ARINC Channels

LOOK-UP TABLE MODE:

In the Look-up table mode, the word's LABEL is used by the board as an offset to a 256-word look-up table. The table is programmed by the user with address pointers as to where to write the Receiver Data Block. Each block contains the 32-bit ARINC word, 32-bit Time Tag, an error count, and status word. The 256-word table can be placed anywhere within the memory via a user-programmable Receiver Look-up Table Pointer.

The user has the ability to monitor the operational status of each channel and to be interrupted on various events. In addition, there exist pollable registers which can be used in conjunction with or instead of interrupt processing.

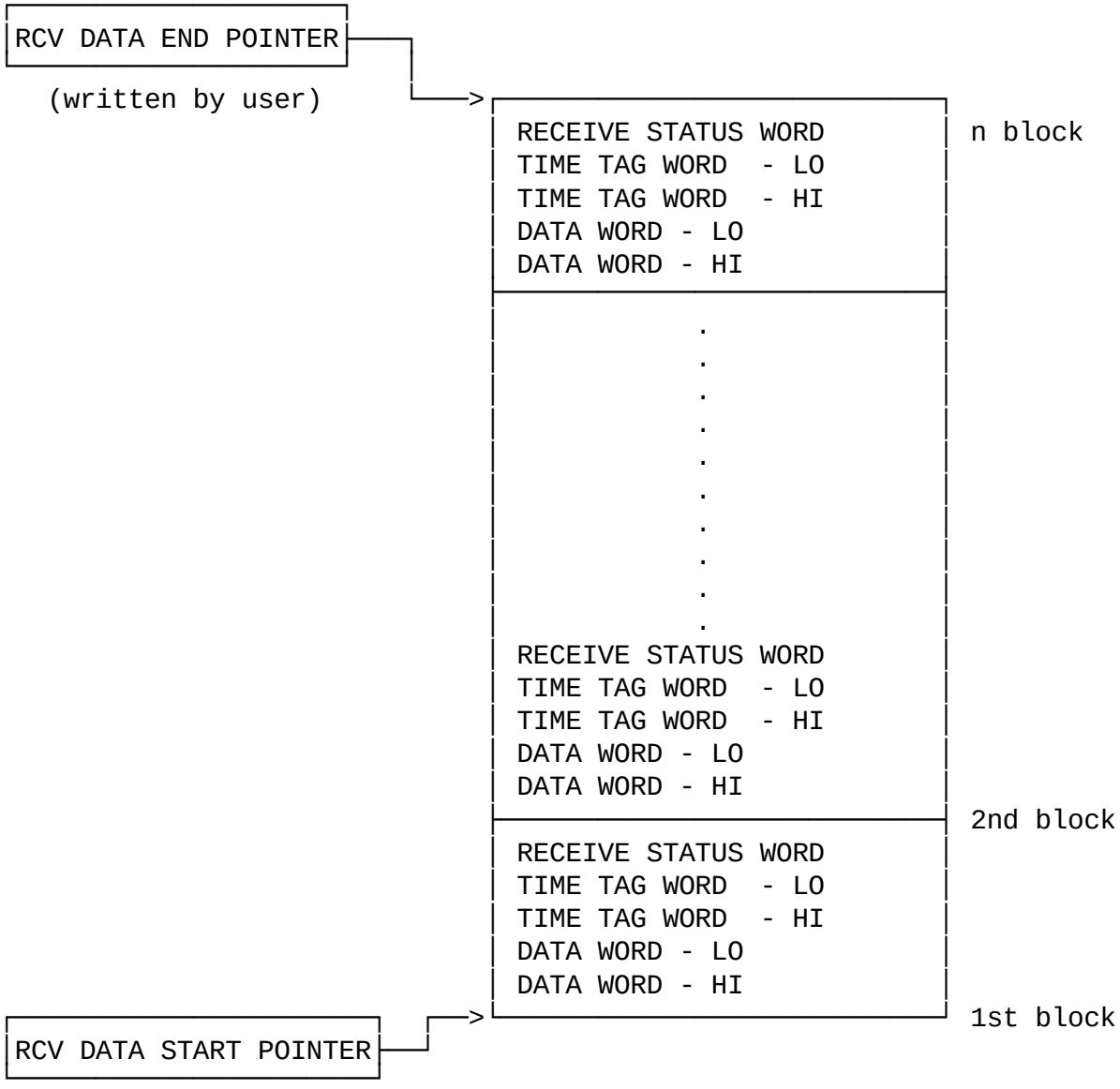
MERGE MODE:

The Merge Mode operates in the same manner as the Sequential Mode except that all receive channels are merged into one data buffer area. The control registers for the Merge Mode are located and defined in the global register section. In this mode the receive data blocks are stored in sequential order and each receive Status Word is tagged with a Channel ID - indicating on which channel the data was received. Each data block contains a Time Tag word as in the standard Sequential Mode of operation.

SEQUENTIAL MODE

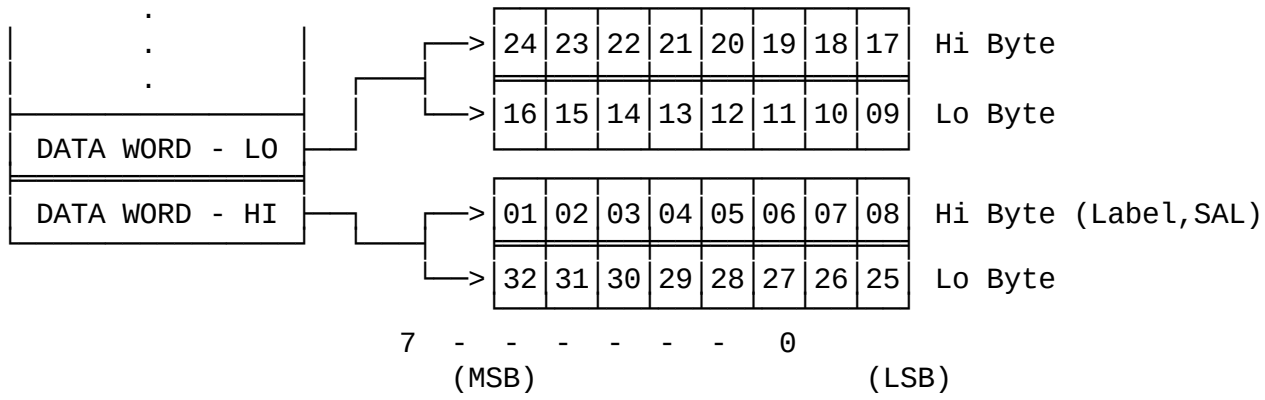
RECEIVE BUFFER STORAGE SEQUENCE

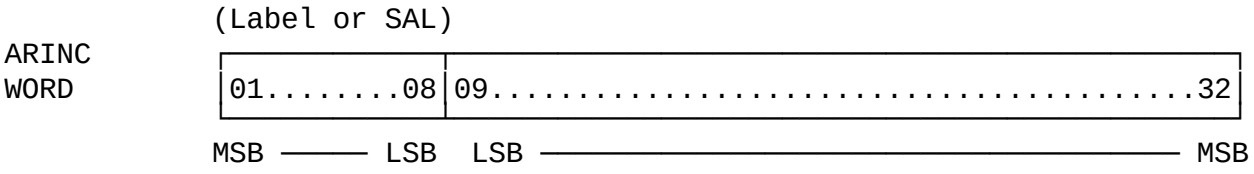
The drawing below illustrates the way in which the receive data blocks are stored within the dual-port RAM while in the sequential mode of operation. The Receiver Channel Error Count Register is updated with every invalid word which is stored. The Start and End pointers set up the buffer size. The receive data storage will stop when the end pointer is reached or will wrap around to the beginning of the buffer - depending upon the condition of the Receiver Wrap Around bit within the Channel Configuration Register. The Time Tag resolution is 10 µsec/bit. The Time Tag is set to zero upon power up or software reset. The contents of the Receiver Status Word are described within this section.



RCV DATA WORD FORMAT

The received, 32-bit, ARINC word is stored as two 16-bit words within the memory (Hi-Word followed by Lo-Word). The data bytes are shown below. The numbers shown within the four bytes represent the ARINC bit locations within the 32-bit word.



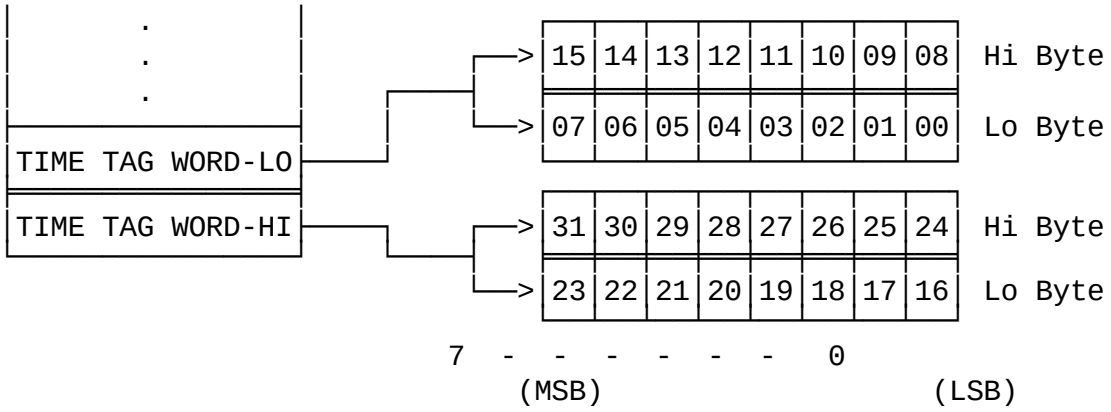


Note that bits 09 through 32 are ordered from LSB to MSB (opposite from the Label field which is organized MSB to LSB). It is for this reason that the data block is built the way it is (Hi-Word followed by Lo-Word) with the Label and the ARINC field 32 through 25 in the Hi-Word and bits 24 through 09 in the Lo-Word.

In Williamsburg words the label field is replaced by the system address label (SAL).

TIME TAG WORD FORMAT

As stated above, the Time Tag is a 32-bit word made up of two 16-bit words; Time Tag-Hi followed by Time Tag-Lo. The resolution of the time tag is 10 μsec/bit.

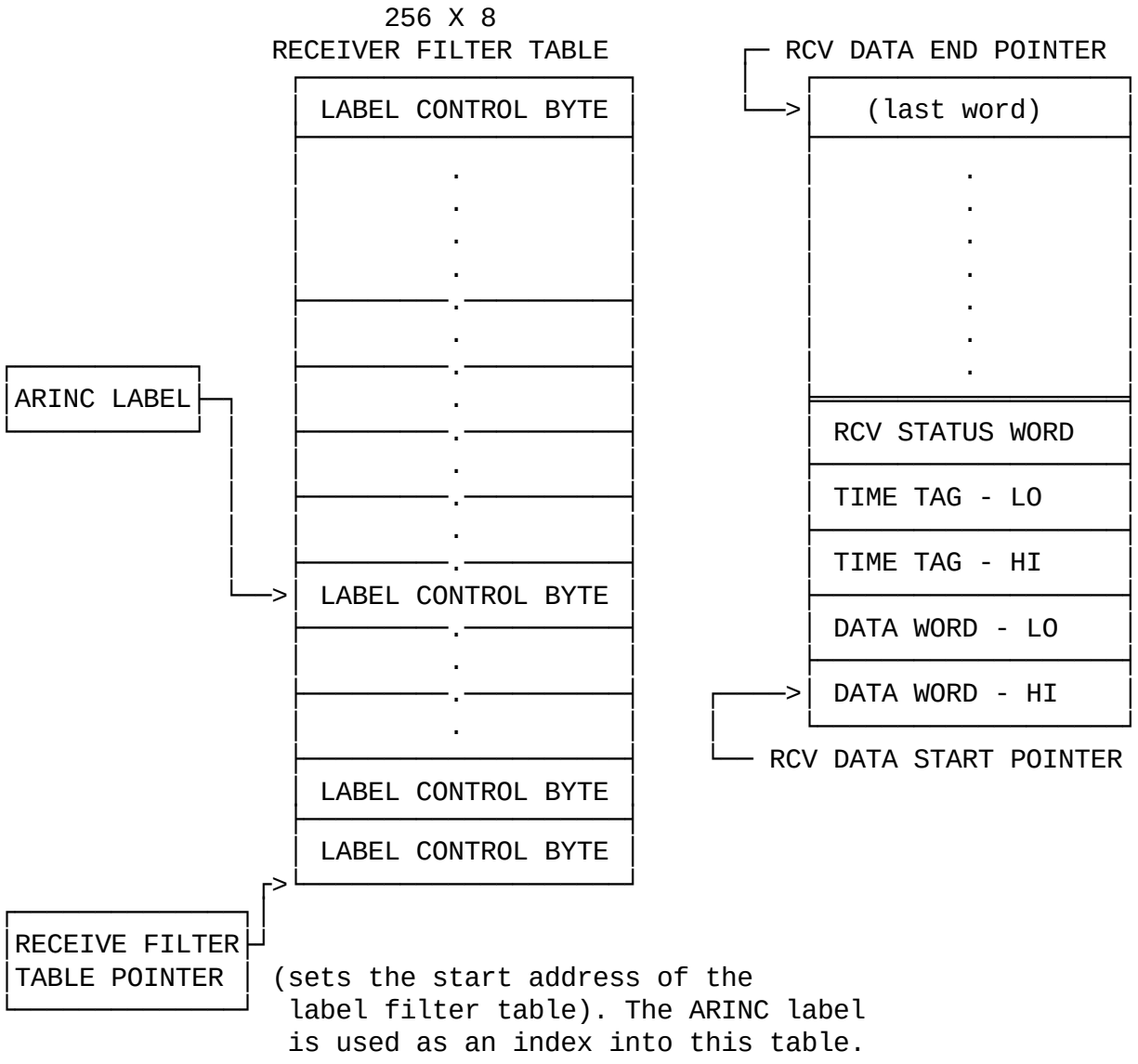


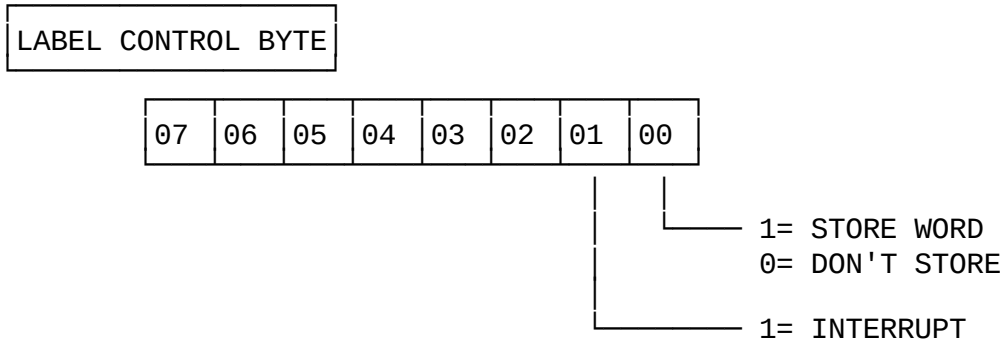
Note:

Each bit in the receive time tag represents 10 microseconds. There is a latency between the time a word is received on the bus and the time that word is recorded in dual port ram. This latency is affected by the number of channels and data rate of these channels. The time tag reflects the time the word is written to dual port ram rather than the time the word is received over the bus. In no event shall this latency exceed a single hi-speed word time e.g. 360 microseconds.

RCV SEQUENTIAL MODE DIAGRAM

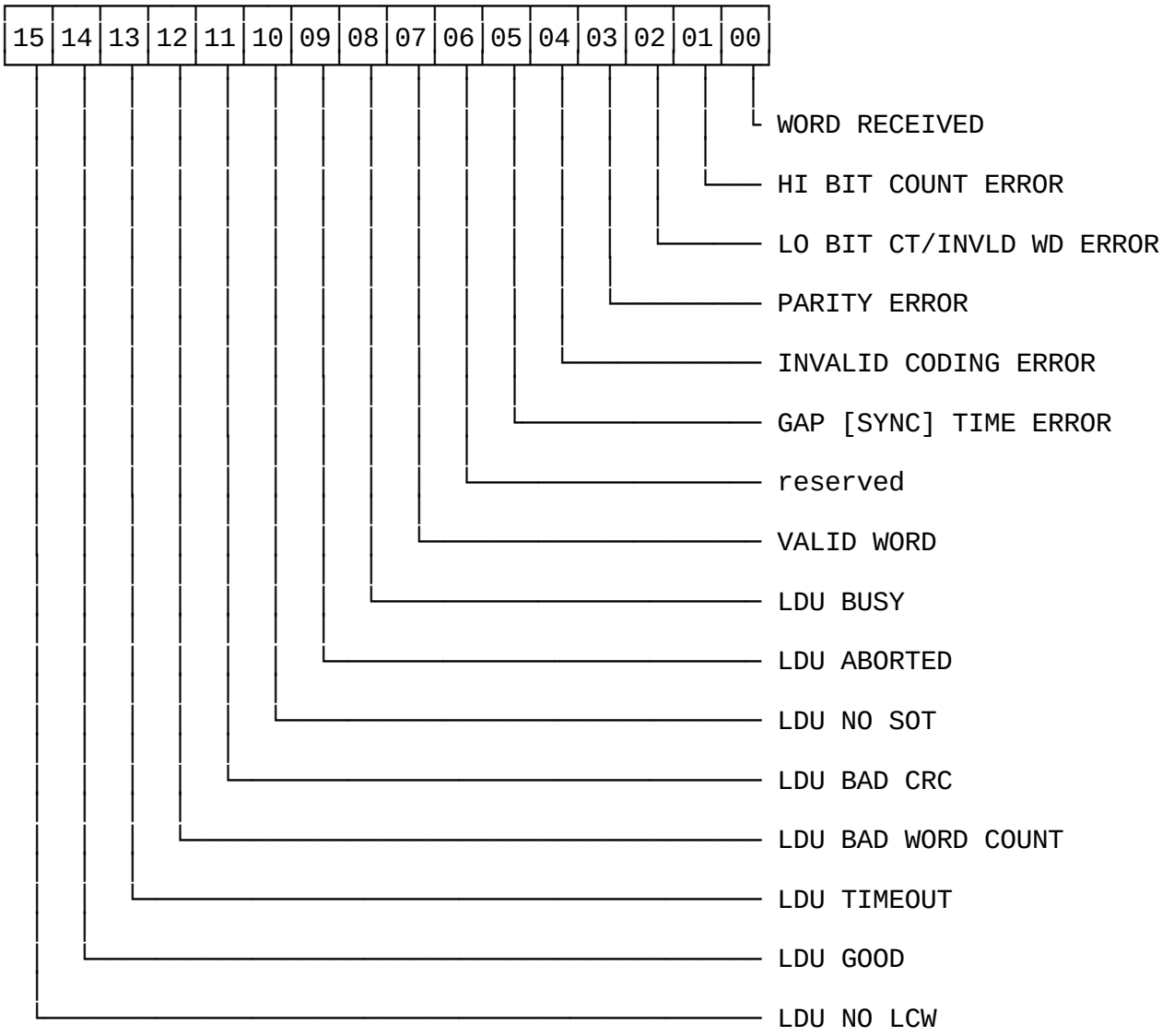
This diagram illustrates ONE receiver channel.





0= DON'T INTERRUPT

RECEIVE SEQUENTIAL MODE; STATUS WORD

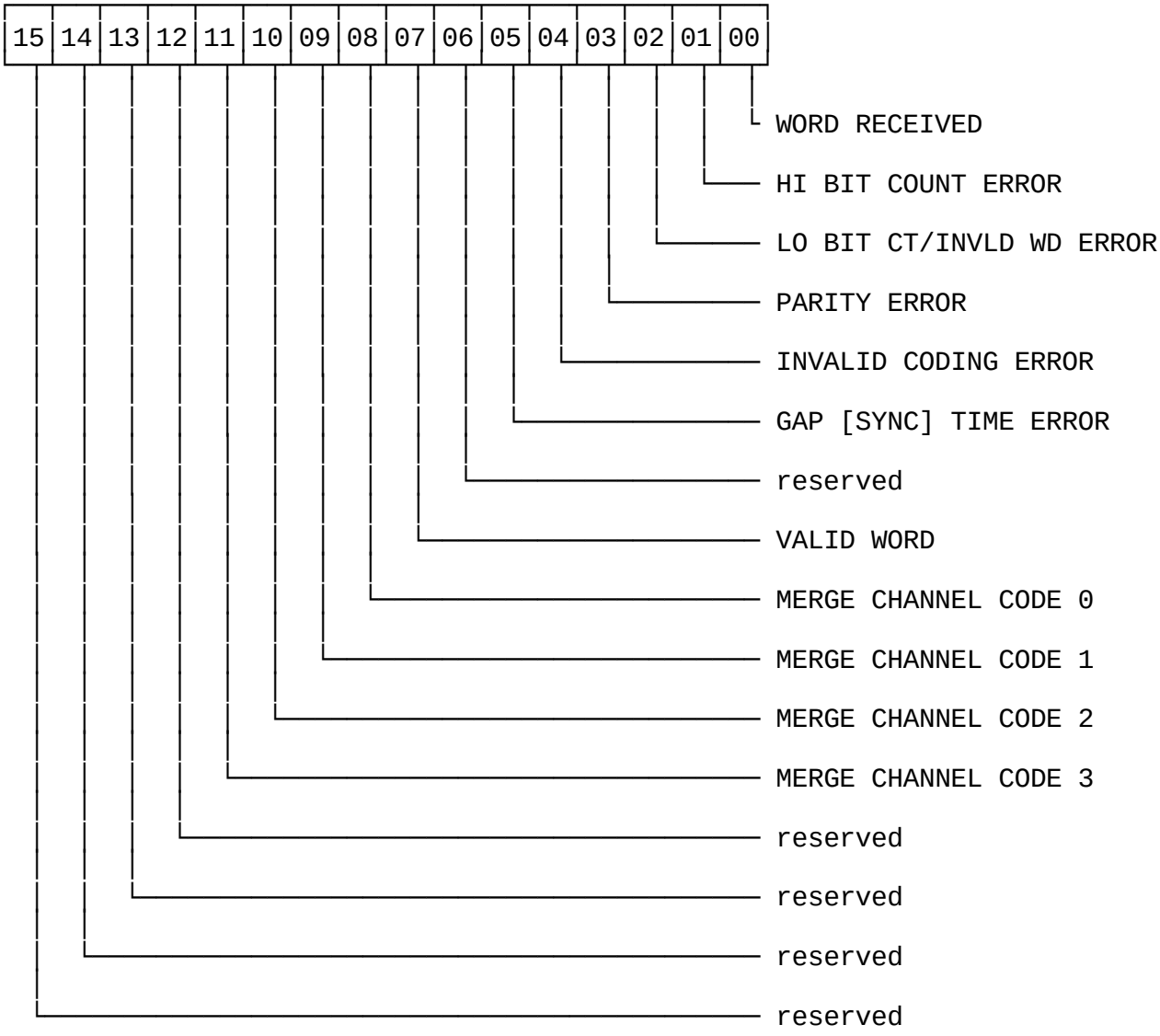


Note: bits 08 - 15 are relevant to Williamsburg channels only. They appear in the Status Word of RTS words only and are set when the reception of the associated LDU is completed.

