

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

COMM422

Communication Board
User Manual

Document No. B-T-MU-COMM422#-A-0-A2

FOREWORD

The information in this document has been carefully checked and is believed to be accurate; however, no responsibility is assumed for inaccuracies. SYSTRAN reserves the right to make changes without notice.

SYSTRAN makes no warranty of any kind with regard to this printed material including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

The schematic diagrams contained in this publication are included for reference only. The information was current at the time the printed circuit board was last revised. This information is not necessarily current or complete manufacturing data, nor is it part of the product specification. Schematics may not be reproduced without the express permission of SYSTRAN Corp.

The STARTECH ST16C554D data sheet contained in appendix B has been reproduced with the express authorization of EXAR Corporation.

©Copyright 1998 SYSTRAN Corp. All Rights Reserved.

ALTERA® and Altera® are registered trademarks of Altera Corporation.
HP™ is a trademark of the Hewlett-Packard Corporation.
Motorola® is a registered trademark of the Motorola Corporation.
NuBus™ is a trademark of Texas Instruments, Inc.
OS-9® is a registered trademark of Microware Systems Corporation.
TRI-STATE® is a registered trademark of National Semiconductor Corporation.
ViewDraw® and VIEWlogic® are registered trademarks of Viewlogic Systems, Inc.

ALTERA: Code © 1998, SYSTRAN Corp., All Rights Reserved. The code (programs) contained in this product are proprietary to SYSTRAN Corp. and copying or other use of the bit-stream programs, except as expressly authorized by SYSTRAN Corp., is expressly prohibited.

Revised: March 3, 1998

SYSTRAN Corporation
4126 Linden Avenue
Dayton, OH 45432-3068 USA
(937) 252-5601

FCC

This product is intended for use in industrial, laboratory or military environments. This product uses and emits electromagnetic radiation which may interfere with other radio and communication devices. The user may be in violation of FCC regulations if this device is used in other than the intended market environments.

CE

Please note: As a component part of another system, this product has no intrinsic function and is therefore not subject to the European Union CE EMC directive 89/336/EEC.

TABLE OF CONTENTS

1.0 INTRODUCTION 1-1
- 1.1 Purpose 1-1
- 1.2 Scope 1-1
- 1.3 Overview 1-2
- 1.4 Features 1-2
- 1.5 Specifications 1-2
- 1.6 Related Product 1-3
- 1.7 Related Publications 1-4
- 1.8 Technical Support 1-4
- 1.9 Reliability 1-4
- 1.10 Ordering Process 1-5

2.0 DESCRIPTION 2-1
- 2.1 Block Diagram Description 2-1
- 2.2 Theory of Operation 2-3
  - 2.2.1 ID Reads 2-3
  - 2.2.2 I/O Reads and Writes 2-3
- 2.3 Schematics 2-7
- 2.4 Cleaning 2-7

3.0 INSTALLATION 3-1
- 3.1 Unpack the COMM422 3-1
- 3.2 Inspect the COMM422 3-1
- 3.3 Install the COMM422 3-1

4.0 PROGRAMMING GUIDE 4-1
- 4.1 Overview 4-1
- 4.2 Description 4-1
- 4.3 IP Module ID ROM Listing 4-1
- 4.4 IP Module I/O Address Map 4-2
- 4.5 Power-Up and System Reset Defaults 4-7
- 4.6 Word Access Address Translation 4-7
- 4.7 Byte Access Address Translation 4-8
- 4.8 Register Definitions 4-9
  - 4.8.1 Receive Holding Register (RHR) 4-9
  - 4.8.2 Transmit Holding Register (THR) 4-10
  - 4.8.3 Interrupt Enable Register (IER) 4-10
  - 4.8.4 Divisor Latch Low Byte (DLL) 4-11
  - 4.8.5 Divisor Latch High Byte (DLM) 4-11
  - 4.8.6 FIFO Control Register (FCR) 4-12
  - 4.8.7 Interrupt Status Register (ISR) 4-13
  - 4.8.8 Line Control Register (LCR) 4-14
  - 4.8.9 Modem Control Register (MCR) 4-15
  - 4.8.10 Line Status Register (LSR) 4-16
  - 4.8.11 Modem Status Register (MSR) 4-17
  - 4.8.12 Scratchpad Register (SPR) 4-17
  - 4.8.13 Port Interrupt Steering/Driver Disable Register 4-18
  - 4.8.14 Interrupt Vector Register 4-18
- 4.9 Programming Examples 4-19

4.9.1 Open 4-19
4.9.2 Write 4-20
4.9.3 Read 4-20
4.9.4 Ioctl 4-20
4.9.5 ISR 4-21
4.9.6 Close 4-22

5.0 PERFORMANCE 5-1
5.1 Overview 5-1
5.2 Timing Measurements 5-1
5.3 Oscilloscope Measurements 5-2
5.4 State Timing Diagrams 5-7
5.4.1 ID Read Cycle 5-7
5.4.2 I/O Write and Read Cycle 5-7
5.4.3 Interrupt Request 5-7
5.4.4 Interrupt Select Cycle 5-7

6.0 TYPICAL APPLICATIONS 6-1
6.1 Overview 6-1
6.2 Configuration and Interfacing for the COMM422 IP Module 6-1
6.3 COMM422 Configuration Settings 6-1
6.3.1 Balanced-Serial-Port Modes 6-1
6.3.2 RS-422 Settings 6-4
6.3.3 RS-485 Half Duplex Settings 6-5
6.3.4 RS-485 Full Duplex Settings 6-5
6.4 Interfacing the COMM422 6-6
6.4.1 Common Cable and Connector Types 6-6
6.4.2 Carrier to Device Interconnect 6-6
6.4.3 Receiver Biasing 6-6

APPENDICES

Appendix A - COMM422 Schematics
Appendix B - STARTECH ST16C554D Data Sheet

FIGURES

Figure 1-1 COMM422 Board 1-1
Figure 2-1 COMM422 Block Diagram 2-1
Figure 2-2 Detailed Block Diagram 2-5
Figure 2-3 Interface Block Diagram 2-6
Figure 3-1 COMM422 Installed on a VME IP Module Carrier 3-2
Figure 3-2 COMM422 Installed on an ISA IP Module Carrier 3-2
Figure 5-1 Test Configuration for Oscilloscope and Analyzer Measurements 5-1
Figure 5-2 Rise-Time/Amplitude for Port A TX(-) at Driver Out 5-3
Figure 5-3 Fall-Time/Amplitude for Port A TX(-) at Driver Out 5-3
Figure 5-4 Rise-Time/Amplitude for Port A TX(+) at Driver Out 5-4
Figure 5-5 Fall-Time/Amplitude for Port A TX(+) at Driver Out 5-4
Figure 5-6 Rise-Time/Amplitude for Port A RX(-)at Receiver Input 5-5
Figure 5-7 Fall-Time/Amplitude for Port A RX(-) at Receiver Input 5-5
Figure 5-8 Rise-Time/Amplitude for Port A RX(+)at Receiver Input 5-6
Figure 5-9 Fall-Time/Amplitude for Port A RX(+) at Receiver Input 5-6
Figure 5-10 ID Read Cycle from IP Module Address 00 *hex* 5-8
Figure 5-11 I/O Write Cycle IP Module Address 07 *hex* (Port A Scratchpad Register) 5-8
Figure 5-12 I/O Read Cycle IP Module Address 07 *hex* (Port A Scratchpad Register) 5-9
Figure 5-13 I/O Write to IP Module Address 01 *hex* (Port A Interrupt-Enable Register) Causing an Interrupt Request to be Generated 5-9
Figure 5-14 Interrupt Request Followed by an IP Module Interrupt Select Cycle 5-10
Figure 5-15 IP Module Interrupt-Select Cycle Details 5-10
Figure 5-16 I/O Read of IP Module Address 02 *hex* (UART-Interrupt-Information Register) for removal of interrupt Request 5-11
Figure 6-1 Typical RS-422 Driver/Receiver Interconnect 6-2
Figure 6-2 Typical RS-485 Half Duplex Device Interconnect 6-3
Figure 6-3 Typical RS-485 Full Duplex Device Interconnect 6-3
Figure 6-4 Jumper Locations on the COMM422 6-4
Figure 6-5 Jumper Positions vs. Mode for the COMM422 6-5
Figure 6-6 Common Connectors and Phoenix Contact Block 6-8
Figure A-1 Schematic #1 ..... A-1
Figure A-2 Schematic #2 ..... A-2
Figure A-3 Schematic #3 ..... A-3
Figure A-4 Schematic #4 ..... A-4

TABLES

Table 3-1 Contents of COMM422 Shipping Packages 3-1
Table 3-2 COMM422 Installation Tools 3-2
Table 3-3 IP Module Logic Bus Pin Assignments 3-4
Table 3-4 IP Module I/O Connector Pin Assignments 3-5
Table 4-1 COMM422 ID Address Space Listing 4-2
Table 4-2 COMM422 I/O Address Map 4-5
Table 4-3 Word Access Address Translation Table 4-7
Table 4-4 Byte Access Address Translation Table 4-8
Table 4-5 Receive Holding Register 4-9
Table 4-6 Transmit Holding Register 4-10
Table 4-7 Interrupt Enable Register 4-10
Table 4-8 Divisor Latch Low Byte 4-11
Table 4-9 Divisor Latch High Byte 4-11
Table 4-10 FIFO Control Register 4-12
Table 4-11 Interrupt Status Register 4-13
Table 4-12 Line Control Register 4-14
Table 4-13 Modem Control Register 4-15
Table 4-14 Line Status Register 4-16
Table 4-15 Modem Status Register (MSR) 4-17
Table 4-16 Scratchpad Register 4-17
Table 4-17 Port Interrupt Steering/Driver Disable Register 4-18
Table 4-18 Interrupt Vector Register 4-18
Table 4-19 DLL and DLM Register Baud Rate Values 4-19
Table 6-1 Typical Signal Pinouts for DB25 and DB9 Connectors 6-7

GLOSSARY

[x:y] ---------- Nomenclature designating a bit-range, where “x” is the left-most bit and “y” is the right-most bit. (e.g. Data bus [7:0] refers to the least-significant eight bits).

Byte ---------- 8 bits.

byte-lane ---------- 8 bits of a data bus on octal boundaries.

CAE ---------- Computer Aided Engineering.

doublewide ---------- An IP Module that is twice the size of the singlewide board.

EPLD ---------- Erasable Programmable Logic Device.

ID ROM ---------- The circuitry that presents the proper data patterns to the low 8 bits of the IPDbus, with upper-byte zero fills during the ID (READ) transfers.

IPDbus ---------- IP Module Data Bus.

IP Module ---------- Business-card size mezzanine-type subsystems designed with a common digital interface known as the IP bus. These field-installable plug-and-play modules are automatically recognized by system software. An open industry standard defines the mechanical and electrical interface to the carrier board.

IP Module logic bus ---------- A synchronous, 4 MTransfers/sec, 16- or 32-bit-wide bus that includes I/O, memory, ID PROM, and interrupts. The address bus is 6 bits wide, except in memory mode. Then the data bus is multiplexed for the upper portion of the address bus, resulting in 22 bits of address. This results in up to 4 Mwords of memory space per IP Module.

I/O ---------- Input/Output.

ISR ---------- Interrupt Service Routine.

LSB ---------- Least Significant Bit.

MTBF ---------- Mean Time Between Failures.

ns, µs, ms ---------- Nanoseconds, microseconds, and milliseconds, respectively.

PC ---------- Personal Computer.

QSOP ---------- Quarter-sized Small Outline Packages.

singlewide ---------- A standard IP Module printed circuit board (3.9" by 1.8") with two 50-pin connectors.

VHDL ---------- Very high-speed integrated circuit Hardware Description Language.

This page intentionally left blank

1.0 INTRODUCTION

1.1 Purpose

This is a reference manual for SYSTRAN's four-channel, balanced serial communication IP Module referred to in this document as the COMM422 (part number BHAS-COMM422).

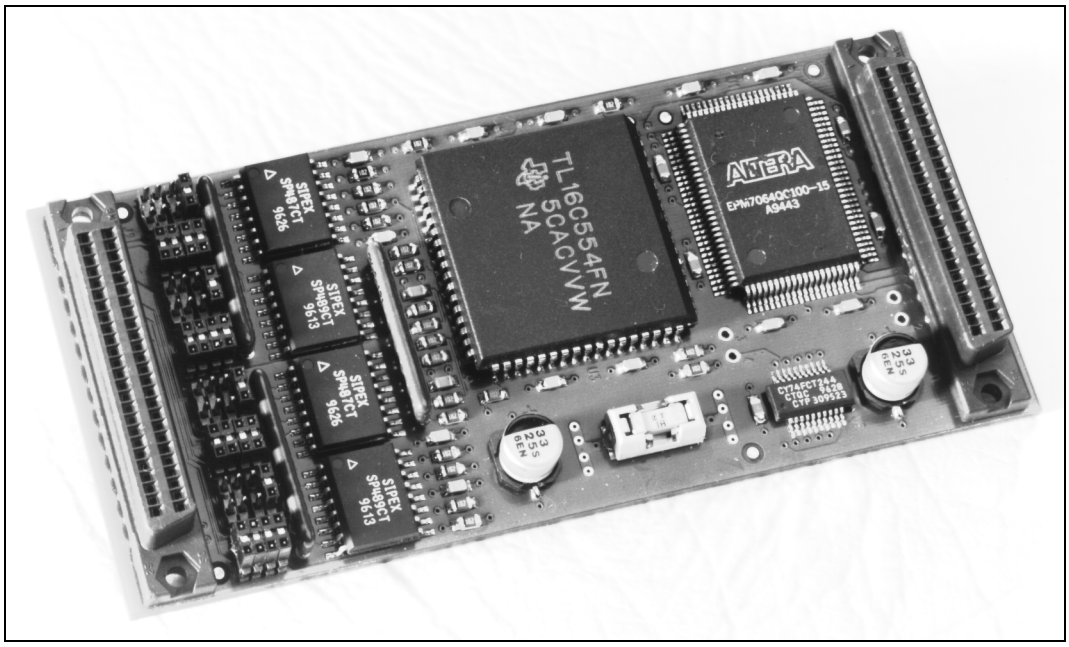


Figure 1-1 COMM422 Board

1.2 Scope

This reference manual covers the physical and operational description of the COMM422, both from hardware and software perspectives. This manual also contains detailed technical information about the COMM422's performance characteristics, and some typical applications. The reader should have a general understanding of computer processing, software and/or hardware applications experience, and a working knowledge of using IP Modules on their carrier(s) of choice. Citing equipment from other vendors within this document does not constitute an endorsement of their product(s).

1.3 Overview

The COMM422 is a singlewide IP Module board, containing a quad Asynchronous Communications Element (ACE) and Receiver and Driver circuits that conform to the TIA/EIA-422B and EIA-485 standards. This board was developed using guidelines in the ANSI/VITA 4-1995 IO Module Interface Specification.

For specific information regarding this specification, a copy can be obtained by contacting the VMEbus International Trade Association (VITA) at the following address:

VMEbus International Trade Association
7825 E. Gelding Dr., Suite 104
Scottsdale, AZ 85260

Telephone: (602) 951-8866

1.4 Features

Provides the ideal low-cost solution for applications requiring multiple RS-422 or 485 multi-drop Serial ports. The COMM422 IP Module has the following features:

- RTS*/CTS* handshake support in RS-422 mode.
- User prioritization of interrupts (by port) via software using single-interrupt-vector register and port-steering register.
- +5V only operation. Driver disable register allows power savings.
- Flexible operation with programmable baud rates, FIFO trigger level and programmable interface characteristics.

1.5 Specifications

MECHANICAL: SINGLEWIDE IP MODULE (TYPE I)

- Measurements: 1.800 x 3.900 x 0.303 inches (above board)
  4.572 x 9.906 x 0.770 cm (above board)
- Weight: 1.224 oz. - 34.70 grams
- Board thickness: 0.062 inches, 0.157 cm, nominally (6 layers)

ELECTRICAL: SINGLEWIDE IP MODULE TRANSFERS

- 8 MHz clock rate
- One wait cycle on all transfers
- Hold cycles supported on all valid transfer types
- Complete address decoding
- I/O transfers: 8- and 16-bit writes (upper byte ignored for 16) and 8- and 16-bit reads (upper byte is zero-filled for non-UART transactions only)
  32 address locations in the Quad UART 48 registers total
  One 8 bit vector register
  One 8 bit CSR register: 4 bit Port Steering Register (R/O)
  and 4 bit Driver Disable Reg. (R/W)
  Maximum READ/WRITE rate: 2.67M Transfers/sec
- ID transfers: 8 bit READ (lower byte only)
  Upper byte zero-filled
  Twelve bytes, including the Cyclic Redundancy Check (CRC)
  SYSTRAN's manufacturer's ID = 45 *hex* = E ASCII
  COMM422's model number = 73 *hex*

CRC value = DD *hex*
Maximum READ rate: 2.67 M Transfers/sec

- Memory Transfers: - not supported
- Interrupt Vector Transfers:
  8 bit READ on lower byte only
  Upper byte zero-filled
  Maximum READ rate: 2.67 M Transfers/sec
- Interrupt requests:
  INTREQ0 driven upon UART interrupt
  INTREQ1 is not used
- DMA activity: none supported
- Non-supported transfer types (no acknowledgment):
  Memory READ and WRITE
  I/O reads and writes beyond IP Module address 21
  ID WRITE
- Supports RS-422/RS-485 data rates up to 512 Kbps

POWER REQUIREMENTS

- +5 Volt supply - 500 mA max, 260 mA typical
- + 12 Volt supply - 0 mA
- -12 Volt supply - 0 mA

PERFORMANCE CHARACTERISTICS

- Electrical Characteristics: Refer to Section 5.0 on PERFORMANCE

ABSOLUTE MAXIMUM RATINGS

- Supply Voltage (+5 Volt IP bus): -0.5 V (min) to +7.0 V (max)
- I/O port signals (RS-422/485 inputs): -14 V (min) to +14 V (max)

RECOMMENDED OPERATING CONDITIONS

- Main supply voltage (IP bus 5V supply): +4.75 to +5.25 V
- Signals on all serial inputs should be from -7 V (typical) to +12 V (typical)
- Cable lengths > 4000 feet are not recommended.

ENVIRONMENTAL SPECIFICATIONS

- Temperature (Operating): 0°C $\rightarrow$ +70°C
- Temperature (No bias, storage): -40°C $\rightarrow$ +85°C
- Humidity (Noncondensing): 5% $\rightarrow$ 95%

MEAN TIME BETWEEN FAILURES (MTBF)

- 626,096 hours

1.6 Related Product

- OS-9 SCF device-driver routines and documentation.

1.7 Related Publications

- American National Standard for IP Modules (ANSI/VITA 4 - 1995) published by the VMEbus International Trade Association 7825 Gelding Dr., Suite 104, Scottsdale, AZ 85260
- TIA/EIA-422B *Characteristics of Balanced Voltage Digital Interface Circuits* published by Telecommunications Industry Association Standards/Technology Department 2001 Pennsylvania Ave. NW, Washington DC 20006
- EIA-485 *Standard for Electrical Characteristics of Generators and Receivers For Use in Balanced Digital Multipoint Systems* published by Electronic Industries Association Engineering Department 2500 Wilson Boulevard Arlington, VA 22201
- STI6C554D Quad Asynchronous Receiver/Transmitter With FIFO's data document by Startech Division of EXAR. This document is included as appendix B in this manual and has been reproduced with permission of EXAR.

1.8 Technical Support

Technical documentation provided with the product discusses the technology, its performance characteristics, and some typical applications. It includes tutorial material, with comprehensive support information, designed to answer any technical questions that might arise concerning the use of this product. SYSTRAN also publishes technical briefs and application notes that cover a wide assortment of topics. The applications selected are derived from real scenarios, but do not cover all possible circumstances.

Direct questions not satisfactorily answered by this document, or concerns about the functional-fit of this product for your particular application, or programming questions, to the factory at **(937) 252-5601**, or send an e-mail message to **support@systran.com** for additional assistance. Our goal is to help solve your problem.

1.9 Reliability

SYSTRAN Corporate policy is to provide the highest quality products in support of customer's needs. In addition to the physical product, the company provides documentation, sales and marketing support, hardware and software technical support, and timely product delivery. The SYSTRAN commitment to quality begins with product concept, and continues after receipt of the purchased product.

SYSTRAN has developed a Quality System which conforms to the ISO 9001 international standard for quality systems. ISO 9001 is the model for quality assurance in design, development, production, installation and servicing. The ISO 9001 standard is the most comprehensive of the conformance standards, in that it addresses all 20 clauses of the ISO quality system requirements.

SYSTRAN's Quality System addresses the following basic quality objectives:

- Achieve, maintain and continually improve the quality of SYSTRAN products
- Improve the quality of its own operations to meet the needs of SYSTRAN customers and stakeholders
- Provide confidence internally that quality is being fulfilled, maintained and improved
- Provide confidence to the customer and other stakeholders that requirements for quality will be achieved in the delivered product.

SYSTRAN's Quality System was assessed by BSI QA, which is the certification division of British Standards Institution, the largest and most respected standardization authority in the world. SYSTRAN's Quality System was found to meet or exceed the international standards in all areas, and Certificate of Registration number FM 31468 was issued to SYSTRAN on May 16, 1995.

The scope of SYSTRAN's registration is: "Design, manufacture and service of high technology hardware and software computer communications products." The registration is maintained under BSI QA's program of continuing assessment, under which an audit of the quality system is performed by BSI QA every six months.

An integral part of meeting SYSTRAN quality and reliability goals is customer feedback. Customers are encouraged to contact the factory with any questions or suggestions regarding unique quality requirements, or to obtain additional information about our programs. SYSTRAN's commitment to customers includes, but is not limited to:

- Professional and quick response to customer problems using SYSTRAN's extensive resources.
- Incorporation of established procedures for product design, test, and production operations, with documented milestones. Procedures are constantly reviewed and improved, ensuring the highest possible quality.

1.10 Ordering Process

To learn more about SYSTRAN products or to place an order, the following contacts are available:

- Phone: **(937) 252-5601**
- E-mail address: **info@systran.com**
- World Wide Web address: **http://www.systran.com**

This page intentionally left blank

2.0 DESCRIPTION



NOTE: The IP Module signal references, transfer and cycle types discussed in this document are explained in detail in the ANSI/VITA 4-1995 IP Module specification.

2.1 Block Diagram Description

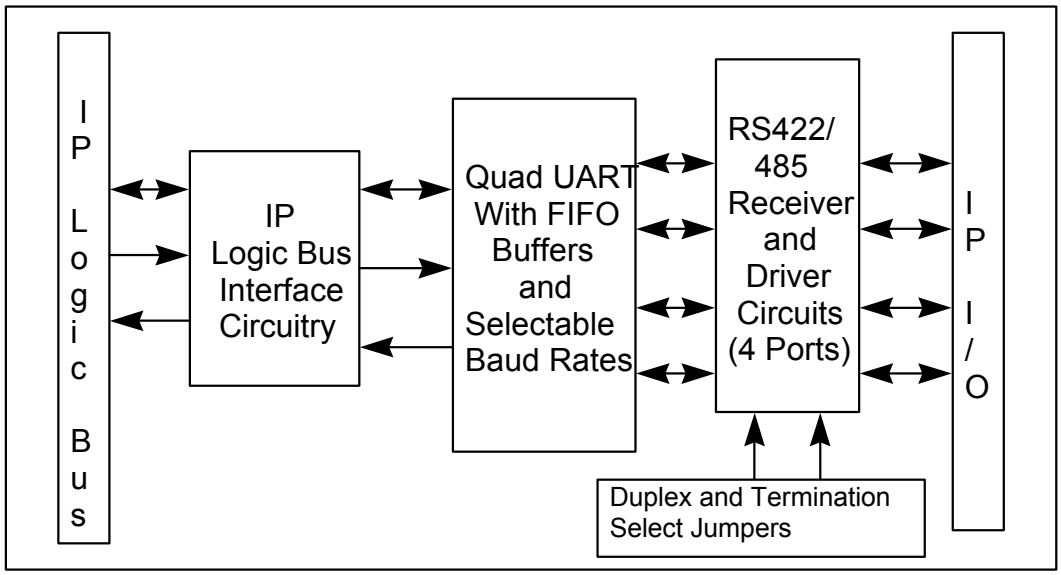


Figure 2-1 COMM422 Block Diagram

Figure 2-1 is a simplified block diagram of the COMM422 representing the signal flow between the IP Module Logic Bus and I/O connectors.

On the left side of the diagram is the IP Module's Logic Bus connector through which all transfers between the IP Module carrier and the COMM422's registers and data sources are conducted.

The block labeled "IP Logic Bus Interface Circuitry" contains one 8-bit READ/WRITE register (interrupt vector register) and an 8-bit dual function (4 bit port steering and 4 bit driver disable) register as well. This block also contains the ID ROM emulator, an interrupt request driver and all necessary logic for providing an interface between the quad UART circuit and the IP Module bus.

The block labeled "Quad UART With FIFO Buffers and Selectable Baud Rates" consists of an industry standard 16C554 Asynchronous Communication Element (ACE) in a 68-pin PLCC package. This circuit is the operational equivalent to four 16C550 UARTs (Universal Asynchronous Receiver Transmitters) and has a 16C450-compatible mode where the internal-data FIFOs are not used. All necessary serial-to-parallel and parallel-to-serial conversions are performed inside this unit as well as baud rate generation and temporary storage of data received or data to be transmitted.

The block labeled “RS-422/485 Receiver and Driver Circuits” contains the necessary level-shifter/drivers to convert the TTL levels from the UART to that necessary for standard balanced RS-422 and 485 levels. The receiver circuits must convert the RS-422/485 balanced voltage translations back to single-ended TTL levels for inputs to the UART.

Figure 2-1 presents a detailed block diagram of the COMM422, which, when used with the schematics, provides a good view of the signal flow between the functional areas on the board.

The IP Module logic bus connector is at the left side of the diagram and supports all necessary signals between the COMM422 IP Module and the host carrier to allow all supported transfer types to take place. A line buffer provides clock isolation as well as buffering for the address bits.