RECEIVE SEQUENTIAL MODE STATUS WORD; BIT DEFINITION

00	Word Received	A logic `1'. This bit is set when the status word is written into memory.
01	Hi Bit Ct Error	Indicates that a Hi-bit count error was detected in the ARINC word.
02	Lo Bit Ct Invalid Wd	Indicates that a Lo-bit count or a Null bit error was detected in the ARINC word.
03	Parity Error	Indicates that a parity error was detected in the ARINC word.
04	Invalid Coding Err	Indicates that a bit level decoding error was detected in the ARINC word.
05	Gap [Sync] Time Error	Indicates that a Gap [Sync] Time error occurred between words (less than 4 bit times between words for ARINC 429/575/582-2wire channels and less than 1 bit time between words for ARINC 561/568/582-6wire channels).
06	reserved	SET TO "0"
07	Valid Word	Global Bit. Indicates that the received ARINC word was valid in all respects.
08	LDU Busy	A response of BUSY was sent to received RTS
09	LDU Aborted	Reception aborted due to arrival of a second RTS or an ALO
10	LDU No SOT	A data word or EOT word was received when an SOT word was expected
11	LDU Bad CRC	A bad CRC was received in the EOT word
12	LDU Bad Word Count	An EOT was received when a data word was expected or the received RTS contained an illegal word count
13	LDU Timeout	An EOT word was not received before the timeout interval elapsed
14	LDU Good	The received LDU is valid
15	LDU No LCW	A data word or SOT or EOT was received when an LCW word was expected in a Buckhorn window

RECEIVE "MERGE MODE" STATUS WORD



RECEIVE "MERGE MODE" STATUS WORD; BIT DEFINITION

00	Word Received	A logic `1'. This bit is set when the status word is written into memory.
01	Hi Bit Ct Error	Indicates that a Hi-bit count error was detected in the ARINC word.
02	Lo Bit Ct Invalid Wd	Indicates that a Lo-bit count error or a Null bit error was detected in the ARINC word.
03	Parity Error	Indicates that a parity error was detected in the ARINC word.
04	Invalid Coding Err	Indicates that a bit level decoding error was detected in the ARINC word.
05	Gap [Sync] Time Error	Indicates that a Gap [Sync] Time error occurred between words (less than 4 bit times between words for ARINC 429/575/582-2wire channels and less than 1 bit time between words for ARINC 561/568/582-6wire channels).
06	reserved	SET TO "0"
07	Valid Word	Global Bit. Indicates that the received ARINC word was valid in all respects.
08	Merge CH. ID Code 0	0000 - Channel 0 0001 - Channel 1 0010 - Channel 2 0011 - Channel 3 0100 - Channel 4 0101 - Channel 5 0110 - Channel 6 0111 - Channel 7 1000 - Channel 8 1001 - Channel 9 Note: CH ID Code 0=LSB " " " 3=MSB
09	Merge CH. ID Code 1	
10	Merge CH. ID Code 2	
11	Merge CH. ID Code 3	
12	reserved	SET TO "0"
13	reserved	SET TO "0"
14	reserved	SET TO "0"
15	reserved	SET TO "0"

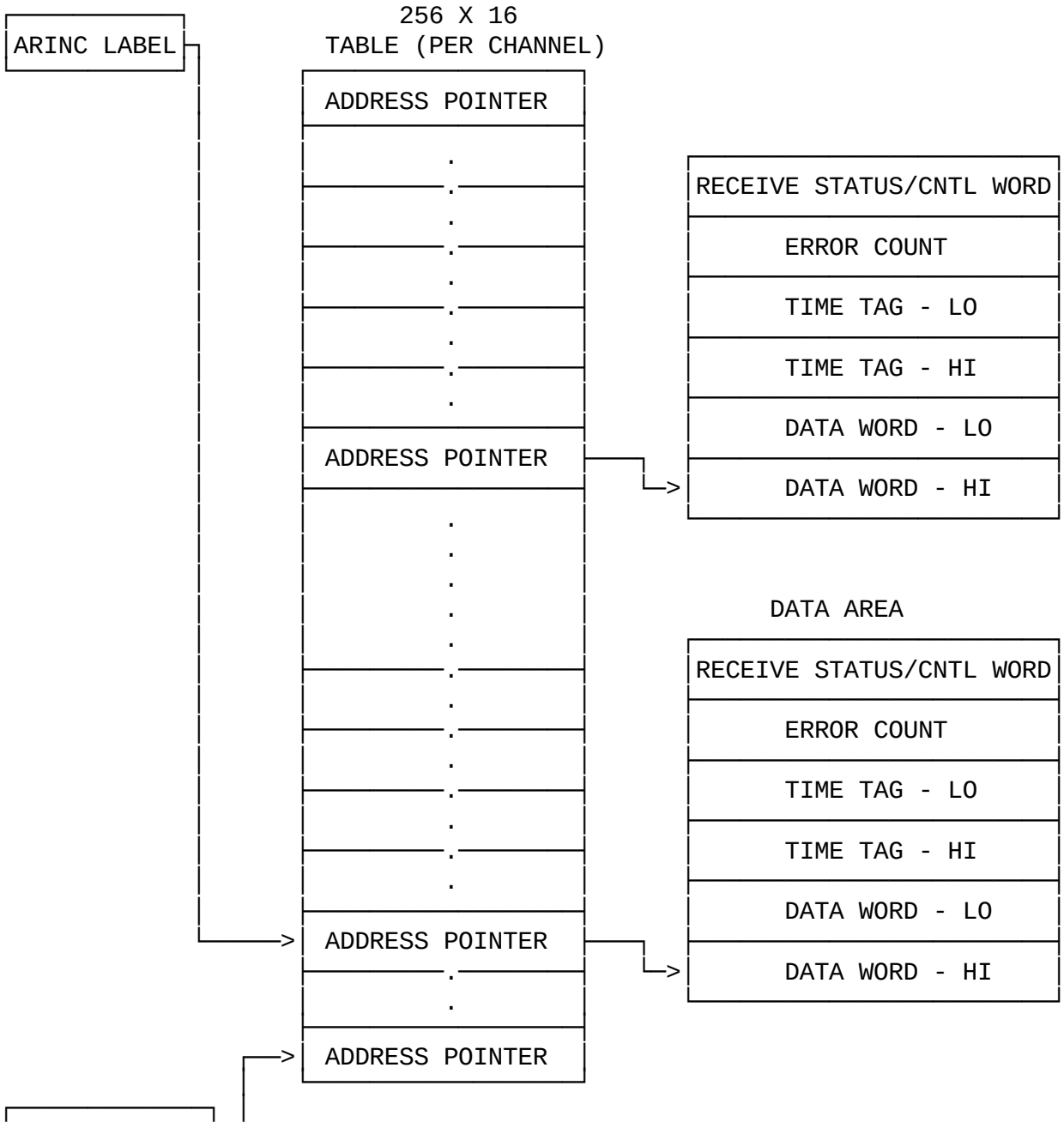
ARINC Channels

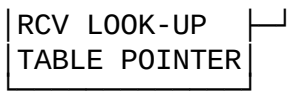
LOOK-UP TABLE MODE OPERATION

LOOK-UP TABLE MODE:

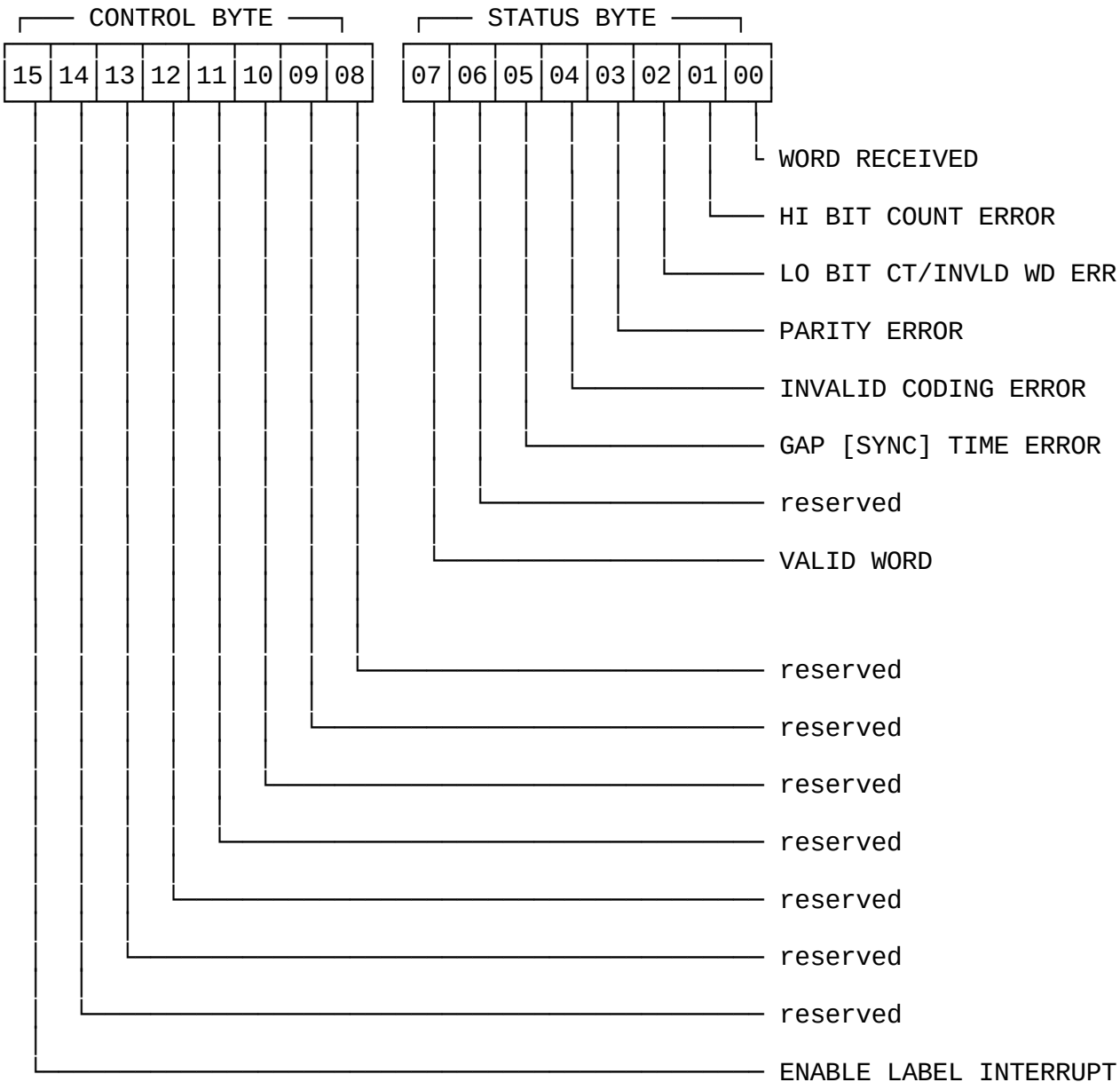
In the Look-up table mode, the word's LABEL is used by the board as an offset to a 256-word look-up table. The table is programmed by the user with address pointers as to where to write the Receiver Data Block. Each block contains the 32-bit ARINC word, 32-bit Time Tag, an error count, and status word. The 256-word table can be placed anywhere within the memory via a user-programmable Receiver Look-up Table Pointer.

The user has the ability to monitor the operational status of each channel and to be interrupted on various events. In addition, there exist pollable registers which can be used with or instead of interrupt processing.




RCV LOOK-UP
TABLE POINTER

LOOK-UP TABLE; RECEIVE STATUS/CONTROL WORD (RD/WR)



LOOK-UP TABLE; RECEIVE STATUS/CONTROL WORD

00	Word Received	A logic `1'. This bit is set when the status word is written into memory.
01	Hi Bit Ct Error	Indicates that a Hi-bit count error was detected in the ARINC word.
02	Lo Bit Ct Invalid Wd	Indicates that a Lo-bit count error or a Null bit error was detected in the ARINC word.
03	Parity Error	Indicates that a parity error was detected in the ARINC word.
04	Invalid Coding Err	Indicates that a bit level decoding error was detected in the ARINC word.
05	Gap [Sync] Time Error	Indicates that a Gap [Sync] Time error occurred between words (less than 4 bit times between words for ARINC 429/575/582-2wire channels and less than 1 bit time between words for ARINC 561/568/582-6wire channels).
06	reserved	SET TO "0"
07	Valid Word	Global Bit. Indicates that the received ARINC word was valid in all respects.

08	reserved	SET TO "0"
09	reserved	SET TO "0"
10	reserved	SET TO "0"
11	reserved	SET TO "0"
12	reserved	SET TO "0"
13	reserved	SET TO "0"
14	reserved	SET TO "0"
15	Enable Label Int.	Enables the`interrupt on Label received' capability. This bit is used in conjunction with the Interrupt/ Trigger Condition Register.

ARINC Channels

TRANSMITTER OPERATION - NON-WILLIAMSBURG CHANNELS

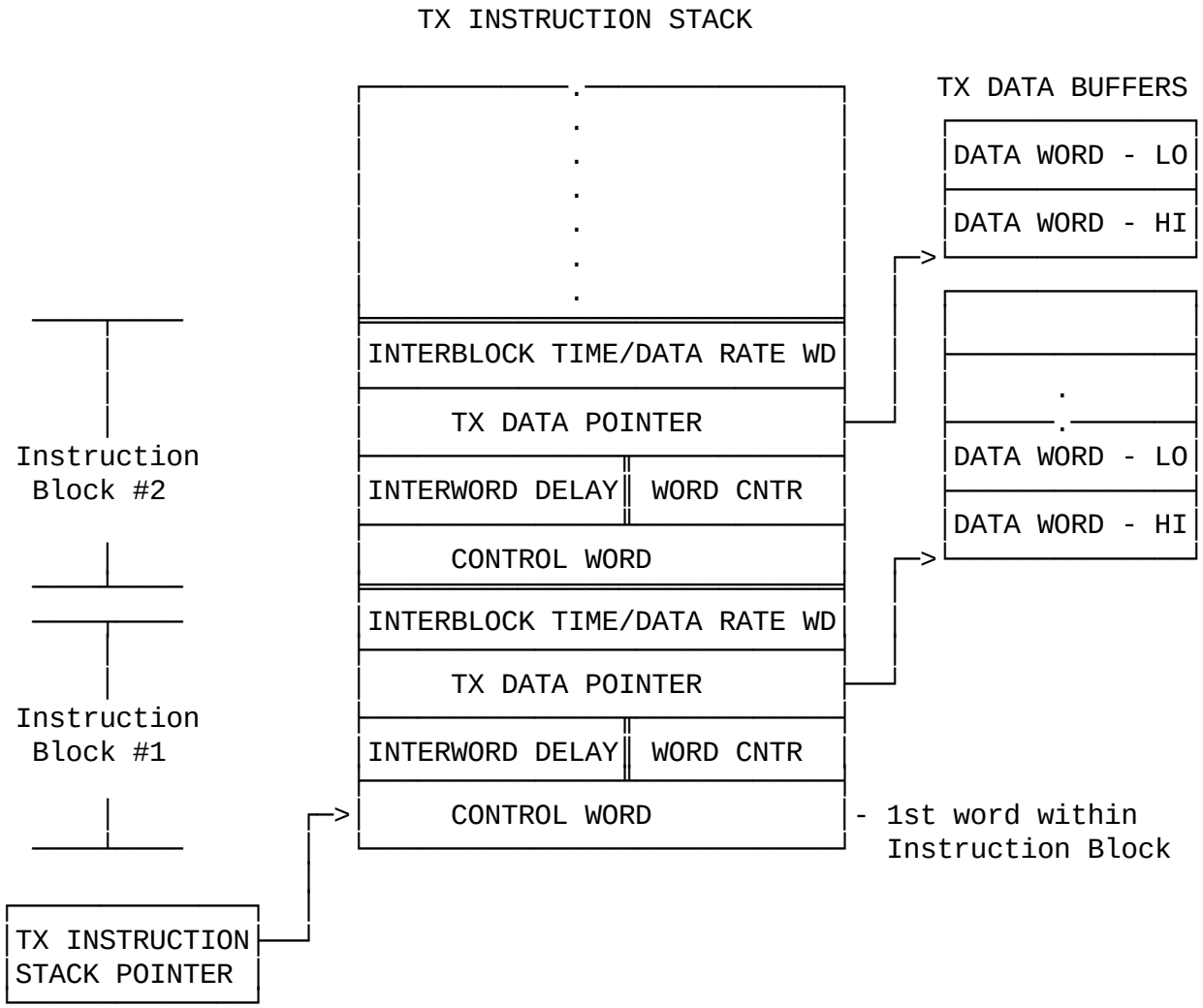
The method implemented in the transmit mode requires the user to create an instruction stack for the transmitter channel, write the data into the Dual-Port RAM and start transmission by writing to the Start Register found within the Global Register area. The sequence of writes to memory is not important except for the write to the Global Start Register, which is performed last.

TRANSMIT INSTRUCTION STACK

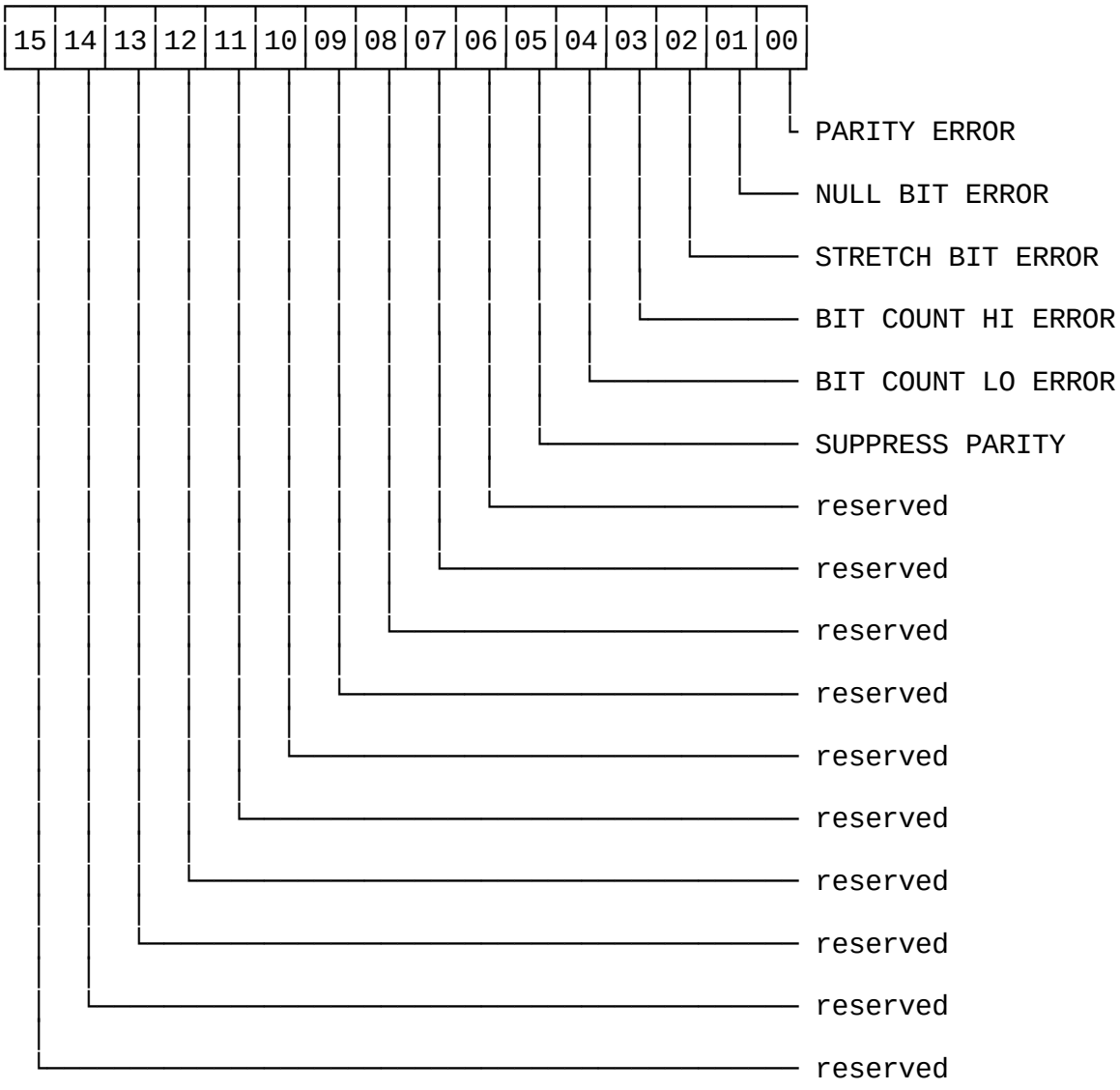
The Transmit Instruction Stack is divided into Instruction Blocks - each containing 4 words. Each Instruction Block relates to a Data Buffer. A Data Buffer contains one or more ARINC words which the user desires to transmit with the same amount of delay time between each word. The stack is sequential, so that the first Instruction Block relates to the first data block, the second to the second data block, etc.

As stated, each Instruction Block contains 4 words. The first word is the Control Word which contains error injection parameters. The second word contains an 8-bit Word Count which instructs the board as to the number of ARINC words to transmit within a particular block and an 8-bit, inter-word delay value which programs the time between words within the same buffer. The third word contains a 16-bit, user-supplied data pointer. This is a 16-bit address (must be even) which points to the beginning of the data words within the memory. The fourth word is the Interblock Time/Data Rate value and is used to program the time between data block transfers or the transmission period for the specific data block (see diagram for details).

TRANSMIT BLOCK DIAGRAM



CONTROL WORD DEFINITION (1st word within Instruction Block)



CONTROL WORD DEFINITION

00	Parity Error	A PARITY ERROR IS INSERTED WITHIN ALL ARINC WORDS WITHIN THE BLOCK.
01	Null Bit Error	A NULL BIT ERROR IS INSERTED WITHIN THE SECOND BIT OF EACH WORD WITHIN THE BLOCK (ARINC BIT 02) THIS BIT IS VALID FOR ARINC-429/575/582-2wire ONLY
02	Stretch Bit Error	THE SECOND ARINC BIT WITHIN EACH WORD WITHIN THE BLOCK IS "STRETCHED" CAUSING A MANCHESTER CODING ERROR. THIS BIT IS VALID FOR ARINC-429/575/582-2wire ONLY
03	Bit Count HI Error	THIRTY THREE ARINC BITS ARE TRANSMITTED WITHIN EACH WORD WITHIN THE BLOCK
04	Bit Count LO Error	THIRTY ONE ARINC BITS ARE TRANSMITTED WITHIN EACH WORD WITHIN THE BLOCK
05	Suppress Parity	FORCES A NO-PARITY CONDITION WITHIN THE WORD EVEN THOUGH "PARITY-ON" HAS BEEN SELECTED IN THE CHANNEL CONFIGURATION REGISTER
06	reserved	SET TO "0"
07	reserved	SET TO "0"
08	reserved	SET TO "0"
09	reserved	SET TO "0"
10	reserved	SET TO "0"
11	reserved	SET TO "0"
12	reserved	SET TO "0"
13	reserved	SET TO "0"
14	reserved	SET TO "0"
15	reserved	SET TO "0"

Note: The SUPPRESS PARITY bit can be useful in a situation where most of the data buffers contain standard binary data with parity and a few buffers contain BCD type data without parity. In such a situation, the parity can be set ON for all words within the Channel Configuration Register while exceptional cases (ie BCD data word buffer) can be forced to NO PARITY using this bit.

ARINC Channels

INTERWORD DLY/ WORD CNTR

This word is divided into two bytes, The Interword Delay and the Word Counter. The Interword delay byte specifies the time between blocks within this data block. The resolution is in the form of "bit times" according to the transmission bit rate. At the Lo- Speed setting, the resolution will be 80 usec/bit while at the Hi-Speed setting, it would be 10 usec/bit. The programmable setting would be according to its bit rate.

The Word Counter is used to specify the number of data words within this data block (1-255).

TX DATA POINTER

This register is used to set the start address of the transmit data buffer. The size of the buffer is determined by the Word Count value.

INTERBLOCK TIME/DATA RATE WORD

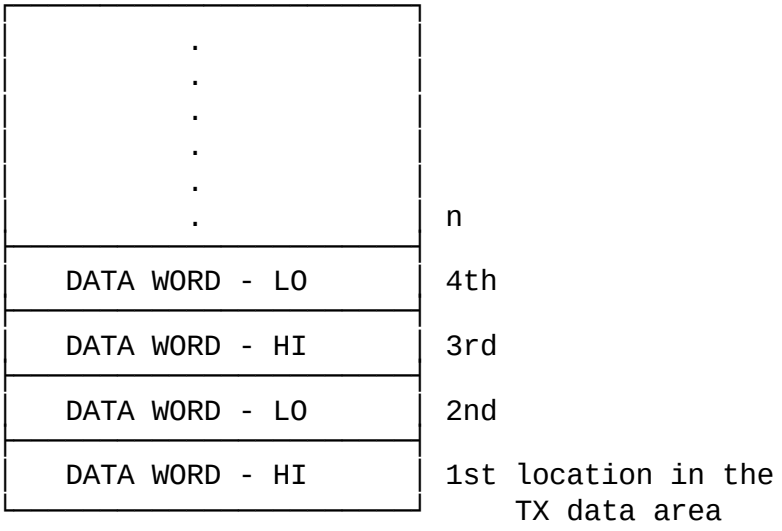
This word has two functions. In the standard transmit mode, the Instruction Blocks are accessed sequentially and their associated data words transmitted according to this sequential order. The Interblock Time allows the user to specify the time between data blocks. The resolution of this 16-bit word is according to the transmission bit rate. At the Lo-Speed setting, the resolution will be 80 usec/bit while at the Hi-Speed setting, it would be 10 usec/bit. The programmable setting would be according to its bit rate.
The interblock time is inserted AFTER the block transmission.

In Data Rate mode the user can specify the transmission period of the particular data block. If the number 'n' is written to this location, then the message will be transmitted every n bit times. At the Lo-Speed setting, one bit time is 80 usec while at the Hi-Speed setting, it is 10 usec. At the programmable setting a bit time is according to the specified bit rate.

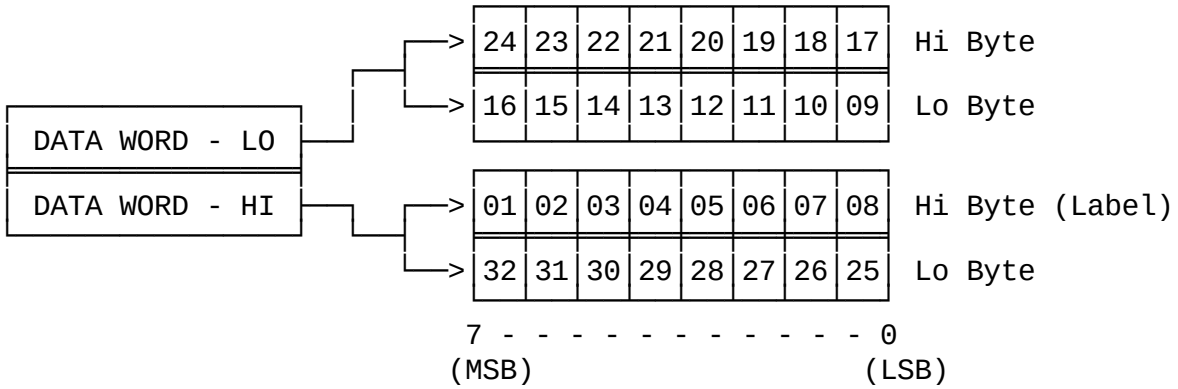
The selection of the modes is determined by the TX Mode Select bit within the Channel Configuration Register. Note that in Data Rate mode, a scratch buffer for the firmware must be allocated via the Channel Buffer Start and Channel Buffer End Registers.