Three major interfacing functions all reside in the same programmable part, namely, the IP bus control, the ID ROM emulator and the interrupt controller. The IP bus interface provides address decoding and READ/WRITE control between the host via the carrier and the quad UART. Address decoding and READ/WRITE control are also provided for the registers inside the programmable part. The ID ROM emulator is a fixed circuit that presents the low byte of the IP Module 16-bit data bus with pertinent information about the COMM422 board during ID transfers for the first 12 address locations. The upper byte is zero-filled during ID READ transfers.

The interrupt controller section consists of the interrupt driver that signals an INTREQ0* to the IP bus when any interrupt from the UART is present, a 4-bit port-steering register and an 8-bit interrupt-vector register. This register determines the priority of the four ports relative to each other under software control.

The block at the center of the detailed block diagram is the 16C554 standard quad UART, which, when accessed by the address decode circuits in the IP bus controller, has four separate serial ports that can be configured and monitored by system software. The following control and status registers are accessible for each port:

- transmit and receive holding registers
- interrupt enable and status registers
- FIFO control register
- line control and status registers
- modem control and status registers
- scratchpad register
- MSB and LSB registers for the divisor used to select the baud rate.

As with any standard UART integrated circuit, the 16C554 contains a parallel data interface on one end, and necessary serial-to-parallel and parallel-to-serial conversion circuits on the other.

Each “Driver & Receiver” block contains a single balanced driver/receiver pair. The “HDX/FDX select jumpers” block contains four jumpers that allow connection of the driver/receiver pairs for TX and RX as half duplex or full duplex. The “Term. Select Jumpers” block contains a jumper for transmitter termination, receiver termination and for shunting in bias termination for the RX receiver circuit.

2.2 Theory of Operation



NOTE: Refer to the COMM422 schematics located in Appendix A while reading this material since individual components are referred to by function name, schematic reference designator, or industry-standard device part number.

The COMM422 was developed using VHDL and synthesis, targeted to an Altera EPLD, schematic-captured and integrated to standard logic devices, connectors and discrete components using ViewDraw by Viewlogic and associated tools. This top-down design approach is evident in the schematics in Appendix A.

Refer to figures 2-2 and 2-3 and the schematics in Appendix A when reading the following two sections.

2.2.1 ID Reads

When the carrier performs an ID READ of the COMM422 IP Module card, the pertinent control signals used are IDSEL*, R/W*, BS0* and ACK*. When the twelve supported ID address locations (IP address 00 *hex* to 0B *hex*) are accessed (IDSEL* = 0 while all other select signals are inactive), the IP bus interface address decoder enables the READ/WRITE state machine which, in turn, enables the selected location in the ID space lookup table to drive the lower byte of the IP data bus while the high byte is filled with zeros. If a WRITE attempt is made (R/W* = 0) during an ID access, or an address beyond 0B *hex* is decoded or if a high-byte-only transaction is attempted (BS0* = 1), the address decoder and READ/WRITE state machine will not become active and the attempted transaction will be ignored (the data bus will be tristated and ACK* will remain in the inactive state).

2.2.2 I/O Reads and Writes

READs and WRITEs to the I/O space can be broken into two addressable areas: the UART space and the CSR (control and status register) space.

The UART is directly mapped into the I/O space in the IP Module address range 00 *hex* through 1F *hex* . This means that the UART alone resides in the bottom half of the address space while the CSR's reside in the upper half. During an I/O transaction, the address decoder determines whether the access is to the UART or the registers in the IP bus interface area, and enables the appropriate functional area. If the address is for the UART space, the address decoder will then determine which of the four ports are to be accessed. This is done by decoding the upper three address bits with the lower three bits connected directly to the 16C554 UART. With a valid access to the UART detected, the READ/WRITE control state machine will produce a READ or WRITE cycle to the UART and produce ACK* at the appropriate time as well as IOR* or IOW* and one of the four port-chip selects at the appropriate time. A complete listing of the UART registers that can be accessed in the IP Module I/O space can be found in chapter four.

Accesses to the two internal CSR's is performed in similar manner only the IOR*, IOW* and chip-select outputs are inactive.

The Port Steering/Driver Disable register at IP Module address 20 *hex* is an 8-bit dual purpose register that presents the status of the four UART interrupts as ports A through D on bits 0 through 3 respectively and also allows control of the balanced driver circuits for

ports A through D on bits 4 through 7 respectively. The Port Steering/Driver Disable Register is only accessed during a valid I/O cycle to address 20 *hex* .

The interrupt vector register at IP Module address location 21 *hex* can be accessed in three ways. This register can be written to and read from in the I/O space at address 21 *hex* during a valid I/O access as determined by the address decoder and IP bus interface section. This register is also read during a valid interrupt-acknowledge cycle when the following occur:

- INTSEL* = 0 while all other IP bus cycle selects are inactive
- R/W* = 1
- BS0* = 0
- address bit A1 = 0, indicating a READ of the interrupt vector register.

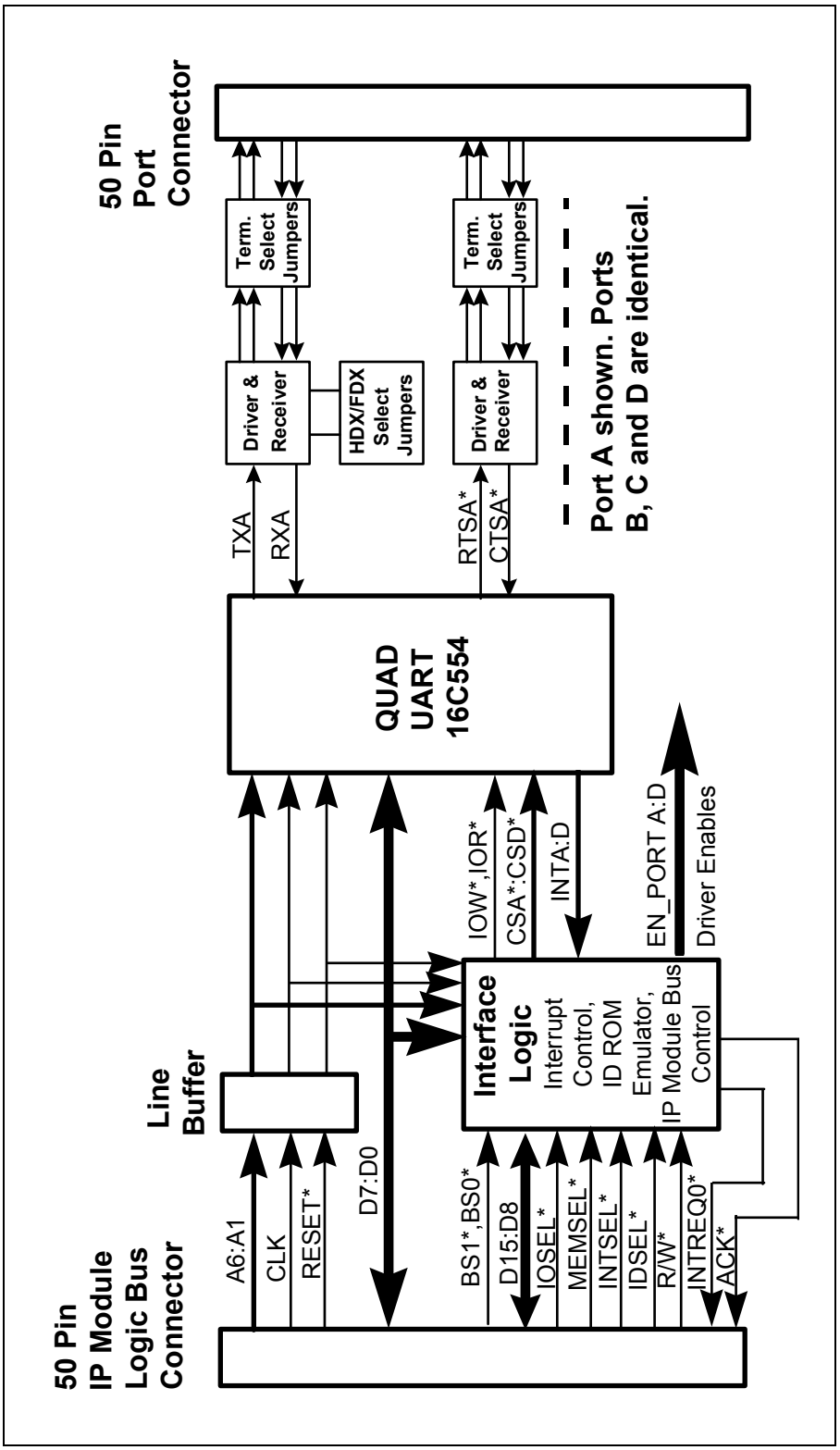


Figure 2-2 Detailed Block Diagram

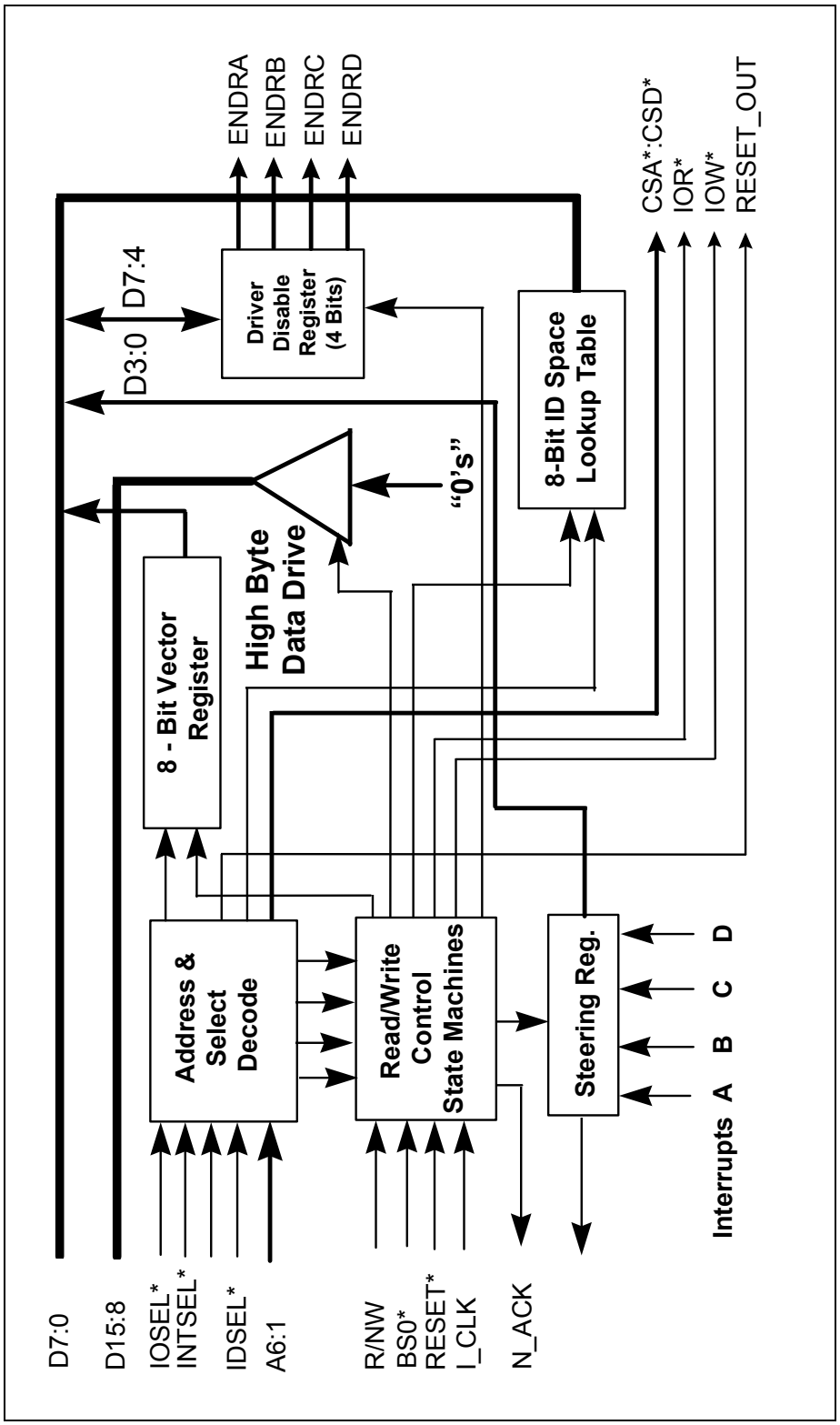


Figure 2-3 Interface Block Diagram

2.3 Schematics

The following is a short travel-map through the schematics of the COMM422 which emphasizes the top-down design methodology used

SCHEMATIC #1

The first page of the schematic, labeled Schematic #1, consists of the buffer circuit U2 and the IP Module interface controller (IPCOM422) in slot U1. The buffer provides isolation for the IP Module clock, the six dedicated address lines and the lower-byte strobe. The IPCOM422 part is an Altera EPM7064QC100 part programmed to provide all of the following:

- Address decoding and byte-lane control for the 16C554 UART registers.
- Determination of, and acknowledgment to valid IP Module bus cycles.
- Zero-fill for the upper byte on all valid 16-bit non-UART cycles.
- INTREQ0 driver to the host via the carrier on UART interrupts.
- Indication of which UART port(s) have active interrupts via a Port Steering Register.
- Control of balanced driver circuits through Driver Disable Register.
- Interrupt Vector Register.
- ID PROM emulator active only during ID READs.

SCHEMATIC #2

The page labeled Schematic #2 contains the 16C554 quad UART INTEGRATED CIRCUIT in slot U3, the necessary pullup resistors for unused serial handshake inputs and a pullup resistor for the interrupt enable input for the quad UART INTEGRATED CIRCUIT

SCHEMATIC #3

The page labeled Schematic #3 contains the RS-422/485 drivers in slots U4 and U6 and the necessary termination resistors and jumper connectors to allow selectable 120 Ohm line termination for the TX driver outputs and the RX receiver inputs. The resistors for termination are housed in SIP resistor packs R23 and R24 while the jumper connectors are J3, J5, J7 and J9. J2 is the 50-pin I/O connector with all the balanced interface signals and FH1 is a fuseholder with a 1 Amp fuse to protect the +5V lines provided for external biasing.

SCHEMATIC #4

The page labeled Schematic #4 contains the RS-422/485 receivers in slots U5 and U7. R6 through R21 are 1 K Ohm discrete resistors used for receiver biasing while jumpers J4, J6, J8 and J10 are used to select the bias resistors for the RX lines on each port. SIP resistor pack R22 is used to terminate the CTS* balanced inputs. Jumpers J11, J13, J15 and J17 connect the balanced receive inputs to the balanced driver outputs for half-duplex operation. Jumpers J12, J14, J16 and J18 connect the balanced receiver inputs to the I/O connector for full-duplex operation.

2.4 Cleaning

If, for some reason, this product requires cleaning after delivery, most solvents are safe to use that are based on: Fluorine, Chlorine, Aqueous, and Alcohol. Do NOT use gasoline or thinner solvents on this product.

This page intentionally left blank

3.0 INSTALLATION

3.1 Unpack the COMM422

The contents of the COMM422 shipping packages are listed in Table 3-1

Table 3-1 Contents of COMM422 Shipping Packages

Qty	Description
1	COMM422 Printed Circuit Assembly
1	COMM422 User Manual *
1	Spare Fuse (1 Amp)

* One manual is shipped for each board ordered. Extra manuals may be purchased by calling SYSTRAN or by mail. Use the prefix "BTMR-" followed by the product order part number. (e.g. BTMR-COMM422).

The Printed Circuit Assembly is enclosed in an anti-static box. The box and the manual are packaged together in a larger box. Save the shipping material in case the board needs to be returned.

3.2 Inspect the COMM422

Examine the COMM422 to determine if any damage occurred during shipping.

3.3 Install the COMM422



NOTE: Since the COMM422 is an Electrostatic Sensitive Device (ESD), install the COMM422 hardware using a good anti-static workbench to protect the IP Module and carrier boards. The IP Module carrier board must be removed from the host system using good ESD practices and moved to an ESD-controlled area where the COMM422 can be installed.

The tools required for the COMM422 installation are listed in Table 3-2.

Table 3-2 COMM422 Installation Tools

Qty	Description
1	ESD Static Control Kit/Ground Strap/Etc.
1	Standard Flat-Head Screwdriver (Optional)

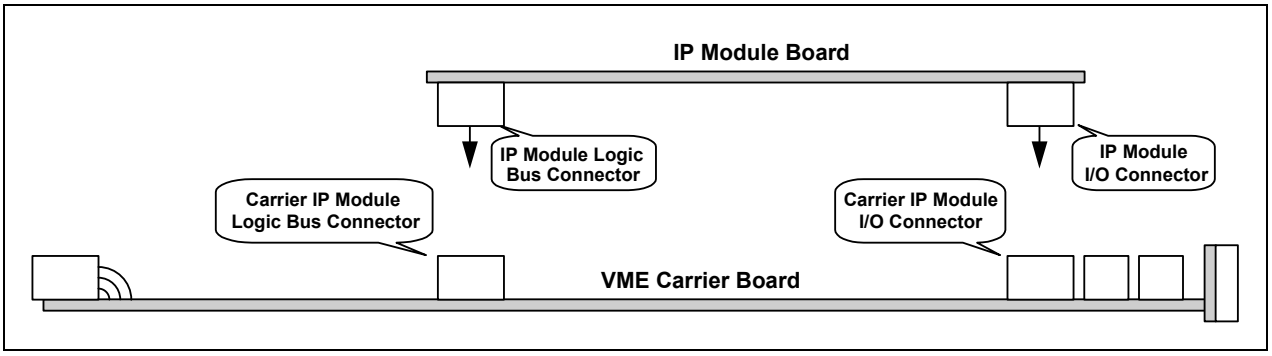


Figure 3-1 COMM422 Installed on a VME IP Module Carrier

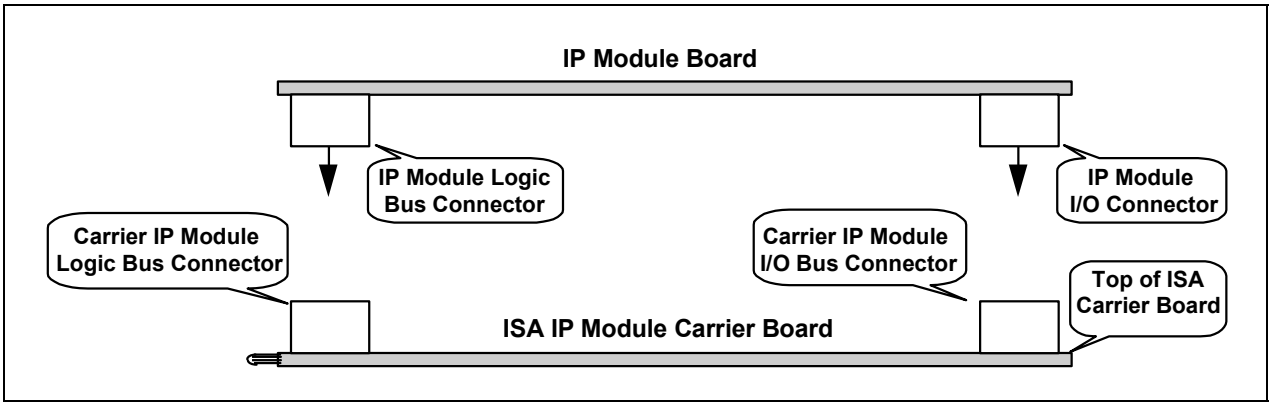


Figure 3-2 COMM422 Installed on an ISA IP Module Carrier

Figures 3-1 and 3-2 show the COMM422 installed on a VME IP Module carrier and on an ISA carrier, respectively.

The pin assignments for the IP Module Logic Bus connector are shown in Table 3-3. The signals in the left column are the original IP Module signal nomenclature, and the signals in the right column are those used by SYSTRAN Corp.

Table 3-4 shows the pin assignments for the I/O connector. Refer to the IP Module carrier board User Manual for more information. For information on mode-select jumper settings, refer to Section 6.3.

Referring to the appropriate figures and table described above, perform the following steps to install the COMM422. The asterisk (*) denotes optional items.

1. Turn OFF all power to the host system.
1. Remove the target IP Module carrier and move it to the ESD-controlled area where the COMM422 can be installed
1. Remove the COMM422 from the shipping package and place it on the ESD bench.
1. Install the COMM422 onto the carrier board by applying adequate and equal pressure to the COMM422 board at both ends.

* 1. Install four M2x5 mm flat-head machine screws onto the carrier board's IP Module connectors.

Check to ensure that all jumpers on the COMM422 board are set to the factory default if RS-422 operation is desired. If RS-485 half duplex or RS-485 full duplex operation is desired, see the detailed information and diagrams in section 6.3 for proper settings.

A 1A fuse should be installed in fuse holder FH1. If external biasing for the balanced receivers is not going to be necessary, the fuse need not be installed. A spare fuse is shipped with the COMM422.

This completes the installation of the COMM422 to the carrier board.

Table 3-3 IP Module Logic Bus Pin Assignments

Original IP Module Signal Names	IP Module Logic Bus Pin #	SYSTRAN Signal Names
GND	50	GND
reserved	49	RESERVED1
Ack*	48	N_ACK
A6	47	IPA6
Strobe*	46	N_STROBE
A5	45	IPA5
IntReq1*	44	N_INTREQ1
A4	43	IPA4
IntReq0*	42	N_INTREQ0
A3	41	IPA3
Error*	40	N_ERROR
A2	39	IPA2
DMAEnd*	38	N_DMAEND
A1	37	IPA1
reserved	36	RESERVED2
IOSel*	35	N_IOSEL
DMAck0*	34	N_DMACK0
IntSel*	33	N_INTSEL
DMAReq1*	32	N_DMAREQ1
MemSel*	31	N_MEMSEL
DMAReq0*	30	N_DMAREQ0
IDSel*	29	N_IDSEL
R/W*	28	IPR_N_W
+5V	27	+5VDC
GND	26	GND
GND	25	GND
+5V	24	+5VDC
+12V	23	+12VDC
-12V	22	-12VDC
BS1*	21	N_BS1
BS0*	20	N_BS0
D15	19	IPD15
D14	18	IPD14
D13	17	IPD13
D12	16	IPD12
D11	15	IPD11
D10	14	IPD10
D9	13	IPD9
D8	12	IPD8
D7	11	IPD7
D6	10	IPD6
D5	9	IPD5
D4	8	IPD4
D3	7	IPD3
D2	6	IPD2
D1	5	IPD1
D0	4	IPD0
Reset*	3	N_RESET
CLK	2	ICLK
GND	1	GND

Table 3-4 IP Module I/O Connector Pin Assignments

IP Module I/O Pin #	Signal Names
50	Not Used
49	Not Used
48	Not Used
47	Not Used
46	Not Used
45	Not Used
44	Not Used
43	Port D common
42	Port D CTS -
41	Port D RTS +
40	Port D CTS +
39	Port D RTS -
38	Port D RX +
37	Port D TX +
36	Port D RX -
35	Port D TX -
34	Port C Common
33	Port C CTS -
32	Port C RTS +
31	Port C CTS +
30	Port C RTS -
29	Port C RX +
28	Port C TX +
27	Port C RX -
26	Port C TX -
25	GND
24	GND
23	VDD (fused)
22	VDD (fused)
21	VDD (fused)
20	VDD (fused)
19	GND
18	Port B Common
17	Port B CTS -
16	Port B RTS +
15	Port B CTS +
14	Port B RTS -
13	Port B RX +
12	Port B TX +
11	Port B RX -
10	Port B TX -
9	Port A Common
8	Port A CTS -
7	Port A RTS +
6	Port A CTS +
5	Port A RTS -
4	Port A RX +
3	Port A TX +
2	Port A RX -
1	Port A TX -

This page intentionally left blank

4.0 PROGRAMMING GUIDE

4.1 Overview

This section of the manual describes the operation of the COMM422 from the software perspective, detailing the COMM422 registers and providing programming examples. A more detailed description of the hardware can be found in Sections 2.0: DESCRIPTION and 6.0: TYPICAL APPLICATIONS.

4.2 Description

The COMM422 is a quad, balanced serial-communications card featuring RTS/CTS handshake support for each port in 422 mode. The four serial ports are contained inside the 16C554 device which is mapped directly into the IP Module I/O space from IP Module addresses 00 *hex* to 1F *hex* . The I/O space consists of the following:

- Port A (00 *hex* - 07 *hex*)
- Port B (08 *hex* - 0F *hex*)
- Port C (10 *hex* - 17 *hex*)
- Port D (18 *hex* - 1F *hex*)
- Interrupt steering/driver disable register (20 *hex*)
- Interrupt vector register (21 *hex*)

Each of the four serial ports consists of twelve registers mapped into eight address locations. This is possible since some are READ-only, some are WRITE-only, and two (the divisor-latch registers for the baud-rate generator) can only be accessed when the divisor-latch-enable bit in the Line Control Register is set high. This gives a total of forty-eight registers that can be accessed in the quad-UART circuit, and two external CSR registers, for a total of fifty in the COMM422 I/O space. The interrupt-vector register is the only register accessed during an interrupt acknowledge cycle and a READ-only ID ROM function is enabled during ID accesses to the COMM422.

4.3 IP Module ID ROM Listing

Emulating the ID ROM function is possible due to the never-changing information presented by this function. This emulation saves space, lowers costs, improves reliability, and enables this IP Module to provide “hard-coded” READ accesses of this information.



NOTE: The ID address space is fully decoded. Therefore, attempted accesses to IPA addresses at and above IPA = 0C *hex* , or any attempt to WRITE to ID space results in no acknowledgment and its subsequent bus time-out error on the carrier upon which this IP Module resides.

The IP Module ID data is presented only on byte-lane 0 (IPD[7:0]). The upper byte-lane 1 (IPD[15:8]) READs as all zeros during valid ID READ-only accesses.

Table 4-1 is the COMM422 emulated ID Address Space ROM listing.

Table 4-1 COMM422 ID Address Space Listing

IP Module Address	Description	Data Read
IPA = 00 *hex*	ASCII 'I'	49 *hex*
IPA = 01 *hex*	ASCII 'P'	50 *hex*
IPA = 02 *hex*	ASCII 'A'	41 *hex*
IPA = 03 *hex*	ASCII 'C'	43 *hex*
IPA = 04 *hex*	SYSTRAN's ID	45 *hex*
IPA = 05 *hex*	COMM422's Model Number	73 *hex*
IPA = 06 *hex*	Revision Level	30 *hex*
IPA = 07 *hex*	Reserved	00 *hex*
IPA = 08 *hex*	Low Byte Driver ID	00 *hex*
IPA = 09 *hex*	High Byte Driver ID	00 *hex*
IPA = 0A *hex*	Number of Bytes Used	0C *hex*
IPA = 0B *hex*	CRC	DD *hex*

4.4 IP Module I/O Address Map

The COMM422 I/O Address Map was designed to have efficiently located data, control and interrupt registers in the IP Module I/O address space. The method for locating most of these registers is unique since they are contained in the 16C554 Quad UART which is mapped directly onto the IP Module I/O address space. The 16C554 UART is located in addresses IPA=00 through IPA=1F while the Interrupt Steering/Driver Disable Register is at IPA=20 and the Interrupt Vector Register is at IPA=21.

Interfacing to the UART is simplified since twelve registers necessary for data, status and control purposes are condensed into eight address locations which, in turn, are redundant since there are four identical serial ports. This address structure is possible since there are some READ-only and WRITE-only registers at the same address and two registers (the divisor latch - high and low byte) which can only be accessed when bit 7 in the LCR register is set to a '1'. Since there are four serial ports, these twelve registers are repeated four times for a total of 48 registers in the UART circuitry. Along with the vector and steering/disable register, there are a total of 50 registers that can be accessed on the COMM422. Below is a brief description of the COMM422 registers. A more detailed description follows in section 4.8.

The UART registers are address mapped as follows:

- Port A registers - IPA00 through IPA07
- Port B registers - IPA08 through IPA0F
- Port C registers - IPA10 through IPA17
- Port D registers - IPA18 through IPA1F

The twelve registers are listed below with the base address of each register being defined by IP Module bus address bits 4 and 5 as follows:

- Port A - IPA5=0, IPA4=0
- Port B - IPA5=0, IPA4=1
- Port C - IPA5=1, IPA4=0
- Port D - IPA5=1, IPA4=1

Read Holding Register (RHR):	This 8-bit, READ-only register at IPA= base+000b is used to temporarily hold incoming data arriving from the serial port.
Transmit Holding Register (THR):	This 8-bit, WRITE-only register at IPA= base+000b is used to temporarily hold data to be written out to the driver circuits.
Interrupt Enable Register (IER):	This 4-bit, WRITE-only register at IPA= base+001b allows the masking of incoming interrupts from receiver ready, transmitter empty, line status and modem status. The upper four bits of this register are not used and are set to zero.
FIFO Control Register (FCR):	This 6-bit, WRITE-only register at IPA= base+010b is used to enable the on-board 16-byte FIFOs, clear them, set the FIFO interrupt trigger level and select the type of DMA signaling. Bits 4 and 5 are not used in this register and are set to zero.
Interrupt Status Register (ISR):	This 6-bit, READ-only register at IPA= base+010b gives information on UART mode (450 mode vs. 550 mode) and provides information on the source of an interrupt in prioritized manner. Bits 4 and 5 are not used and are read as zero.
Line Control Register (LCR):	This 8-bit READ/WRITE register at IPA= base+011b allows for control of common modem settings such as word length, stop bits, parity and break control. All bits are used in this register and can be read as well as written.
Modem Control Register (MCR):	This 5-bit READ/WRITE register at IPA= base+100b is use to change the status of the hardware control lines between the modem or peripheral and the IP Module bus as well as enable/disable the local loopback mode for testing. Bits 5 through 7 of this register are not used and are set to zero.
Line Status Register (LSR):	This 8-bit READ-only register at IPA= base+101b provides the status of data transfer to the host via the IP Module bus. Functions such as receive holding register, overrun status, parity

	error status, break condition, etc. are monitored from this register.
Modem Status Register (MSR):	This 8-bit READ-only register at IPA= base+110b provides the current status of the control lines between the serial port and the IP Module bus side of the UART. A change of state of the CTS lines can be detected by a READ of this register.
ScratchPad Register (SPR):	This 8-bit READ/WRITE register at IPA= base+111b can be used as a general-purpose register.
Divisor Latch LSB (DLL):	This 8-bit WRITE-only register at IPA= base+000b (with LCR7 = "1") represents the lower 8 bits of the 16-bit divisor value for the programmable baud rate generator. This register is used with the DLM register to form a whole divisor.
Divisor Latch MSB (DLM):	This 8-bit WRITE-only register at IPA= base+001b (with LCR7 = "1") represents the upper bits of the 16-bit divisor value for the programmable baud rate generator. This register is used with the DLL register to form a whole divisor.
Interrupt Steering/Driver Disable Register:	(Not in UART) This 8-bit multi-function register is divided into two 4-bit sections. The lower four bits D3 through D0 are the port-interrupt-steering register and are used to determine which, if any, of the four UART channels are generating an interrupt. These four bits are READ-only and any WRITE to them will be ignored. The upper four bits D7 throughD4 are the driver-disable register. This section allows the enabling or disabling of the transmit drivers for each port where a '1' written to each bit enables the driver and a '0' written will disable the driver. This section can be written or read. The interrupt-steering/driver-disable register is at address 20 *hex.*
Interrupt Vector Register:	(not in UART) This 8-bit READ/WRITE register at IP Module address 21 *hex* can be written to in the IP Module I/O space and read from in the I/O space as well as read during an interrupt acknowledge cycle.

Table 4-2 is the COMM422 I/O Address Map and contains the data ports, control registers, and interrupt registers. (Note: [X] means not used.)

Table 4-2 COMM422 I/O Address Map

IP Module Address	Byte-Lane 1	Byte-Lane 0	Description
IPA = 00 *hex*	[XXXX,XXXX]	Recv. data	Receive Holding Register - Port A 8 bits, READ-only, LCR7 = 0
IPA = 00 *hex*	[XXXX,XXXX]	Tran. data	Transmit Holding Register - Port A 8 bits, WRITE-only, LCR7 = 0
IPA = 00 *hex*	[XXXX,XXXX]	control data	Divisor Latch Low Byte - Port A 8 bits, WRITE only, LCR7 = 1
IPA = 01 *hex*	[XXXX,XXXX]	control data	Interrupt Enable Register - Port A 8 bits (4 active), WRITE-only, LCR7 = 0
IPA = 01 *hex*	[XXXX,XXXX]	control data	Divisor Latch High Byte - Port A 8 bits, WRITE-only, LCR7 = 1
IPA = 02 *hex*	[XXXX,XXXX]	control data	FIFO Control Register - Port A 8 bits (6 active), WRITE-only, LCR7 = X
IPA = 02 *hex*	[XXXX,XXXX]	status data	Interrupt Status Register - port A 8 bits (6 active), READ-only, LCR7 = X
IPA = 03 *hex*	[XXXX,XXXX]	control data	Line Control Register - Port A 8 bits, READ/WRITE, LCR7 = X
IPA = 04 *hex*	[XXXX,XXXX]	control data	Modem Control Register - Port A 8 bits (4 active), READ/WRITE, LCR7 = X
IPA = 05 *hex*	[XXXX,XXXX]	status data	Line Status Register - Port A 8 bits, READ-only, LCR7 = X
IPA = 06 *hex*	[XXXX,XXXX]	status data	Modem Status Register - Port A 8 bits, READ-only, LCR7 = X
IPA = 07 *hex*	[XXXX,XXXX]	general data	Scratchpad Register - Port A 8 bits, READ/WRITE, LCR7 = X
IPA = 08 *hex*	[XXXX,XXXX]	Recv. data	Receive Holding Register - Port B 8 bits, READ-only, LCR7 = 0
IPA = 08 *hex*	[XXXX,XXXX]	Tran. data	Transmit Holding Register - Port B 8 bits, WRITE-only, LCR7 = 0
IPA = 08 *hex*	[XXXX,XXXX]	control data	Divisor Latch Low Byte - Port B 8 bits, WRITE only, LCR7 = 1
IPA = 09 *hex*	[XXXX,XXXX]	control data	Interrupt Enable Register - Port B 8 bits (4 active), WRITE-only, LCR7 = 0
IPA = 09 *hex*	[XXXX,XXXX]	control data	Divisor Latch High Byte - Port B 8 bits, WRITE-only, LCR7 = 1
IPA = 0A *hex*	[XXXX,XXXX]	control data	FIFO Control Register - Port B 8 bits (6 active), WRITE-only, LCR7 = X
IPA = 0A *hex*	[XXXX,XXXX]	status data	Interrupt Status Register - port B 8 bits (6 active), READ-only, LCR7 = X
IPA = 0B *hex*	[XXXX,XXXX]	control data	Line Control Register - Port B 8 bits, READ/WRITE, LCR7 = X
IPA = 0C *hex*	[XXXX,XXXX]	control data	Modem Control Register - Port B 8 bits (4 active), READ/WRITE, LCR7 = X
IPA = 0D *hex*	[XXXX,XXXX]	status data	Line Status Register - Port B 8 bits, READ-only, LCR7 = X
IPA = 0E *hex*	[XXXX,XXXX]	status data	Modem Status Register - Port B 8 bits, READ-only, LCR7 = X
IPA = 0F *hex*	[XXXX,XXXX]	general data	Scratchpad Register - Port B 8 bits, READ/WRITE, LCR7 = X
IPA = 10 *hex*	[XXXX,XXXX]	Recv. data	Receive Holding Register - Port C 8 bits, READ-only, LCR7 = 0

IP Module Address	Byte-Lane 1	Byte-Lane 0	Description
IPA = 10 *hex*	[XXXX,XXXX]	Tran. data	Transmit Holding Register - Port C 8 bits, WRITE-only, LCR7 = 0
IPA = 10 *hex*	[XXXX,XXXX]	control data	Divisor Latch Low Byte - Port C 8 bits, WRITE only, LCR7 = 1
IPA = 11 *hex*	[XXXX,XXXX]	control data	Interrupt Enable Register - Port C 8 bits (4 active), WRITE-only, LCR7 = 0
IPA = 11 *hex*	[XXXX,XXXX]	control data	Divisor Latch High Byte - Port C 8 bits, WRITE-only, LCR7 = 1
IPA = 12 *hex*	[XXXX,XXXX]	control data	FIFO Control Register - Port C 8 bits (6 active), WRITE-only, LCR7 = X
IPA = 12 *hex*	[XXXX,XXXX]	status data	Interrupt Status Register - port C 8 bits (6 active), READ-only, LCR7 = X
IPA = 13 *hex*	[XXXX,XXXX]	control data	Line Control Register - Port C 8 bits, READ/WRITE, LCR7 = X
IPA = 14 *hex*	[XXXX,XXXX]	control data	Modem Control Register - Port C 8 bits (4 active), READ/WRITE, LCR7 = X
IPA = 15 *hex*	[XXXX,XXXX]	status data	Line Status Register - Port C 8 bits, READ-only, LCR7 = X
IPA = 16 *hex*	[XXXX,XXXX]	status data	Modem Status Register - Port C 8 bits, READ-only, LCR7 = X
IPA = 17 *hex*	[XXXX,XXXX]	general data	Scratchpad Register - Port C 8 bits, READ/WRITE, LCR7 = X
IPA = 18 *hex*	[XXXX,XXXX]	Recv. data	Receive Holding Register - Port D 8 bits, READ-only, LCR7 = 0
IPA = 18 *hex*	[XXXX,XXXX]	Tran. data	Transmit Holding Register - Port D 8 bits, WRITE-only, LCR7 = 0
IPA = 18 *hex*	[XXXX,XXXX]	control data	Divisor Latch Low Byte - Port D 8 bits, WRITE only, LCR7 = 1
IPA = 19 *hex*	[XXXX,XXXX]	control data	Interrupt Enable Register - Port D 8 bits (4 active), WRITE-only, LCR7 = 0
IPA = 19 *hex*	[XXXX,XXXX]	control data	Divisor Latch High Byte - Port D 8 bits, WRITE-only, LCR7 = 1
IPA = 1A *hex*	[XXXX,XXXX]	control data	FIFO Control Register - Port D 8 bits (6 active), WRITE-only, LCR7 = X
IPA = 1A *hex*	[XXXX,XXXX]	status data	Interrupt Status Register - port D 8 bits (6 active), READ-only, LCR7 = X
IPA = 1B *hex*	[XXXX,XXXX]	control data	Line Control Register - Port D 8 bits, READ/WRITE, LCR7 = X
IPA = 1C *hex*	[XXXX,XXXX]	control data	Modem Control Register - Port D 8 bits (4 active), READ/WRITE, LCR7 = X
IPA = 1D *hex*	[XXXX,XXXX]	status data	Line Status Register - Port D 8 bits, READ-only, LCR7 = X
IPA = 1E *hex*	[XXXX,XXXX]	status data	Modem Status Register - Port D 8 bits, READ-only, LCR7 = X
IPA = 1F *hex*	[XXXX,XXXX]	general data	Scratchpad Register - Port D 8 bits, READ/WRITE, LCR7 = X
IPA = 20 *hex*	[0000,0000]	status data	Port Interrupt Steering/Driver Disable Register, bits 0 through 3 are READ-only (writes ignored) and bits 4 through 7 are READ/WRITE.
IPA = 21 *hex*	[0000,0000]	interrupt vector	Interrupt Vector Register, READ/WRITE, 8 bits

4.5 Power-Up and System Reset Defaults

All control registers in the COMM422 are set to their inactive state during a power-on reset from the host via the IP Module carrier. All outputs are disabled at this time including the interrupts. The transmitter output for each port as well as the receiver inputs are disabled as well. The driver outputs to the I/O connector are disabled upon reset and must be programmed to be enabled. The vector register is also cleared to all zeroes and must be programmed to the desired value before interrupts are enabled.

4.6 Word Access Address Translation

Table 4-3 shows the relationship between VME, ISA, and NuBus local bus addresses, and the IP Module address for word accesses.

In the table, the word BASE represents the I/O or ID base address. All addresses are in hexadecimal.

Table 4-3 Word Access Address Translation Table

VMEbus Address	ISA Bus Address	NuBus Address	IP Module Address
BASE + 0	BASE + 0	BASE + 2	IPA = 00 *hex*
BASE + 2	BASE + 2	BASE + 6	IPA = 01 *hex*
BASE + 4	BASE + 4	BASE + A	IPA = 02 *hex*
BASE + 6	BASE + 6	BASE + E	IPA = 03 *hex*
BASE + 8	BASE + 8	BASE + 12	IPA = 04 *hex*
BASE + A	BASE + A	BASE + 16	IPA = 05 *hex*
BASE + C	BASE + C	BASE + 1A	IPA = 06 *hex*
BASE + E	BASE + E	BASE + 1E	IPA = 07 *hex*
BASE + 10	BASE + 10	BASE + 22	IPA = 08 *hex*
BASE + 12	BASE + 12	BASE + 26	IPA = 09 *hex*
BASE + 14	BASE + 14	BASE + 2A	IPA = 0A *hex*
BASE + 16	BASE + 16	BASE + 2E	IPA = 0B *hex*

4.7 Byte Access Address Translation

Table 4-4 shows the relationship between VME, ISA, and NuBus local bus addresses, and the IP Module address for byte accesses.

In the table, the word BASE represents the I/O or ID* base address. All addresses are in hexadecimal.



NOTE: The addresses for Byte-lane 1 are not listed since this IP Module will not respond to high-byte accesses.

Table 4-4 Byte Access Address Translation Table

VMEbus Address	ISA Bus Address	NuBus Address	IP Module Address	Byte-Lane
BASE + 1	BASE + 0	BASE + 3	IPA = 00 *hex*	0
BASE + 3	BASE + 2	BASE + 7	IPA = 01 *hex*	0
BASE + 5	BASE + 4	BASE + B	IPA = 02 *hex*	0
BASE + 7	BASE + 6	BASE + F	IPA = 03 *hex*	0
BASE + 9	BASE + 8	BASE + 13	IPA = 04 *hex*	0
BASE + B	BASE + A	BASE + 17	IPA = 05 *hex*	0
BASE + D	BASE + C	BASE + 1B	IPA = 06 *hex*	0
BASE + F	BASE + E	BASE + 1F	IPA = 07 *hex*	0
BASE + 11	BASE + 10	BASE + 23	IPA = 08 *hex*	0
BASE + 13	BASE + 12	BASE + 27	IPA = 09 *hex*	0
BASE + 15	BASE + 14	BASE + 2B	IPA = 0A *hex*	0
BASE + 17	BASE + 16	BASE + 2F	IPA = 0B *hex*	0

4.8 Register Definitions

The COMM422 has a register-rich architecture with fifty accessible registers, forty-eight of which are contained within the Startech 16C554 Quad UART. The other two registers are contained within the logic interface part, IPCOM422.

The forty-eight registers in the UART represent a 4X replication of twelve basic registers of which some are READ/WRITE, WRITE-only or READ-only. Additionally, two of the registers can only be accessed when bit 7 in the LCR register is set to a '1'. This allows the addressing of all twelve registers to be performed with only the three lowest IP Module address bits: IPA1 through IPA3. Bits IPA4 and 5 are used by the interface controller to enable only one of the four UART ports at a time. The WRITE-only registers are primarily used for control purposes while READ-only registers are used for status and the READ/WRITE registers for READ-modify-WRITE operations.

The straightforward addressing mentioned above allows for one of each register type in the Quad UART to be accessed for each of the four ports. In the register definitions below, the IP Module address for each port will be listed as well as the register type using "W" for WRITE-only, "R" for READ-only and "R/W" for READ/WRITE. The necessary setting for bit 7 in the LCR register will also be listed. A '0' will denote a bit in a register that is hard-wired to a low state.

4.8.1 Receive Holding Register (RHR)

The primary function of this register is to allow a READ of the incoming data after being converted to parallel form from the incoming serial bit stream. In "16C450" mode , this register is read for every incoming character. In "FIFO" mode, this register's contents are transferred to the receive FIFO until the trigger level is reached.

Table 4-5 Receive Holding Register

IP Module addresses: Port A - 00\h Port B - 08\h Port C - 10\h Port D - 18\h **LCR bit 7**: = '0'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R/W	R	R	R	R	R	R	R	R
Reset to:	X	X	X	X	X	X	X	X

4.8.2 Transmit Holding Register (THR)

The primary function of the Transmit Holding register is to hold data to be converted to a serial bit stream before leaving the card via the driver circuits. Writing to this register transfers the contents of the low byte of the IP Module data bus into this register. Note that the Transmit Holding Register empty bit in the LSR register is set to a '1' when the transmitter is empty or data is transferred to the transmit shift register.

Table 4-6 Transmit Holding Register

IP Module addresses: Port A - 00\h Port B - 08\h Port C - 10\h Port D - 18\h **LCR bit 7**: = '0'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	Bit7	Bit7	Bit7	Bit7	Bit7	Bit7	Bit7	Bit7
R/W	W	W	W	W	W	W	W	W
Reset to:	X	X	X	X	X	X	X	X

4.8.3 Interrupt Enable Register (IER)

The main function of the Interrupt Enable register is to allow enabling of the following incoming interrupts: receiver ready, transmitter ready, line status and modem status. This can be done individually or as a group where '0' = disable and '1' = enable.

Table 4-7 Interrupt Enable Register

IP Module addresses: Port A - 01\h Port B - 09\h Port C - 11\h Port D - 19\h **LCR bit 7**: = '0'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	'0'	'0'	'0'	'0'	Modem Status Interrupt	Receive Line Status Interrupt	Transmit Holding Register	Receive Holding Register
R/W	W	W	W	W	W	W	W	W
Reset to:	X	X	X	X	X	X	X	X

4.8.4 Divisor Latch Low Byte (DLL)

This register is used to program the low byte of the 16-bit divisor for the baud rate generator. This value will be used with the high byte to program the bit rate for serial port operation.

Table 4-8 Divisor Latch Low Byte

IP Module addresses: Port A - 00\h Port B - 08\h Port C - 10\h Port D - 18\h **LCR bit 7**: = '1'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R/W	W	W	W	W	W	W	W	W
Reset to:	X	X	X	X	X	X	X	X

4.8.5 Divisor Latch High Byte (DLM)

This register is used to program the high byte of the 16-bit divisor for the baud rate generator. This value is used with the low byte to program the bit rate for serial port operation.

Table 4-9 Divisor Latch High Byte

IP Module addresses: Port A - 01\h Port B - 09\h Port C - 11\h Port D - 19\h **LCR bit 7**: = '1'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
R/W	W	W	W	W	W	W	W	W
Reset to:	X	X	X	X	X	X	X	X



NOTE: Possible values for the DLL and DLM registers are given in Table 4-19.

4.8.6 FIFO Control Register (FCR)

This register is used to program all aspects of UART FIFO operation including: FIFO enabling, FIFO resetting, DMA mode selection and receive FIFO trigger level as shown in the chart below.

Table 4-10 FIFO Control Register

IP Module addresses: Port A - 02\h Port B - 0A\h Port C - 12\h Port D - 1A\h **LCR bit 7**: = 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	RCV Trigger (MSB)	RCV Trigger (LSB)	0	0	DMA Mode Select	Transmit FIFO Reset	Receiver FIFO Reset	FIFO Enable
R/W	W	W	W	W	W	W	W	W
Reset to:	0	0	0	0	0	0	0	0

Receive FIFO trigger level:

Bit7	Bit6	Trigger Level
0	0	1 byte
0	1	4 bytes
1	0	8 bytes
1	1	14 bytes



NOTE: If the DMA-mode-select bit does not remain clear, unpredictable results may occur.

4.8.7 Interrupt Status Register (ISR)

This register is used to indicate the presence of an interrupt at the highest priority level currently pending. The priority levels are as follows:

1 - Receiver line status
1 - Receiver data ready or character time-out
1 - Transmitter Holding register empty
1 - Modem status.

Table 4-11 Interrupt Status Register

IP Module addresses: Port A - 02\h Port B - 0A\h Port C - 12\h Port D - 1A\h **LCR bit 7**: = 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	*	*	0	0	Interrupt Priority bit 2	Interrupt Priority bit 1	Interrupt Priority bit 0	Interrupt Status
R/W	R	R	R	R	R	R	R	R
Reset to:	0	0	0	0	0	0	0	0

* Bits 7 and 6 are set when FCR0 is set to '1'

Priority levels from bits 3 to 0:

P	D3	D2	D1	D0	Interrupt Source
1	0	1	1	0	LSR (Receiver Line Status Register)
2	0	1	0	0	RXRDY (Receive Data Ready)
2*	1	1	0	0	RXRDY (Receive Data Time-out)
3	0	0	1	0	TXRDY (Transmitter Holding Register Empty
4	0	0	0	0	MSR (Modem Status Register)

* A receive-data time-out occurs only when the Quad UART is operating in the FIFO mode. Additionally, it will not occur if the receive FIFO is empty. See the Startech data sheet in Appendix B for more information.

4.8.8 Line Control Register (LCR)

This register is used to program the characteristics of data to be transmitted and received in the UART. This includes standard settings such as word length, stop bits, parity enable and sense as well as setting a break and controlling access to the DLL and DLM registers. A full description follows.

Table 4-12 Line Control Register

IP Module addresses: Port A - 03\h Port B - 0B\h Port C - 13\h Port D - 1B\h **LCR bit 7**: = 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	Divisor Latch Enable	Set Break	Set Parity	Even Parity	Parity Enable	Stop Bits	Word Length Bit 1	Word Length Bit 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset to:	0	0	0	0	0	0	0	0

Bit explanations:

Bit 1 and 0 - These bits specify the length of the data word to be transmitted or received as described below:

Bit 1	Bit 0	Word Length
0	0	5 bits
0	1	6 bits
1	0	7 bits
1	1	8 bits

Bit 2 specifies the number of stop bits as follows:

Bit 2	Word Length	Stop Bit(s)
0	5,6,7,8	1
1	5	1.5
1	6,7,8	2

Bit 3 - '0' = no parity, '1' = parity used
Bit 4 - If parity is enabled, '0' = odd, '1' = even
Bit 5 - If parity is enabled, this bit selects the forced parity format. LCR5 = '1' and LCR4 = '0' will force a '1' in the parity bit while LCR5 = '1' and LCR4 = '1' will force a '0' into the data.
Bit 6 - Setting this bit to '1' causes a break condition to be transmitted to the communication terminal.
Bit 7 - The internal baud rate counter latch enable which must be set to a "1" to access the DLL and DLM registers.

4.8.9 Modem Control Register (MCR)

This register is used to control the interface with a modem or peripheral device to the RS-422 circuitry. It will control the assertion of RTS* , enable or disable the interrupt outputs and specify normal vs. internal loopback mode. A detailed description follows.

Table 4-13 Modem Control Register

IP Module addresses: Port A - 04\h Port B - 0C\h Port C - 14\h Port D - 1C\h **LCR bit 7**: = 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	0 (MSB)	0 (LSB)	0	Loopback Enable	INTX Enable Select	Not used Reset	RTS* Reset	DTR*
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset to:	0	0	0	0	0	0	0	0

Bit explanations:

Bit 0 - Setting this bit to '1' forces DTR* output to go low (not used for RS-422/485 operation).

Bit 1 - Setting this bit to '1' forces RTS* output to go low.

Bit 2 - This bit is only used during the internal loopback mode.

Bit 3 - This bit has no effect since the INTN* pin on the UART is set to always enable the interrupt driver outputs.

Bit 4 - Setting this bit to a '1' will enable the wire loopback mode. In this mode, the transmitter output is set high (mark) while the receiver inputs RX, CTS*, DSR*, CD* and RI* are all disabled. Internally, the transmitter output is connected to the receiver input and DTR*, RTS*, MCR2 and INT enable are connected to the modem control inputs. This allows an internal test that excludes external receiver and driver circuits. Note that DSR*, CD* and RI* are pulled high external to the UART and are therefore not used on this board regardless of operation mode.

Bits 5-7 - Not used and set to '0'.

4.8.10 Line Status Register (LSR)

This register provides the current status such as receive data ready, errors, break interrupt or empty transmit holding register. A detailed description follows.

Table 4-14 Line Status Register

IP Module addresses: Port A - 05\h Port B - 0D\h Port C - 15\h Port D - 1D\h **LCR bit 7: =** 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	FIFO Error	Trans Empty	Trans Holding Empty	Break Interrupt	Framing Error	Parity Error	Overrun Error	Receive Data Ready
R/W	R	R	R	R	R	R	R	R
Reset to:	0	0	0	0	0	0	0	0

Bit explanations:

Bit 0 - A '1' for this bit shows that data has been saved in the receive holding register or FIFO.