TX DATA BLOCK FORMAT

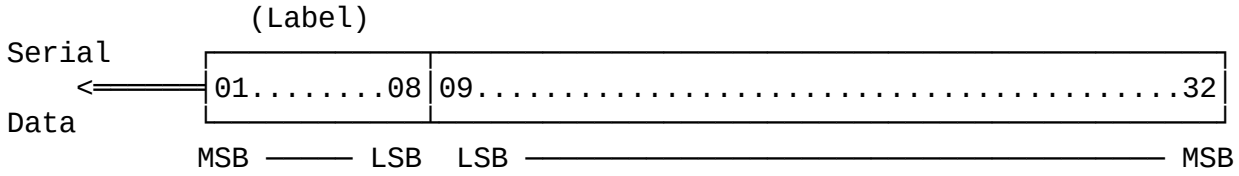
The figure below illustrates the format of the TX data words within the memory.



The figure below defines the locations and bit definitions of the data bytes within the memory.



Note: The numbers contained within the bytes above represent the ARINC bit locations within the 32-bit word.



Note that bits 09 through 32 are ordered from LSB to MSB (opposite from the Label field which is organized MSB to LSB). It is for this reason that the data block

is built the way it is (Hi-Word followed by Lo-Word) with the Label and the ARINC field 32 through 25 in the Hi-Word and bits 24 through 09 in the Lo-Word.

TRANSMITTER OPERATION - WILLIAMSBURG/BUCKHORN CHANNELS

The MAGICard Williamsburg channel sets support the Williamsburg protocol (section 2.5 of the ARINC-429 specification) as well as the fast Buckhorn protocol (section 2.6 of the ARINC-429 spec). Both protocols are supported at both high and low speed.

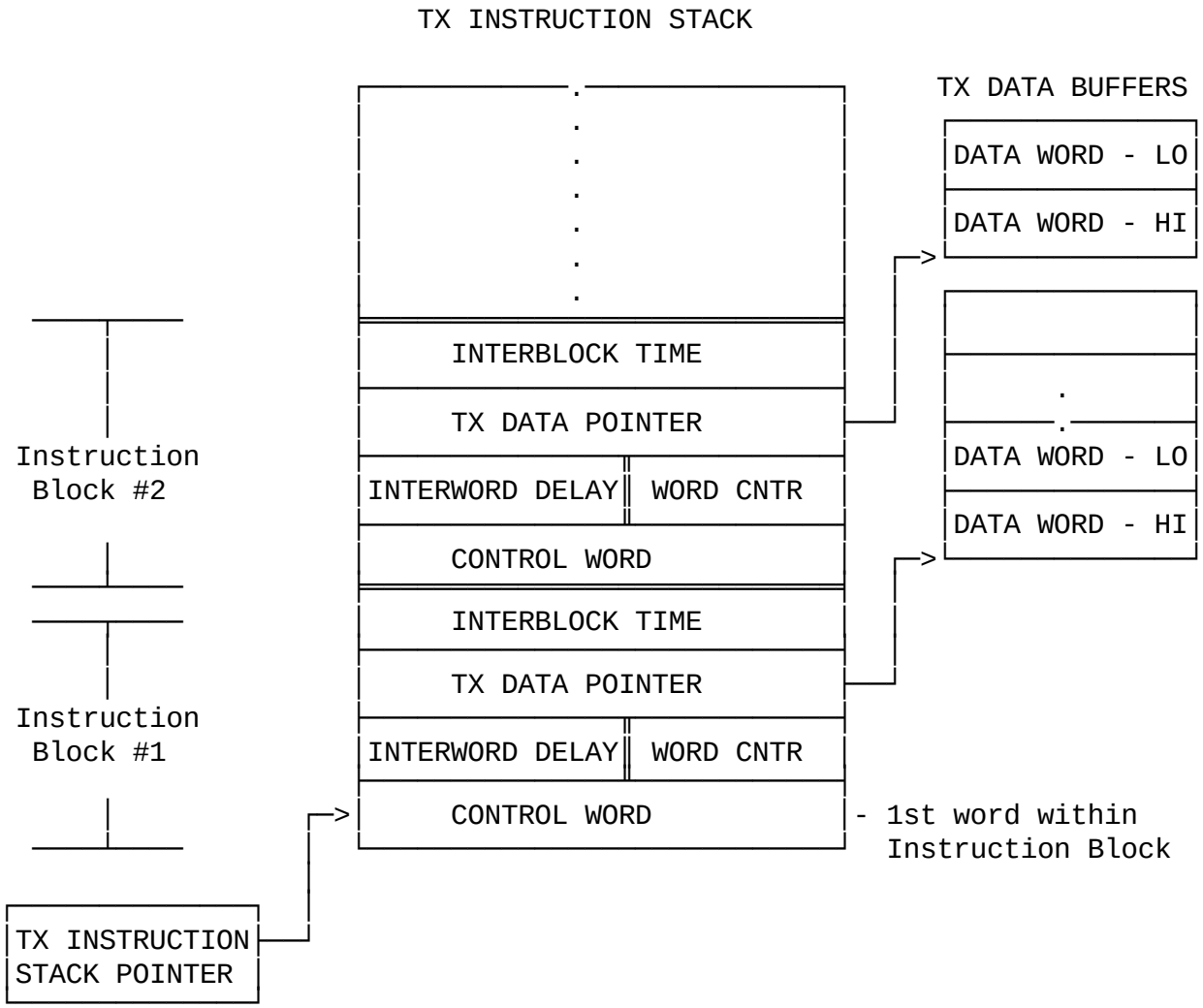
The method implemented in the transmit mode requires the user to create an instruction stack for the transmitter channel, write the data into the Dual-Port RAM and start transmission by writing to the Start Register found within the Global Register area. The sequence of writes to memory is not important except for the Write to "Start" Register operation, which is performed last.

TRANSMIT INSTRUCTION STACK

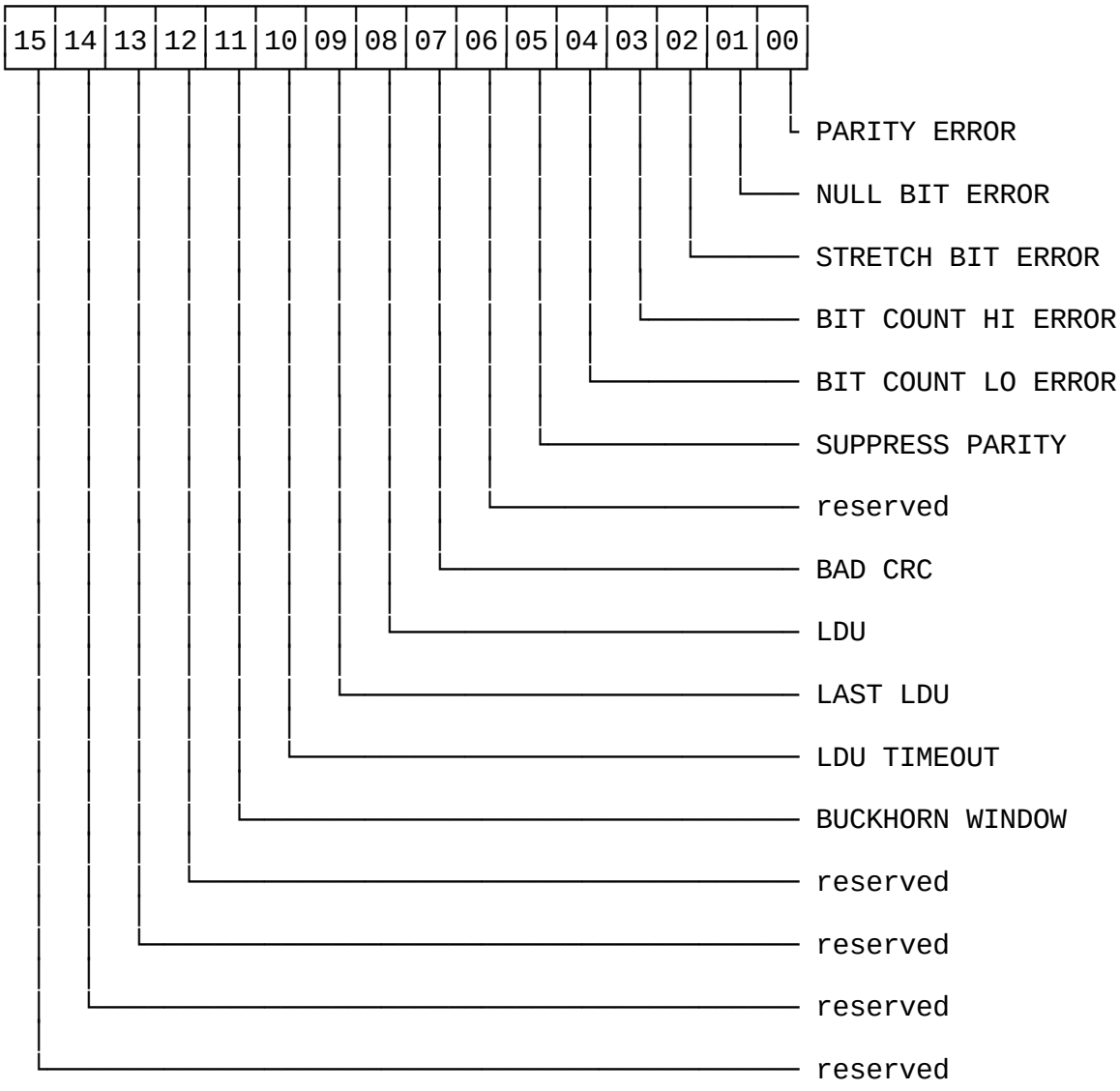
The Transmit Instruction Stack is divided into Instruction Blocks - each containing 4 words. Each Instruction Block relates to a Data Buffer. A Data Buffer contains one or more ARINC words which the user desires to transmit with the same amount of delay time between each word. The stack is sequential, so that the first Instruction Block relates to the first data block, the second to the second data block, etc.

As stated, each Instruction Block contains 4 words. The first word is the Control Word which contains error injection parameters and Williamsburg control. The second word contains an 8-bit Word Count which instructs the board as to the number of ARINC words to transmit within a particular block and an 8-bit, inter-word delay value which programs the time between words within the same buffer. The third word contains a 16-bit, user-supplied data pointer. This is a 16-bit address (must be even) which points to the beginning of the data words within the memory. The fourth word is the Interblock Time value and is used to program the time between data block transfers (see diagram for details).

TRANSMIT BLOCK DIAGRAM



CONTROL WORD DEFINITION (1st word within Instruction Block)



CONTROL WORD DEFINITION

00	Parity Error	A PARITY ERROR IS INSERTED WITHIN ALL ARINC WORDS WITHIN THE BLOCK.
01	Null Bit Error	A NULL BIT ERROR IS INSERTED WITHIN THE SECOND BIT OF EACH WORD WITHIN THE BLOCK (ARINC BIT 02)
02	Stretch Bit Error	THE SECOND ARINC BIT WITHIN EACH WORD WITHIN THE BLOCK IS "STRETCHED" CAUSING A MANCHESTER CODING ERROR.
03	Bit Count HI Error	THIRTY THREE ARINC BITS ARE TRANSMITTED WITHIN EACH WORD WITHIN THE BLOCK
04	Bit Count LO Error	THIRTY ONE ARINC BITS ARE TRANSMITTED WITHIN EACH WORD WITHIN THE BLOCK
05	Suppress Parity	FORCES A NO-PARITY CONDITION WITHIN THE WORD EVEN THOUGH "PARITY-ON" HAS BEEN SELECTED IN THE CHANNEL CONFIGURATION REGISTER
06	reserved	SET TO "0"
07	Bad CRC	FORCES TRANSMISSION OF AN INCORRECT CRC IN EOT WORD
08	LDU	INDICATES MESSAGE THAT REQUIRES RESPONSE (LDU, TEST WORD, OR ALO WORD)
09	Last LDU	INDICATES LAST LDU IN FILE (FOR INSERTION INTO EOT)
10	LDU Timeout	FORCES A TIMEOUT ERROR IN THE TRANSMITTED LDU BY NOT SENDING THE EOT WORD
11	Buckhorn Window	CAUSES AN LDU TO BE SENT AS A SINGLE-LDU BUCKHORN WINDOW BY ADDING A LCW WORD BEFORE THE SOT
12	reserved	SET TO "0"
13	reserved	SET TO "0"
14	reserved	SET TO "0"
15	reserved	SET TO "0"

Note: The SUPPRESS PARITY bit can be useful in a situation where most of the data buffers contain standard binary data with parity and a few buffers contain BCD type data without parity. In such a situation, the parity can be set ON for all words within the Channel Configuration Register while exceptional cases (ie BCD data word buffer) can be forced to NO PARITY using this bit.

INTERWORD DLY/ WORD CNTR

This word is divided into two bytes: the Interword Delay and the Word Counter. The Interword delay byte specifies the time between blocks within this data block. The resolution is in the form of bit times according to the transmission bit rate. At the Lo-Speed setting, the resolution will be 80 usec/bit while at the Hi-Speed setting, it would be 10 usec/bit. The programmable setting would be according to its bit rate.

The Word Counter is used to specify the number of data words within this data block (1-255).

TX DATA POINTER

This register is used to set the start address of the transmit data buffer. The size of the buffer is determined by the Word Count value.

INTERBLOCK TIME

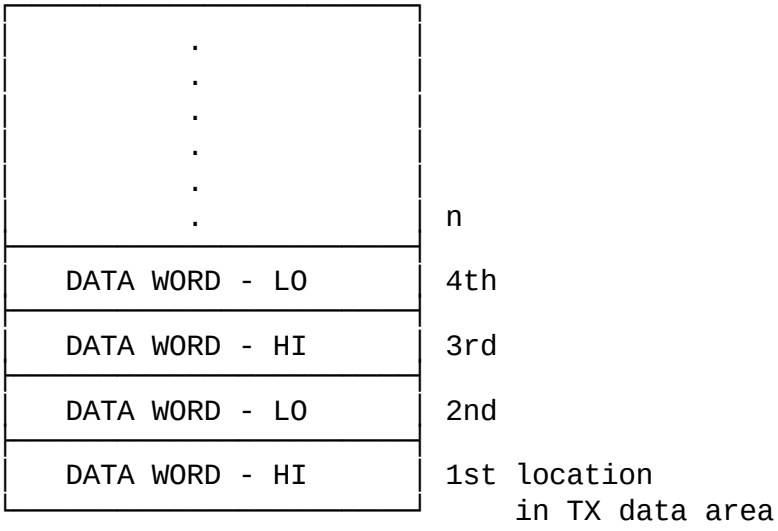
During transmission, Instruction Blocks are accessed sequentially and their associated data words transmitted according to this sequential order. The Interblock Time allows the user to specify the time between data blocks.

For blocks that require no response (LDU bit of the control word in the command block set to 0) the resolution of this 16-bit word is according to the transmission bit rate. At the Lo- Speed setting, the resolution will be 80 usec/bit while at the Hi-Speed setting, it would be 10 usec/bit. The programmable setting would be according to its bit rate. The interblock time is inserted AFTER the block transmission.

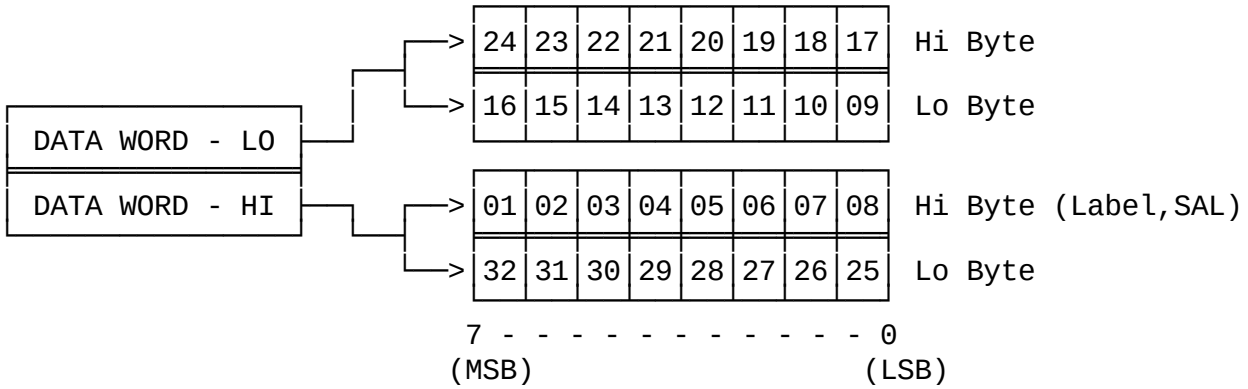
For blocks that do require a response, the resolution of this 16-bit word is always 80 usec/bit, regardless of the speed setting. The interblock time is inserted AFTER reception of the expected response, or after the maximum number of timeouts. For these blocks, the delay may be up to 1 ms longer than requested.

TX DATA BLOCK FORMAT

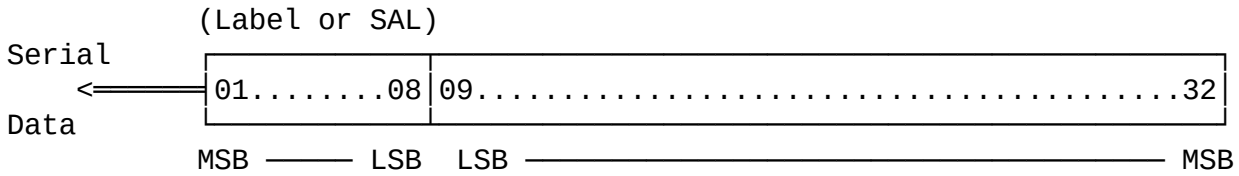
The figure below illustrates the format of the TX data words within the memory.



The figure below defines the locations and bit definitions of the data bytes within the memory.



Note: The numbers contained within the bytes above represent the ARINC bit locations within the 32-bit word.



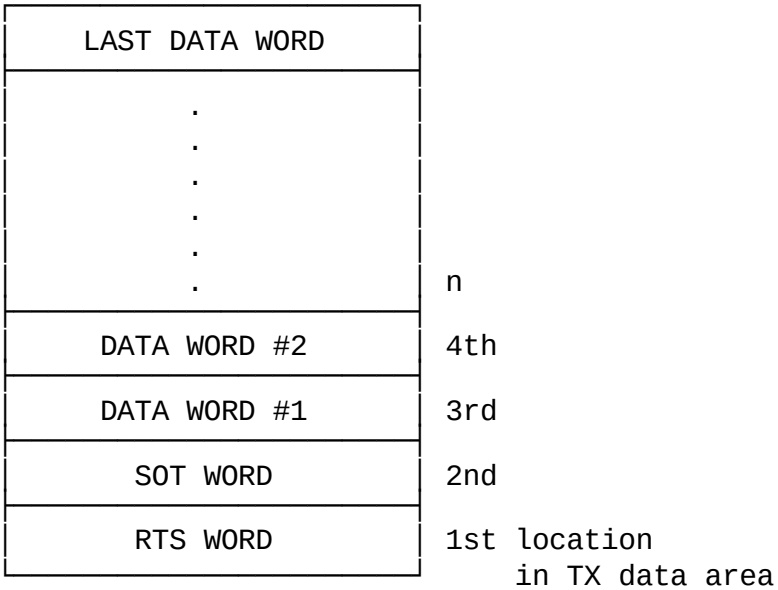
Note that bits 09 through 32 are ordered from LSB to MSB (opposite from the Label field which is organized MSB to LSB). It is for this reason that the data block is built the way it is (Hi-Word followed by Lo-Word) with the Label and the ARINC

field 32 through 25 in the Hi-Word and bits 24 through 09 in the Lo-Word.

TX DATA BLOCK

There are two types of blocks of data for transmission. The first type is blocks that require no handshaking. They consist of any combination of ARINC 429 data word with labels, and Williamsburg (bit oriented) SOLO data words. This type of block requires the LDU bit of the instruction block's control word to be set to 0.

The second type of block requires a response. This type of block requires the LDU bit of the instruction block's control word to be set to 1. It can be a single-word block containing an ALO word, a single-word block containing a TEST word, or an LDU. In each case, all words must contain in the label field the destination SAL which is found in the Channel Transmit SAL Register. An LDU block must start with an RTS word whose word count field matches the word count in the block's instruction block. This must be followed by an SOT word, and then any combination of Williamsburg full data words, partial data words, and character data words. In the case of a Buckhorn window (Buckhorn Window bit of the message Control Word set to 1), the LCW is not included in the block. It is generated and sent automatically before the SOT. An EOT word is also not included in the block. It is generated and sent automatically after the last data word.

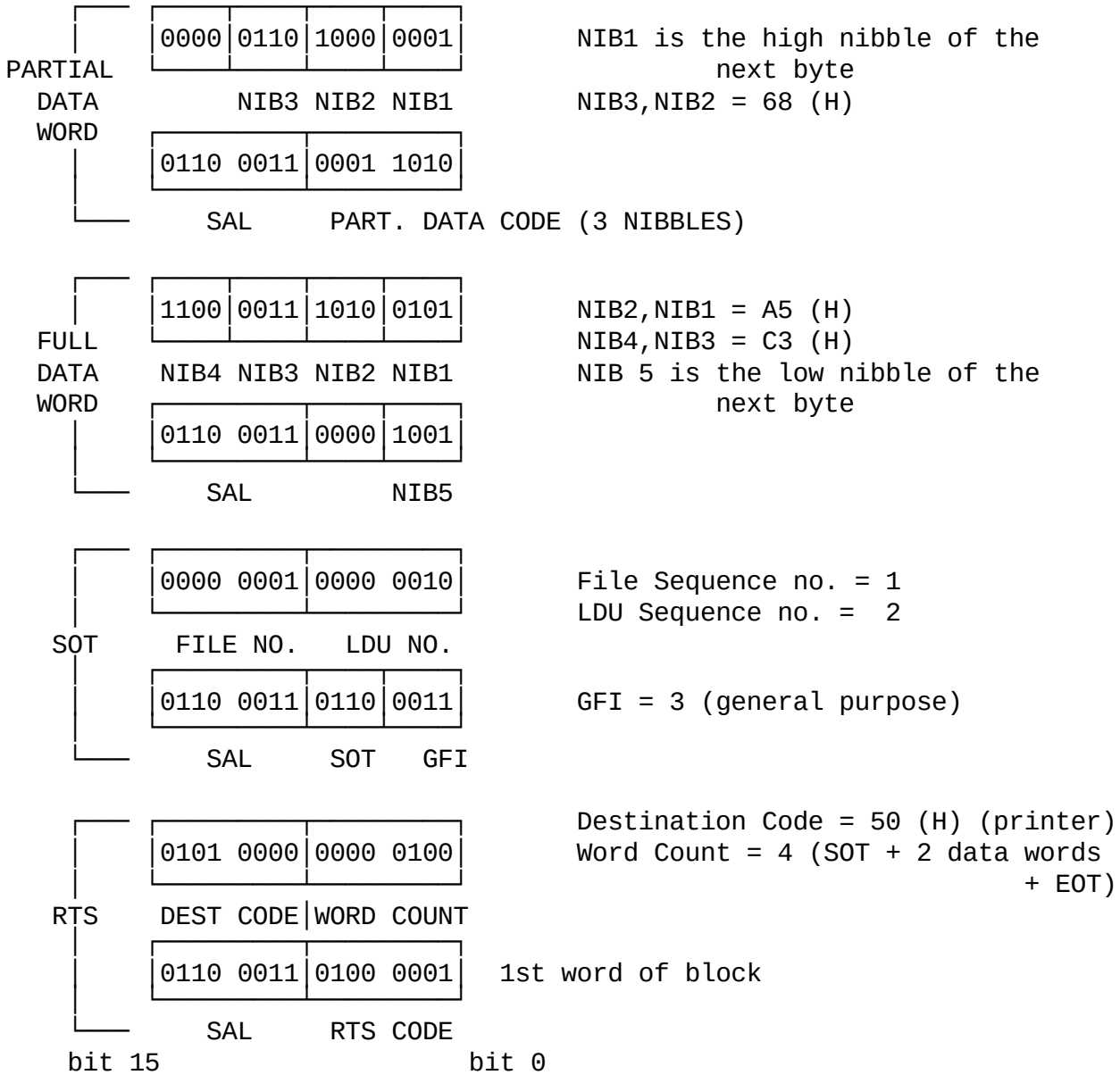


LDU block

LDU EXAMPLE

This is an example of how to build an LDU block to send the following 4 byte sequence to a device whose SAL is 63 (H):

A5 (H)
C3 (H)
19 (H)
68 (H)



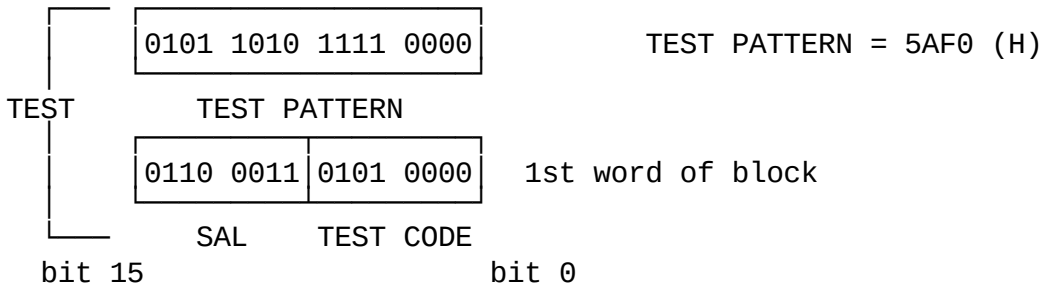
The word count in the block's instruction block must be 4, and the LDU bit in the control word must be set to 1. The Channel Tx SAL Register in the Channel Control Register Block for the Williamsburg channel must match the SAL field (63 (H) in this example).

After this sequence, an EOT would be automatically generated and sent. For these particular bytes the CRC field in the EOT would be 788D (H).

If the Buckhorn Window bit is set in the message's Control Word, a LCW word will be automatically inserted before the SOT with a remaining LDU count of 1, and a word count which matches the word count field of the RTS.

TEST EXAMPLE

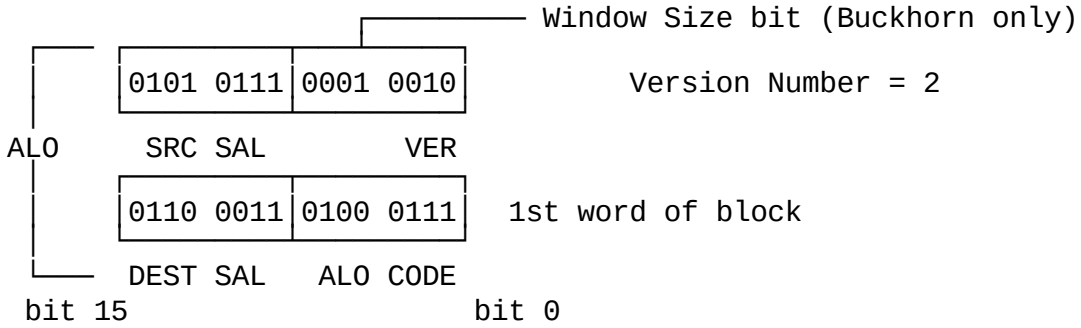
This is an example of how to build a TEST block to test the integrity of a Williamsburg interface with a device whose SAL is 63 (H). The block consists of a single TEST word as shown.



The block's instruction block must have a word count of 1 and the LDU bit of the control word must be set to 1. The Channel Tx SAL Register in the Channel Control Register Block for the Williamsburg channel must match the SAL field (63 (H) in this example).

ALO EXAMPLE

This is an example of how to build an ALO block to establish Williamsburg communication with a device whose SAL is 63 (H). The block consists of a single ALO word as shown. The SAL of the source is 57 (H).



The block's instruction block must have a word count of 1 and the LDU bit of the control word must be set to 1. The Channel Tx SAL Register in the Channel Control Register Block for the Williamsburg channel must match the Destination SAL field (63 (H) in this example), and the Rx SAL Register must match the Source SAL field.

DUTY CYCLE

The MAGICard card has a processing overhead time of approximately 50 microseconds per channel (150 microseconds per Williamsburg channel) during actual reception or transmission. This permits 10 channels to run at hi speed with an interword delay of 18 bit times resulting in an effective throughput of 1 word per channel every 500 microseconds. If fewer channels are used a smaller inter-word delay can be realized. If the duty cycle is exceeded in the Transmit Mode, all data will be sent but the interword times may be elongated. In the Receive Mode, exceeding the maximum duty cycle will result in occasional loss of data.

MAGICARD IMPLEMENTATION OF
THE WILLIAMSBURG PROTOCOL

FULL/HALF-DUPLEX

A Williamsburg channel can be configured for full- or half-duplex operation by writing to the Channel Full-Duplex Register in the Channel Control Register Block.

FULL-DUPLEX

In full-duplex operation, the channel is capable of simultaneously handling a block of incoming data and transmitting a block.

HALF-DUPLEX

In half-duplex operation, a channel will only accept incoming data if it is between transmission of consecutive blocks, or has no blocks to transmit. If a channel receives data in the Williamsburg format while waiting between blocks it will not initiate transmission of the next block until it has completed reception of the incoming block, and has responded to it. This can result in larger than expected intermessage gaps.

If an RTS word is received after transmission of an RTS (i.e. a race condition exists between 2 channels) the channel waits before acting. The wait varies randomly between 62.5 ms and 500 ms in steps of 62.5 ms. If during this period an RTS is received it is handled normally, and transmission of the pending transmit block is put off until the incoming LDU is processed. Otherwise, the RTS is retransmitted. This process is repeated until the conflict is resolved.

Notes:

1) An incoming ALO word will be accepted and handled properly at any time, even in half-duplex mode during transmission of a block.

2) During the interblock delay after a non-LDU block only, the Williamsburg channel will not send responses. This is true in both full- and half-duplex mode.

TRANSMIT

There are different types of blocks of data to be transmitted, each of which is handled as descibed below.

BLOCKS NOT REQUIRING RESPONSE

A block consisting only of ARINC 429 data words and Williamsburg SOLO words requires no response. This is indicated in the block's instruction block's control word by setting the LDU bit to 0. Such a block will be transmitted with the specified interword delay and intermessage delay before the next block.

BLOCKS REQUIRING RESPONSE

If the LDU bit of a block's control word (in its instruction block) is set, that indicates that the block requires response. For such a block, the intermessage delay is timed from the reception of the expected response, or after the maximum number of timeouts. The label field of every word within a block requiring a response must contain the destination SAL found in the Channel TX SAL Register. There are three types of blocks that require responses which are described below.

TEST BLOCK

The TEST-LOOP sequence is used to test the integrity of a Williamsburg communication link. A block containing a Williamsburg TEST word must have a word count of 1. It requires as response a Williamsburg LOOP word with the identical 16-bit test pattern within 200 ms. If a timeout occurs, the TEST word is not retransmitted. The timing of the interblock delay begins immediately upon reception of a legal LOOP word or a timeout.

ALO BLOCK

The ALO-ALR sequence is used to establish Williamsburg communiciation between two devices. A block containing a Williamsburg ALO word must have a word count of 1. It requires as response a Williamsburg ALR word within 200 ms. If a timeout occurs, the ALO is retransmitted a maximum of two more times. The timing of the interblock delay begins immediately upon reception of a legal ALR word or after

the third timeout.

LDU BLOCK

A block containing an LDU must contain as its first word a Williamsburg RTS word. When this word is transmitted, a response of CTS, NCTS, or BUSY is expected within 150 ms for high speed operation, or within 500 ms for low or variable speed operation.

If a NCTS or illegal CTS is received, the RTS is retransmitted up to 4 more times before passing on to the next block. The delay is 100 ms for high speed operation, and 500 ms for low or variable speed.

If a BUSY is received, the RTS is retransmitted up to 19 more times before passing on to the next block. The delay is 1 sec for high speed operation, and 15 sec for low or variable speed operation.

If a timeout occurs, the RTS is retransmitted up to 4 more times before passing on to the next block.

If a legal CTS word is received, the next word sent depends on the state of the Buckhorn Window bit of the message's Control Word. If this bit is set to 1, a LCW is constructed and transmitted. It contains a remaining LDU field of 1, and the word count from the RTS. Otherwise a LCW is not sent. The SOT (the second word in the transmit block) is then transmitted followed by the data words with the specified interword delay between consecutive words. After the last data word is transmitted, the EOT is constructed and transmitted. It contains the calculated CRC, and the last LDU bit as read from the command word in the block's instruction block. A response of ACK is expected within 220 ms. If a timeout occurs, the LDU is retransmitted starting with the RTS word a maximum of 4 more times. The timing of the interblock delay begins immediately upon reception of a legal ACK word, or after the fifth timeout.

If at any time during the transmission of an LDU (between transmission of an RTS and reception of an ACK word) a NAK word is received, transmission of the LDU is repeated beginning with the RTS word. If a NAK is received 3 times during the course of a single LDU, transmission of the next block is initiated after the interblock delay has passed.

If at any time during the transmission of an LDU a SYN word is received, transmission of the frame starting with the block pointed to by the first instruction block is immediately initiated (the loop count is not altered).

If at any time during transmission of an LDU an ALO word is received, the required response, an ALR word, is transmitted, and transmission of the LDU is repeated beginning with the RTS word.

RECEIVE

If data is receive which does not contain the SAL of the Williamsburg channel (as found in the Channel Rx SAL Register) or has a parity error the data is stored sequentially in the receive data area, and no further action is taken.

If an RTS word is received, the channel enters LDU mode. If the word count field of the RTS word is less than the legal minimum of 3, a response of NCTS is sent. If the destination code field of the RTS matches the (non-zero) contents of the Channel Destination Code Busy Register, a response of BUSY is sent (this is for test purposes only). Otherwise, a response of CTS is sent, and reception of an LCW word (Buckhorn only), SOT word, data words and an EOT word is expected. If an LCW word is received, the expected number of LDUs is specified in the "remaining LDUs" field. If the LDU is received with no errors, a response of ACK is sent. A NAK word is sent in the following cases.

-A data word, LCW word or EOT word is received when expecting SOT
     (status code = 80 (H))
-An expected LCW is not received
-Remaining LDUs field of received LCW is not 1 less than in previous LCW
     (status code = 7F (H))
-EOT is received earlier or later than expected
-RTS or LCW indicates word count less than 3
     (status code = 88 (H))
-CRC field in the EOT does not agree with the calculated CRC
     (status code = 85 (H))
-EOT not received within 2.5 secs of having sent CTS (low or var speed)
-EOT not received within 400 ms of having sent CTS (high speed operation)
     (status code = 86 (H))

If an RTS is received during reception of an LDU, the response is sent as described above, and reception of the entire LDU from the beginning is expected. However, the data already stored in the receive data area is not overwritten by the repeated LDU. The repeated LDU is recorded after the last word received.

Whenever an ALO word is received, an ALR word is sent in response. If it received during reception of an LDU, the LDU is expected to be repeated beginning with the RTS.

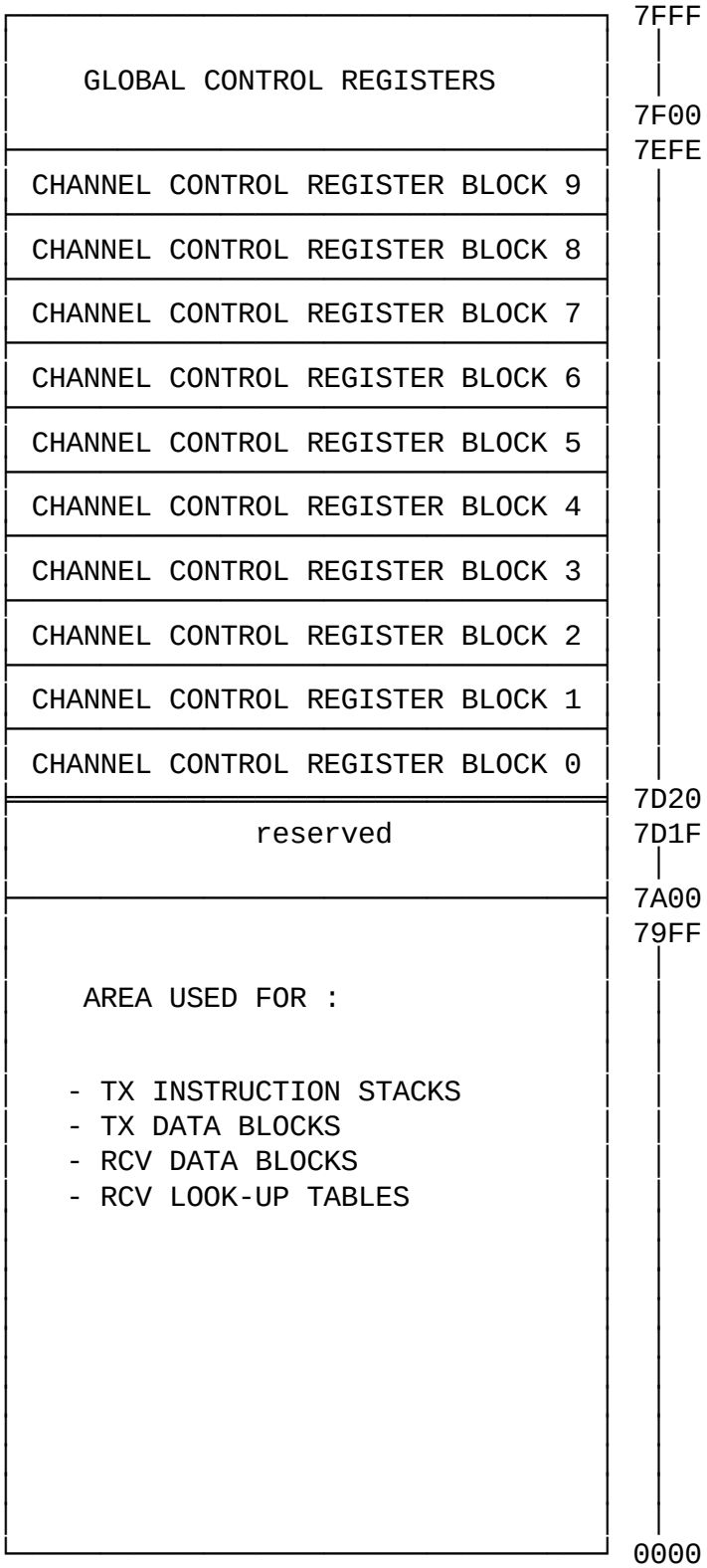
If a TEST word is received during reception of an LDU, it is stored and no further action is taken. If reception of an LDU is not in progress when a TEST word is received, a LOOP word is sent in response.

Notes:
1) A received LDU whose RTS word contains a non-zero destination code is not automatically transferred to the indicated final destination. If it is desired to transfer the LDU the user must stop the channel (by setting to zero the Global Start Bit associated with the Williamsburg receive module), set up a transmission instruction block and restart the channel.

2) The MAGICard does not check the sequence of file or LDU sequence numbers in a received SOT word.

OPERATION MANUAL FOR THE RS-232/422/485/423 CHANNELS

GENERAL MEMORY MAP



GLOBAL CONTROL REGISTERS

reserved	7FFF | | 7F86
SOFTWARE RESET REGISTER	7F84
reserved	7F82
reserved	7F80
reserved	7F7E
RESET TIME TAG REGISTER	7F7C
START/STOP REGISTER	7F7A
reserved	7F78
BOARD READY REGISTER	7F76
BOARD STATUS REGISTER	7F74
FIRMWARE REVISION REGISTER	7F72
INTERRUPT STATUS REGISTER	7F70
reserved	7F6E
CHANNEL 9 CONFIGURATION STATUS REG.	7F6C
CHANNEL 8 CONFIGURATION STATUS REG.	7F6A
CHANNEL 7 CONFIGURATION STATUS REG.	7F68
CHANNEL 6 CONFIGURATION STATUS REG.	7F66
CHANNEL 5 CONFIGURATION STATUS REG.	7F64
CHANNEL 4 CONFIGURATION STATUS REG.	7F62
CHANNEL 3 CONFIGURATION STATUS REG.	7F60
CHANNEL 2 CONFIGURATION STATUS REG.	7F5E
CHANNEL 1 CONFIGURATION STATUS REG.	7F5C
CHANNEL 0 CONFIGURATION STATUS REG.	7F5A
RECEIVER DATA STORAGE MODE REGISTER	7F58
. .	

RS Channels

RS-232/422/485/423 OPERATING MODE REG	7F3A

SOFTWARE RESET REGISTER 7F84 (H)

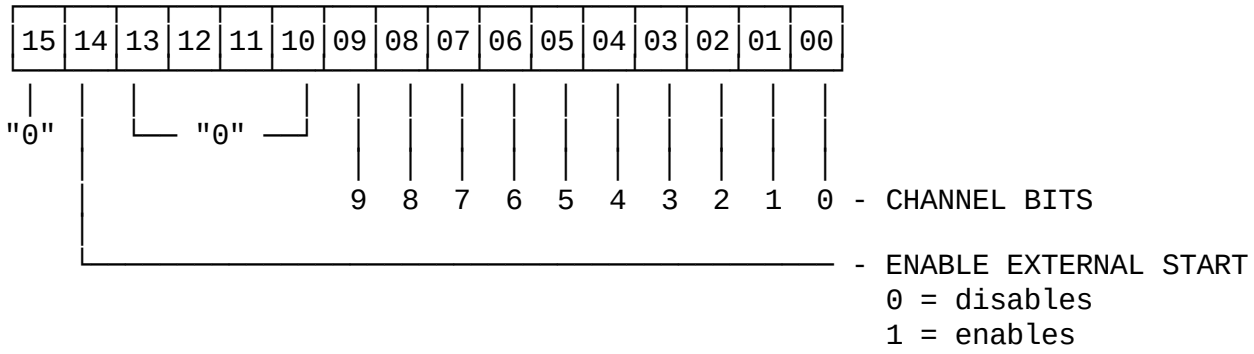
Writing a 0 to this register resets the board. Following a reset, the board will execute a self-test (both memory and channels), clear all the on-board memory and then update the Board Status Register. The board indicates that it is ready by writing a value of 3000(H) to the Board Ready Register.

RESET TIME TAG REGISTER 7F7C (H)

Writing any non-zero value to this register will cause the time tag to be reset to `0'. Upon completion of the Time Tag Reset operation, this register will be cleared by the board.

START/STOP REGISTER 7F7A (H)

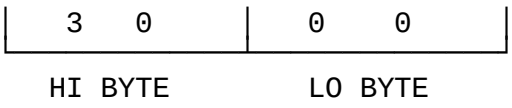
The user can start one or more channels at the same time. Writing a 1 to bit "00" starts channel 0 operation, writing a 1 to the next location starts channel 1, etc. Writing a 0 to a bit location will stop that channel's operation. The user should wait a minimum of 350 µsec between writes to the START/STOP register. The Enable External Start bit is used in conjunction with on-board jumpers to start the board's operation from an external event (see section on VME/VXI Jumpers and External Triggers for details).



BOARD READY REGISTER 7F76 (H)

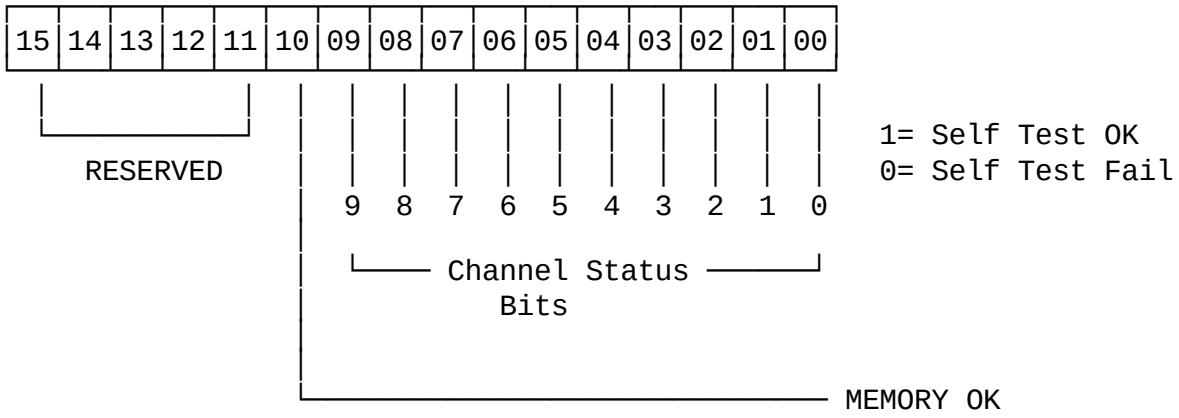
This register indicates that the board has finished its Power-on sequence and that the board is ready to be accessed by the Host. The board will write the value: 3000 (H) into this register when ready.

RS Channels


3 0
0 0
HI BYTE
LO BYTE

BOARD STATUS REGISTER 7F74 (H)

Indicates the result of the Power-on, self-test of the board.



Notes:

1. The "Self Test Fail" is set when the channel self-test fails or when the channel is not present on the board.

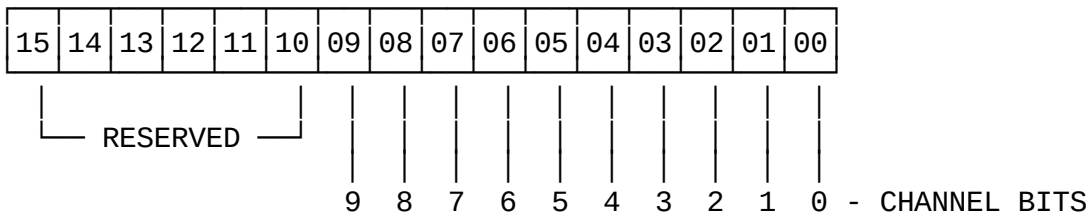
2. The board will continue to operate on condition of Channel Self-Test Failures BUT will not continue to operate on condition of a Memory failure.

FIRMWARE REVISION REGISTER 7F72 (H)

Indicates the revision level of the firmware (ie: 0114 (H) = Rev 1.14)

INTERRUPT STATUS REGISTER 7F70 (H)

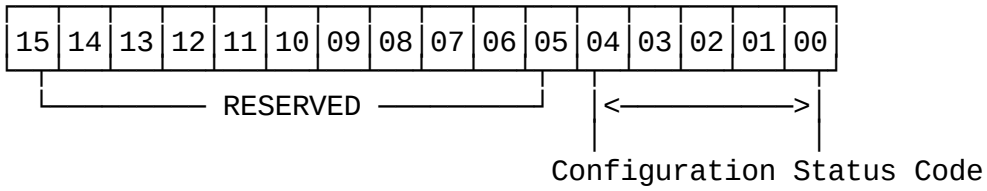
Indicates which channel issued the interrupt (`1' = Active). The status bits are only reset by the user.



RS Channels

CHANNEL x CONFIGURATION STATUS REGISTER (see Global Memory Map)

This register indicates to the host the type of channel configured in each channel socket on the board.

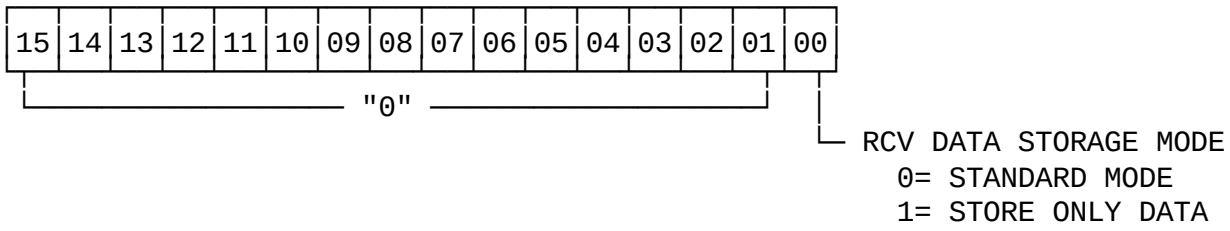


4 3 2 1 0	Configuration Status Code
0 0 0 0 0	Undefined Channel
0 0 0 0 1	ARINC-429 Receive Channel
0 0 0 1 0	ARINC-429 Transmit Channel
0 0 0 1 1	ARINC-561 Receive Channel
0 0 1 0 0	ARINC-561 Transmit Channel
0 0 1 0 1	ARINC-568 Receive Channel
0 0 1 1 0	ARINC-568 Transmit Channel
0 0 1 1 1	ARINC-575 Receive Channel
0 1 0 0 0	ARINC-575 Transmit Channel
0 1 0 0 1	ARINC-582 2-Wire Receive Channel
0 1 0 1 0	ARINC-582 2-Wire Transmit Channel
0 1 0 1 1	ARINC-582 6-Wire Receive Channel
0 1 1 0 0	ARINC-582 6-Wire Transmit Channel
0 1 1 0 1	reserved
0 1 1 1 0	reserved
0 1 1 1 1	*
1 0 0 0 0	RS-232 Channel
1 0 0 0 1	RS-422 Channel
1 0 0 1 0	RS-485 Channel
1 0 0 1 1	RS-423 Channel (Note 2)
1 0 1 0 0	*
1 0 1 0 1	*
1 0 1 1 0	*
1 0 1 1 1	*
1 1 0 0 0	*
1 1 0 0 1	*
1 1 0 1 0	*
1 1 0 1 1	*
1 1 1 0 0	*
1 1 1 0 1	ARINC-429 Williamsburg Rx Channel
1 1 1 1 0	ARINC-429 Williamsburg Tx Channel
1 1 1 1 1	ARINC-429 Williamsburg CRC Channel

Notes: 1) * - Free Codes for future channels
2) The RS-423 channel requires firmware Revision 1.31 and up.