Bit 1 - A '1' for this bit shows that another character arrived before the receive holding register or FIFO could be emptied. Note that the "extra" character will be written into the shift register but not written into the FIFO.

Bit 2 - A '1' for this bit shows that the received data does not have correct parity. In FIFO mode, this error will pertain to the character at the top of the FIFO.

Bit 3 - A '1' for this bit shows that received data failed to contain a valid stop bit. In FIFO mode, this error pertains to the character at the top of the FIFO.

Bit 4 - A '1' for this bit shows that a break condition was received. In FIFO mode, one zero character will be loaded into the FIFO.

Bit 5 - A '0' for this bit denotes a full transmit holding register while a '1' allows the host bus to load another character.

Bit 6 - A '0' for this bit denotes a full transmit holding register and shift register. A '1' on this bit denotes empty registers. In FIFO mode, a "1" will also denote an empty transmit FIFO.

Bit 7 - A '1' on this bit shows that at least one parity error, framing error or break condition was found in the FIFO. It is cleared to a "0" for normal operation when the LSR is read.

4.8.11 Modem Status Register (MSR)

This register will provide information to the host regarding the current state of the control lines from the modem or peripheral. A detailed description of the bits follows the table.

Table 4-15 Modem Status Register (MSR)

IP Module addresses: Port A - 06\h Port B - 0E\h Port C - 16\h Port D - 1E\h **LCR bit 7:** = 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	CD	RI	DSR	CTS	Delta CD*	Delta RI*	Delta DSR*	Delta CTS*
R/W	R	R	R	R	R	R	R	R
Reset to:	X	X	X	X	0	0	0	0

Bit explanations:

Bits 3,2,1, and 0 - These bits, when read, show that the status of the associated receive lines has changed since the register was last read.

Bits 7,6,5, and 4 - These bits when read, show the current status of the associated receive lines as the complement of the actual input. In local loopback mode, bit 7 will show the equivalent to the INT enable in the MCR while bit 6 will be the equivalent to MCR bit 2, bit 5 will be equivalent to the DTR in the MCR and bit 4 will be equivalent to the RTS in the MCR. When not in loopback mode, bits 7,6,5 and 4 represent CD*, RI*, DSR* and CTS*, respectively. Only bits 0 and 4 will be actively used on the COMM422.

4.8.12 Scratchpad Register (SPR)

This 8-bit register is a general-purpose READ/WRITE storage element.

Table 4-16 Scratchpad Register

IP Module addresses: Port A - 07\h Port B - 0F\h Port C - 17\h Port D - 1F\h **LCR bit 7:** = 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset to:	0	0	0	0	0	0	0	0

4.8.13 Port Interrupt Steering/Driver Disable Register

This 8-bit, dual function register uses the lower four bits (READ-only) to communicate to the host which of the four communication ports is asserting an interrupt. The upper 4 bits of this register are used to enable or disable the driver circuits for each individual port. The upper four bits can be written or read.

Table 4-17 Port Interrupt Steering/Driver Disable Register

IP Module addresses: All ports - IPA = 20\h **LCR bit 7: =** 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	ENDRD	ENDRC	ENDRB	ENDRA	Port D Interrupt	Port C Interrupt	Port B Interrupt	Port A Interrupt
R/W	R/W	R/W	R/W	R/W	R	R	R	R
Reset to:	0	0	0	0	0	0	0	0

Bit explanations:

Bits 0 through 3 - Each of these bits will be a '1' when the associated port interrupt is active. Note that it is possible for more than one bit to be active at a time.

Bits 4 through 7 - These bits are set to a '1' when the desired port is to be enabled for data transmit. Bits 4 through 7 will enable ports A through D, respectively. At reset, these values are cleared to zeros and therefore a WRITE from the host is necessary before any data is to be transmitted.

4.8.14 Interrupt Vector Register

This register holds the 8-bit vector value that is read during an interrupt-acknowledge cycle. This register can be written in the I/O space and read from in the I/O space as well as during an interrupt acknowledge.

Table 4-18 Interrupt Vector Register

IP Module addresses: (All ports) IPA = 21 *hex* **LCR bit 7: =** 'X'								
Bit #	7	6	5	4	3	2	1	0
Bit Name	Vector Bit 7	Vector Bit 6	Vector Bit 5	Vector Bit 4	Vector Bit 3	Vector Bit 2	Vector Bit 1	Vector Bit 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset to:	0	0	0	0	0	0	0	0

4.9 Programming Examples

The following example illustrates how to program the COMM422 device. It does not include any platform or carrier-specific information, such as configuring the host interrupt level. The example assumes that interrupts will be used as they are required for reliable operation.

In addition to using the hardware FIFOs present in the UART, many device drivers also utilize software buffers to provide additional levels of storage. To simplify the explanation, only a single serial port is used in this example. These concepts can be readily extended to additional serial ports, although the implementation is dependent on the software design. For example the entire COMM422 device can appear as a single device, or it may be a separate logical device for each serial port.

The following programming example is broken down into the separate entry points typically used in a device driver. It does not address any protocol used by higher-level software to prevent multiple ports from simultaneously transmitting when RS-485 mode is used.

4.9.1 Open

The purpose of this routine is to provide any initialization needed by the driver, configure the hardware, and install the interrupt service routine.

- Allocate any storage required by the software transmit and receive buffers
- Initialize any memory locations as required by the software
- Initialize the interrupt vector (IPA = 0x21) to the desired value
- Write a ‘80h’ to the Line Control Register (LCR) to set the divisor latch enable bit
- WRITE the appropriate value to the DLL and DLM registers to set the baud rate to the appropriate value. The following table lists the values to WRITE to these registers for each baud rate:

Table 4-19 DLL and DLM Register Baud Rate Values

Baud Rate	DLM Value	DLL Value
50	27h	10h
75	1Ah	0Bh
110	11h	C1h
134.5	0Eh	85h
150	0Dh	05h
300	06	83h
600	03	73h
1200	01	A1h
1800	01	15h
2000	00	FAh
2400	00	D0h
3600	00	8Bh
4800	00	68h
7200	00	45h
9600	00	34h
19200	00	1Ah
38400	00	0Dh
56000	00	09
128000	00	04
256000	00	02
512000	00	01

- WRITE the appropriate value to the Line Control Register based on desired data size, stop size, and parity. Clear the divisor latch bit in order to re-enable access to the transmit and receive registers.
- WRITE a value to the FIFO Control Register (FCR) to configure the FIFOs as the application requires. The DMA-mode-select bit should remain clear in all situations.
- WRITE the appropriate value to the Modem Control Register (MCR) to set the value of RTS* as desired. Ensure that the ENDRA through ENDRD bits in the interrupt steering/driver disable register are cleared to zeros until ready to transmit data. Typically, the INT enable bit in this register remains clear and interrupts are enabled/disabled by the Interrupt Enable Register (IER). The loopback bit also typically remains clear since it is usually used for hardware-confidence tests, production testing by the factory, etc.
- Install the interrupt-service routine as appropriate for the platform.
- Enable the interrupts via the Interrupt Enable Register (IER) as the application requires. While the need for a modem status or line status interrupt may vary according to the customer's requirements (e.g. hardware handshaking), the transmit holding register and receive holding register interrupts should be used to minimize the CPU loading by fully utilizing interrupts.
- Enable the output drivers via the appropriate bit in the interrupt steering/driver disable register.

4.9.2 Write

This routine may be used to transfer an entire string of characters to the software's transmit buffer.

- If insufficient storage is available in the transmit buffer, take appropriate action (i.e. return error, allocate additional storage, etc.)
- Insert data to be transmitted into transmit buffer
- If transmit-holding-register interrupts are disabled, reenable them via the Interrupt Enable Register to allow this new data to be transmitted.

4.9.3 Read

This routine may be used to retrieve any received data present in the software receive buffers.

- Check to see if any data is present in software receive buffers. If not, take appropriate action (i.e. return error, put calling process to sleep, etc.)
- Check to see if errors were logged by the interrupt service routine. If so, return error.
- Return next character(s) from the receive buffer to the calling process and update receive buffer.

4.9.4 Ioctl

This routine may be used to query device status, reconfigure the hardware, etc. It is largely dependent on operating system and user requirements. Possible uses include:

- Checking to see if any data is present in software receive buffers.
- Reconfiguring the serial port to new parameters (data size, stop size, parity, baud rate, etc.).

- Setting/clearing the driver enable bits in the interrupt steering/driver disable register as required.
- Returning the current status of the port's configuration (data size, stop size, parity, baud rate, etc.).
- Specifying signal levels for RTS*.
- Returning the status of the CTS* handshake line.
- Specifying type of flow control used by the driver (software, hardware, none).
- Specifying new characters to be used for software flow control (XON/XOFF).

4.9.5 ISR

The interrupt-service routine deals with any interrupts generated by the COMM422 device. It is not called directly by the user's process, but is instead executed after an interrupt acknowledge by the host CPU. Some of its primary functions include: transmitting any data available in the software-transmit buffer, emptying the UART's receive buffer or FIFO, and placing this data in the software's receive buffer.

This ISR example only considers interaction with a single port. There are several methods of extending this to multiple ports, including handling all four ports within a single ISR or utilizing the operating system to chain multiple interrupt handlers (one per port) off of a single interrupt source.

- Identify source of interrupt within port by reading Interrupt Source Register.
- If interrupt source is receiver-line-status, READ Line Status Register (LSR) and record the new line status if appropriate. The driver could then send a signal to the user process via the operating system or indicate the change in status on a subsequent call to the READ routine.
- If interrupt source is received-data-available, READ the data in the Receive Holding Register (RHR) and place the data in the software receive buffer. This process could then be repeated until no received data is available, which can be determined by examining bit 0 in the LSR. Use this process for character-time-out interrupts as well.
- If interrupt source is transmit-holding-register-empty, examine the status of the transmit buffer. If this software buffer is empty, disable transmit-holding-register-empty interrupts by clearing this bit in the Interrupt Enable Register. If not, remove a character from the transmit buffer and WRITE it to the Transmit Holding Register (THR). Then READ line status to check the status of the transmit holding bit (bit #5). As long as this bit is set, this process can be repeated (assuming sufficient data exists in the transmit buffer).
- If interrupt source is modem-status, READ the Modem Status Register (MSR) and record the new modem status if appropriate. The driver could then send a signal to the user process or use these inputs to implement hardware flow control.

4.9.6 Close

The purpose of this routine is to provide any termination needed by the driver, configure the hardware to a benign state, and uninstall the interrupt service routine.

- Disable all interrupts except output enable interrupts via the Interrupt Enable Register
- Wait until output buffer has been emptied by ISR
- Wait until all data has been shifted out of the port (the 'Trans Empty" bit of Line Status Register is set by the UART when this occurs)
- Write an appropriate value to the Modem Control Register (typically RTS* and DTR* negated)
- Disable the output drivers for this port via the appropriate bit in the Interrupt Steering/Driver Disable Register
- Disable all interrupt sources via the Interrupt Enable Register
- Uninstall the interrupt-service routine as appropriate for the platform
- Free any storage allocated to the driver during the Open entry point

5.0 PERFORMANCE

5.1 Overview

The purpose of this section is to provide several sets of empirical data that present typical performance parameters beyond those provided in the specification. The important feature is that these are typical responses for the configuration cited, and do not supplant the maximum and minimum envelopes presented in Section 1.0: INTRODUCTION. The figures in this section are a collection of typical rise/fall, and board response times to various system stimuli and IP Module cycles.

5.2 Timing Measurements

Timing measurements were conducted on the COMM422 using an HP model 16500A Logic Analyzer with a 1 GHz sampling Master Module and a 400 MHz Digitizing Oscilloscope Module. Two COMM422 IP Modules were used, one in slot A of a VME6U Carrier and the other in slot D. The I/O ends of boards A and D were interconnected with three-foot ribbon cables and a transition board with null-modem connections. An IP Module breakout board and cable were used with the board in slot A to connect the logic analyzer to board A I/O. The probe from the Digitizing Oscilloscope was used to monitor the balanced driver outputs TXA- and TXA+ at pins 1 and 3 respectively, and the RXA- and RXA+ balanced receive inputs at pins 2 and 4 respectively.

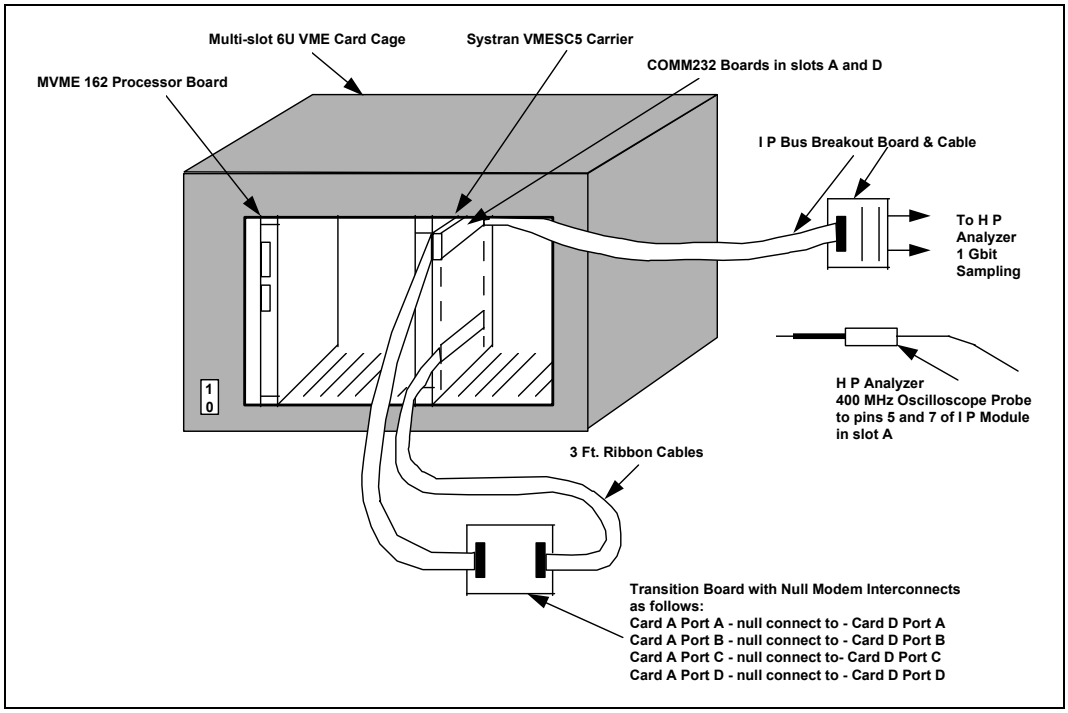


Figure 5-1 Test Configuration for Oscilloscope and Analyzer Measurements

5.3 Oscilloscope Measurements

The measurements shown in Figures 5-2 through 5-9 were made with the 400 MHz Digitizing Oscilloscope built into the HP Model 16500A Logic Analyzer. The probe was attached to the Transmit Data "+" and "-" signals, and the Receive Data "+" and "-" signal pins for port A. The signals in figures 5-2 and 5-3 represent the TXA- output for both the rising and falling edges. The signals in figures 5-4 and 5-5 represent the TXA+ output for both rising and falling edges.

Figures 5-6, 5-7, 5-8 and 5-9 give the same information as figures 5-2 through 5-5, respectively, except that the measurements were taken at the Receive-Data pins being driven.



NOTE: Necessary characteristics of drivers used in any RS-485 serial communications interface according to the EIA-485 standard include the following:

- The driver will produce an output voltage of at least 1.5 Volts between "+" and "-" pins into a load of 54 Ohms.
- The driver will not allow a current at the output terminals of greater than 250 mA.
- The driver must be capable of transmitting data at rates of up to 10 megabits per second.

Since the RS-422 specifications are basically a subset of EIA-485, drivers and receivers that meet or exceed 485 specifications will also meet or exceed 422 specifications as well.

For more information on the RS-422 or 485 standard, a copy of the TIA/EIA-422-B or EIA-485 Specifications can be obtained from Global Engineering Documents at the following address:

Global Engineering Documents
15 Inverness Way East
Englewood, CO 80112
Phone. (303)-792-2181