RECEIVER DATA STORAGE MODE REGISTER 7F58 (H)

This register is used to select the Receiver Data Storage Mode. Data bytes can be stored with Time Tag and Status Byte appended to the data block or without these additional bytes. Set bit "00" to a logic 0 in order to select the standard mode which appends both Time Tag and Status Byte to each data byte stored in memory. Set register bit "00" to a logic 1 to select Data Only mode.



Notes:

1) There is no Merge Mode option for RS type channels.

2) A change in this register is only noted by the firmware after the Start/Stop Register contains a value of "0" for at least 1 msec.

RS-232/422/485/423 OPERATING MODE REGISTER 7F3A (H)

This register is used to select the mode of operation for all RS-232/422/485/423 channels. The channels will operate in regular mode if 0 is written to this register, and in fast mode if 1 is written. In regular mode, operation together with high speed ARINC channels is supported. However, at high baud rates data will be lost. In fast mode, operation at high baud rates is supported, but high speed ARINC receive modules may lose data. In addition, in fast mode received bytes are stored without status and time tag. For more information on these modes see the section "Baud Rate Limitations".

CHANNEL CONTROL REGISTER BLOCKS

CHANNEL 0 CONTROL REGISTER BLOCK

CHANNEL 0 CONFIGURATION REGISTER	7D4E
CHANNEL 0 BAUD RATE GENERATOR	7D4C
CHANNEL 0 RCV DATA START POINTER	7D4A
CHANNEL 0 RCV DATA END POINTER	7D48
CHANNEL 0 RCV DATA CURRENT POINTER	7D46
reserved	7D44
reserved	7D42
CHANNEL 0 RCV DATA BYTE COUNT REGISTER	7D40
CHANNEL 0 RCV BUFFER WRAPAROUND REGISTER	7D3E
CHANNEL 0 RCV BYTE COUNTER TRIGGER RGISTER	7D3C
reserved	7D3A
CHANNEL 0 RCV INTERVAL COUNTER TRIGGER	7D38
CHANNEL 0 RCV ERROR COUNT REGISTER	7D36
CHANNEL 0 RCV BYTE TRIGGER REGISTER	7D34
CHANNEL 0 TX INSTRUCTION STACK POINTER	7D32
CHANNEL 0 TX INSTRUCTION COUNTER	7D30
CHANNEL 0 TX LOOP COUNTER	7D2E
reserved	7D2C
CHANNEL 0 INTERRUPT/TRIGGER CONDITION REG.	7D2A
CHANNEL 0 STATUS REGISTER	7D28
reserved	7D26
reserved	7D24
CHANNEL 0 START TRANSMIT REGISTER	7D22
CHANNEL 0 START RECEIVE REGISTER	7D20

CHANNEL 1 CONTROL REGISTER BLOCK

CHANNEL 1 CONFIGURATION REGISTER	7D7E
CHANNEL 1 BAUD RATE GENERATOR	7D7C
CHANNEL 1 RCV DATA START POINTER	7D7A
CHANNEL 1 RCV DATA END POINTER	7D78
CHANNEL 1 RCV DATA CURRENT POINTER	7D76
reserved	7D74
reserved	7D72
CHANNEL 1 RCV DATA BYTE COUNT REGISTER	7D70
CHANNEL 1 RCV BUFFER WRAPAROUND REGISTER	7D6E
CHANNEL 1 RCV BYTE COUNTER TRIGGER	7D6C
reserved	7D6A
CHANNEL 1 RCV INTERVAL COUNTER TRIGGER	7D68
CHANNEL 1 RCV ERROR COUNT REGISTER	7D66
CHANNEL 1 RCV BYTE TRIGGER REGISTER	7D64
CHANNEL 1 TX INSTRUCTION STACK POINTER	7D62
CHANNEL 1 TX INSTRUCTION COUNTER	7D60
CHANNEL 1 TX LOOP COUNTER	7D5E
reserved	7D5C
CHANNEL 1 INTERRUPT/TRIGGER CONDITION REG.	7D5A
CHANNEL 1 STATUS REGISTER	7D58
reserved	7D56
reserved	7D54
CHANNEL 1 START TRANSMIT REGISTER	7D52
CHANNEL 1 START RECEIVE REGISTER	7D50

CHANNEL 2 CONTROL REGISTER BLOCK

CHANNEL 2 CONFIGURATION REGISTER	7DAE
CHANNEL 2 BAUD RATE GENERATOR	7DAC
CHANNEL 2 RCV DATA START POINTER	7DAA
CHANNEL 2 RCV DATA END POINTER	7DA8
CHANNEL 2 RCV DATA CURRENT POINTER	7DA6
reserved	7DA4
reserved	7DA2
CHANNEL 2 RCV DATA BYTE COUNT REGISTER	7DA0
CHANNEL 2 RCV BUFFER WRAPAROUND REGISTER	7D9E
CHANNEL 2 RCV BYTE COUNTER TRIGGER	7D9C
reserved	7D9A
CHANNEL 2 RCV INTERVAL COUNTER TRIGGER	7D98
CHANNEL 2 RCV ERROR COUNT REGISTER	7D96
CHANNEL 2 RCV BYTE TRIGGER REGISTER	7D94
CHANNEL 2 TX INSTRUCTION STACK POINTER	7D92
CHANNEL 2 TX INSTRUCTION COUNTER	7D90
CHANNEL 2 TX LOOP COUNTER	7D8E
reserved	7D8C
CHANNEL 2 INTERRUPT/TRIGGER CONDITION REG.	7D8A
CHANNEL 2 STATUS REGISTER	7D88
reserved	7D86
reserved	7D84
CHANNEL 2 START TRANSMIT REGISTER	7D82
CHANNEL 2 START RECEIVE REGISTER	7D80

CHANNEL 3 CONTROL REGISTER BLOCK

CHANNEL 3 CONFIGURATION REGISTER	7DDE
CHANNEL 3 BAUD RATE GENERATOR	7DDC
CHANNEL 3 RCV DATA START POINTER	7DDA
CHANNEL 3 RCV DATA END POINTER	7DD8
CHANNEL 3 RCV DATA CURRENT POINTER	7DD6
reserved	7DD4
reserved	7DD2
CHANNEL 3 RCV DATA BYTE COUNT REGISTER	7DD0
CHANNEL 3 RCV BUFFER WRAPAROUND REGISTER	7DCE
CHANNEL 3 RCV BYTE COUNTER TRIGGER	7DCC
reserved	7DCA
CHANNEL 3 RCV INTERVAL COUNTER TRIGGER	7DC8
CHANNEL 3 RCV ERROR COUNT REGISTER	7DC6
CHANNEL 3 RCV BYTE TRIGGER REGISTER	7DC4
CHANNEL 3 TX INSTRUCTION STACK POINTER	7DC2
CHANNEL 3 TX INSTRUCTION COUNTER	7DC0
CHANNEL 3 TX LOOP COUNTER	7DBE
reserved	7DBC
CHANNEL 3 INTERRUPT/TRIGGER CONDITION REG.	7DBA
CHANNEL 3 STATUS REGISTER	7DB8
reserved	7DB6
reserved	7DB4
CHANNEL 3 START TRANSMIT REGISTER	7DB2
CHANNEL 3 START RECEIVE REGISTER	7DB0

CHANNEL 4 CONTROL REGISTER BLOCK

CHANNEL 4 CONFIGURATION REGISTER	7E0E
CHANNEL 4 BAUD RATE GENERATOR	7E0C
CHANNEL 4 RCV DATA START POINTER	7E0A
CHANNEL 4 RCV DATA END POINTER	7E08
CHANNEL 4 RCV DATA CURRENT POINTER	7E06
reserved	7E04
reserved	7E02
CHANNEL 4 RCV DATA BYTE COUNT REGISTER	7E00
CHANNEL 4 RCV BUFFER WRAPAROUND REGISTER	7DFE
CHANNEL 4 RCV BYTE COUNTER TRIGGER	7DFC
reserved	7DFA
CHANNEL 4 RCV INTERVAL COUNTER TRIGGER	7DF8
CHANNEL 4 RCV ERROR COUNT REGISTER	7DF6
CHANNEL 4 RCV BYTE TRIGGER REGISTER	7DF4
CHANNEL 4 TX INSTRUCTION STACK POINTER	7DF2
CHANNEL 4 TX INSTRUCTION COUNTER	7DF0
CHANNEL 4 TX LOOP COUNTER	7DEE
reserved	7DEC
CHANNEL 4 INTERRUPT/TRIGGER CONDITION REG.	7DEA
CHANNEL 4 STATUS REGISTER	7DE8
reserved	7DE6
reserved	7DE4
CHANNEL 4 START TRANSMIT REGISTER	7DE2
CHANNEL 4 START RECEIVE REGISTER	7DE0

CHANNEL 5 CONTROL REGISTER BLOCK

CHANNEL 5 CONFIGURATION REGISTER	7E3E
CHANNEL 5 BAUD RATE GENERATOR	7E3C
CHANNEL 5 RCV DATA START POINTER	7E3A
CHANNEL 5 RCV DATA END POINTER	7E38
CHANNEL 5 RCV DATA CURRENT POINTER	7E36
reserved	7E34
reserved	7E32
CHANNEL 5 RCV DATA BYTE COUNT REGISTER	7E30
CHANNEL 5 RCV BUFFER WRAPAROUND REGISTER	7E2E
CHANNEL 5 RCV BYTE COUNTER TRIGGER	7E2C
reserved	7E2A
CHANNEL 5 RCV INTERVAL COUNTER TRIGGER	7E28
CHANNEL 5 RCV ERROR COUNT REGISTER	7E26
CHANNEL 5 RCV BYTE TRIGGER REGISTER	7E24
CHANNEL 5 TX INSTRUCTION STACK POINTER	7E22
CHANNEL 5 TX INSTRUCTION COUNTER	7E20
CHANNEL 5 TX LOOP COUNTER	7E1E
reserved	7E1C
CHANNEL 5 INTERRUPT/TRIGGER CONDITION REG.	7E1A
CHANNEL 5 STATUS REGISTER	7E18
reserved	7E16
reserved	7E14
CHANNEL 5 START TRANSMIT REGISTER	7E12
CHANNEL 5 START RECEIVE REGISTER	7E10

CHANNEL 6 CONTROL REGISTER BLOCK

CHANNEL 6 CONFIGURATION REGISTER	7E6E
CHANNEL 6 BAUD RATE GENERATOR	7E6C
CHANNEL 6 RCV DATA START POINTER	7E6A
CHANNEL 6 RCV DATA END POINTER	7E68
CHANNEL 6 RCV DATA CURRENT POINTER	7E66
reserved	7E64
reserved	7E62
CHANNEL 6 RCV DATA BYTE COUNT REGISTER	7E60
CHANNEL 6 RCV BUFFER WRAPAROUND REGISTER	7E5E
CHANNEL 6 RCV BYTE COUNTER TRIGGER	7E5C
reserved	7E5A
CHANNEL 6 RCV INTERVAL COUNTER TRIGGER	7E58
CHANNEL 6 RCV ERROR COUNT REGISTER	7E56
CHANNEL 6 RCV BYTE TRIGGER REGISTER	7E54
CHANNEL 6 TX INSTRUCTION STACK POINTER	7E52
CHANNEL 6 TX INSTRUCTION COUNTER	7E50
CHANNEL 6 TX LOOP COUNTER	7E4E
reserved	7E4C
CHANNEL 6 INTERRUPT/TRIGGER CONDITION REG.	7E4A
CHANNEL 6 STATUS REGISTER	7E48
reserved	7E46
reserved	7E44
CHANNEL 6 START TRANSMIT REGISTER	7E42
CHANNEL 6 START RECEIVE REGISTER	7E40

CHANNEL 7 CONTROL REGISTER BLOCK

CHANNEL 7 CONFIGURATION REGISTER	7E9E
CHANNEL 7 BAUD RATE GENERATOR	7E9C
CHANNEL 7 RCV DATA START POINTER	7E9A
CHANNEL 7 RCV DATA END POINTER	7E98
CHANNEL 7 RCV DATA CURRENT POINTER	7E96
reserved	7E94
reserved	7E92
CHANNEL 7 RCV DATA BYTE COUNT REGISTER	7E90
CHANNEL 7 RCV BUFFER WRAPAROUND REGISTER	7E8E
CHANNEL 7 RCV BYTE COUNTER TRIGGER	7E8C
reserved	7E8A
CHANNEL 7 RCV INTERVAL COUNTER TRIGGER	7E88
CHANNEL 7 RCV ERROR COUNT REGISTER	7E86
CHANNEL 7 RCV BYTE TRIGGER REGISTER	7E84
CHANNEL 7 TX INSTRUCTION STACK POINTER	7E82
CHANNEL 7 TX INSTRUCTION COUNTER	7E80
CHANNEL 7 TX LOOP COUNTER	7E7E
reserved	7E7C
CHANNEL 7 INTERRUPT/TRIGGER CONDITION REG.	7E7A
CHANNEL 7 STATUS REGISTER	7E78
reserved	7E76
reserved	7E74
CHANNEL 7 START TRANSMIT REGISTER	7E72
CHANNEL 7 START RECEIVE REGISTER	7E70

CHANNEL 8 CONTROL REGISTER BLOCK

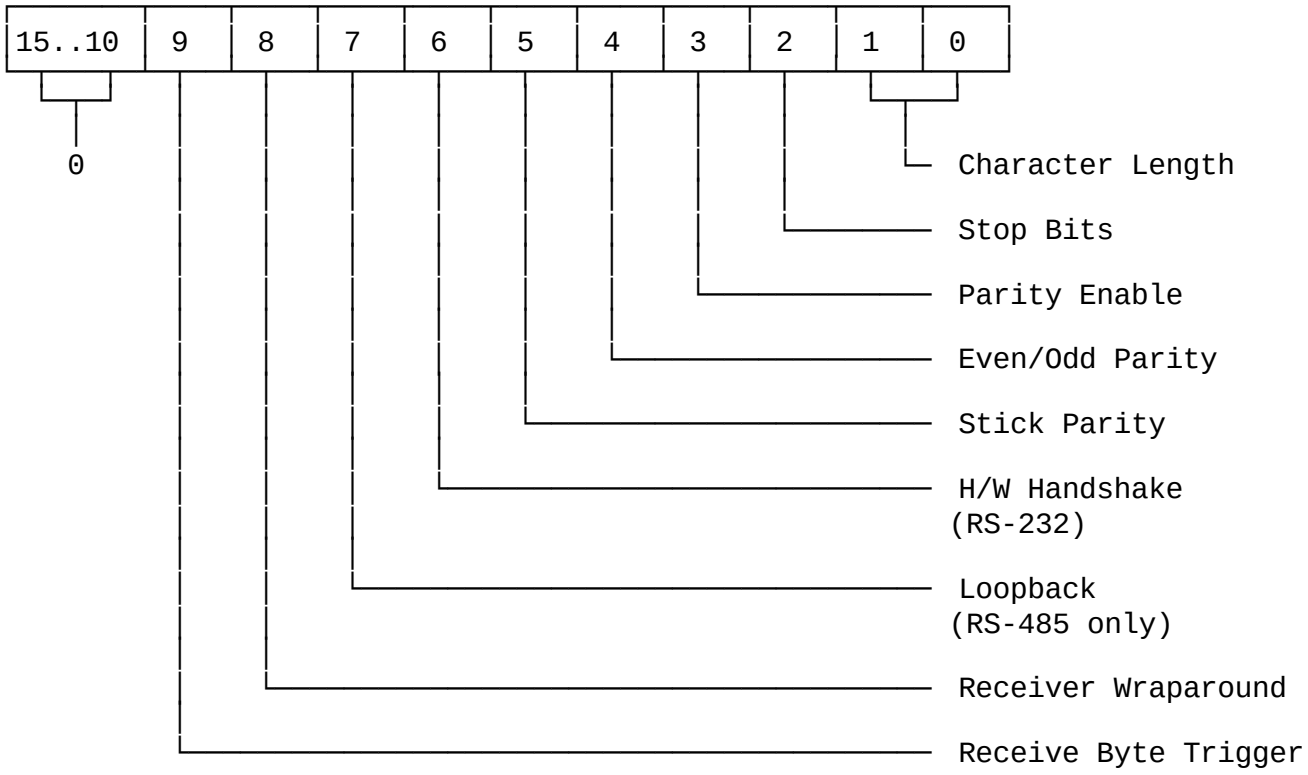
CHANNEL 8 CONFIGURATION REGISTER	7ECE
CHANNEL 8 BAUD RATE GENERATOR	7ECC
CHANNEL 8 RCV DATA START POINTER	7ECA
CHANNEL 8 RCV DATA END POINTER	7EC8
CHANNEL 8 RCV DATA CURRENT POINTER	7EC6
reserved	7EC4
reserved	7EC2
CHANNEL 8 RCV DATA BYTE COUNT REGISTER	7EC0
CHANNEL 8 RCV BUFFER WRAPAROUND REGISTER	7EBE
CHANNEL 8 RCV BYTE COUNTER TRIGGER	7EBC
reserved	7EBA
CHANNEL 8 RCV INTERVAL COUNTER TRIGGER	7EB8
CHANNEL 8 RCV ERROR COUNT REGISTER	7EB6
CHANNEL 8 RCV BYTE TRIGGER REGISTER	7EB4
CHANNEL 8 TX INSTRUCTION STACK POINTER	7EB2
CHANNEL 8 TX INSTRUCTION COUNTER	7EB0
CHANNEL 8 TX LOOP COUNTER	7EAE
reserved	7EAC
CHANNEL 8 INTERRUPT/TRIGGER CONDITION REG.	7EAA
CHANNEL 8 STATUS REGISTER	7EA8
reserved	7EA6
reserved	7EA4
CHANNEL 8 START TRANSMIT REGISTER	7EA2
CHANNEL 8 START RECEIVE REGISTER	7EA0

CHANNEL 9 CONTROL REGISTER BLOCK

CHANNEL 9 CONFIGURATION REGISTER	7EFE
CHANNEL 9 BAUD RATE GENERATOR	7EFC
CHANNEL 9 RCV DATA START POINTER	7EFA
CHANNEL 9 RCV DATA END POINTER	7EF8
CHANNEL 9 RCV DATA CURRENT POINTER	7EF6
reserved	7EF4
reserved	7EF2
CHANNEL 9 RCV DATA BYTE COUNT REGISTER	7EF0
CHANNEL 9 RCV BUFFER WRAPAROUND REGISTER	7EEE
CHANNEL 9 RCV BYTE COUNTER TRIGGER	7EEC
reserved	7EEA
CHANNEL 9 RCV INTERVAL COUNTER TRIGGER	7EE8
CHANNEL 9 RCV ERROR COUNT REGISTER	7EE6
CHANNEL 9 RCV BYTE TRIGGER REGISTER	7EE4
CHANNEL 9 TX INSTRUCTION STACK POINTER	7EE2
CHANNEL 9 TX INSTRUCTION COUNTER	7EE0
CHANNEL 9 TX LOOP COUNTER	7EDE
reserved	7EDC
CHANNEL 9 INTERRUPT/TRIGGER CONDITION REG.	7EDA
CHANNEL 9 STATUS REGISTER	7ED8
reserved	7ED6
reserved	7ED4
CHANNEL 9 START TRANSMIT REGISTER	7ED2
CHANNEL 9 START RECEIVE REGISTER	7ED0

CHANNEL x CONFIGURATION REGISTER

This register sets up various run parameters for both the receive and transmit channels.



Notes:

1. This register can only be written to when the respective channel is turned OFF (via the Start Register). (For ARINC channels, it is only possible to change this register if ALL channels are turned off. That is not the case for RS channels.)

2. It is recommended that ALL active channel Configuration Registers be set up immediately following the Board Handshake procedure before programming any other parameters.

CHANNEL x CONFIGURATION REGISTER; BIT DEFINITIONS

0,1	Char Length	This field determines the number of bits in each character sent.	Bit 1 0 0 1 1	Bit 0 0 1 0 1	Length 5 bits 6 bits 7 bits 8 bits
2	Stop bits	This bit specifies the number of Stop bits transmitted or received with each character. 0 = one stop bit is used. 1 = two stop bits are used for lengths 6,7 and 8. 1.5 stop bits are used if 5 bit chars were chosen.			
3	Parity	1 = Parity 0 = No Parity			
4	Even/Odd	1 = Even parity 0 = Odd parity (if bit 3 is 1)			
5	Stick Parity	If bits 3, 4 and 5 are all 1, the parity bit will always be set to zero. If 3 and 5 are 1 and 4 is zero, the parity bit will always be 1. If bit 5 is zero, regular Even and Odd parity will be used.			
6	H/W Protocol	1 = Transmit only when CTS is high, set DTR hi if in danger of overrun. 0 = Ignore CTS and DTR			
7	Loopback (RS-485 only)	1 = Channel receives even during transmission 0 = Channel receives at all times except during transmission (when receive is enabled)			
8	Receiver Wrap Around	1 = Data storage is halted when buffer full 0 = Receiver wraps around data within the block			
9	Receiver Byte	1 = Start data storage upon receipt of Byte xx. 0 = Receiver stores data without Start Byte Trigger			

Note:

1. Receiver Byte Trigger. See description of the RCV Start Byte Trigger Register.

2. If H/W protocol is enabled for RS-232, there will be an interbyte delay of at least 1 bit time.

RS Channels

CHANNEL x BAUD RATE GENERATOR REGISTER

This is a 16 bit register that must be filled in prior to starting the channel. The value in this register should be chosen as follows.
For a module with a 8.0000 MHz oscillator, the value is
value = 8000000 / (16 x baud).

For a module with a 7.3728 MHz oscillator, the value is
value = 7372800 / (16 x baud).

Example: Desired baud rate is 38,400 from a module with a 8 MHz oscillator.
8000000 / (16 x 38400) = 13.0208333
The Baud Rate Register should be set to 13 (decimal)

Minimum allowed baud rate is 50 baud.

The following tables show representative baud rates available for the two available oscillators.

7.37280 Mhz oscillator (AA/BB/CC/ZZ Modules)		
Desired Baud Rate	Contents of BR Generator Reg.	% Error
50	9216	-
75	6144	-
134.5	3426	0.001
150	3072	-
300	1536	-
600	768	-
1000	461	0.043
1200	384	-
1800	256	-
2000	230	0.043
2400	192	-
3600	128	-
4800	96	-
7200	64	-
9600	48	-
10000	46	0.174
19200	24	-
38400	12	-
76800	6	-
153600	3	-
230000	2	0.174

8.00000 Mhz oscillator (A/B/C/Z Modules)		
Desired Baud Rate	Contents of BR Generator Reg.	% Error
50	10000	-
75	6667	0.005
110	4545	0.010
134.5	3717	0.013
150	3333	0.010
300	1667	0.020
600	833	0.040
1000	500	-
1200	417	0.080
1800	277	0.080
2000	250	-
2400	208	0.160
3600	139	0.080
4800	104	0.160
7200	69	0.644
9600	52	0.160
19200	26	0.160
38400	13	0.160
56000	9	0.790
128000	4	2.344
256000	2	2.344

Note: depending on the configuration of the MAGICard, for very high baud rates there may be lost bytes on receive if DTR and CTS are not used, and an inter-byte

gap on transmit. See the section "Baud Rate Limitations".

RS Channels

CHANNEL x RECEIVE DATA START POINTER (WR)

This register is used to set the start address of the receive data buffer. The address must be even.

CHANNEL x RECEIVE DATA END POINTER (WR)

Sets the end address of the receiver data buffer. The data will wrap around or stop depending upon the Receiver Wrap Around control bit within the Configuration Register.

CHANNEL x RECEIVE DATA CURRENT POINTER (RD)

Indicates the address where the next byte is to be placed within the buffer. This pointer value is incremented after the entire receiver block (data byte, time tag, and status) is written into memory.

CHANNEL x RCV DATA BYTE COUNTER REGISTER (RD)

Indicates the number of bytes received (0-64k). This register wraps around. It may be reset to zero by the user only when the channel is stopped.

CHANNEL x RCV BUFFER WRAPAROUND REGISTER (WR)

This register contains 2 bits for synchronization with the "C" drivers. If bit 14 is set to 1, the receive buffer has wrapped around once since the last data read. If bit 15 is set to 1, there have been multiple wraparounds.

CHANNEL x RCV DATA BYTE COUNTER TRIGGER REGISTER (WR)

This register allows the user to generate an interrupt and set a flag when a certain number of bytes have been received (1-64k). The appropriate bit must also be set in the Channel x Interrupt/Trigger Condition Register.

CHANNEL x RCV INTERVAL COUNTER TRIGGER REGISTER (WR)

This register allows the user to generate an interrupt and set a flag upon reception of every "N" number of bytes. The appropriate bit must also be set in

the Channel x Interrupt/Trigger Condition Register.