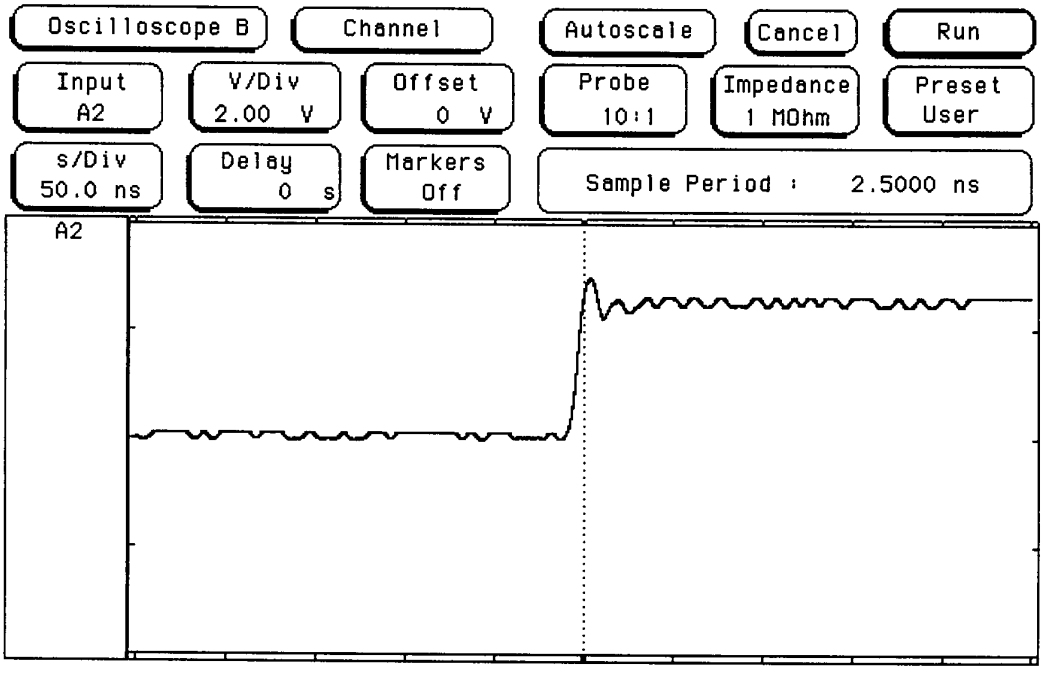


Figure 5-2 Rise-Time/Amplitude for Port A TX (-) at Driver Out

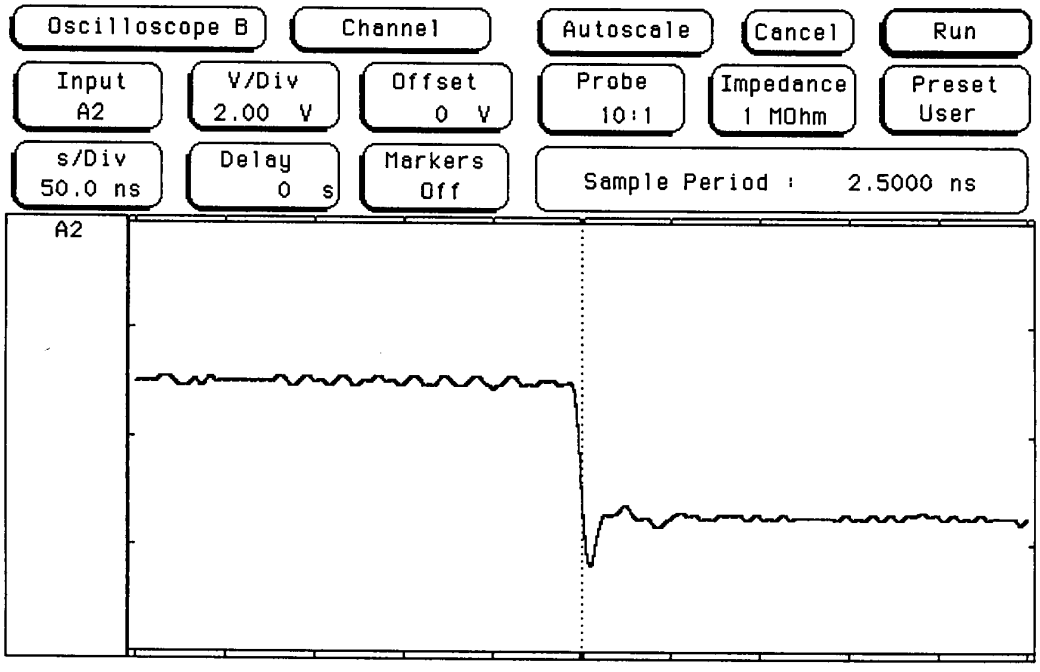


Figure 5-3 Fall-Time/Amplitude for Port A TX(-) at Driver Out

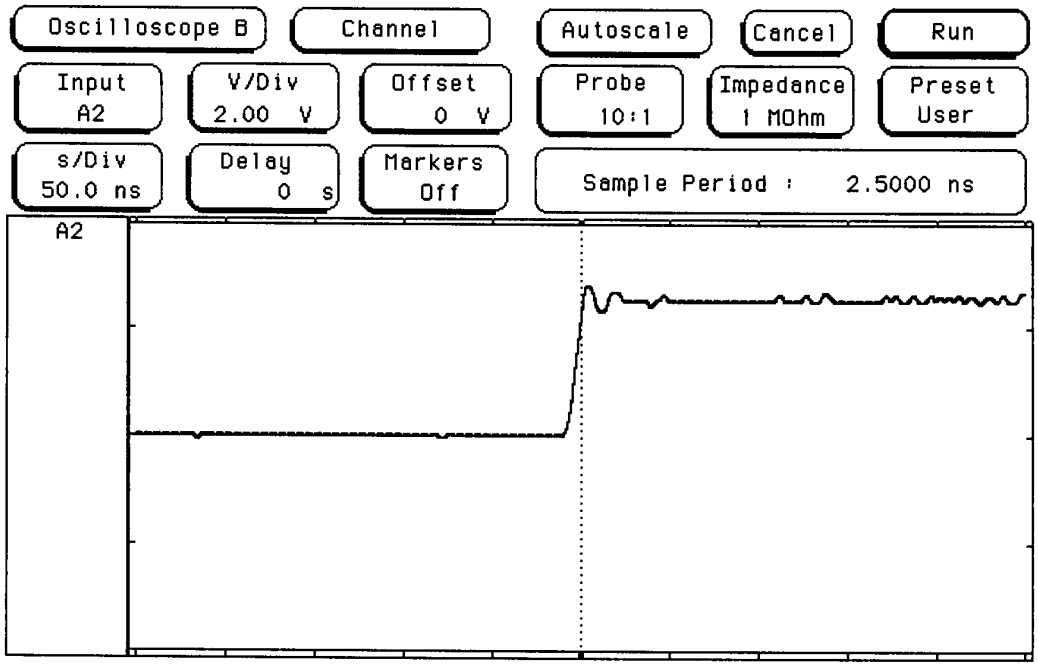


Figure 5-4 Rise-Time/Amplitude for Port A TX(+) at Driver Out

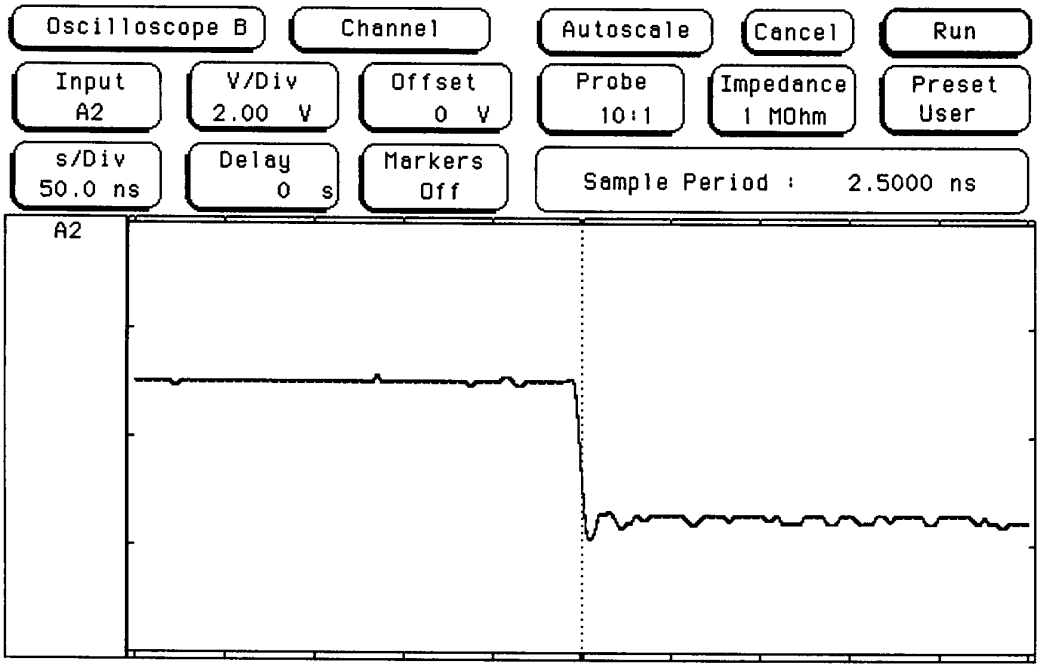


Figure 5-5 Fall-Time/Amplitude for Port A TX(+) at Driver Out

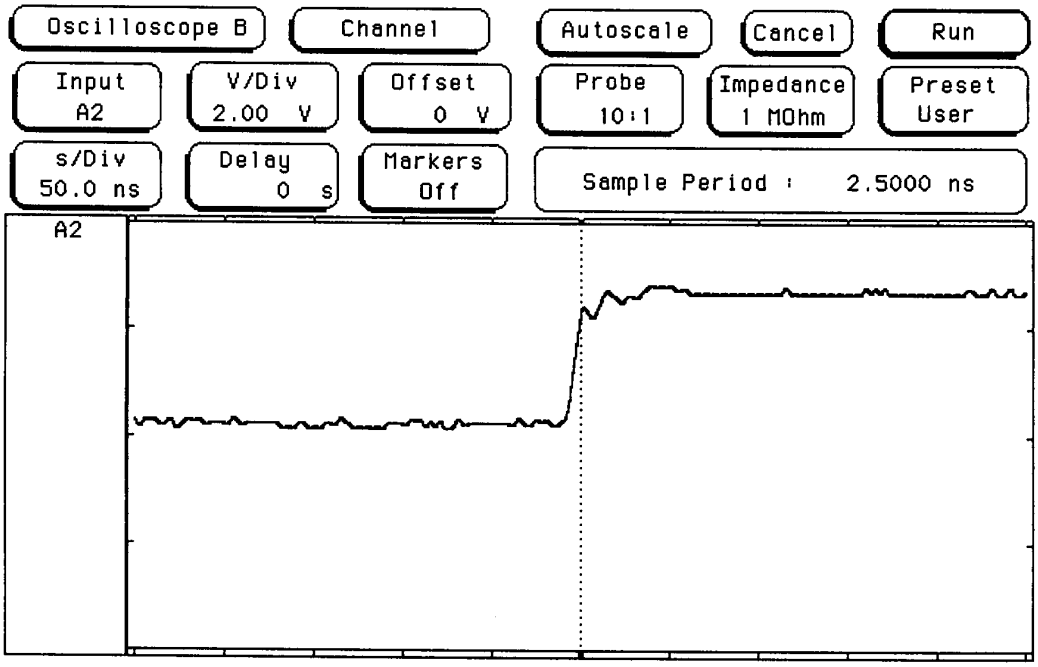


Figure 5-6 Rise-Time/Amplitude for Port A RX(-) at Receiver In

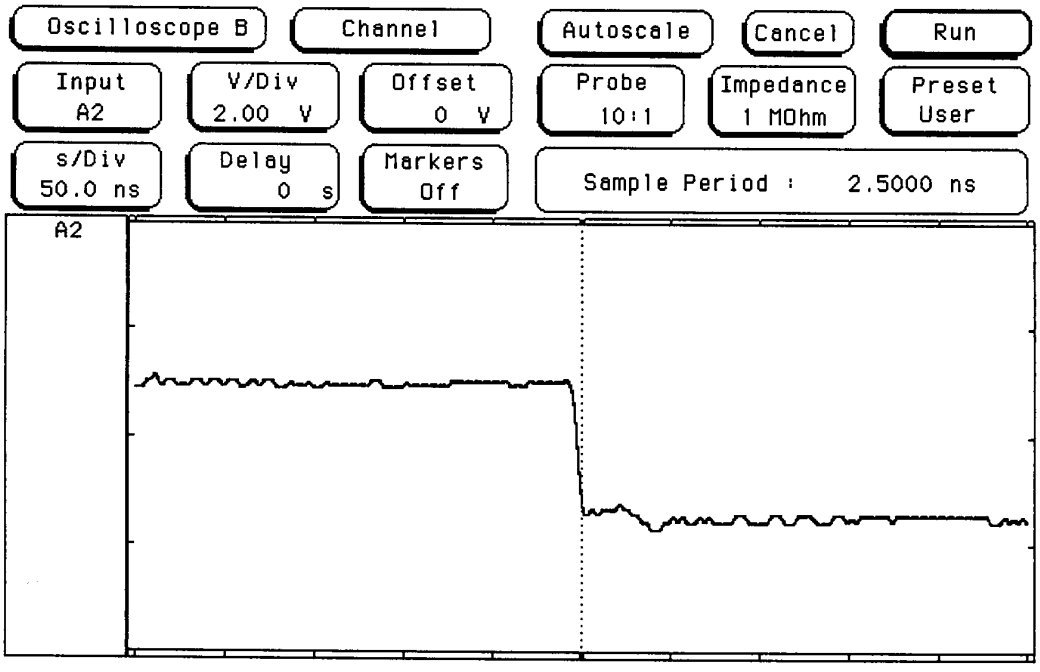


Figure 5-7 Fall-Time/Amplitude for Port A RX(-) at Receiver In

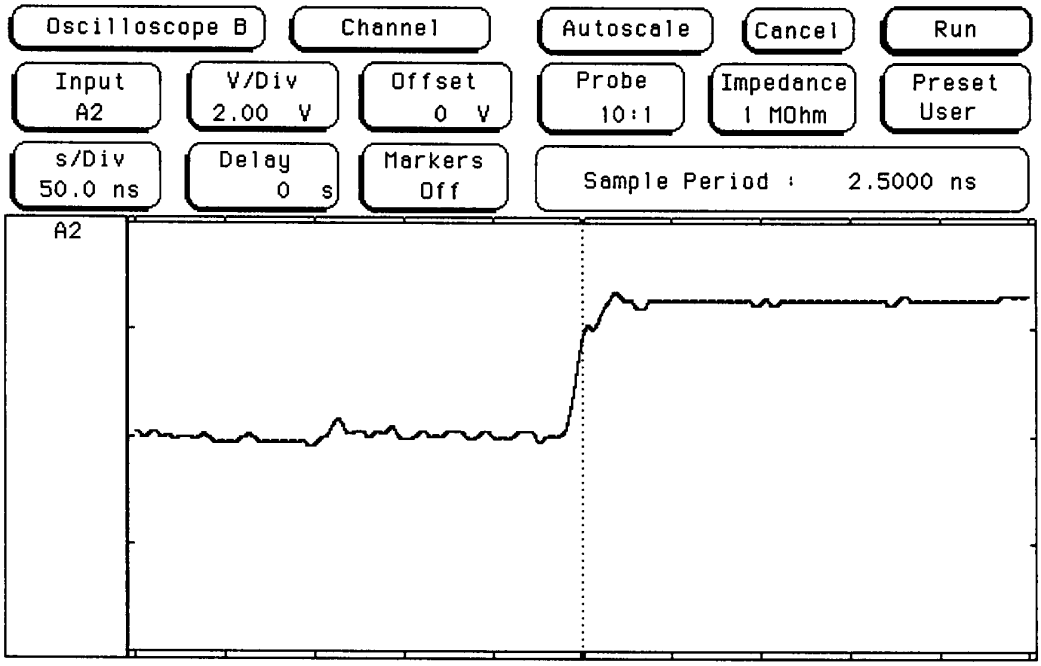


Figure 5-8 Rise-Time/Amplitude for Port A RX(+) at Receiver In

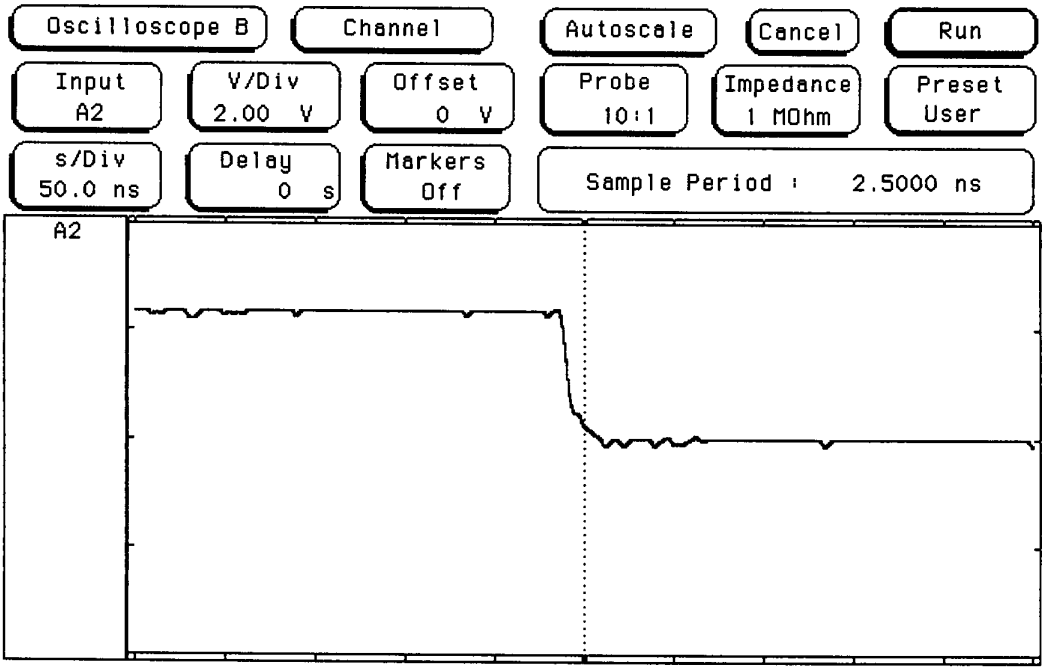


Figure 5-9 Fall-Time/Amplitude for Port A RX(+) at Receiver In

5.4 State Timing Diagrams

The state waveforms shown in Figures 5-10 through 5-16 were made using the HP 1 GHz Timing Master Module and the SYSTRAN IP Module Logic Bus Extender (IPLBE) with the IP Module Logic Bus Breakout Board (IPLBB). Several of the signals on the logic analyzer‘s screen are active low, and are represented by a ‘/’ as the first character as opposed to the ‘N_’ used elsewhere, due to the limited amount of character spacing.

5.4.1 ID Read Cycle

Figure 5-10 is a complete COMM422 ID READ cycle of IP address 00 *hex* . The signals of interest are /IDSEL, N_ACK, and the four IPD signals (3,2,1 and 0). The data being driven to the host carrier is the lower four bits of the ASCII character “I” or a “9” *hex* . The data is typically driven within about 22 to 25 ns from the rising edge of the IP bus clock which is well within the limit of 40 ns specified in the ANSI/VITA 4-1995 specification.



NOTE: For all COMM422 transactions with the host carrier, one wait state will be asserted by the COMM422 to allow for extra hold time for WRITE data to the quad UART and ample time for data to be read from the UART, given maximum access times. Therefore, N_ACK will be held off for one clock cycle following the initialization of the select cycle. On carriers that assert one or more hold states, the wait state will occur simultaneously with the first hold state.

5.4.2 I/O Write and Read Cycle

Figure 5-11 is a complete I/O WRITE cycle of IP address 07 *hex* which is the Scratchpad register for UART Port A. The signals of interest are /IOSEL, N_ACK, R/NW and the four IPD bits 3,2,1 and 0.

Figure 5-12 is a complete I/O READ cycle at the same address as the WRITE above. The same signals of interest are active with R/NW now being high and the data being read back to the host carrier (a READ of the reset value of 00 *hex*) occurring within about 18 ns.

5.4.3 Interrupt Request

Figure 5-13 shows an I/O WRITE to IP address 01 *hex* (the quad-UART Interrupt-Enable Register - Port A) producing an interrupt within about 87 ns. Figure 5-14 shows the processor returning via the carrier with an IP interrupt-select cycle in about 2 μs.

5.4.4 Interrupt Select Cycle

Figure 5-15 shows the details of an interrupt select cycle. The signals of interest for this cycle are /INTSL, N_ACK, R/NW, IPA1 and the four IPD bits 3,2,1 and 0. Note that since the COMM422 only supports INTREQ0, IPA1 will be ‘0’ during the access. Also, the vector value is returned to the carrier within about 19 ns of the IP CLK; 24 ns is typical.

Figure 5-16 shows a READ of IP address 02 *hex* (quad-UART Interrupt-Information Register - Port A) which clears the interrupt request from the UART and therefore removes the interrupt request from the COMM422 board.

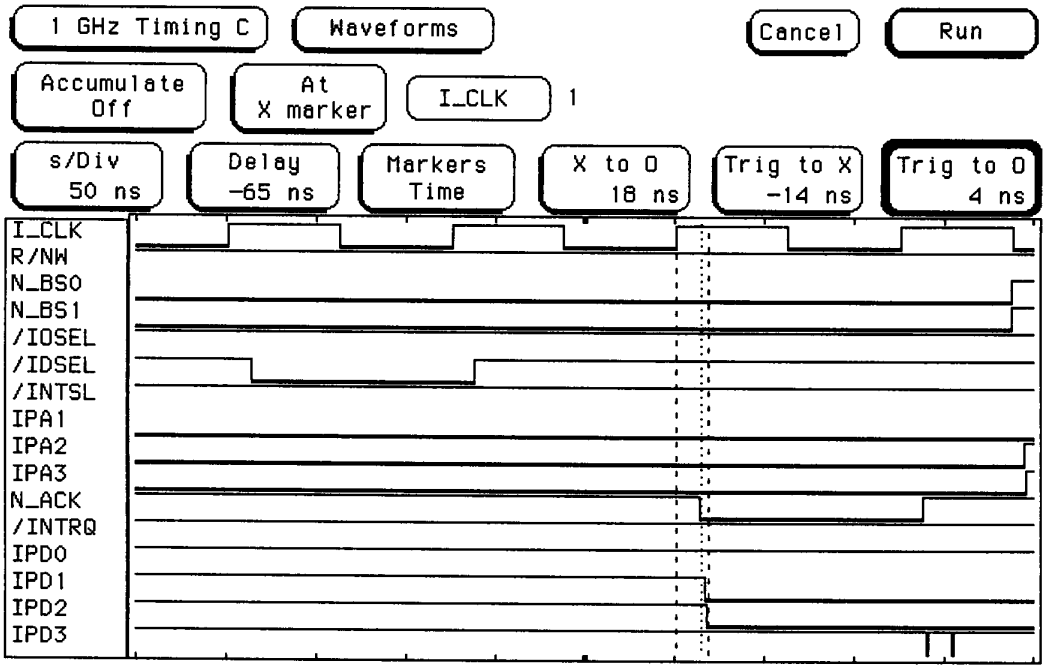


Figure 5-10 ID Read Cycle from IP Module Address 00 hex

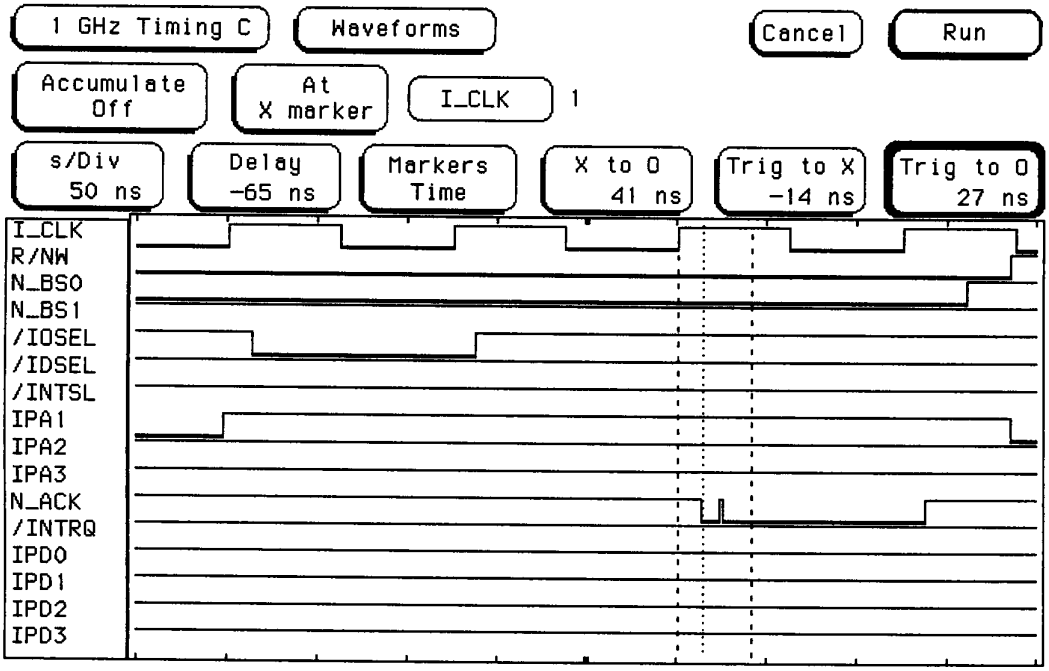


Figure 5-11 I/O Write Cycle IP Module Address 07 hex (Port A Scratchpad Register)

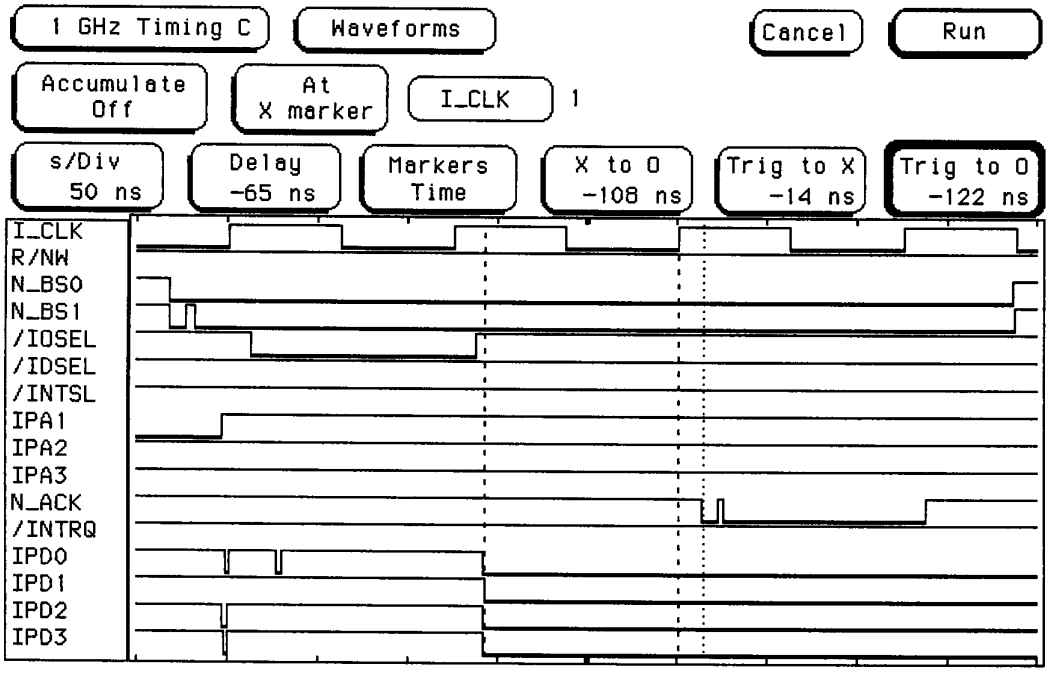


Figure 5-12 I/O Read Cycle IP Module Address 07 *hex* (Port A Scratchpad Register)

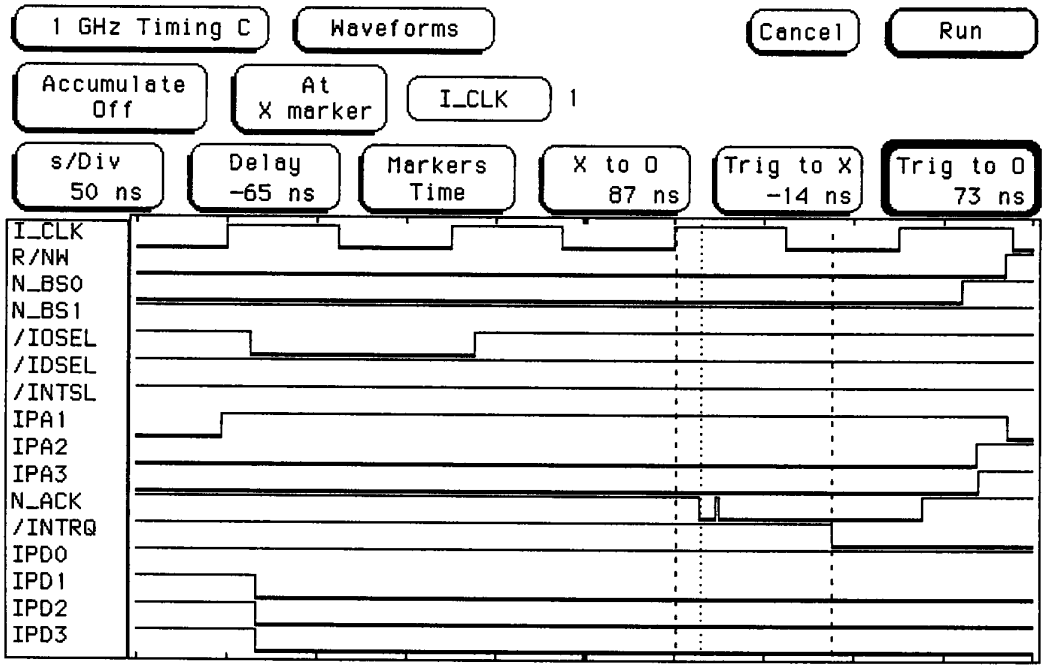


Figure 5-13 I/O Write to IP Module Address 01 *hex* (Port A Interrupt-Enable Register) Causing an Interrupt Request to be Generated

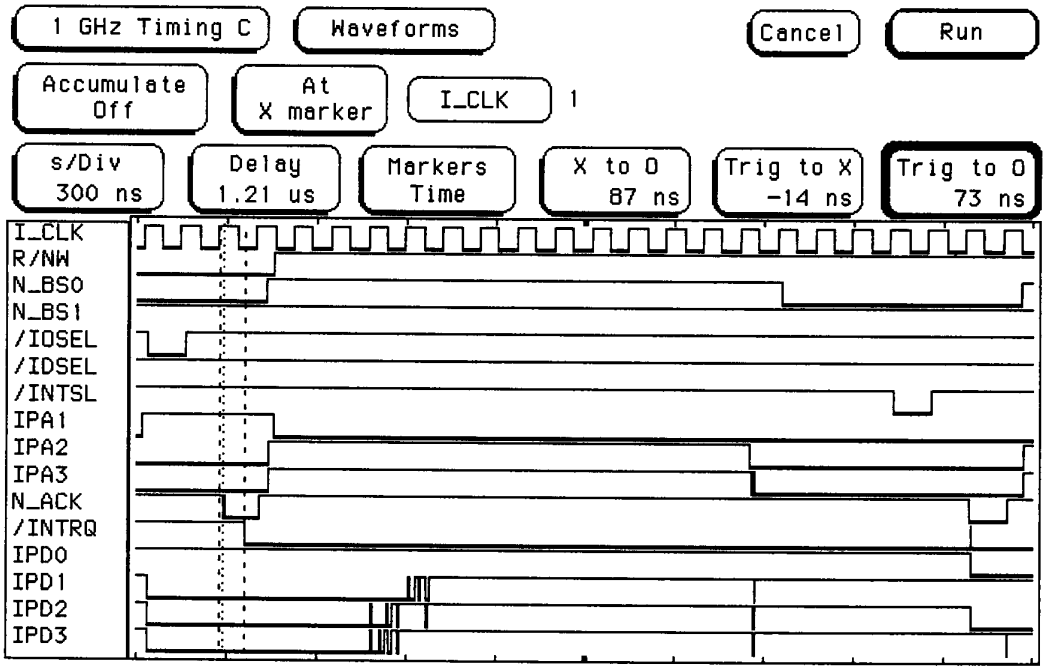


Figure 5-14 Interrupt Request Followed by an IP Module Interrupt Select Cycle

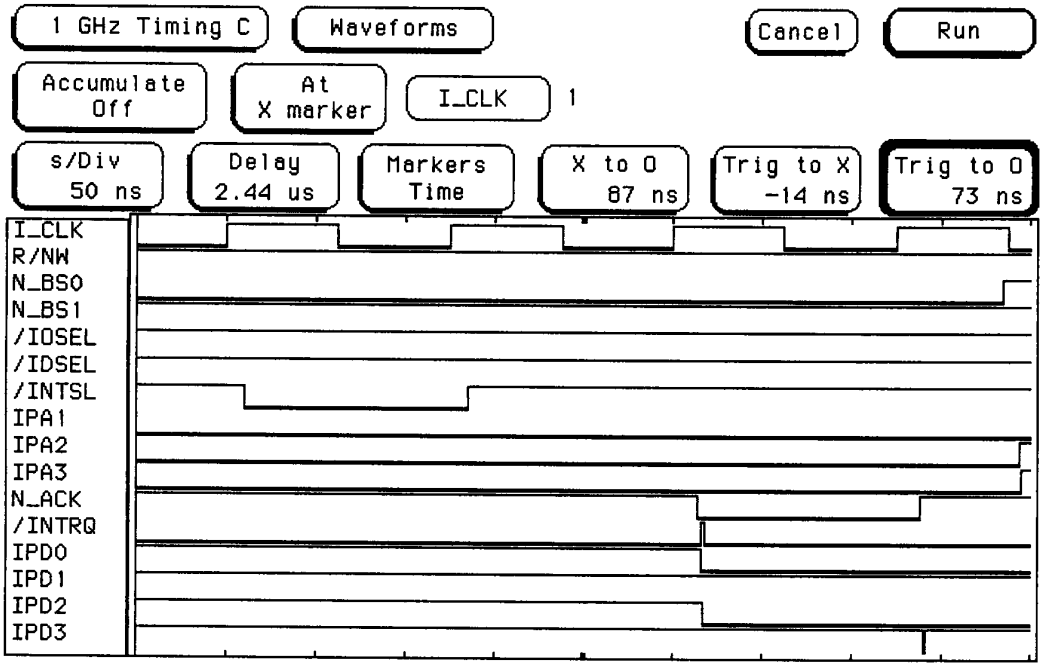


Figure 5-15 IP Module Interrupt-Select Cycle Details

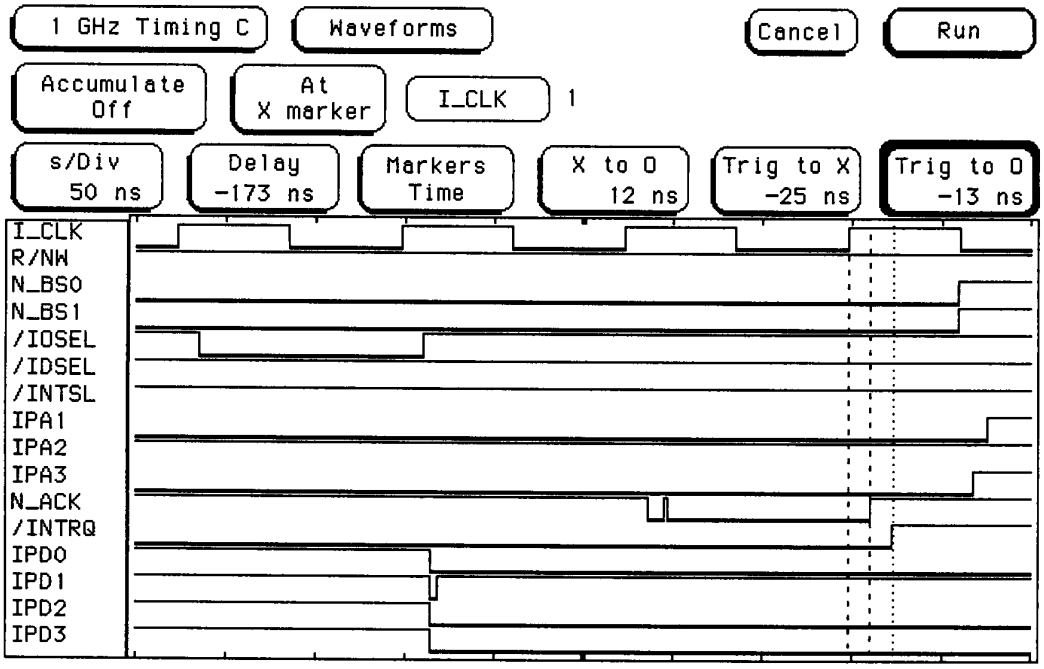


Figure 5-16 I/O Read of IP Module Address 02 *hex* (UART-Interrupt-Information Register) for removal of interrupt Request

This page intentionally left blank

6.0 TYPICAL APPLICATIONS

6.1 Overview

SYSTRAN extends an open invitation to all users to freely submit their applications that might, or do, use the COMM422 IP Module to solve a problem. This section of the manual will be revised periodically to include new application ideas for all users to consider. Help advance the level of technology by participating with the SYSTRAN team, while simultaneously publishing your ideas.