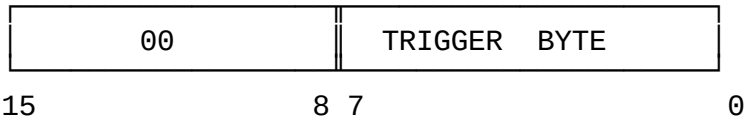
RS Channels

CHANNEL x RCV ERROR COUNT REGISTER (RD)

This 16-bit counter indicates the number of errors received on a particular channel. This register wraps around. The user can reset this register by writing 0 to it even while the channel is running.

CHANNEL x RCV START BYTE TRIGGER REGISTER (WR)

This register is used in conjunction with the Receiver Byte Trigger bit within the Configuration Register to enable the reception and storage of data upon receipt of a specific byte.



CHANNEL x TX INSTRUCTION STACK POINTER (WR)

Sets the starting address of the TX Instruction Stack. The address must be even.

CHANNEL x TX INSTRUCTION COUNTER (WR)

Sets the number of TX Instruction blocks to process.

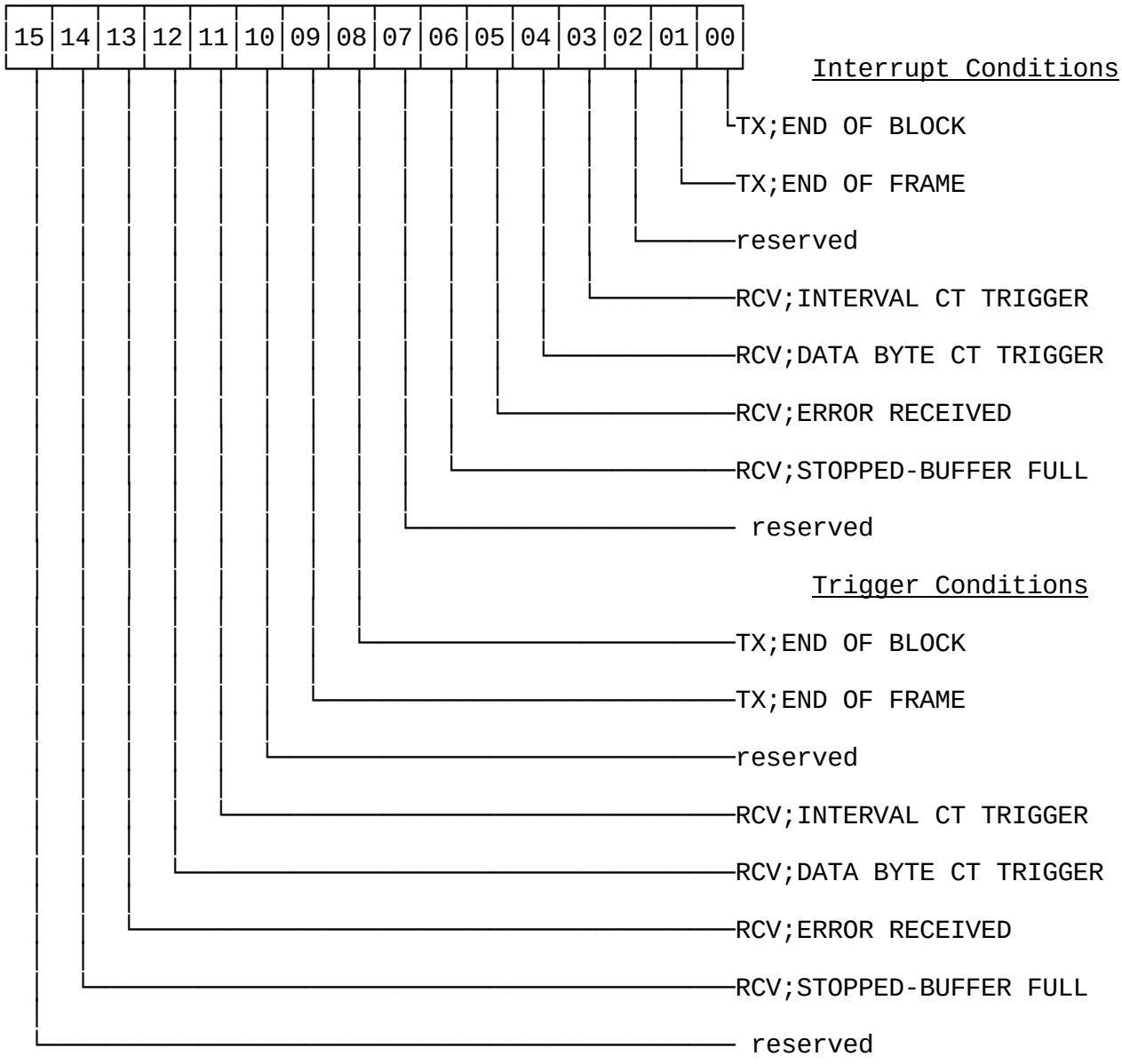
CHANNEL x TX LOOP COUNTER (WR)

Sets the number of times to execute the TX instruction blocks; `N' Times or Continuous Loop. If the continuous value is selected, the channel's operation can be terminated by setting the related channel bit within the Global Start Register to a 0, or its Start Transmit Register to 0.

Value:
0000 = CONTINUOUS
0001 = One Time
0002 = Two Times
.
.
FFFF = 65535 Times

CHANNEL x INTERRUPT/TRIGGER CONDITION REGISTER (WR)

Sets the Interrupt and Trigger condition(s) of the board. Bits 00-07 are the interrupt condition bits while bits 08-15 relate to the hardware trigger bits. The trigger conditions set a pulse on the trigger [subminiature BNC] J1 connector or the VXI "TRIG0$" TTL TRIGGER output signal.



Notes:

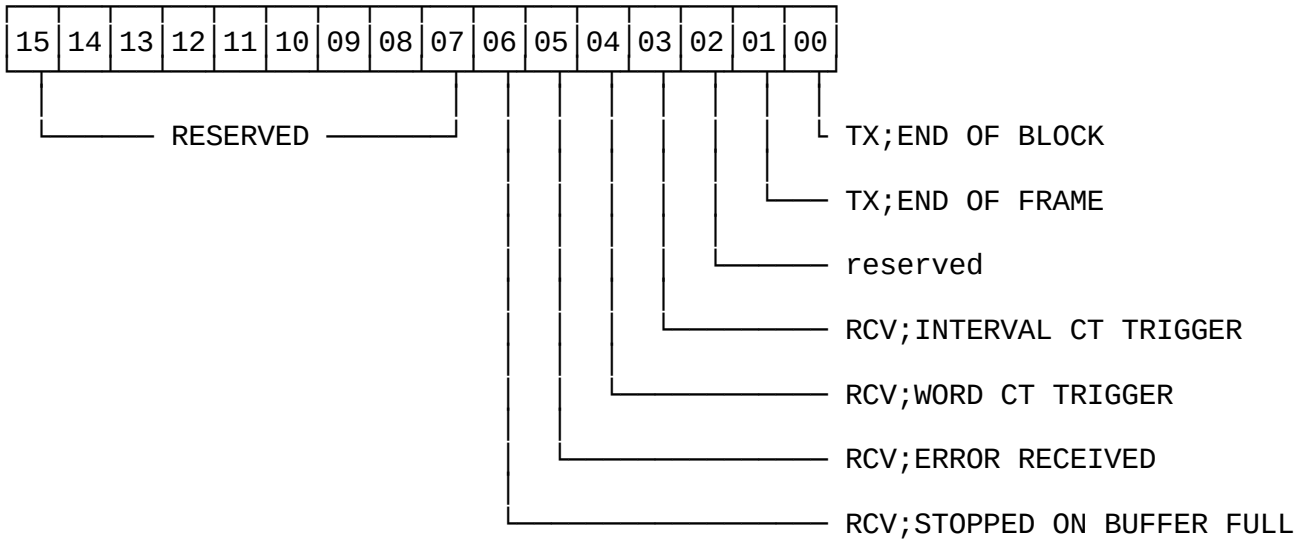
- To activate the RCV;INTERVAL CT TRIGGER interrupt or trigger, the Channel x Rcv Interval Counter Trigger Register must also be set.

RS Channels

- To activate the RCV;DATA WORD CT TRIGGER interrupt or trigger, the Channel x Rcv Word Counter Trigger Reg must also be set.

CHANNEL x STATUS REGISTER (RD)

This register indicates the operational status of the channel. This register can be used to poll the status of the channel purposes or can be used with interrupts. In this case, the register indicates the condition(s) which caused the interrupt. A logic "1" indicates an active bit. Status bits are only reset by the user or a reset.



CHANNEL x START TRANSMIT REGISTER (WR)

When the channel is enabled via the Global Start Register, writing a one to this register initiates transmission according to the transmit instruction block and data which has been previously set up. Setting it to zero will terminate the transmission, but allow the receive operation to continue. It is permissible to set this register to zero, update the transmit command block and data, and restore the register to one while the channel is enabled. When the requested transmit operation is completed, this register is reset to zero.

CHANNEL x START RECEIVE REGISTER (WR)

When the channel is enabled via the Global Start Register, writing a one to this register initiates receive according to the receive parameters which have been previously set up. Setting it to zero will terminate the receive operation, but allow transmission to continue. It is permissible to set this register to zero, update the receive parameters, and restore the register to one while the channel is enabled. When the requested receive operation is completed, this register is reset to zero.

RS Channels

RECEIVER / MONITOR OPERATION

GENERAL INFORMATION:

The user sets up each channel's mode of receive operation by writing to the various Channel Control Registers (one set per channel). In sequential mode, the data bytes are stored with a status byte and a 32-bit time tag value. In Data Only Mode and Fast Operation Mode, only data is stored.

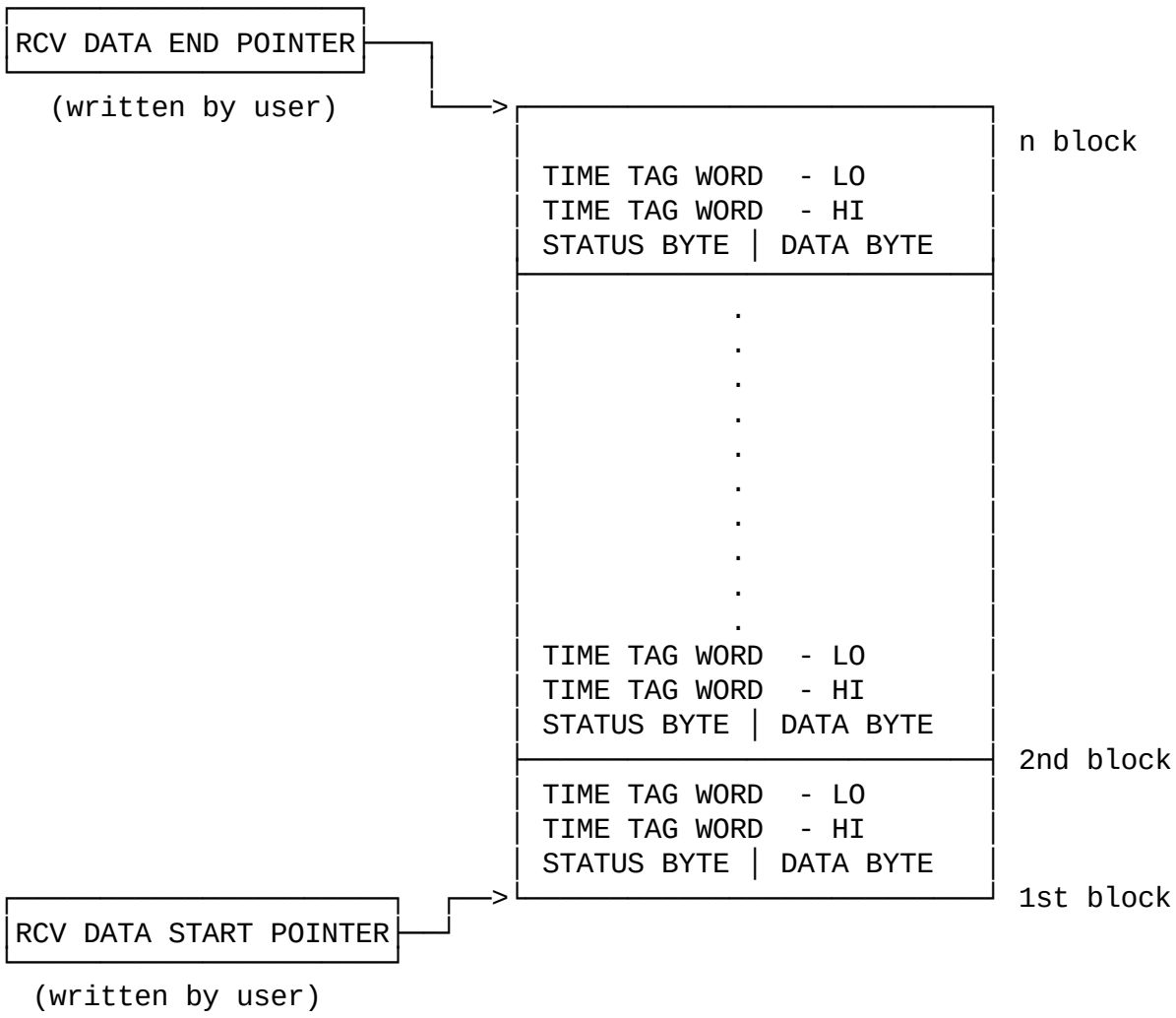
OPERATION:

The board is initialized in a wait loop - looking for a START command from the computer. This command, issued by writing to the Global Start Register, instructs the board to begin operation on the ACTIVE channel(s). Additionally, the channel Start Receive Register must be written to.

The data is stored in sequential order. The data buffer's size and location within the memory is programmed via a Start and End pointer. Each received data byte has an accompanying Status Byte and a 32-bit Time Tag value. These six bytes make up a single receive data block. In addition, the data can be stored without the Time Tag or Status. This is global selection which affects all receiver channels (see: Receiver Data Storage Mode Register in the Global Register section of this manual). Interrupts and pollable status registers allow for numerous event recognition and are described in the Channel Register section of this manual).

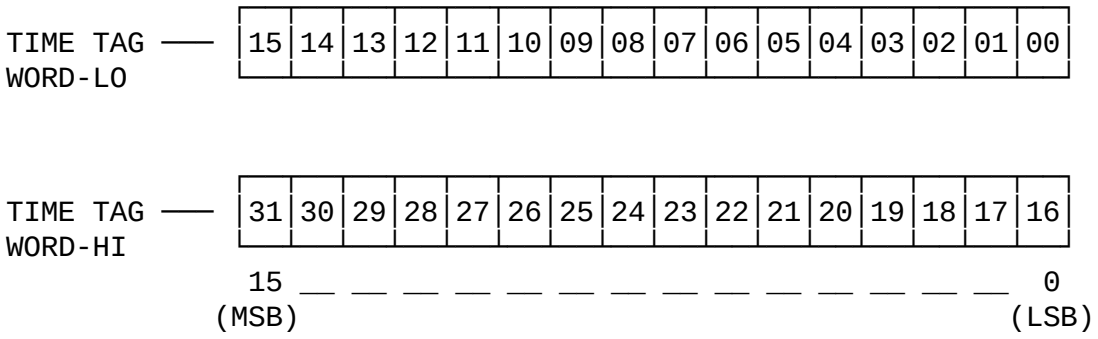
RECEIVE BUFFER STORAGE SEQUENCE

The drawing below illustrates the way in which the receive data blocks are stored within the dual-port RAM while in the sequential mode of operation. The Start and End pointers set up the buffer size. The receive data storage will stop when the end pointer is reached or will wrap around to the beginning of the buffer - depending upon the condition of the Receiver Wrap Around bit within the Channel Configuration Register. The Time Tag resolution is 10 µsec/bit. The contents of the Receiver Status Byte are described within this section.

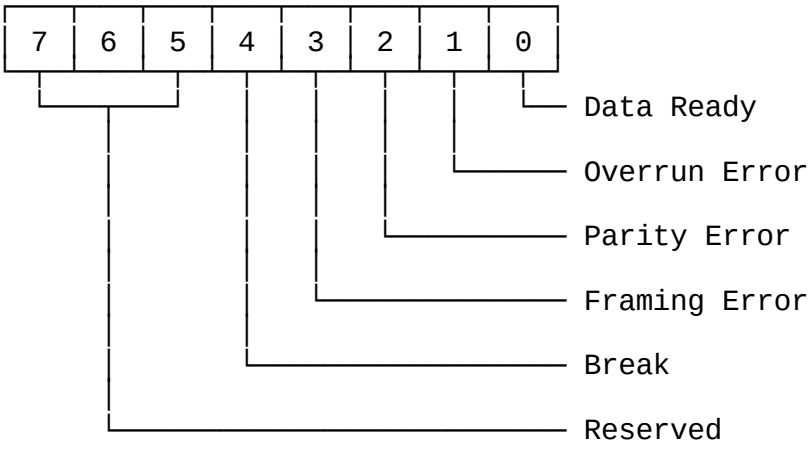


TIME TAG WORD FORMAT

As stated above, the Time Tag is a 32-bit word made up of two 16-bit words; Time Tag-Hi followed by a Time Tag-Lo. The resolution of the time tag is 10 µsec/bit.



RECEIVER STATUS BYTE



Status Byte Bit Definitions

0	Data Ready	Indicates a byte has been received
1	Overrun	Indicates a byte was lost. This can occur at speeds greater than 19200 baud when many channels are working simultaneously.
2	Parity Err	Indicates received parity did not match parity chosen in Configuration Register.
3	Frame Err	Indicates received character did not have a valid stop bit
4	Break	Indicates receipt of a Break character
5-7	Reserved	Set to 0

TRANSMITTER OPERATION

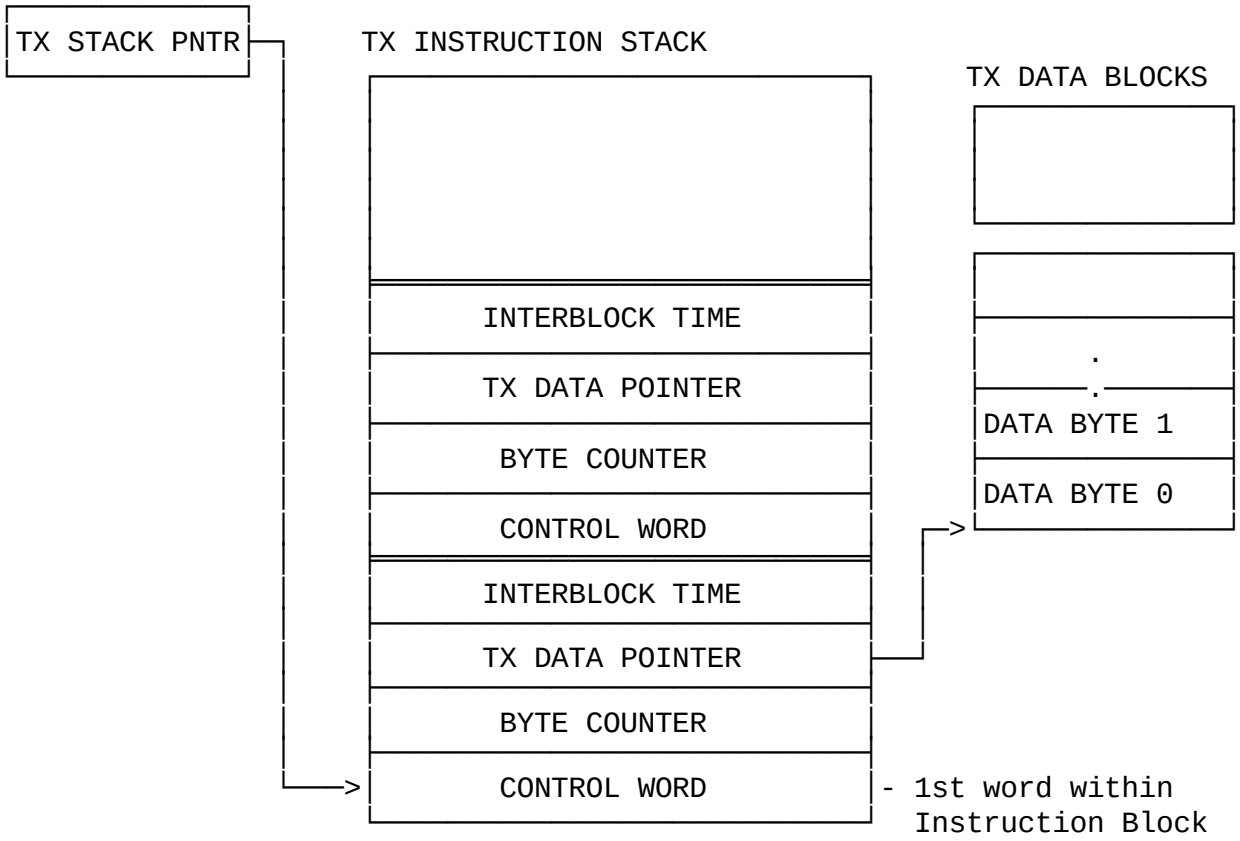
In order to initiate a transmission, both the Global Start Register and the Channel Start Transmit Register must be written to. The user must create an instruction stack for the transmitter channel and write the data into the Dual-Port RAM before writing to the Start Transmit Register. The Channel Configuration Register and the Channel Baud Rate Generator must be set before writing to the Global Start Register. It is permissible to write a zero to the Channel Start Transmit Register, update the instruction stack and data, then restart transmission by restoring it to one without writing a zero to the Global Start Register. This is necessary in order not to interfere with the receive operation.

TRANSMIT INSTRUCTION STACK

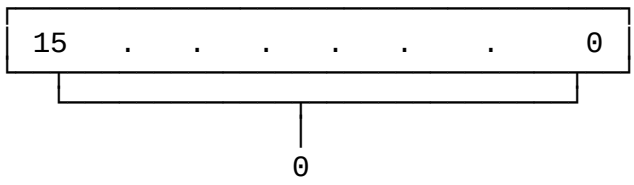
The Transmit Instruction Stack is divided into instruction blocks - each containing 4 words. Each section relates to a data block. A data block is composed of one or more bytes which the user desires to transmit contiguously. The stack is sequential, so that the first instruction block relates to the first data block, the second to the second data block, etc.

As stated, each section contains 4 words. The first word is the Control Word which is reserved for future use. The second word contains a 16-bit byte_count which instructs the board as to the number of serial bytes to transmit within a particular block. The third word contains a 16-bit, user-supplied data pointer. This is a 16-bit address (must be even) which points to the beginning of the data within the memory. The fourth word is the Interblock Time value and is used to program the time between blocks.

TRANSMIT BLOCK DIAGRAM



CONTROL WORD DEFINITION (1st word within Instruction Block)



This field is reserved for future use.

RS Channels

BYTE COUNTER

The Byte Counter is used to specify the number of data bytes within this data block (1 - 65535).

TX DATA POINTER

This register is used to set the start address of the transmit data block. The size of the block is determined by the Byte Count value.

INTERBLOCK TIME

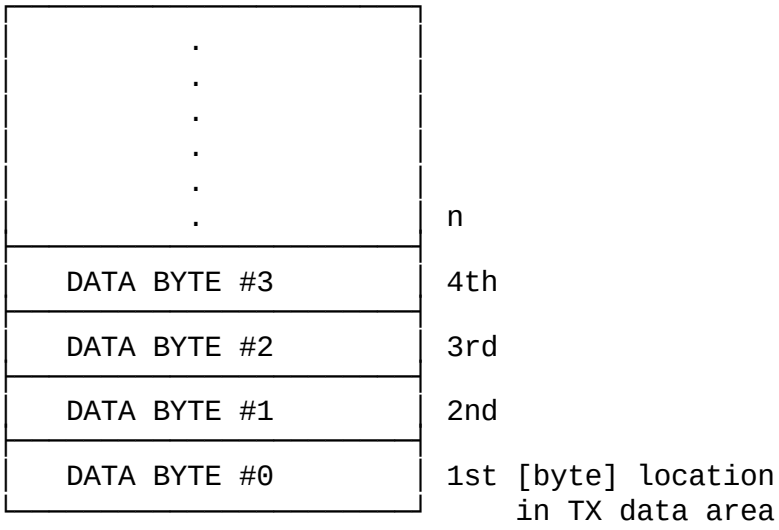
Instruction Blocks are accessed sequentially and their associated data bytes transmitted according to this sequential order. The Interblock Time allows the user to specify the time between data blocks. The resolution of this 16-bit word is one bit time according to the transmission bit rate. The minimum allowed value is 1.

RS-485 OPERATION

RS-485 transmitters go into tristate whenever they are not transmitting a block in order to provide other transmitters with an opportunity to safely transmit data. This includes the Interblock Time and the time between frames. If there are gaps between bytes within a block (which can happen at high baud rates) the transmitter does not enter tristate. The transmitter goes into tristate immediately after the last stop bit of the last byte of a block has been transmitted, and remains tristated until the beginning of the transmission of the start bit of the next byte.

TX DATA BLOCK FORMAT

The figure below illustrates the format of the TX data bytes within the memory.