Submission constitutes permission to publish without additional consent or compensation, and SYSTRAN reserves the right to modify submissions to provide for more generic appeal, when necessary.

6.2 Configuration and Interfacing for the COMM422 IP Module

The COMM422 Balanced Quad Communication IP Module is a four-channel, configurable, serial-communication device. It is a singlewide Type I IP Module that combines the functions of RS-422, RS-485 Half Duplex and RS-485 Full Duplex (or a mixture of these) on a single board.

The following two sections cover configuration of the COMM422 for the three interface types and offer suggestions for interfacing to other systems. Section 6.3 gives information on each specific type of interface and how to reconfigure the board to operate in that mode. Section 6.4 covers interfacing of the COMM422 to other systems via cables, transition modules or contact blocks.

6.3 COMM422 Configuration Settings

6.3.1 Balanced-Serial-Port Modes

Figures 6.1, 6.2 and 6.3 show the typical interconnections for RS-422, RS-485 Half Duplex and RS-485 Full Duplex, respectively.

RS-422 allows for one driver to be connected to as many as 10 receivers. However, RS-422 devices are typically used “point-to-point” as is RS-232. Figure 6.1 shows that termination is used only on the receiver. If multiple receivers are used, then only the one furthest from the driver needs to be terminated. Also note that the connections shown in Figure 6-1 represent an RS-422 DTE to DCE connection. For any application involving a DTE to DTE interconnect, a null modem similar to that used in RS-232 will be required. Null-modem connections can be performed on a transition board or with a Phoenix Contact Block (model no. FLKMSO.)

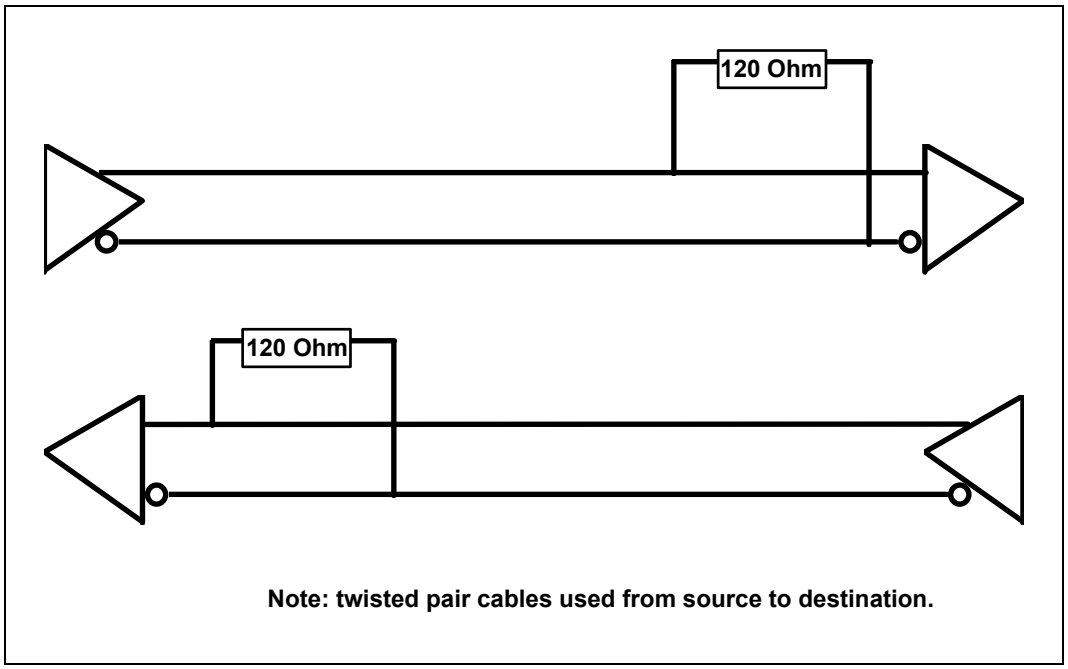


Figure 6-1 Typical RS-422 Driver/Receiver Interconnect

RS-485 Half Duplex is also known as a “two wire” multi-drop serial interface in that up to 32 devices can all be connected together in parallel with only two wires for each driver, receiver, or combination of both. An optional third ground wire can be run between devices as well. In this configuration, termination resistors are used at the extreme ends of the line only. Since the value of 120 Ohms is commonly used at each end, the drivers will see 60 Ohms DC resistance across the line. See figure 6.2 for a typical RS-485 Half Duplex interconnect. Hardware handshaking such as RTS* and CTS* are generally not used for RS-485.

RS-485 Full Duplex is also known as a “four wire” multi-drop serial interface in that up to 32 devices can be connected together with the driver on the master device driving the receiver on all slaves and the drivers on all slave devices signaling the receiver on the master. In this configuration, it is possible to have communication to and from the master device at the same time. Also note that termination will be used on both the receiver and driver at the extreme ends of the interconnect. See figure 6.3 for a typical RS-485 Full Duplex interconnect.

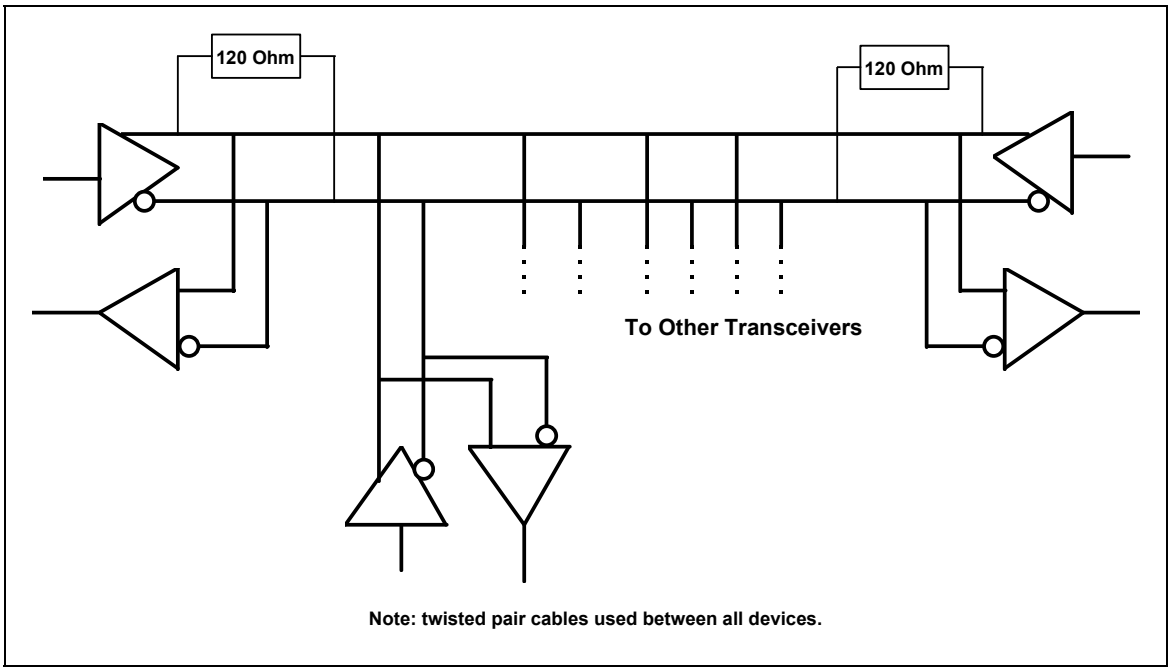


Figure 6-2 Typical RS-485 Half Duplex Device Interconnect

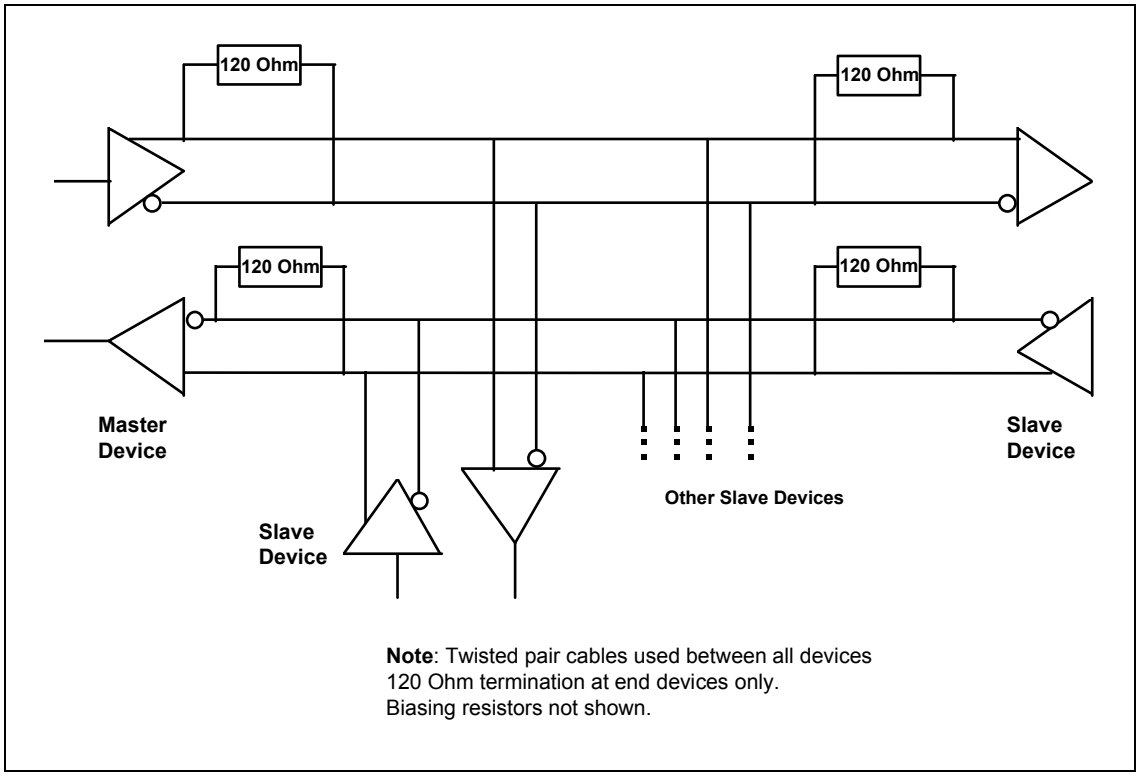


Figure 6-3 Typical RS-485 Full Duplex Device Interconnect

6.3.2 RS-422 Settings

The default (shipping) configuration for the COMM422 is with all four ports set for RS-422 operation. However, at times it may be necessary to reconfigure from another mode. Therefore, all three jumper configurations are shown in Figure 6-5.

Figure 6-4 shows the location of all sixteen jumper headers on the COMM422 used to configure the board to all modes. Figure 6-5 shows that each of the four ports on the COMM422 have four jumper headers assigned to that particular port. Each jumper header has a left slot and a right slot such that two jumpers can be inserted side-by-side. Jumper positions for RS-422 operation are shown in the leftmost diagram in figure 6-5.

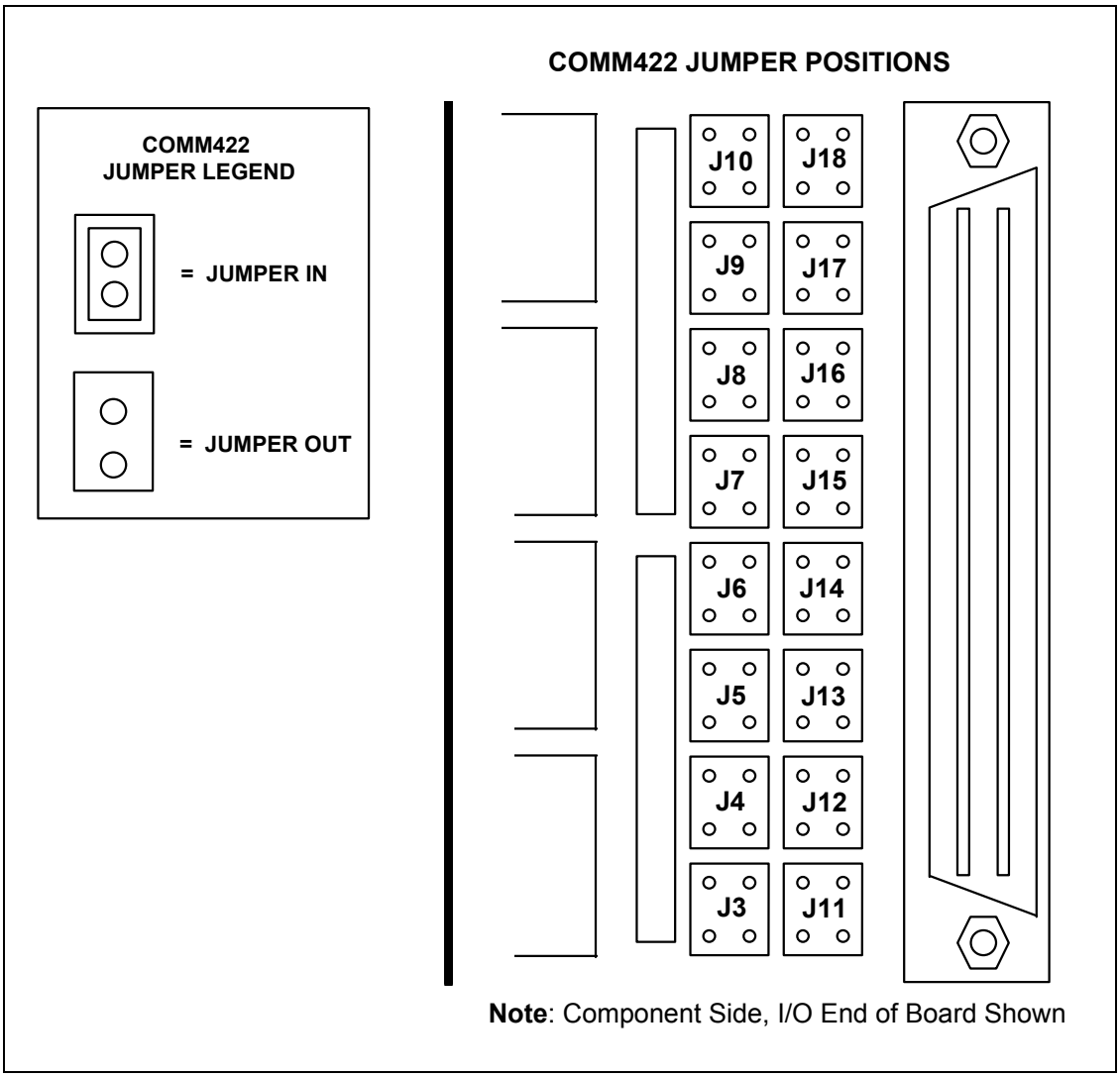


Figure 6-4 Jumper Locations on the COMM422

6.3.3 RS-485 Half Duplex Settings

The jumper positions for RS-485 Half Duplex operation for each port are shown in the upper middle section of Figure 6-5. These settings connect the receiver and driver lines together as a "transceiver" while removing the lines between the receiver and the I/O connector. The receive termination will be "in" for this configuration as well as bias resistors to prevent indeterminate levels if no transmitter on the line is enabled. If a given COMM422 port is to be used in the center of a multi-drop chain whereby the termination is provided on other nodes, then the biasing and termination can be removed by pulling all jumpers shown in headers "C" and "A" for the desired port.

6.3.4 RS-485 Full Duplex Settings

The jumper positions for RS-485 Full Duplex operation for each port are shown in the upper right portion of Figure 6-5. With these settings, both the driver and receiver are attached to the I/O connector in isolated fashion as with RS-422. The only difference between RS-485 Full Duplex and RS-422 settings is that on 485 FDX the driver termination is used such that both jumpers are installed in header "A". If a COMM422 port in this mode is to be connected in the center of a multi-drop chain whereby the termination is provided on other nodes, the termination and bias resistors can be removed by pulling all jumpers in headers "C" and "A" for the desired port.

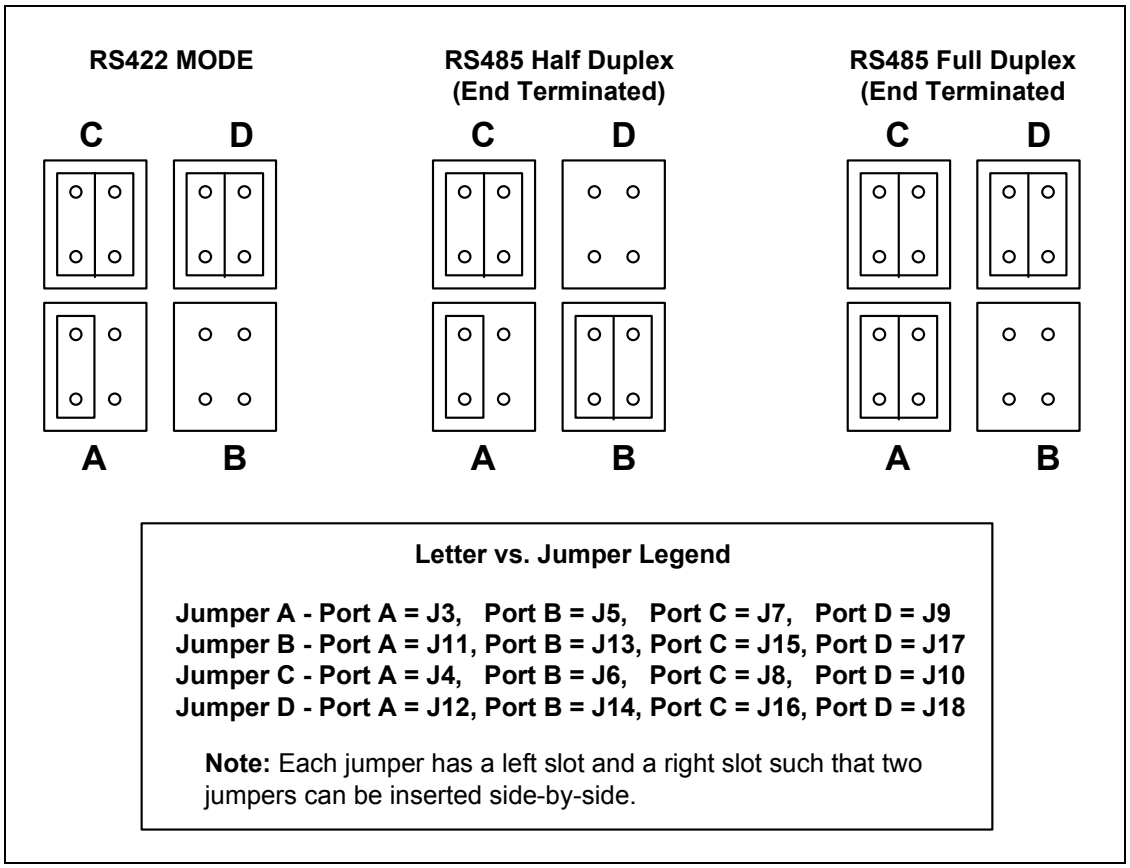


Figure 6-5 Jumper Positions vs. Mode for the COMM422

6.4 Interfacing the COMM422

6.4.1 Common Cable and Connector Types

Figure 6-6 shows two of the most common connectors used for interfacing RS-422 and RS-485 balanced serial devices: DB25 and DB9 female bulkhead connectors with male pins. These connector types can be used directly with the opposite gender versions usually found on RS-422 and 485 interface boards. Also, since ready-made, twisted-pair communication cables are not commonly found, it may be desired to customize an interface cable. This is especially true if the cable is to be run long distances.

For RS-422 applications, computer cable such as Belden #9829 (2 twisted pairs) can be used if only the TX and RX lines are to be connected. Belden #9831 (4 twisted pairs) can be used if RTS and CTS are to be used in addition to TX and RX lines. For RS-485 Half Duplex applications, Belden #9841 (1 twisted pair) can be used while #9842 (2 twisted pairs) can be used for Full Duplex.

6.4.2 Carrier to Device Interconnect

Refer to Figure 6-6 and Table 6-1 for the following discussion.
A variety of ways exist to connect the COMM422 to other common devices using RS-422 or RS-485 balanced serial interfaces. One way is to connect a short 50-pin flat ribbon cable between the IP Carrier board and a Phoenix contact block like the one shown in Figure 6-6. Another way to connect the IP carrier to external devices is to construct a transition board using two 50-pin ribbon cable headers with wire wrap pins and a piece of solderless breadboard large enough for the two connectors. This arrangement also allows for an easy “null modem” interconnect for RS-422. Table 6-1 and Table 3-4 can be referenced when building cables to go from the IP Carrier to other devices.

6.4.3 Receiver Biasing

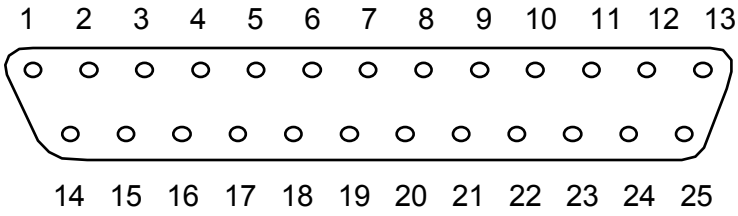
In RS-485 networks, up to 32 devices can be connected together. For this reason, all drivers connected to a common interface must be disabled when not driving useful data. With all drivers “off” and 120 Ohm termination resistors at both ends of the line, the resulting 60 Ohms across the receiver inputs can produce an effect sometimes referred to as “swamping”. This occurs when the internal bias circuits in the receivers become loaded to the point where the bias voltage drops below the level necessary to maintain stability. For this reason the COMM422 has shunt selectable bias resistors on all receivers. In some instances, however, additional biasing may be necessary on a given port. If this is necessary, four pins on the I/O connector provide fused +5 Volt power to be used for additional bias resistors external to the COMM422. These resistors will typically be in the range of 1.0 K Ohms to 1.5 K Ohms and will be connected as follows: one resistor will be connected between a fused power pin and the non-inverting receive line while a second one will be connected between the inverting receive line and ground.



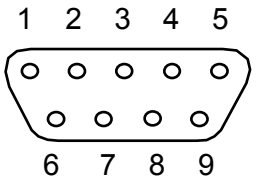
NOTE: The four fused power pins (20, 21, 22 and 23) on the COMM422 I/O connector are meant to be used for biasing only. These pins should not be used as a power supply for any device external to the COMM422.