BAUD RATE LIMITATIONS

There are configurations of the MAGICard for which the baud rate of the RS-232/422/485/423 channels must be less than the maximum of 250 Kbaud. If not, there will be interbyte gaps in transmission, and a possibility of losing received data. There are two modes of operation (see the description of the RS-232/422/485/423 Channel Operation Mode Register). In regular mode, each received byte may be stored with status and time tag. The allowed baud rates are relatively low, but concurrent high speed ARINC operation is not disturbed. In fast mode, each received byte is stored without status and time tag. High baud rates are acceptable, but high speed ARINC channels may lose data. The following equations are meant to be a guide to allow the user to determine which baud rates may be used without data loss. The maximum baud rates are approximate since these values are dependent upon many factors such as the size of transmitted blocks, and whether or not interrupts are used.
In all equations, BR is the baud rate in Khz, R is the number of RS-232/422/485/423 channels running on the board, and A is the number of ARINC channels running on the board (a Williamsburg channel set counts as 3 ARINC channels).

For fast operation with channels working half-duplex (receiving only), the approximate maximum baud rate which will not result in data loss is

$$BR(Khz) = \frac{160{,}000}{120 + 188xR + 50xA}$$

For fast operation with channels working full-duplex (transmitting and receiving), the approximate maximum baud rate which will not result in data loss is

$$BR(Khz) = \frac{160{,}000}{120 + 372xR + 50xA}$$

For the regular operation mode with channels working full-duplex (transmitting and receiving), the approximate maximum baud rate which will not result in data loss is

$$BR(Khz) = \frac{10{,}000}{120 + 51xR + 50xA}$$

Note: The maximum allowed baud rate for RS-232/422/485/423 channels is 250 Kbaud even if the calculated value of BR is greater.

RS Channels

VME/VXI INTERFACE

The MAGICard board interfaces to the computer via a 16-bit data bus which can be accessed in bytes or words. The board may be accessed by using addresses in the form:

For accessing VME/VXI Configuration Registers:
XXXX (H) (A16 mode) - with ADDRESS MODIFIER CODES: 29, 2D

For accessing Data Storage Area and Control Registers:
XX XXXX (H) (A24 mode) - with ADDRESS MODIFIER CODES: 39, 3A, 3D, 3E
XXXX XXXX (H) (A32 mode) - with ADDRESS MODIFIER CODES: 09, 0A, 0D, 0E

selectable via jumper JP1.

The MAGICard memory map is divided into two distinct blocks:

1. VXI/VME Configuration Registers.
2. Data Storage Area and Control Registers.

The VXI/VME Configuration Registers are used for mapping and setting parameters of the MAGICard within the user's VME or VXI system. The Data Storage Area and Control Registers are used to control the operation of the MAGICard.

VXI/VME Configuration Registers

The VXI/VME Configuration registers are located within a 64 byte block in the A16 address space between the addresses 49152 (dec.) and 65472 (dec.). The base address of the Configuration registers is determined by the following equation:

```
Base Address (dec.) = V*64 + 49152 (dec.)
```

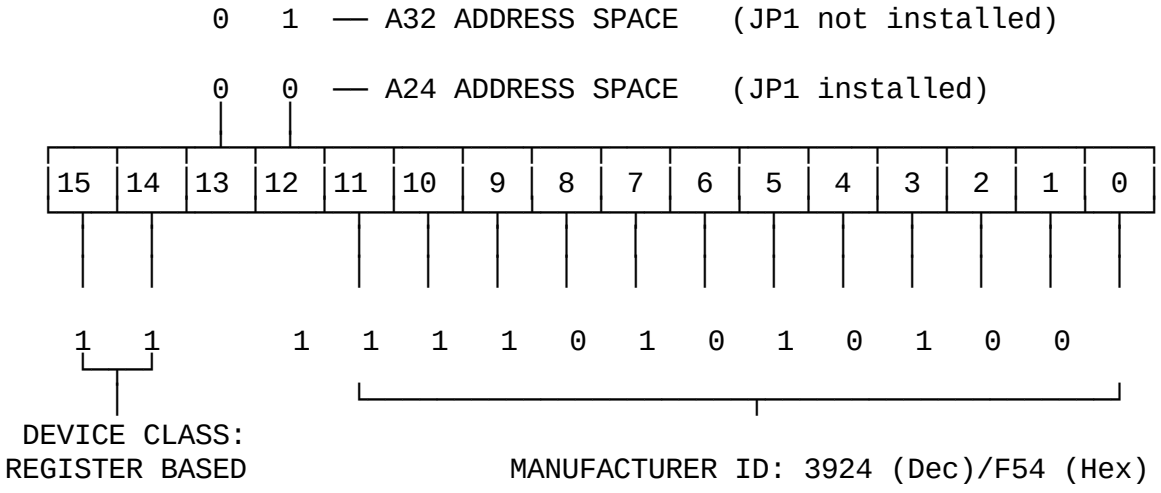
V, the Logical Address of the card, is an integer which varies between 0 and 255 and is defined by the user via the 8 pole dipswitch SW1 (see the section on dipswitch setup at the end of this manual). In order to ensure correct operation of the MAGICard within the user's VME or VXI system, the configuration registers must be (re-)initialized after power up or after assertion of SYSRESET*. For a full explanation of the VXI Configuration registers and other topics relating to operation of the VXI bus refer to the "VXI Bus System Specification"

Configuration Register Memory Map

ADAPTER BOARD IDVECT	BASE + 22 (H)
IDVECT	BASE + 20 (H)
OFFSET REGISTER	BASE + 06 (H)
STATUS/CONTROL REGISTER	BASE + 04 (H)
DEVICE TYPE	BASE + 02 (H)
ID REGISTER	BASE + 00 (H)

ID REGISTER (VXI only) BASE + 00 (Read)

The contents of this 16-bit register provides the following information about the MAGICard's configuration.

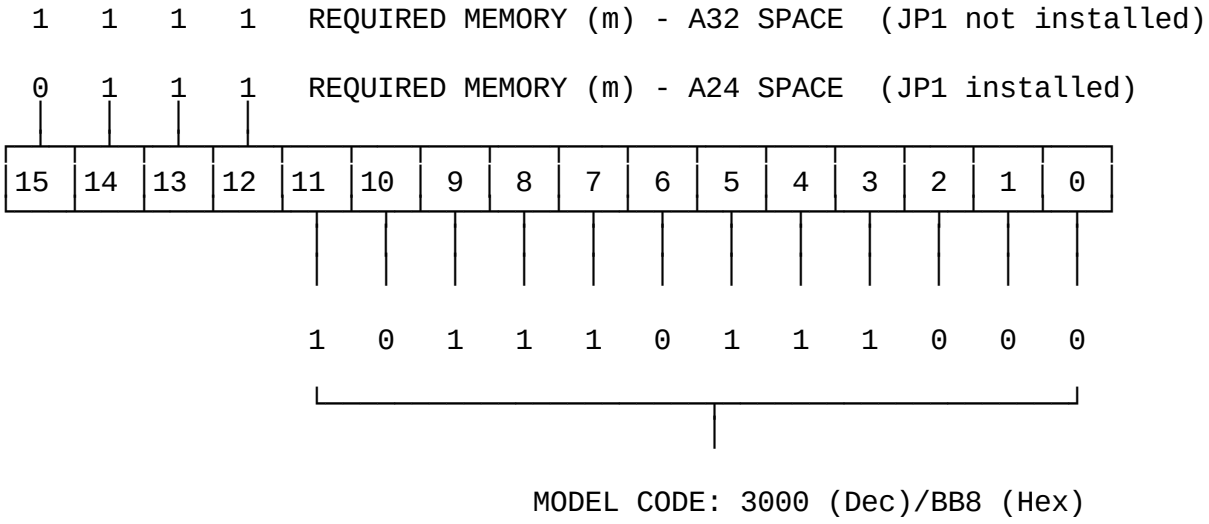


Note: This register contains the same value whether set up for VME or VXI installation. The VXI specification requires all VXI devices to identify themselves via the above fields. This location is not defined under the VME specification.

DEVICE TYPE REGISTER (VXI only)

BASE + 02 (Read)

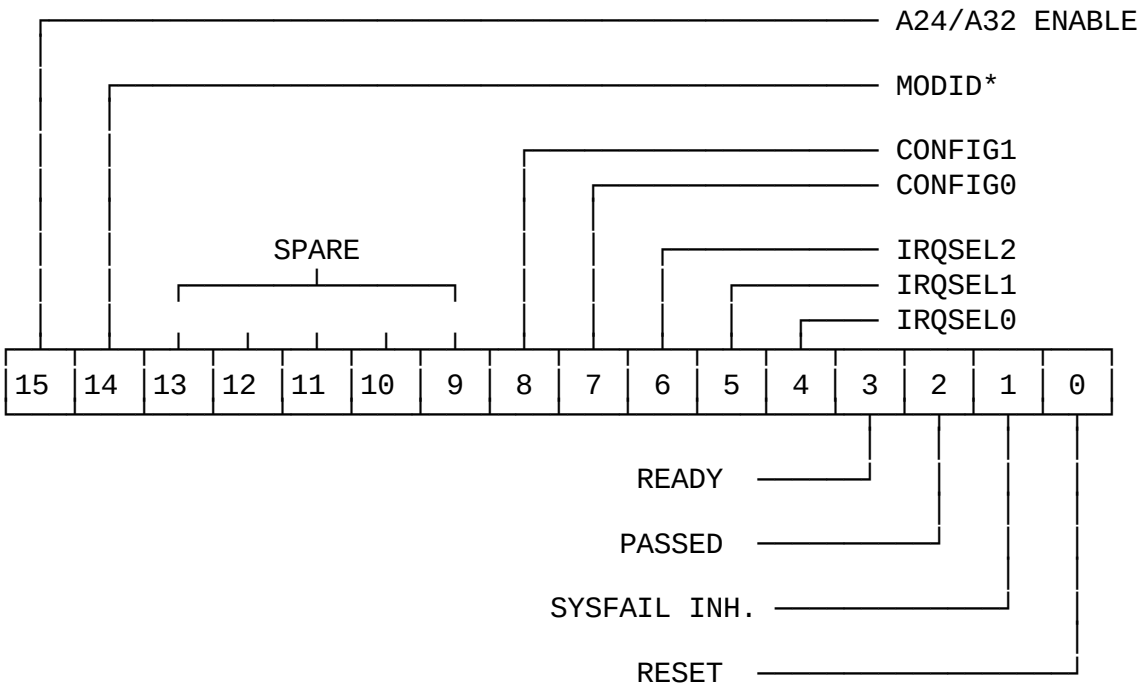
This 16 bit register contains a fixed Device Type Identifier as well as a four bit field which reflects the Required Memory usage of the card.



Note: This register contains the same value whether set up for VME or VXI installation. The VXI specification requires all VXI devices to identify themselves via the above fields. This location is not defined under the VME specification.

STATUS REGISTER (VXI and VME)	BASE + 04 (Read)

A read of this 16 bit register provides information about the card's
as defined below.

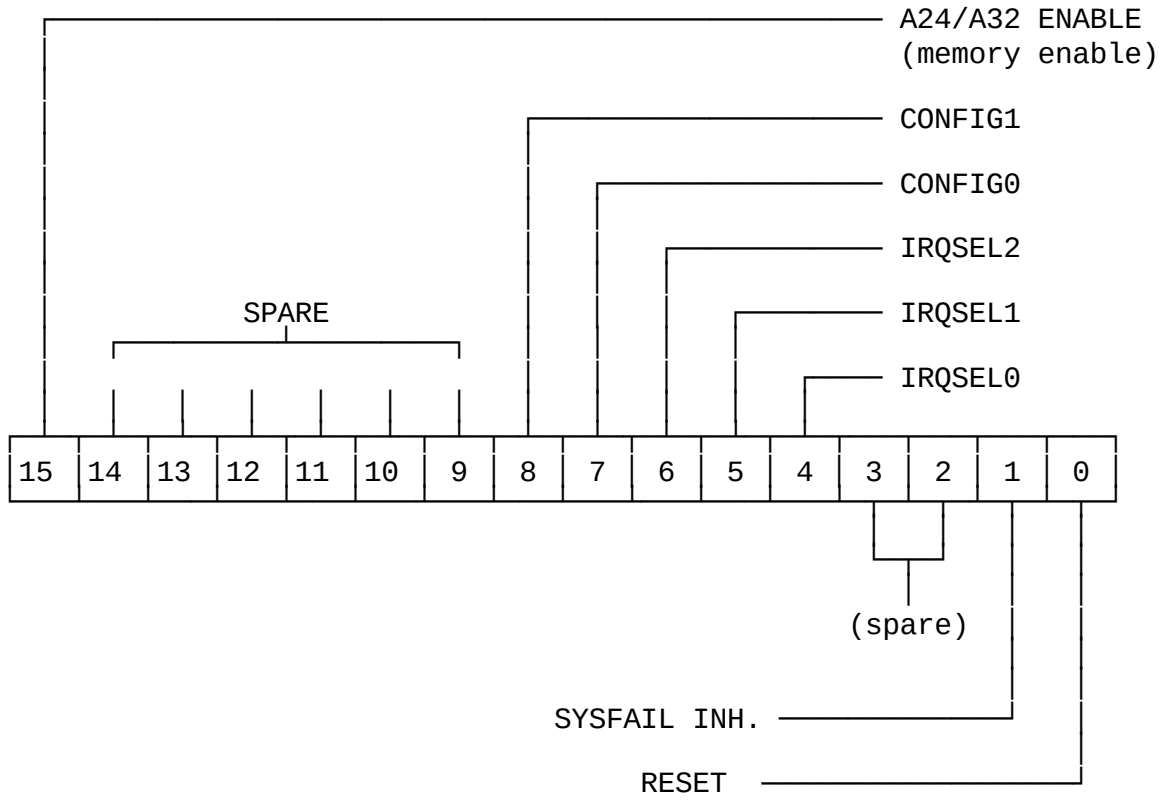


Bit	Name	Definition
0	RESET	Indicates the state of the RESET bit in the Control Reg.
1	SYSFAIL INHIBIT	Indicates the state of the SYSFAIL INHIBIT bit in the Control Register.
2	PASSED	A "0" indicates that the board is either executing or has failed it's self test. A "1" indicates that the self test has successfully completed.
3	READY	A "1" indicates that the card is ready to accept commands
4-6	IRQSEL 2-0	Indicates the state of the IRQSEL2-0 bits in the Control Register
7	CONFIG0	Indicates the state of the CONFIG0 bit in the Control Reg
8	CONFIG1	Indicates the state of the CONFIG1 bit in the Control Reg
14	MODID*	Indicates the inverted value of the VXI bus "MODID" line.
15	A24/A32 ENABLE	Indicates the state of the A24/A32 ENABLE bit in the Control Register.

Note: The RESET, SYSFAIL INH., PASSED, READY and MODID* bits are included to maintain compliance with the VXI specification.

CONTROL REGISTER (VXI and VME) BASE + 04 (Write)

Writing to this 16-bit register causes the actions listed below to be executed by the card. Note that all bits in this register are set to 0 after assertion of VME bus line SYSRESET*.



RESET

Writing a 1 to this bit forces the card into the "RESET" state. The user must not write a 0 into this bit for at least 100 μsec. after writing a 1 into it. That is, once in the "RESET" state, the card must remain in this state for at least 100 μsec. While in the "RESET" state the card is completely inactive and will not respond to any commands. Upon releasing the card from the "RESET" state (write 0 to this bit), the card will perform its self-test routines. The board may also be reset via the Software Reset Register defined whithin the main body of this manual. This second method is the preferred mechanism for resetting the card.

SYSFAIL INHIBIT

Writing a 1 to this bit disables the card from driving the VME bus line SYSFAIL*, in the case when JP5 is shorted (see Jumpers), otherwise it has no effect.

IRQSEL 2-0

Writing to these bits selects which one of the VME bus Interrupt Request lines IRQ1* -- IRQ7* will be driven active when the card generates an interrupt. The following table shows the relationship between IRQSEL 2—0 and IRQ7-1.

SELECTED INTERRUPT LINE	IRQSEL2	IRQSEL1	IRQSEL0
NONE	0	0	0
IRQ1*	0	0	1
IRQ2*	0	1	0
IRQ3*	0	1	1
IRQ4*	1	0	0
IRQ5*	1	0	1
IRQ6*	1	1	0
IRQ7*	1	1	1

USING INTERRUPTS

Note:

The interrupt generated on the selected IRQ* line is the "logical OR" of the two interrupt generating sources on the card. An interrupt which was generated by the board Controller will result in the interrupt routine whose vector resides in the IDVECT register. The card will place the value in the IDVECT register, called the STATUS/ID, onto the VME data lines when issuing the interrupt acknowledge cycle. The user's processor will use this value to determine which entry in the user's interrupt vector table to jump to. Within this interrupt routine the actual cause of the interrupt can be determined by polling the INTERRUPT_STATUS Register. Likewise, an interrupt which was generated by the ADAPTER BOARD will result in the interrupt routine whose vector resides in the ADAPTER_BOARD_IDVECT register.

For all interrupts, the interrupt request is cleared automatically at the end of the interrupt acknowledge cycle. This method is referred to within the VME specification as ROAK (Release On Acknowledge).

CONFIG 0,1

Reserved for future use.

A24/A32 ENABLE (Memory enable)

Writing a 1 to this bit enables access to the card's A24 or A32 VME bus registers and memory. If this bit is set to 0 none of the on card registers and memory which are resident in the A24 or A32 address spaces may be accessed. The Configuration registers, of course remain accessible regardless of the state of this bit, as they reside in the A16 address space of the card.

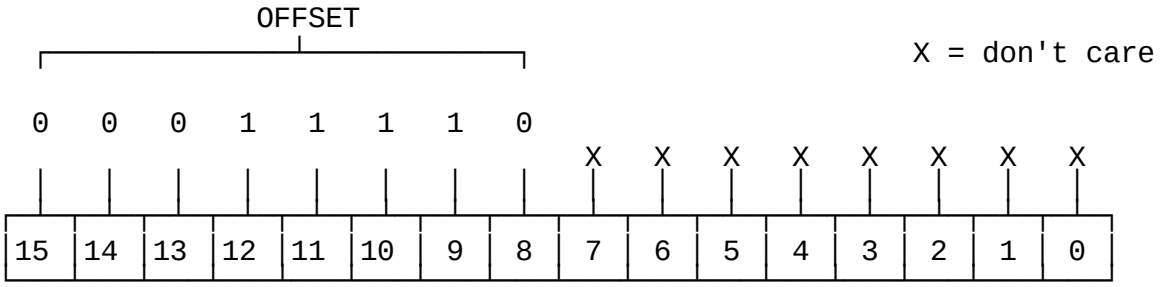
OFFSET REGISTER (VXI and VME)

BASE + 06 (Write/Read)

This 16 bit read/write register defines the base address of the card's A24 and A32 memory and registers. The m+1 most significant bits of the Offset register are the values of the m+1 most significant bits of the card's A24 and A32 memory and register addresses, where "m" is the Required Memory field of the card's Device Type register. The "15-m" least significant bits of the Offset register have no meaning. Thus, the Offset register bits 15 through "15-m" map to the address lines A23 through "A23-m" for the A24 Address Space, and to lines A31 through "A31-m" for the A32 Address Space.

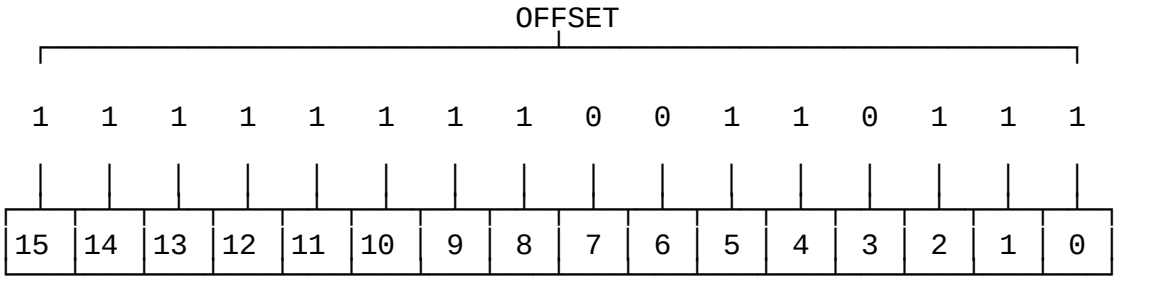
A24 ADDRESSING EXAMPLE

Given:
required base address = 1E 0000(H);
Then: write 1E(H) to Offset register



A32 ADDRESSING EXAMPLE

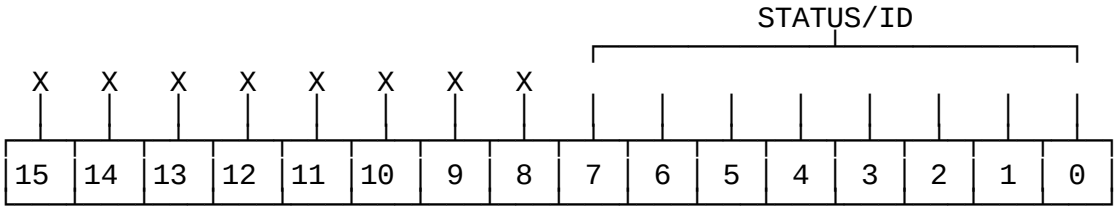
Given:
required base address = FF37 0000(H);
Then: write FF37(H) to Offset register



Note: Once this register is set-up and the Control Register's `A24/A32 ENABLE' bit is set, the card's A24 or A32 memory and registers can be accessed.

IDVECT REGISTER (VXI and VME) BASE + 20 (H) (Write/Read)

In the case of an interrupt generated by the board, the 8 least significant bits of this 16-bit register, known as the STATUS/ID, are used as the interrupt vector during the ensuing interrupt acknowledge cycle. The card is a D08(0) INTERRUPTER, and as a result will place these 8 bits on lines D00-D07 of the VME bus during the interrupt acknowledge cycle.