Table 6-1 Typical Signal Pinouts for DB25 and DB9 Connectors

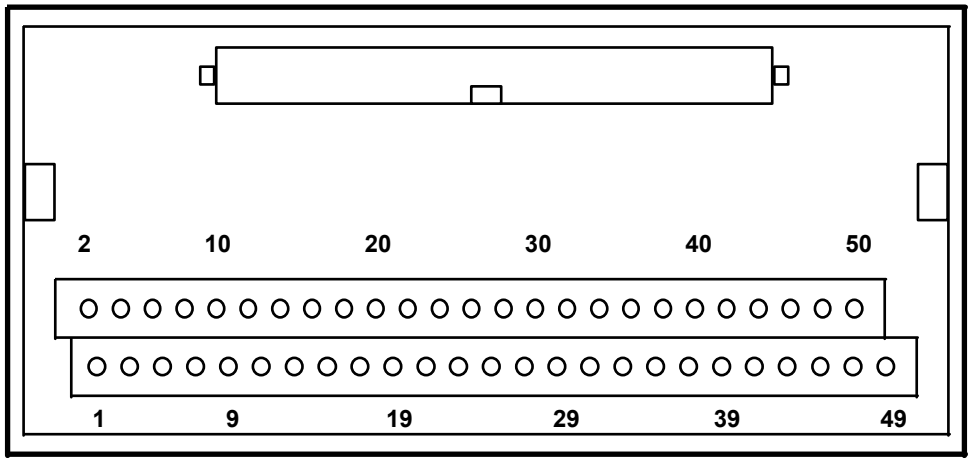
Mode/Signal	DB25 Pin	DB9 Pin
RS-422 TX +	2	4
RS-422 TX -	14	5
RS-422 RX +	3	8
RS-422 RX -	16	9
RS-422 RTS +	4	2
RS-422 RTS -	19	3
RS-422 CTS +	5	7
RS-422 CTS -	13	6
RS-422 Ground	7	1
RS-485 HDX RX/TX +	2	4
RS-485 HDX RX/TX -	14	5
RS-485 HDX Ground	7	1
RS-485 FDX TX +	2	4
RS-485 FDX TX -	14	5
RS-485 FDX RX +	3	8
RS-485 FDX RX -	16	9
RS-485 FDX Ground	7	1



DB25 Female Bulkhead With Male Pins (Looking Into Connector)



DB9 Female Bulkhead With Male Pins (Looking Into Connector)



Phoenix Contact Block (50-pin Ribbon Bulkhead to Contacts)

Figure 6-6 Common Connectors and Phoenix Contact Block

APPENDIX A
SCHEMATICS

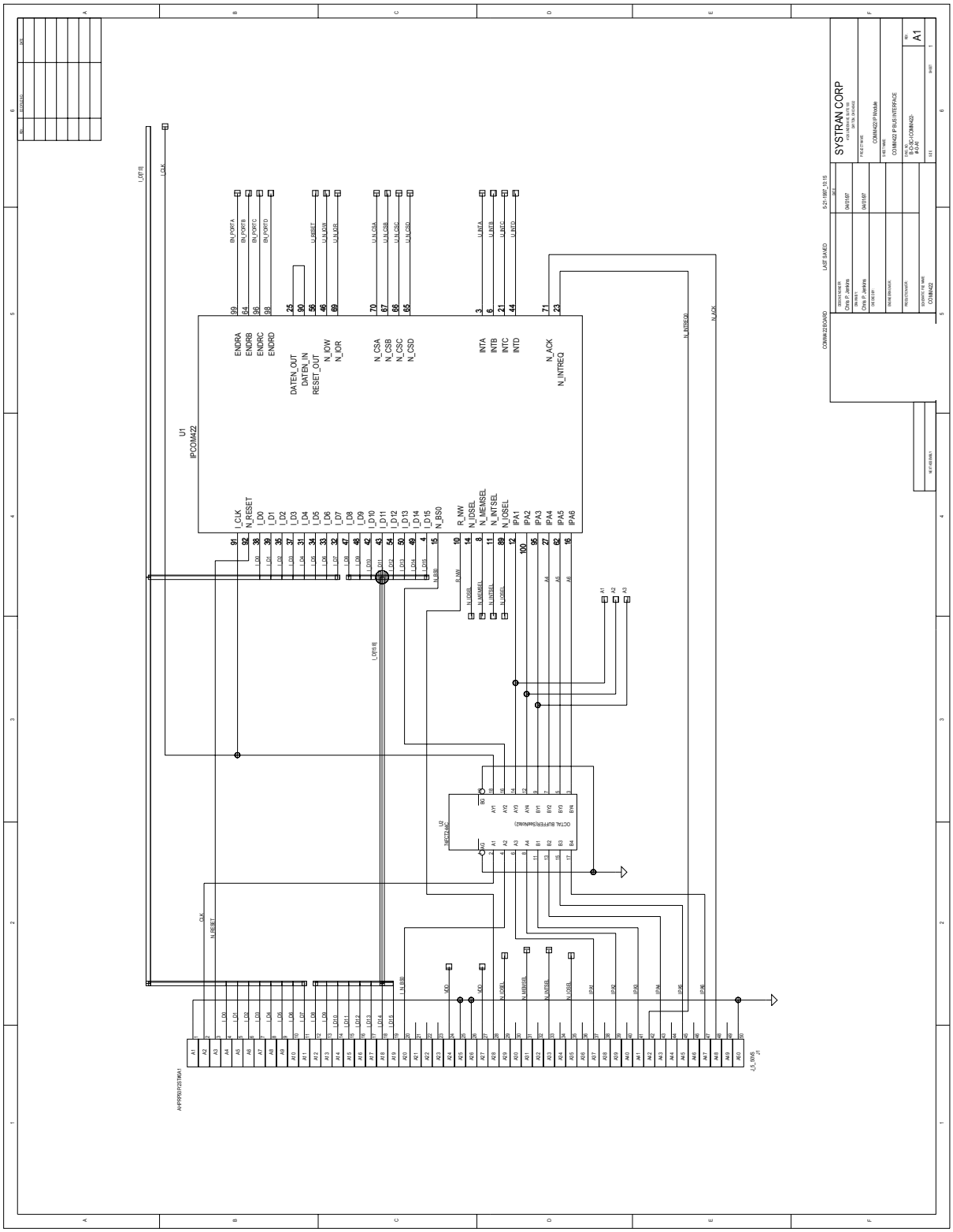


Figure A-1 Schematic #1

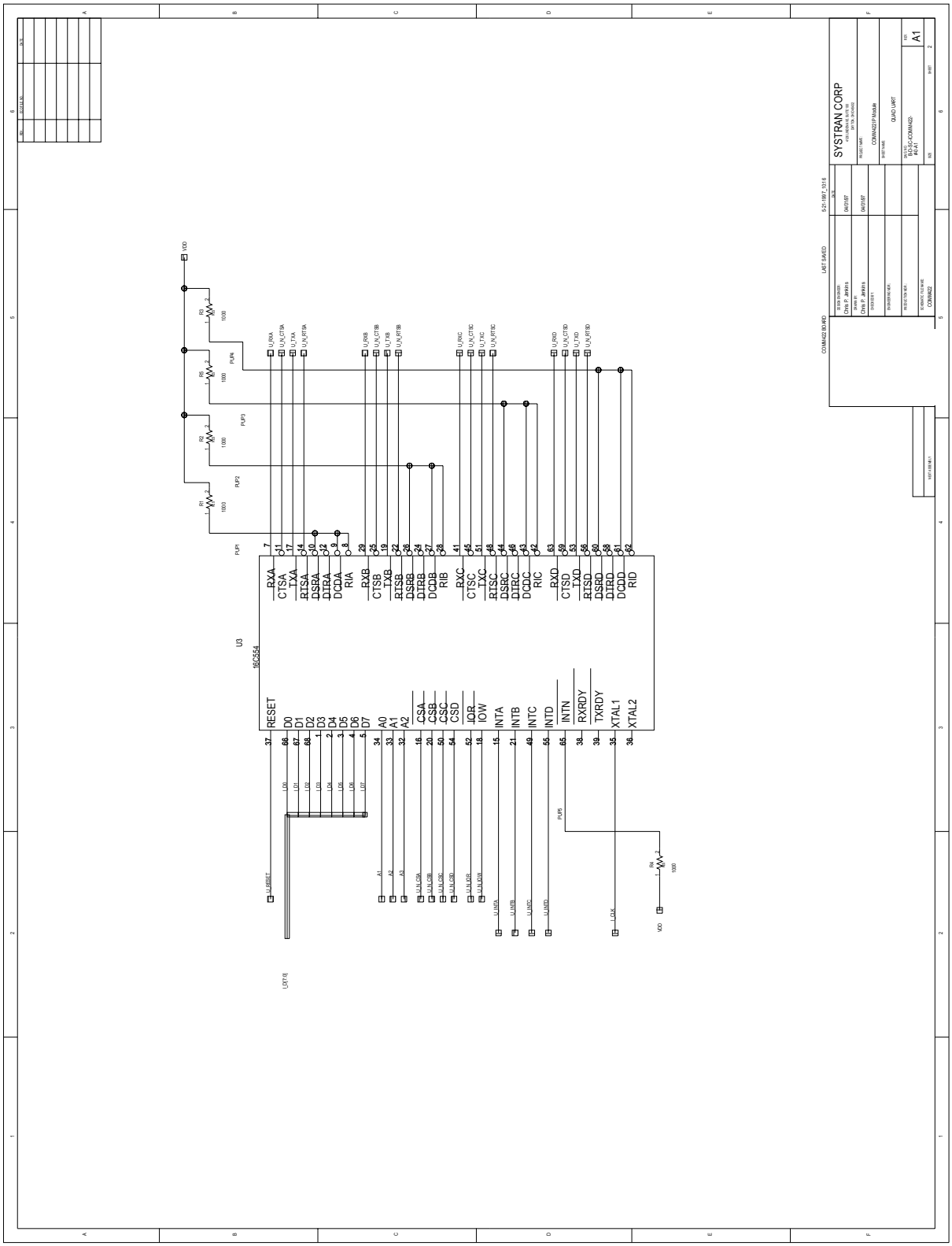


Figure A-2 Schematic #2

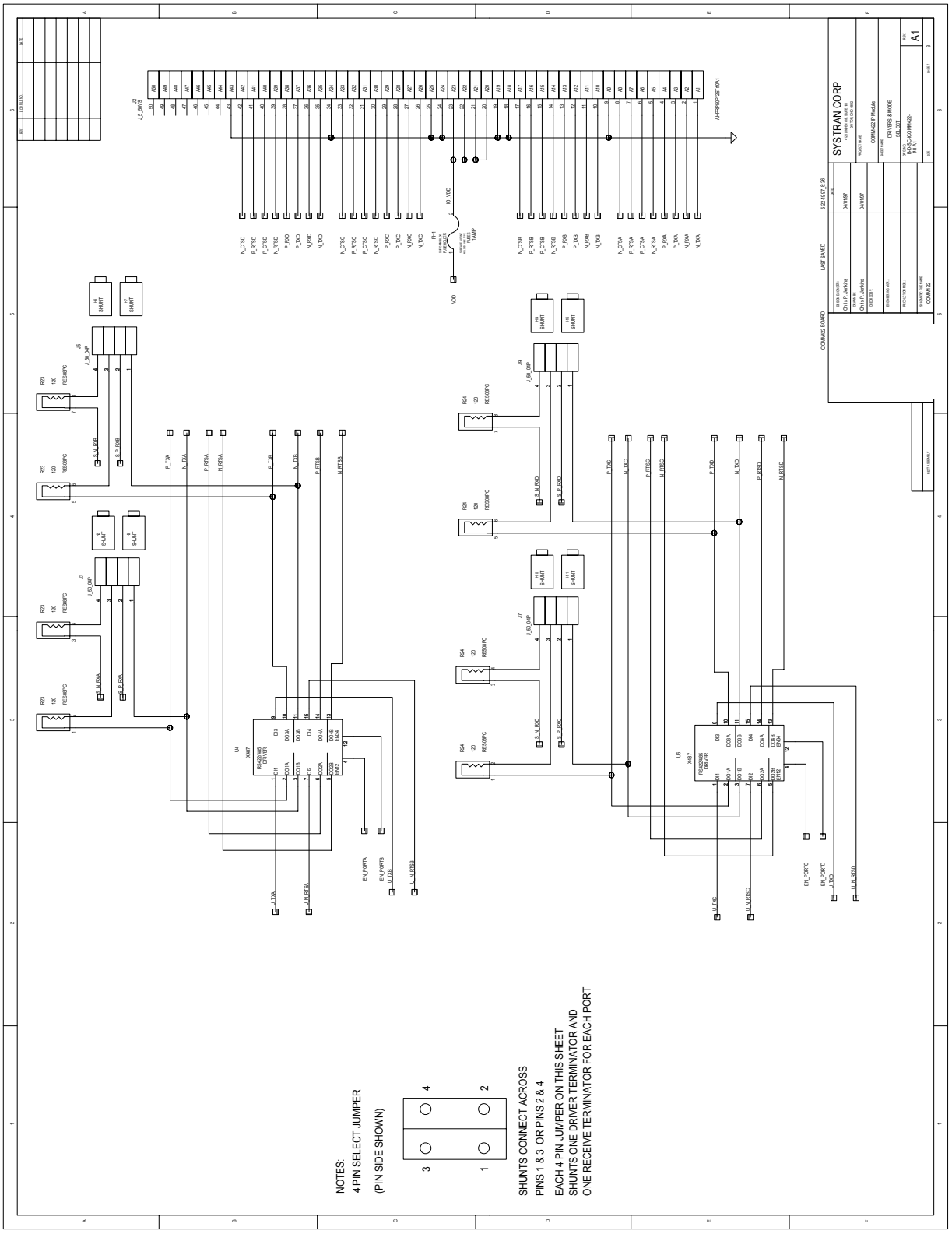


Figure A-3 Schematic #3

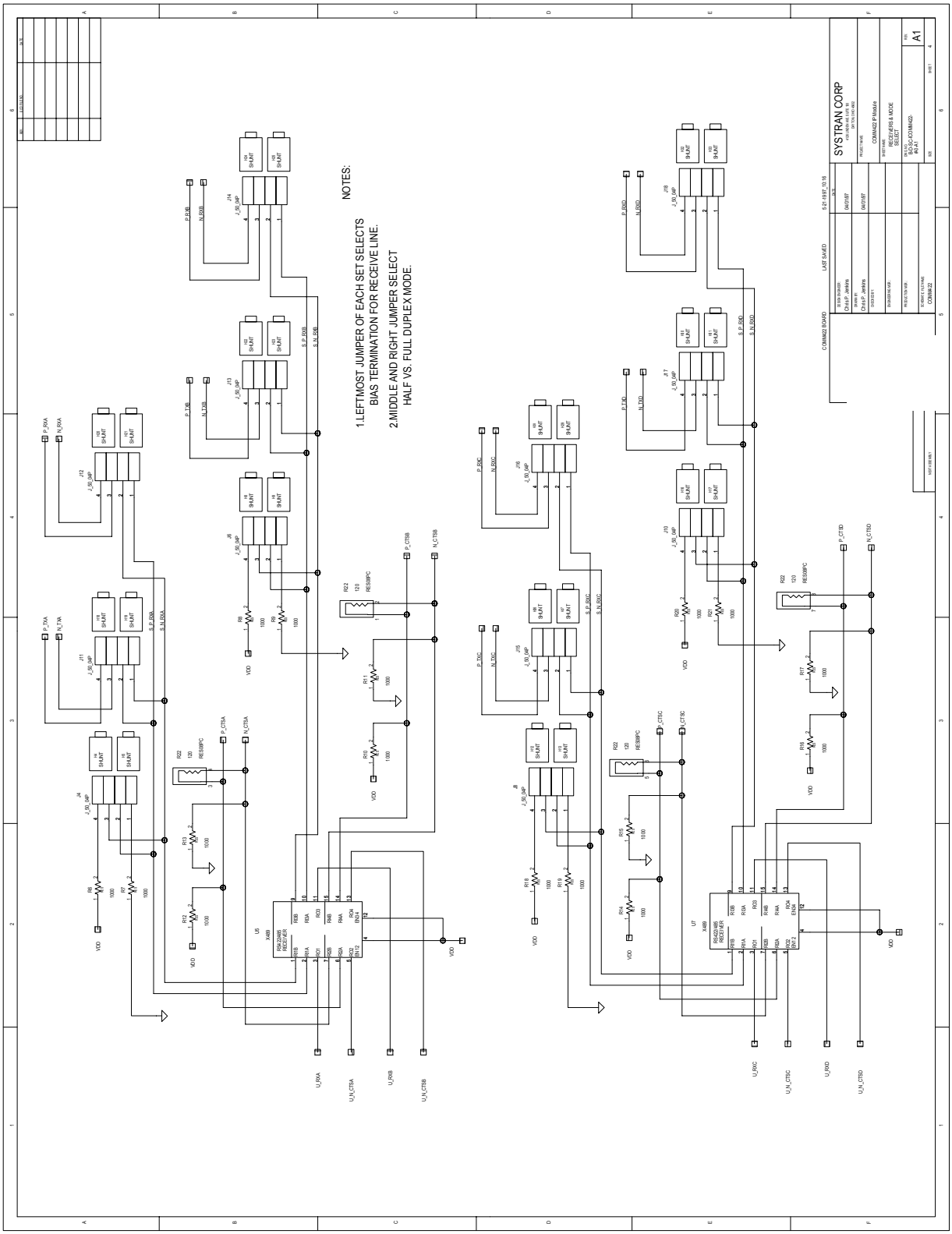


Figure A-4 Schematic #4

APPENDIX B
STARTECH ST16C554D DATA SHEET

STARTECH

An EXAR Company

ST16C554D

Printed August 9, 1996

QUAD ASYNCHRONOUS RECEIVER/TRANSMITTER WITH FIFOs

DESCRIPTION

The ST16C554 is a universal asynchronous receiver and transmitter with 16 byte transmit and receive FIFO. A programmable baud rate generator is provided to select transmit and receive clock rates from 50Hz to 1.5 MHz.

The ST16C554 is an improved version of the NS16C550 UART with higher operating speed and lower access time. The ST16C554 on board status registers provides the error conditions, type and status of the transfer operation being performed. Included is complete MODEM control capability, and a processor interrupt system that may be software tailored to the user's requirements. The ST16C554 provides internal loop-back capability for on board diagnostic testing.

The ST16C554 is fabricated in an advanced 1.2µ CMOS process to achieve low drain power and high speed requirements.

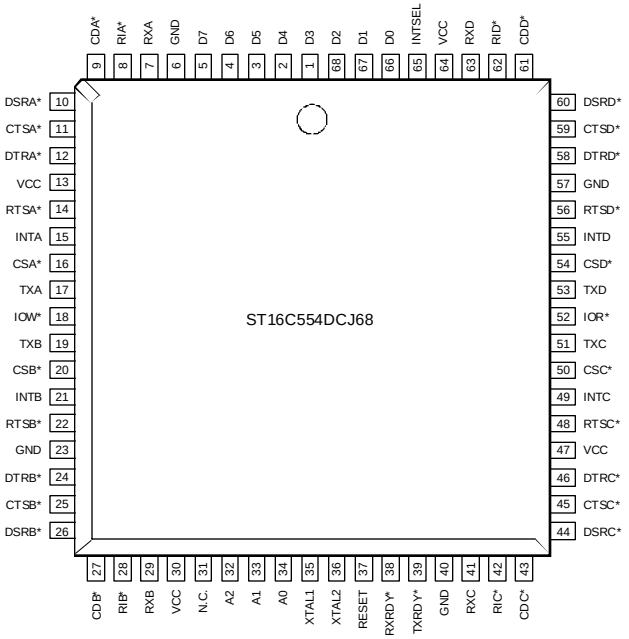
FEATURES

- Pin to pin and functional compatible to ST16C454
- 16 byte transmit FIFO
- 16 byte receive FIFO with error flags
- Modem control signals (CTS*, RTS*, DSR*, DTR*, RI*, CD*)
- Programmable character lengths (5, 6, 7, 8)
- Even, odd, or no parity bit generation and detection
- Status report register
- Independent transmit and receive control
- TTL compatible inputs, outputs
- Software compatible with INS8250, NS16C550
- 460.8 kHz transmit/receive operation with 7.372 MHz crystal or external clock source

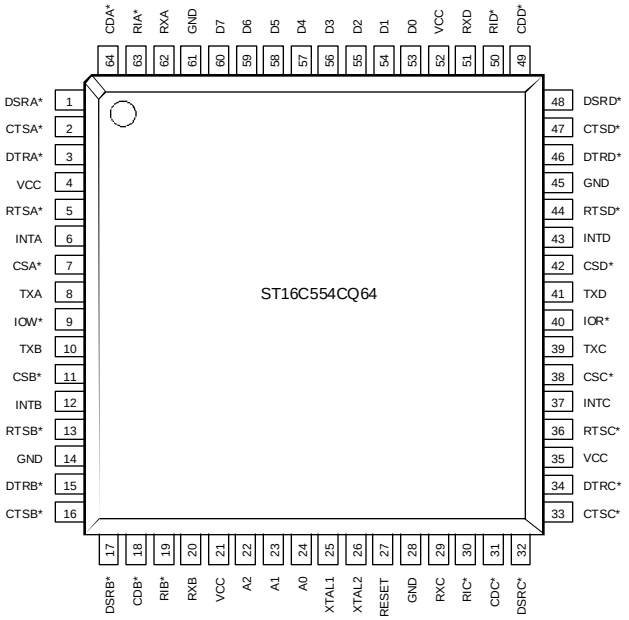
ORDERING INFORMATION

Part number	Package	Operating temperature
ST16C554CQ64	QFP	0° C to + 70° C
ST16C554DCQ64	QFP	0° C to + 70° C
ST16C554DCJ68	PLCC	0° C to + 70° C
ST16C554DIJ68	PLCC	-40° C to + 85° C

PLCC Package



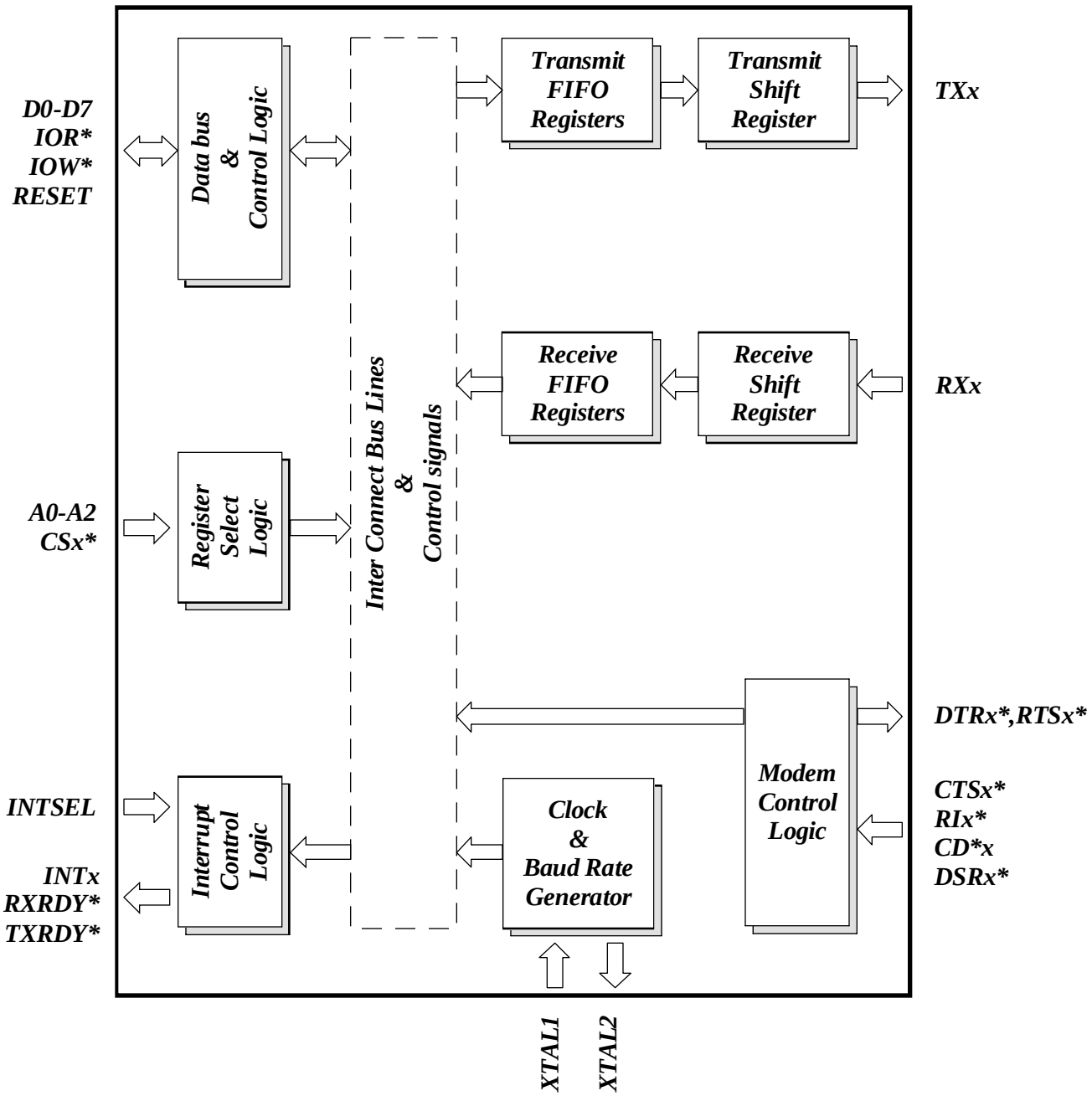
QFP Package



Rev. 2.2

ST16C554D

BLOCK DIAGRAM



ST16C554D

SYMBOL DESCRIPTION

Symbol	Pin	Signal Type	Pin Description
D0-D7	5-66	I/O	Bi-directional data bus. Eight bit, three state data bus to transfer information to or from the CPU. D0 is the least significant bit of the data bus and the first serial data bit to be received or transmitted.
RX A-B RX C-D	7,29 41,63	I	Serial data input. The serial information (data) received from serial port to ST16C554 receive input circuit. A mark (high) is logic one and a space (low) is logic zero. During the local loopback mode the RX input is disabled from external connection and connected to the TX output internally.
TX A-B TX C-D	17,19 51,53	O	Serial data output. The serial data is transmitted via this pin with additional start , stop and parity bits. The TX will be held in mark (high) state during reset, local loopback mode or when the transmitter is disabled.
CS* A-B CS* C-D	16,20 50,54	I	Chip select. (active low) A low at this pin enables the ST16C554 / CPU data transfer operation. Each UART sections of the ST16C554 can be accessed independently.
XTAL1	35	I	Crystal input 1 or external clock input. A crystal can be connected to this pin and XTAL2 pin to utilize the internal oscillator circuit. An external clock can be used to clock internal circuit and baud rate generator for custom transmission rates.
XTAL2	36	O	Crystal input 2 or buffered clock output. See XTAL1.
IOW*	18	I	Write strobe. (active low) A low on this pin will transfer the contents of the CPU data bus to the addressed register.
GND GND	6,23 40,57	O	Signal and power ground.
IOR*	52	I	Read strobe. (active low) A low level on this pin transfers the contents of the ST16C554 data bus to the CPU.

ST16C554D

SYMBOL DESCRIPTION

Symbol	Pin	Signal Type	Pin Description
TXRDY*	39	O	Transmit ready. (active low) This pin goes high when the transmit FIFO of the ST16C554 is full. It can be used as a single or multi-transfer.
A2	32	I	Address select line 2. To select internal registers.
A1	33	I	Address select line 1. To select internal registers.
A0	34	I	Address select line 0. To select internal registers.
RXRDY*	38	O	Receive ready. (active low) This pin goes low when the receive FIFO is full. It can be used as a single or multi-transfer.
INTSEL	65®	I	Interrupt type select. Enable /disable the interrupt three state function. Normal interrupt output can be selected by connecting this pin to VCC (MCR bit-3 does not have any effect on the interrupt output). The three state interrupt output is selected when this pin is left open or connected to GND and MCR bit-3 is set to "1".**Since this pin is not available in 64 pin QFP package, two versions of 64 pin QFP packages are offered. ST16C554DCQ64 with this pin bonded to VCC and ST16C554CQ64 with this pin bonded to GND.**
INT A-B INT C-D	15,21 49,55	O	Interrupt output. (active high) This pin goes high (when enabled by the interrupt enable register) whenever a receiver error, receiver data available, transmitter empty, or modem status condition flag is detected.
RTS* A-B RTS* C-D	14,22 48,56	O	Request to send. (active low) To indicate that the transmitter has data ready to send. Writing a "1" in the modem control register (MCR bit-1) will set this pin to a low state. After the reset this pin will be set to high. Note that this pin does not have any effect on the transmit or receive operation.