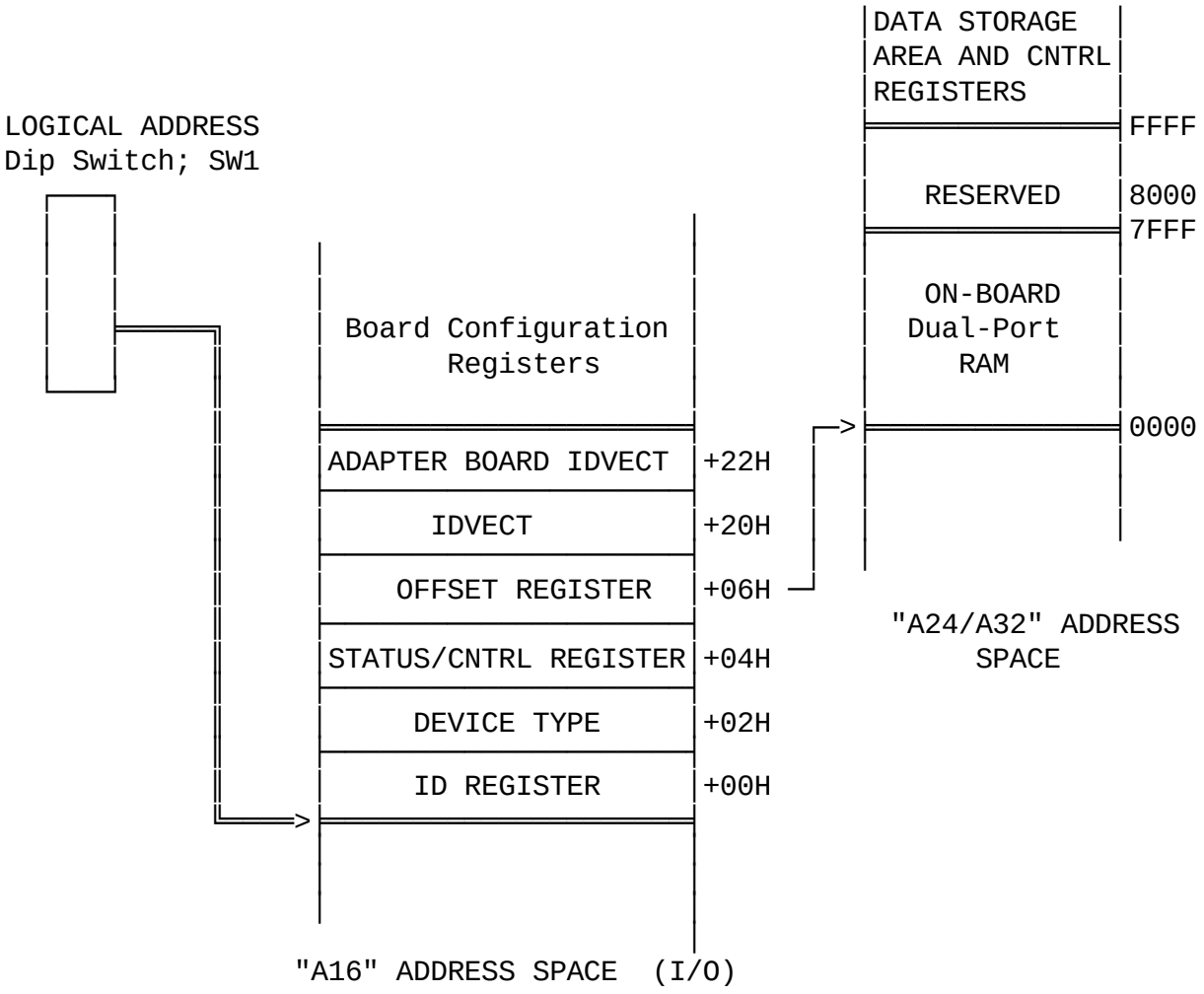


X = don't care

ADAPTER BOARD IDVECT REGISTER (VXI and VME) BASE + 22 (H) (Write/Read)

(SAME AS ABOVE)

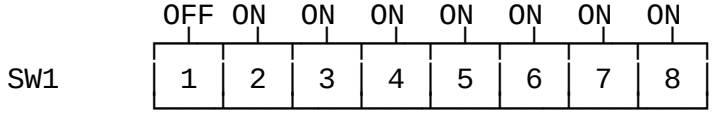
DUAL-PORT RAM ADDRESS MAPPING DIAGRAM



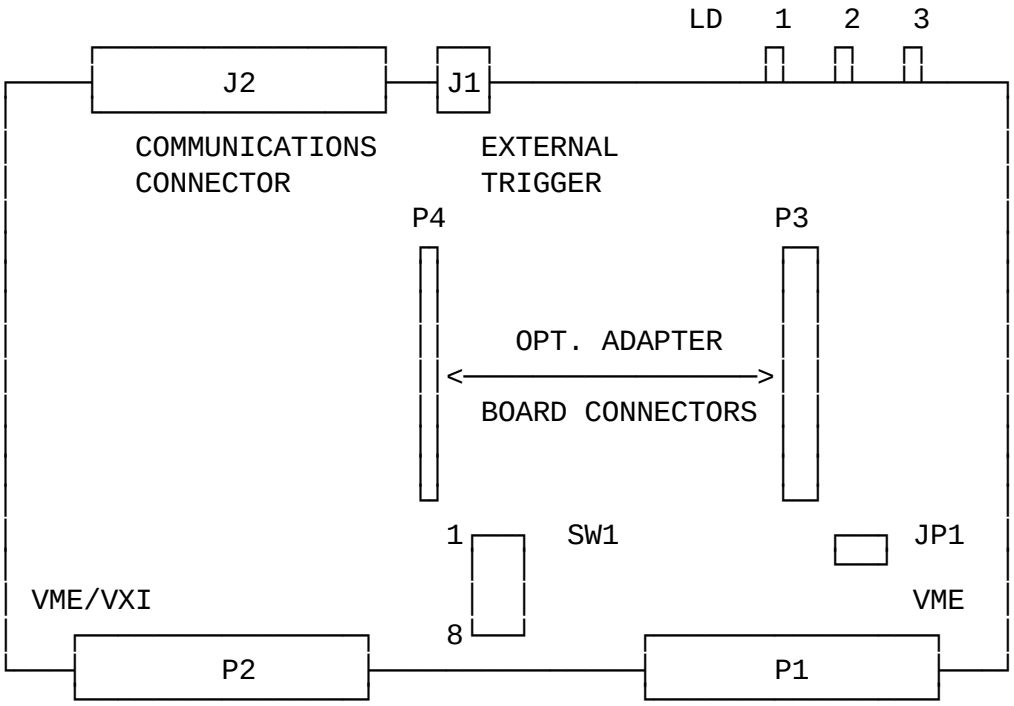
A16 ADDRESSING EXAMPLE

Given:
  required configuration registers base address = E000(H)

Then:
  set dip-switch SW1 to LOGICAL ADDRESS = 80(H)



BOARD LAYOUT



Note: B-size card shown

LEDS

The individual functions of the front panel leds are listed below.

MODID (LD3) - Reflects the state of the MODID pin on the VXI bus (JP19 must be installed). This LED has no function in a VME system.

PASSED (LD2) - Indicates that the card passed the power-on self test routine. (Reflects the state of the same bit in the Configuration Status Registers).

READY (LD1) - Indicates that the card is ready to receive commands. (Reflects the state of the same bit in the Configuration Status Register).

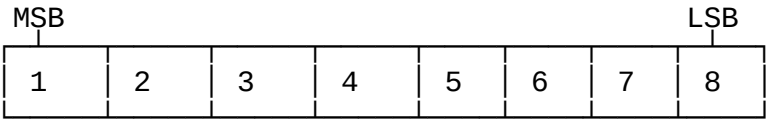
DIP SWITCH SETTINGS

The MAGICard contains 1 Dip Switch which controls the Logical Address of board. The definition of the switch is described below.

CARD LOGICAL ADDRESS DIP SWITCH SETTINGS

Dipwitch SW1 is used to select the card's Logical Address as described in the section "Configuration Registers". The Logical Address is set as shown below.

Logical Address Switch (SW1)



Switch "ON" or "Closed" = logic 0 at bit position
Switch "OFF" or "OPEN" = logic 1 at bit position

Example: for a logical address of 80 Hex, set position "1" to "OFF" or "OPEN" and ALL other switches to "ON" or "CLOSED".

FACTORY DEFAULT DIP SWITCH SETTINGS

SW1 is set to Logical Address 80H (1 OFF, 2 to 8 ON).

JUMPERS

Unless otherwise specified, all jumpers should be normally "out". Because the placement of jumpers is user specific, care should be taken that signals which the user chooses to jumper are not already in use on the VME bus.

VME AND VXI RELATED JUMPERS

JP5	SYSFAIL*	P1-C10	Connects card SYSFAIL* to backplane
JP2		------	Factory set
JP1	A32SEL	------	Jumper in: A24 address space Jumper out: A32 address space
JP30	EXT. TRIG. POLARITY (J1)	------	Pins 1 and 2 shorted : positive pulses. Pins 2 and 3 shorted : negative pulses.

VXI RELATED JUMPERS To continue the VXI Local Bus

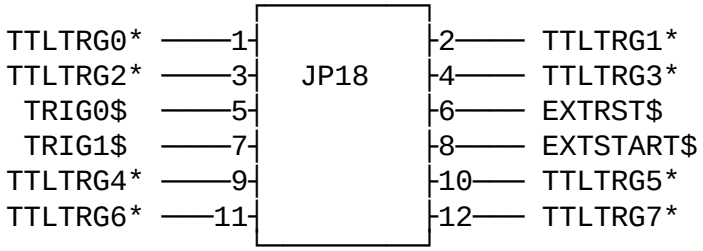
JP15 LBUS00 - (P2-A05) Connects LBUSA00 to LBUSC00 (P2-C05)
JP16 LBUS01 - (P2-A06) Connects LBUSA01 to LBUSC01 (P2-C06)
JP17 LBUS02 - (P2-A08) Connects LBUSA02 to LBUSC02 (P2-C08)
JP6 LBUS03 - (P2-A09) Connects LBUSA03 to LBUSC03 (P2-C09)
JP7 LBUS04 - (P2-A11) Connects LBUSA04 to LBUSC04 (P2-C11)
JP8 LBUS05 - (P2-A12) Connects LBUSA05 to LBUSC05 (P2-C12)
JP9 LBUS06 - (P2-A14) Connects LBUSA06 to LBUSC06 (P2-C14)
JP10 LBUS07 - (P2-A15) Connects LBUSA07 to LBUSC07 (P2-C15)
JP11 LBUS08 - (P2-A17) Connects LBUSA08 to LBUSC08 (P2-C17)
JP12 LBUS09 - (P2-A18) Connects LBUSA09 to LBUSC09 (P2-C18)
JP13 LBUS10 - (P2-A20) Connects LBUSA10 to LBUSC10 (P2-C20)
JP14 LBUS11 - (P2-A21) Connects LBUSA11 to LBUSC11 (P2-C21)

JP19 MODID - P2-A30 Connects card MODID* to backplane
JP4 EXTRST$ ------ External card reset

JP3 EXTSTRT$ ------ External Start (to card)

JP18 TTLTRG0* - TTLTRG7* 12 pin jumper block for TTL TRIGGER SIGNALS, described below.

The jumper block JP18 is provided with wire wrap pins. Each one of the pins (pins 5,7,6,8) may be wired to any one of the TTL TRIGGER LINES 0-7 according to the needs of the user.



TTL TRIGGER OUTPUT SIGNALS

TRIG0$ - Low going pulse (500nsec.) set per condition(s) written to the Interrupt/Trigger Condition Register (same as found on the front panel's External Trigger connector)

TRIG1$ - reserved

TTL TRIGGER INPUT SIGNALS

EXTSTRT$ (External Start)

The card may be started externally. This achieves the same effect as writing a 1 to bit 0 of the Start Register. The card may be started asynchronously depending on the state of jumper JP3. To use the external Start option, the External Start bit within the Start/Stop Register must be set to a logic 1. The desired channel Start bits are then set within this register. The card's operation is then started by pulsing the EXTSTRT$ line.

ASYNCHRONOUS START: JP3 Short pins 1 and 2

EXTRST$ (External Reset)

The card may be initialized by applying a low going pulse (100ns. minimum) on this line. The initialization function performed here is the same as that performed by writing to the Card Initialization Register. If this function is enabled then pins 1 and 2 of JP4 must be shorted. If it is not, then pins 2 and 3 of JP4 must be shorted together.

FACTORY DEFAULT JUMPER SETTINGS

JP2 Installed: A24 Address Space

JP4 Short pins 2 and 3; Disable Ext. Reset Option

JP3 Short pins 2 and 3; Disable External Start Option

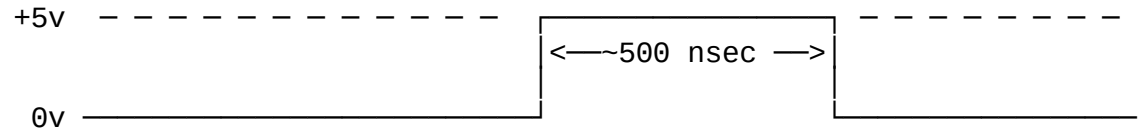
JP19 Installed; Connect MODID to bus (only for the EXC-3000VME-VXI-C/xx board)

JP30 Short pins 1 and 2: Positive pulses on the External Trigger connector.

Note: For this minimum configuration, all other jumpers are left open.

EXTERNAL TRIGGER

The external trigger pulse signal is open-collector with a pull-up resistor and short-circuit protection. The signal polarity and timing is shown below.



Note: In case negative polarity pulses are required, short jumper JP30 pins 2 and 3 (see JUMPERS).

SIZE "C" SHIELD DISASSEMBLE/ASSEMBLE INSTRUCTIONS

If the jumper settings need to be changed or additional channels need to be installed or new revision EPROMs need to be installed, please disassemble the shield cover as follows:

1) remove the 6 screws holding the upper and lower shield covers together.
2) slide out the upper shield cover from under the front panel.
3) carry out the modifications.
4) reassemble the shield covers.

CONNECTORS

The EXC-3000VME-VXI board contains six connectors:
a) all communications I/O signals on one female high density DB-62 connector (J2). In addition, a subminiature BNC connector (J1) is located on the front panel. Mating connectors are supplied for both.
b) two DIN type 96 pin VME/VXI connectors (P1 and P2).
c) two, 0.1" spacing, socket headers (P3 and P4) for optional installation of a plug-in adapter board.

SUBMINIATURE BNC (J1-External Trigger Connector)

This connector supplies an external trigger source. This front-panel connector is under software control and can be activated upon the same conditions as interrupts (see: Interrupt/Trigger Condition Register). The polarity of the pulses can be selected via JP30 (see: Jumpers)

DB-62 CONNECTOR PINOUT (J2-Communications I/O Connector)

The 62-pin connector is defined below. Each channel is allocated 6 pins (refered to as A to E). The function of each pin depends upon the protocol implemented by the channel. A table follows the connector pinout which defines the function per protocol.

PIN #	SIGNAL NAME	PIN #	SIGNAL NAME	PIN #	SIGNAL NAME
1	CH# 0 B	22	CASE GROUND	43	DIGITAL GROUND
2	CH# 0 D	23	CH# 0 A	44	CH# 0 F
3	CH# 1 B	24	CH# 0 C	45	CH# 0 E
4	CH# 1 D	25	CH# 1 A	46	CH# 1 F
5	CH# 2 B	26	CH# 1 C	47	CH# 1 E
6	CH# 2 D	27	CH# 2 A	48	CH# 2 F
7	CH# 3 B	28	CH# 2 C	49	CH# 2 E
8	CH# 3 D	29	CH# 3 A	50	CH# 3 F
9	CH# 4 B	30	CH# 3 C	51	CH# 3 E
10	CH# 4 D	31	CH# 4 A	52	CH# 4 F
11	CH# 5 B	32	CH# 4 C	53	CH# 4 E
12	CH# 5 D	33	CH# 5 A	54	CH# 5 F
13	CH# 6 B	34	CH# 5 C	55	CH# 5 E
14	CH# 6 D	35	CH# 6 A	56	CH# 6 F
15	CH# 7 B	36	CH# 6 C	57	CH# 6 E
16	CH# 7 D	37	CH# 7 A	58	CH# 7 F
17	CH# 8 B	38	CH# 7 C	59	CH# 7 E
18	CH# 8 D	39	CH# 8 A	60	CH# 8 F
19	CH# 9 B	40	CH# 8 C	61	CH# 8 E

20 CH# 9 D	41 CH# 9 A	62 CH# 9 F
21 CH# 9 E	42 CH# 9 C	

In this table the 62-pin connector is sorted by signal name:

SIGNAL NAME	PIN #	SIGNAL NAME	PIN #
CH# 0 A	23	CH# 5 A	33
CH# 0 B	1	CH# 5 B	11
CH# 0 C	24	CH# 5 C	34
CH# 0 D	2	CH# 5 D	12
CH# 0 E	45	CH# 5 E	55
CH# 0 F	44	CH# 5 F	54
CH# 1 A	25	CH# 6 A	35
CH# 1 B	3	CH# 6 B	13
CH# 1 C	26	CH# 6 C	36
CH# 1 D	4	CH# 6 D	14
CH# 1 E	47	CH# 6 E	57
CH# 1 F	46	CH# 6 F	56
CH# 2 A	27	CH# 7 A	37
CH# 2 B	5	CH# 7 B	15
CH# 2 C	28	CH# 7 C	38
CH# 2 D	6	CH# 7 D	16
CH# 2 E	49	CH# 7 E	59
CH# 2 F	48	CH# 7 F	58
CH# 3 A	29	CH# 8 A	39
CH# 3 B	7	CH# 8 B	17
CH# 3 C	30	CH# 8 C	40
CH# 3 D	8	CH# 8 D	18
CH# 3 E	51	CH# 8 E	61
CH# 3 F	50	CH# 8 F	60
CH# 4 A	31	CH# 9 A	41
CH# 4 B	9	CH# 9 B	19
CH# 4 C	32	CH# 9 C	42
CH# 4 D	10	CH# 9 D	20
CH# 4 E	53	CH# 9 E	21
CH# 4 F	52	CH# 9 F	62
CASE GROUND	22	DIGITAL GROUND	43

FUNCTIONAL DEFINITION OF PINS FOR J2 ACCORDING TO PROTOCOL

ARINC-429/575/582-2wire (Transmit or Receive)

A - DATA HI
B - DATA LO
C - unused
D - unused
E - CASE GROUND
F - DIGITAL GROUND

ARINC-561/568/582-6wire (Transmit or Receive)

A - DATA HI
B - SYNC HI
C - CLK HI
D - DATA LO
E - SYNC LO
F - CLK LO

Note:

1. A common CASE GROUND and a common DIGITAL GROUND are available for these protocols (see connector pinout on previous page).
2. For channels containing this type of protocol a jumper has to be disconnected on the soldered side of the printed circuit board as follows: CH#**n** - JP2**n** (n-channel number). For example for channel #**3** cut the jumper JP2**3**.

RS-232

A - TXD
B - RXD
C - CTS INPUT
D - DTR OUTPUT
E - CASE GROUND
F - DIGITAL GROUND

RS-422

A - DATA TRANSMIT HI
B - DATA TRANSMIT LO
C - DATA RECEIVE HI (Note 3)
D - DATA RECEIVE LO (Note 3)
E - CASE GROUND

F - DIGITAL GROUND

Note: 3. The receive inputs are terminated with a 120ohm resistor between HI and LO inputs.

FUNCTIONAL DEFINITION OF PINS FOR J2 ACCORDING TO PROTOCOL (CONTINUED)

RS-485

A - DATA HI
B - DATA LO
C - unused
D - unused
E - CASE GROUND
F - DIGITAL GROUND

RS-423

A - DATA TRANSMIT (Note 4)
B - DATA TRANSMIT INVERTED (OPTION) (Note 4)
C - DATA RECEIVE HI
D - DATA RECEIVE LO
E - CASE GROUND
F - DIGITAL GROUND (CAN BE PAIRED WITH DATA TRANSMIT)

Note: 4. The transmit Driver Rise/Fall time is set to 1µsec approx. This value can be changed by replacing the Waveshape resistor R2 (located on top of Front End Module). Selected values of R2 for different Rise/Fall times are listed in following:

Rise/Fall	R2
5µsec	50Kohm
10µsec	100Kohm
50µsec	500Kohm
100µsec	1Mohm

CONNECTOR P1 PINOUT

Pin #	Signal Name
A1	D00
A2	D01
A3	D02
A4	D03
A5	D04
A6	D05
A7	D06
A8	D07
A9	GND
A10	
A11	GND
A12	DS1*
A13	DS0*
A14	WRITE*
A15	GND
A16	DTACK*
A17	GND
A18	AS*
A19	GND
A20	IACK*
A21	IACKIN*
A22	IACKOUT*
A23	AM4
A24	A07
A25	A06
A26	A05
A27	A04
A28	A03
A29	A02
A30	A01
A31	-12V
A32	+5V

Pin #	Signal Name
B1	
B2	
B3	
B4	BG0IN*
B5	BG0OUT*
B6	BG1IN*
B7	BG1OUT*
B8	BG2IN*
B9	BG2OUT*
B10	BG3IN*
B11	BG3OUT*
B12	
B13	
B14	
B15	
B16	AM0
B17	AM1
B18	AM2
B19	AM3
B20	GND
B21	
B22	
B23	GND
B24	IRQ7*
B25	IRQ6*
B26	IRQ5*
B27	IRQ4*
B28	IRQ3*
B29	IRQ2*
B30	IRQ1*
B31	
B32	+5V

Pin #	Signal Name
C1	D08
C2	D09
C3	D10
C4	D11
C5	D12
C6	D13
C7	D14
C8	D15
C9	GND
C10	SYSFAIL*
C11	
C12	SYSRESET*
C13	LWORD*
C14	AM5
C15	A23
C16	A22
C17	A21
C18	A20
C19	A19
C20	A18
C21	A17
C22	A16
C23	A15
C24	A14
C25	A13
C26	A12
C27	A11
C28	A10
C29	A09
C30	A08
C31	+12V
C32	+5V

CONNECTOR P2 PINOUT

Pin #	Sig. Name		Pin #	Sig. Name		Pin #	Sig. Name	
A1			B1	+5V		C1		
A2			B2	GND		C2		
A3			B3			C3		
A4			B4	A24		C4		
A5	LBUSA00	(x)	B5	A25		C5	LBUSC00	(x)
A6	LBUSA01	(x)	B6	A26		C6	LBUSC01	(x)
A7			B7	A27		C7		
A8	LBUSA02	(x)	B8	A28		C8	LBUSC02	(x)
A9	LBUSA03	(x)	B9	A29		C9	LBUSC03	(x)
A10			B10	A30		C10		
A11	LBUSA04	(x)	B11	A31		C11	LBUSC04	(x)
A12	LBUSA05	(x)	B12	GND		C12	LBUSC05	(x)
A13			B13	+5V		C13		
A14	LBUSA06	(x)	B14			C14	LBUSC06	(x)
A15	LBUSA07	(x)	B15			C15	LBUSC07	(x)
A16			B16			C16		
A17	LBUSA08	(x)	B17			C17	LBUSC08	(x)
A18	LBUSA09	(x)	B18			C18	LBUSC09	(x)
A19			B19			C19		
A20	LBUSA10	(x)	B20			C20	LBUSC10	(x)
A21	LBUSA11	(x)	B21			C21	LBUSC11	(x)
A22			B22	GND		C22		
A23	TTLTRG0*	(x)	B23			C23	TTLTRG1*	(x)
A24	TTLTRG2*	(x)	B24			C24	TTLTRG3*	(x)
A25			B25			C25		
A26	TTLTRG4*	(x)	B26			C26	TTLTRG5*	(x)
A27	TTLTRG6*	(x)	B27			C27	TTLTRG7*	(x)
A28			B28			C28		
A29			B29			C29		
A30	MODID	(x)	B30			C30		
A31			B31	GND		C31		
A32			B32	+5V		C32		

Notes: (x) - VXI signals (each of them is unconnected, unless the specific jumper is shorted. see JUMPERS above)

POWER SUPPLY REQUIREMENTS

The board's power supply requirements are defined below.

MAGICard with no communication channels installed:

+5 volt @ 1.5 Amps
+12 volt @ 50 mA

EACH CHANNEL REQUIRES:

CHANNEL TYPE	+5v	+12v	-12v
ARINC-429 or WILLIAMSBURG TRANSMITTER	150ma	40ma	40ma
ARINC-429 or WILLIAMSBURG RECEIVER	150ma	20ma	20ma
WILLIAMSBURG CRC	150ma	-	-
ARINC-575 TRANSMITTER	150ma	40ma	40ma
ARINC-575 RECEIVER	150ma	20ma	20ma
ARINC-561 TRANSMITTER	150ma	180ma	180ma
ARINC-561 RECEIVER	150ma	20ma	-
ARINC-568 TRANSMITTER	150ma	180ma	180ma
ARINC-568 RECEIVER	150ma	20ma	-
ARINC-582-2wire TRANSMITTER	150ma	40ma	40ma
ARINC-582-2wire RECEIVER	150ma	20ma	20ma
ARINC-582-6wire TRANSMITTER	150ma	180ma	180ma
ARINC-582-6wire RECEIVER	150ma	20ma	-
RS-232 TRANSMIT/RECEIVE	180ma	-	-
RS-422 TRANSMIT/RECEIVE	180ma	-	-
RS-485 TRANSMIT/RECEIVE	180ma	-	-
RS-423 TRANSMIT/RECEIVE	180ma	25ma	25ma

ORDERING INFORMATION

EXC-3000VME-VXI/**option code(s)** "B-size" card

EXC-3000VME-VXI-C/**option code(s)** "C-size" card; complete with shield.

Option Code:	Definition:	No. of Channels Used:	Notes:
Ax	(RS-232 TX/RCV - 8.0000 MHz)	- 1 channel	
AAx	(RS-232 TX/RCV - 7.3728 MHz)	- 1 channel	
Bx	(RS-422 TX/RCV - 8.0000 MHz)	- 1 channel	
BBx	(RS-422 TX/RCV - 7.3728 MHz)	- 1 channel	
Cx	(RS-485 TX/RCV - 8.0000 MHz)	- 1 channel	
CCx	(RS-485 TX/RCV - 7.3728 MHz)	- 1 channel	
Zx	(RS-423 Tx/Rcv - 8.0000MHz)	- 1 channel	3
ZZx	(RS-423 Tx/Rcv - 7.3728MHz)	- 1 channel	3
Dx	(ARINC-429 Transmitter)	- 1 channel	
Ex	(ARINC-429 Receiver)	- 1 channel	
Fx	(ARINC-429 Williamsburg/set)	- 3 channels	
Gx	(ARINC-561 Transmitter)	- 1 channel	
Hx	(ARINC-561 Receiver)	- 1 channel	
Ix	(ARINC-575 Transmitter)	- 1 channel	
Jx	(ARINC-575 Receiver)	- 1 channel	
Kx	(ARINC-568 Transmitter)	- 1 channel	
Lx	(ARINC-568 Receiver)	- 1 channel	
Mx	(ARINC-582-2wire Transmitter)	- 1 channel	
Nx	(ARINC-582-2wire Receiver)	- 1 channel	
Ox	(ARINC-582-6wire Transmitter)	- 1 channel	
Px	(ARINC-582-6wire Receiver)	- 1 channel	
Q1	MB-BCRM-V (1553B BC/RT/Monitor)	- adapter board	
S1	MB-IEEE488-V (IEEE-488 Interface)	- adapter board	
T1	MB-SDLC-V (4 channels of SDLC 232/422)	- adapter board	

Important Notes:

1) The "x" following the Option Code denotes the number of channels per card (Example: D2 = Two ARINC-429 Transmitters).

2) When ordering a card with a number of different protocol channels, the part number must be in the following form: EXC-3000VME-VXI/AxBxCxDx.

3) RS-423 channel requires firmware Revision 1.31 and up.

Part numbers for additional Channel Modules

EXC-3000-**A**M - Part # for additional RS-232 module set

.

.

.

EXC-3000-**P**M - Part # for additional ARINC-582_6wire Receiver module set

The information contained in this document is believed to be accurate. However, no responsibility is assumed by Excalibur Systems, Inc. for its use and no license or rights are granted by implication or otherwise in connection therewith. Specifications are subject to change without notice.

January 1996, Rev. C-2