ST16C554D

SYMBOL DESCRIPTION

Symbol	Pin	Signal Type	Pin Description
DTR* A-B DTR* C-D	12,24 46,58	O	Data terminal ready. (active low) To indicate that ST16C554 is ready to receive data. This pin can be controlled via the modem control register (MCR bit-0). Writing a "1" at the MCR bit-0 will set the DTR* output to low. This pin will be set to high state after writing a "0" to that register or after the reset . Note that this pin does not have any effect on the transmit or receive operation.
RESET	37	I	Master reset. (active high) A high on this pin will reset all the outputs and internal registers. The transmitter output and the receiver input will be disabled during reset time.
CTS* A-B CTS* C-D	11,25 45,59	I	Clear to send. (active low) The CTS* signal is a MODEM control function input whose conditions can be tested by reading the MSR BIT-4. CTS* has no effect on the transmit or receive operation.
DSR* A-B DSR* C-D	10,26 44,60	I	Data set ready. (active low) A low on this pin indicates the MODEM is ready to exchange data with UART. This pin does not have any effect on the transmit or receive operation.
CD* A-B CD* C-D	9,27 43,61	I	Carrier detect. (active low) A low on this pin indicates the carrier has been detected by the modem.
RI* A-B RI* C-D	8,28 42,62	I	Ring detect indicator. (active low) A low on this pin indicates the modem has received a ringing signal from telephone line.
VCC VCC	13,30 47,64	I	Power supply input.

® Has internal pull-down resistor.

ST16C554D

ST16C554 ACCESSIBLE REGISTERS

A2 A1 A0	Register	BIT-7	BIT-6	BIT-5	BIT-4	BIT-3	BIT-2	BIT-1	BIT-0
0 0 0	RHR	bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0
0 0 0	THR	bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0
0 0 1	IER	0	0	0	0	modem status interrupt	receive line status interrupt	transmit holding register	receive holding register
0 1 0	FCR	RCVR trigger (MSB)	RCVR trigger (LSB)	0	0	DMA mode select	XMIT FIFO reset	RCVR FIFO reset	FIFO enable
0 1 0	ISR	0/ FIFOs enabled	0/ FIFOs enabled	0	0	int priority bit-2	int priority bit-1	int priority bit-0	int status
0 1 1	LCR	divisor latch enable	set break	set parity	even parity	parity enable	stop bits	word length bit-1	word length bit-0
1 0 0	MCR	0	0	0	loop back	INT enable	Not used	RTS*	DTR*
1 0 1	LSR	0/ FIFO error	trans. empty	trans. holding empty	break interrupt	framing error	parity error	overrun error	receive data ready
1 1 0	MSR	CD	RI	DSR	CTS	delta CD*	delta RI*	delta DSR*	delta CTS*
1 1 1	SPR	bit-7	bit-6	bit-5	bit-4	bit-3	bit-2	bit-1	bit-0
0 0 0	***DLL***	***bit-7***	***bit-6***	***bit-5***	***bit-4***	***bit-3***	***bit-2***	***bit-1***	***bit-0***
0 0 1	***DLM***	***bit-15***	***bit-14***	***bit-13***	***bit-12***	***bit-11***	***bit-10***	***bit-9***	***bit-8***

DLL and DLM are accessible only when LCR bit-7 is set to “1”.

ST16C554D

PROGRAMMING TABLE

A2	A1	A0	READ MODE	WRITE MODE
0	0	0	Receive Holding Register	Transmit Holding Register
0	0	1		Interrupt Enable Register
0	1	0	Interrupt Status Register	FIFO Control Register
0	1	1		Line Control Register
1	0	0		Modem Control Register
1	0	1	Line Status Register	
1	1	0	Modem Status Register	
1	1	1	Scratchpad Register	Scratchpad Register
0	0	0		LSB of Divisor Latch
0	0	1		MSB of Divisor Latch

REGISTER FUNCTIONAL DESCRIPTIONS

TRANSMIT AND RECEIVE HOLDING REGISTER

The serial transmitter section consists of a Transmit Hold Register (THR) and Transmit Shift Register (TSR). The status of the transmit hold register is provided in the Line Status Register (LSR). Writing to this register (THR) will transfer the contents of data bus (D7-D0) to the Transmit holding register whenever the transmitter holding register or transmitter shift register is empty. The transmit holding register empty flag will be set to “1” when the transmitter is empty or data is transferred to the transmit shift register. Note that a write operation should be performed when the transmit holding register empty flag is set.

On the falling edge of the start bit, the receiver internal counter will start to count 7 1/2 clocks (16x clock) which is the center of the start bit. The start bit is valid if the RX is still low at the mid-bit sample of the start bit. Verifying the start bit prevents the receiver from assembling a false data character due to a low going noise spike on the RX input. Receiver status codes will be posted in the Line Status Register.

FIFO INTERRUPT MODE OPERATION

When the receive FIFO (FCR BIT-0=1) and receive interrupts (IER BIT-0=1) are enabled, receiver interrupt will occur as follows:

A) The receive data available interrupts will be issued to the CPU when the FIFO has reached its programmed trigger level; it will be cleared as soon as the FIFO drops below its programmed trigger level.

B) The ISR receive data available indication also occurs when the FIFO trigger level is reached, and like the interrupt it is cleared when the FIFO drops below the trigger level.

C) The data ready bit (LSR BIT-0) is set as soon as a character is transferred from the shift register to the receiver FIFO. It is reset when the FIFO is empty.

FIFO POLLED MODE OPERATION

When FCR BIT-0=1; resetting IER BIT 3-0 to zero puts the ST16C554 in the FIFO polled mode of operation. Since the receiver and transmitter are controlled separately either one or both can be in the polled mode operation by utilizing the Line Status Register.

ST16C554D

A) LSR BIT-0 will be set as long as there is one byte in the receive FIFO.

B) LSR BIT4-1 will specify which error(s) has occurred.

C) LSR BIT-5 will indicate when the transmit FIFO is empty.

D) LSR BIT-6 will indicate when both transmit FIFO and transmit shift register are empty.

E) LSR BIT-7 will indicate when there are any errors in the receive FIFO.

The ST16C554 requires to have two step FIFO enable operation in order to enable receive trigger levels.

PROGRAMMABLE BAUD RATE GENERATOR

The ST16C554 contains a programmable Baud Rate Generator that is capable of taking any clock input from DC-24 MHz and dividing it by any divisor from 1 to 2^{16} -1. The output frequency of the Baudout* is equal to 16X of transmission baud rate (Baudout*=16 x Baud Rate). Customize Baud Rates can be achieved by selecting proper divisor values for MSB and LSB of baud rate generator.

INTERRUPT ENABLE REGISTER (IER)

The Interrupt Enable Register (IER) masks the incoming interrupts from receiver ready, transmitter empty, line status and modem status registers to the INT output pin.

IER BIT-0:
0=disable the receiver ready interrupt.
1=enable the receiver ready interrupt.

IER BIT-1:
0=disable the transmitter empty interrupt.
1=enable the transmitter empty interrupt.

IER BIT-2:
0=disable the receiver line status interrupt.
1=enable the receiver line status interrupt.

IER BIT-3:
0=disable the modem status register interrupt.
1=enable the modem status register interrupt.

IER BIT 7-4:
All these bits are set to logic zero.

INTERRUPT STATUS REGISTER (ISR)

The ST16C554 provides four level prioritized interrupt conditions to minimize software overhead during data character transfers. The Interrupt Status Register (ISR) provides the source of the interrupt in prioritized matter. During the read cycle the ST16C554 provides the highest interrupt level to be serviced by CPU. No other interrupts are acknowledged until the particular interrupt is serviced. The following are the prioritized interrupt levels:

Priority level

P	D3	D2	D1	D0	Source of the interrupt
1	0	1	1	0	LSR (Receiver Line Status Register)
2	0	1	0	0	RXRDY (Received Data Ready)
2*	1	1	0	0	RXRDY (Receive Data time out)
3	0	0	1	0	TXRDY (Transmitter Holding Register Empty)
4	0	0	0	0	MSR (Modem Status Register)

***RECEIVE TIME-OUT:**
This mode is enabled when STARTECH UART is operating in FIFO mode. Receive time out will not occur if the receive FIFO is empty. The time out counter will be reset at the center of each stop bit received or each time receive holding register is read. The actual time out value is T (**T**ime out length in bits)= 4 X P (**P**rogrammed word length) + 12. To convert time out value to a character value, user has to divide this number to its complete word length + parity (if used) + number of stop bits and start bit.

Example -A: If user programs the word length = 7, and no parity and one stop bit, Time out will be:
T = 4 X 7(programmed word length) +12 = 40 bits
Character time = 40 / 9 [(programmed word length = 7) + (stop bit = 1) + (start bit = 1)] = 4.4 characters.

Example -B: If user programs the word length = 7, with parity and one stop bit, the time out will be:
T = 4 X 7(programmed word length) + 12 = 40 bits
Character time = 40 / 10 [(programmed word length = 7) + (parity = 1) + (stop bit = 1) + (start bit = 1) = 4 characters.

ISR BIT-0:
0=an interrupt is pending and the ISR contents may be used as a pointer to the appropriate interrupt service routine.
1=no interrupt pending.

ISR BIT 1-3:
Logical combination of these bits, provides the highest priority interrupt pending.

ISR BIT 4-7:
These bits are not used and are set to zero in ST16C450 mode. **BIT 6-7:** are set to "1" in ST16C554 mode.

FIFO CONTROL REGISTER (FCR)

This register is used to enable the FIFOs, clear the FIFOs, set the receiver FIFO trigger level, and select the type of DMA signaling.

FCR BIT-0:
0=Disable the transmit and receive FIFO.
1=Enable the transmit and receive FIFO.
This bit should be enabled before setting the FIFO trigger levels.

FCR BIT-1:
0=No change.
1=Clears the contents of the receive FIFO and resets its counter logic to 0 (the receive shift register is not cleared or altered). This bit will return to zero after clearing the FIFOs.

FCR BIT-2:
0=No change.
1=Clears the contents of the transmit FIFO and resets its counter logic to 0 (the transmit shift register is not cleared or altered). This bit will return to zero after clearing the FIFOs.

FCR BIT-3:
0=No change.
1=Changes RXRDY and TXRDY pins from mode "0" to mode "1".

Transmit operation in mode "0":
When ST16C554 is in ST16C450 mode (FCR bit-0=0) or in the FIFO mode (FCR bit-0=1, FCR bit-3=0) when there are no characters in the transmit FIFO or transmit holding register, the TXRDY* pin will go low. Once active the TXRDY* pin will go high (inactive) after the first character is loaded into the transmit holding register.

Receive operation in mode "0":
When ST16C554 is in ST16C450 mode (FCR bit-0=0) or in the FIFO mode (FCR bit-0=1, FCR bit-3=0) and there is at least 1 character in the receive FIFO, the RXRDY* pin will go low. Once active the RXRDY* pin will go high (inactive) when there are no more characters in the receiver.

Transmit operation in mode "1":
When ST16C554 is in ST16C550 mode (FCR bit-0=1, FCR bit-3=1) the TXRDY* pin will become high (inactive) when the transmit FIFO is completely full. It will be low if one or more FIFO locations are empty.

Receive operation in mode "1":
When ST16C554 is in ST16C550 mode (FCR bit-0=1, FCR bit-3=1) and the trigger level or the timeout has been reached, the RXRDY* pin will go low. Once it is activated it will go high (inactive) when there are no more characters in the FIFO.

FCR BIT 4-5:
Not used.

FCR BIT 6-7:
These bits are used to set the trigger level for the receiver FIFO interrupt.

ST16C554D

BIT-7	BIT-6	FIFO trigger level
0	0	01
0	1	04
1	0	08
1	1	14

LINE CONTROL REGISTER (LCR)

The Line Control Register is used to specify the asynchronous data communication format. The number of the word length, stop bits, and parity can be selected by writing appropriate bits in this register.

LCR BIT1-0:
These two bits specify the word length to be transmitted or received.

BIT-1	BIT-0	Word length
0	0	5
0	1	6
1	0	7
1	1	8

LCR BIT-2:
The number of stop bits can be specified by this bit.

BIT-2	Word length	Stop bit(s)
0	5,6,7,8	1
1	5	1-1/2
1	6,7,8	2

LCR BIT-3:
Parity or no parity can be selected via this bit.
0=no parity
1=a parity bit is generated during the transmission, receiver also checks for received parity.

LCR BIT-4:
If the parity bit is enabled, LCR BIT-4 selects the even or odd parity format.
0=ODD parity is generated by forcing an odd number of 1's in the transmitted data, receiver also checks for same format.
1= EVEN parity bit is generated by forcing an even the number of 1's in the transmitted data, receiver also checks for same format.

LCR BIT-5:
If the parity bit is enabled, LCR BIT-5 selects the forced parity format.
LCR BIT-5=1 and LCR BIT-4=0, parity bit is forced to "1" in the transmitted and received data.
LCR BIT-5=1 and LCR BIT-4=1, parity bit is forced to "0" in the transmitted and received data.

LCR BIT-6:
Break control bit. It causes a break condition to be transmitted (the TX is forced to low state).
0=normal operating condition.
1=forces the transmitter output (TX) to go low to alert the communication terminal.

LCR BIT-7:
The internal baud rate counter latch enable (DLAB).
0=normal operation.
1=select divisor latch register.

MODEM CONTROL REGISTER (MCR)

This register controls the interface with the MODEM or a peripheral device (RS232).

MCR BIT-0:
0=force DTR* output to high.
1=force DTR* output to low.

MCR BIT-1:
0=force RTS* output to high.
1=force RTS* output to low.

MCR BIT-2:
Not used, except in internal loop-back mode.

MCR BIT-3:
0=set the INT A-D output pin to three state mode..
1=Enable the INT A-D output pin.

ST16C554D

MCR BIT-4:
0=normal operating mode.
1=enable local loop-back mode (diagnostics). The transmitter output (TX) is set high (Mark condition), the receiver input (RX) , CTS*, DSR*, CD*, and RI* are disabled. Internally the transmitter output is connected to the receiver input and DTR*, RTS*, MCR bit-2 and INT enable are connected to modem control inputs.
In this mode , the receiver and transmitter interrupts are fully operational. The Modem Control Interrupts are also operational, but the interrupts sources are now the lower four bits of the Modem Control Register instead of the four Modem Control inputs. The interrupts are still controlled by the IER .

MCR BIT 5-7:
Not used. Are set to zero permanently.

LINE STATUS REGISTER (LSR)

This register provides the status of data transfer to CPU.

LSR BIT-0:
0=no data in receive holding register or FIFO.
1=data has been received and saved in the receive holding register or FIFO.

LSR BIT-1:
0=no overrun error (normal).
1=overrun error, next character arrived before receive holding register was emptied or if FIFOs are enabled, an overrun error will occur only after the FIFO is full and the next character has been completely received in the shift register. Note that character in the shift register is overwritten, but it is not transferred to the FIFO.

LSR BIT-2:
0=no parity error (normal).
1=parity error, received data does not have correct parity information. In the FIFO mode this error is associated with the character at the top of the FIFO.

LSR BIT-3:
0=no framing error (normal).
1=framing error received, received data did not have a valid stop bit. In the FIFO mode this error is associated with the character at the top of the FIFO.

LSR BIT-4:
0=no break condition (normal).
1=receiver received a break signal (RX was low for one character time frame). In FIFO mode, only one zero character is loaded into the FIFO.

LSR BIT-5:
0=transmit holding register is full. ST16C554 will not accept any data for transmission.
1=transmit holding register (or FIFO) is empty. CPU can load the next character.

LSR BIT-6:
0=transmitter holding and shift registers are full.
1=transmitter holding and shift registers are empty. In FIFO mode this bit is set to one whenever the transmitter FIFO and transmit shift register are empty.

LSR BIT-7:
0=Normal.
1=At least one parity error, framing error or break indication in the FIFO. This bit is cleared when LSR is read.

MODEM STATUS REGISTER (MSR)

This register provides the current state of the control lines from the modem or peripheral to the CPU. Four bits of this register are used to indicate the changed information. These bits are set to “1” whenever a control input from the MODEM changes state. They are set to “0” whenever the CPU reads this register

MSR BIT-0:
Indicates that the CTS* input to the ST16C554 has changed state since the last time it was read.

MSR BIT-1:
Indicates that the DSR* input to the ST16C554 has changed state since the last time it was read.

MSR BIT-2:
Indicates that the RI* input to the ST16C554 has changed from a low to a high state.

ST16C554D

MSR BIT-3:
Indicates that the CD* input to the ST16C554 has changed state since the last time it was read.

MSR BIT-4:
This bit is equivalent to RTS in the MCR during local loop-back mode. It is the compliment of the CTS* input.

MSR BIT-5:
This bit is equivalent to DTR in the MCR during local loop-back mode. It is the compliment of the DSR* input.

MSR BIT-6:
This bit is equivalent to MCR bit-2 during local loop-back mode. It is the compliment of the RI* input.

MSR BIT-7:
This bit is equivalent to INT enable in the MCR during local loop-back mode. It is the compliment to the CD* input.

Note: Whenever MSR BIT3-0: is set to logic “1”, a MODEM Status Interrupt is generated.

SCRATCHPAD REGISTER (SR)

ST16C554 provides a temporary data register to store 8 bits of information for variable use.

BAUD RATE GENERATOR PROGRAMMING TABLE (1.8432 MHz CLOCK):

BAUD RATE	16 x CLOCK DIVISOR	% ERROR
50	2304	
110	1047	0.026
150	768	
300	384	
600	192	
1200	96	
2400	48	
4800	24	
7200	16	
9600	12	
19.2K	6	
38.4K	3	
56K	2	2.77
115.2K	1	

ST16C554 EXTERNAL RESET CONDITION

REGISTERS	RESET STATE
IER	IER BITS 0-7=0
ISR	ISR BIT-0=1, ISR BITS 1-7=0
LCR	LCR BITS 0-7=0
MCR	MCR BITS 0-7=0
LSR	LSR BITS 0-4=0, LSR BITS 5-6=1 LSR, BIT 7=0
MSR	MSR BITS 0-3=0, MSR BITS 4-7=input signals
FCR	FCR BITS 0-7=0

SIGNALS	RESET STATE
TX A-D	High
RTS* A-D	High
DTR* A-D	High
RXRDY*	High
TXRDY*	Low
INT A-D	Three state mode

ST16C554D

AC ELECTRICAL CHARACTERISTICS

T_A=0° - 70° C, Vcc=3.3 - 5.0 V ± 10% unless otherwise specified.

Symbol	Parameter	Limits Min	Typ	Max	Units	Conditions
T_1	Clock high pulse duration	20			ns	
T_2	Clock low pulse duration	20			ns	
T_3	Clock rise/fall time			10	ns	
T_8	Chip select setup time	0			ns	
T_9	Chip select hold time	0			ns	
T_{12}	Data setup time	15			ns	
T_{13}	Data hold time	15			ns	
T_{14}	IOW* delay from chip select	10			ns	
T_{15}	IOW* strobe width	50			ns	
T_{16}	Chip select hold time from IOW*	0			ns	
T_{17}	Write cycle delay	55			ns	
Tw	Write cycle=T_{15}+T_{17}	105			ns	
T_{19}	Data hold time	15		25	ns	
T_{21}	IOR* delay from chip select	10			ns	
T_{23}	IOR* strobe width	65			ns	
T_{24}	Chip select hold time from IOR*	0			ns	
T_{25}	Read cycle delay	55			ns	
Tr	Read cycle=T_{23}+T_{25}	115			ns	
T_{26}	Delay from IOR* to data			35	ns	100 pF load
T_{28}	Delay from IOW* to output			50	ns	100 pF load
T_{29}	Delay to set interrupt from MODEM input			70	ns	100 pF load
T_{30}	Delay to reset interrupt from IOR*			70	ns	100 pF load
T_{31}	Delay from stop to set interrupt			1_{Rclk}	ns	100 pF load
T_{32}	Delay from IOR* to reset interrupt			200	ns	100 pF load
T_{33}	Delay from initial INT reset to transmit start	8		24	*	
T_{34}	Delay from stop to interrupt			100	ns	
T_{35}	Delay from IOW* to reset interrupt			175	ns	
T_{44}	Delay from stop to set RxRdy			1_{RCLK}		
T_{45}	Delay from IOR* to reset RxRdy			100	ns	
T_{46}	Delay from IOW* to set TxRdy			195	ns	
T_{47}	Delay from start to reset TxRdy			8	*	
T_R	Reset pulse width	10			ns	
N	Baud rate devisor	1		2^{16}-1		

Note 1: * = Baudout* cycle

ST16C554D

ABSOLUTE MAXIMUM RATINGS

Supply range	7 Volts
Voltage at any pin	GND-0.3 V to VCC+0.3 V
Operating temperature	0° C to +70° C
Storage temperature	-40° C to +150° C
Package dissipation	500 mW

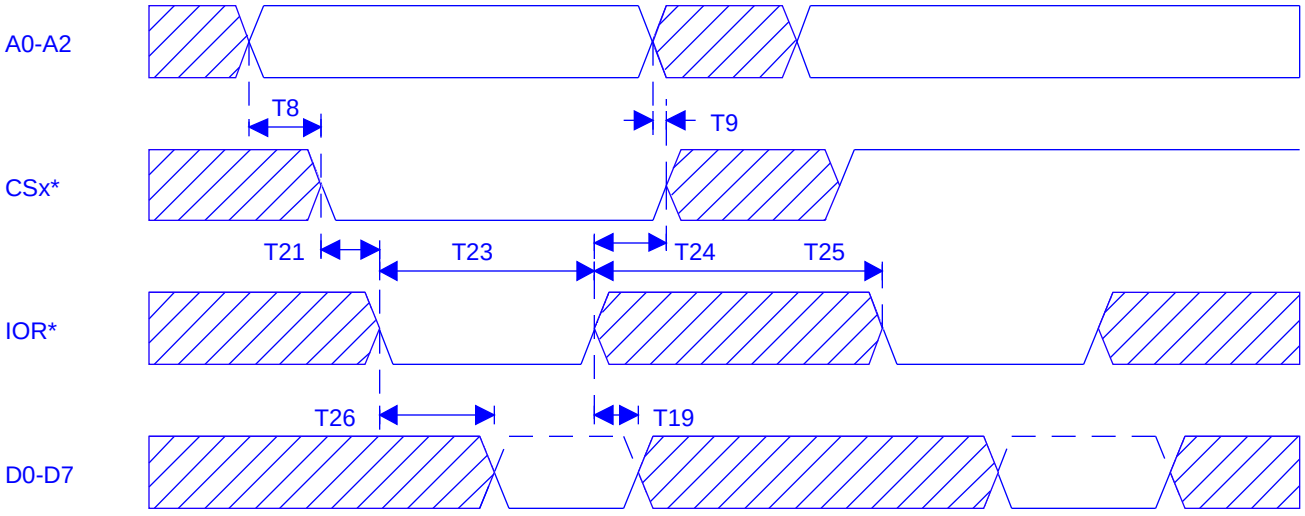
DC ELECTRICAL CHARACTERISTICS

T_A=0° - 70° C, Vcc=3.3 - 5.0 V ± 10% unless otherwise specified.

Symbol	Parameter	Limits Min	Limits Typ	Limits Max	Units	Conditions
V_{ILCK}	Clock input low level	-0.5		0.6	V	
V_{IHCK}	Clock input high level	3.0		VCC	V	
V_{IL}	Input low level	-0.5		0.8	V	
V_{IH}	Input high level	2.2		VCC	V	
V_{OL}	Output low level on all outputs			0.4	V	I_{OL}= 6 mA
V_{OH}	Output high level	2.4			V	I_{OH}= -6 mA
I_{CC}	Avg power supply current		10	14	mA	
I_{IL}	Input leakage			±10	μA	
I_{CL}	Clock leakage			±10	μA	
V_{ILCK}	Clock input low level	-0.3		0.8	V	Vcc=3.0 V
V_{IHCK}	Clock input high level	2.4		VCC	V	Vcc=3.0 V
V_{IL}	Input low level	-0.3		0.8	V	Vcc=3.0 V
V_{IH}	Input high level	2.0		VCC	V	Vcc=3.0 V
V_{OL}	Output low level on all outputs			0.4	V	Vcc=3.0 V, I_{OL}= 8.5 mA
V_{OH}	Output high level	2.0			V	Vcc=3.0 V, I_{OH}= -4 mA
I_{CC}	Avg power supply current		8	12	mA	Vcc=3.0 V

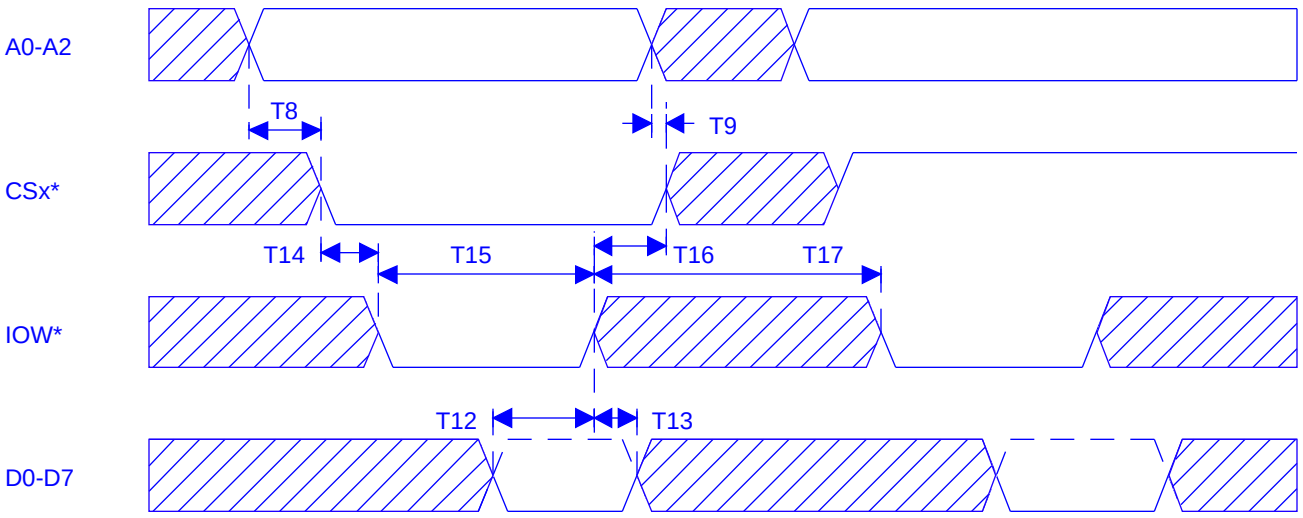
ST16C554D

GENERAL READ TIMING



162450-RD-1

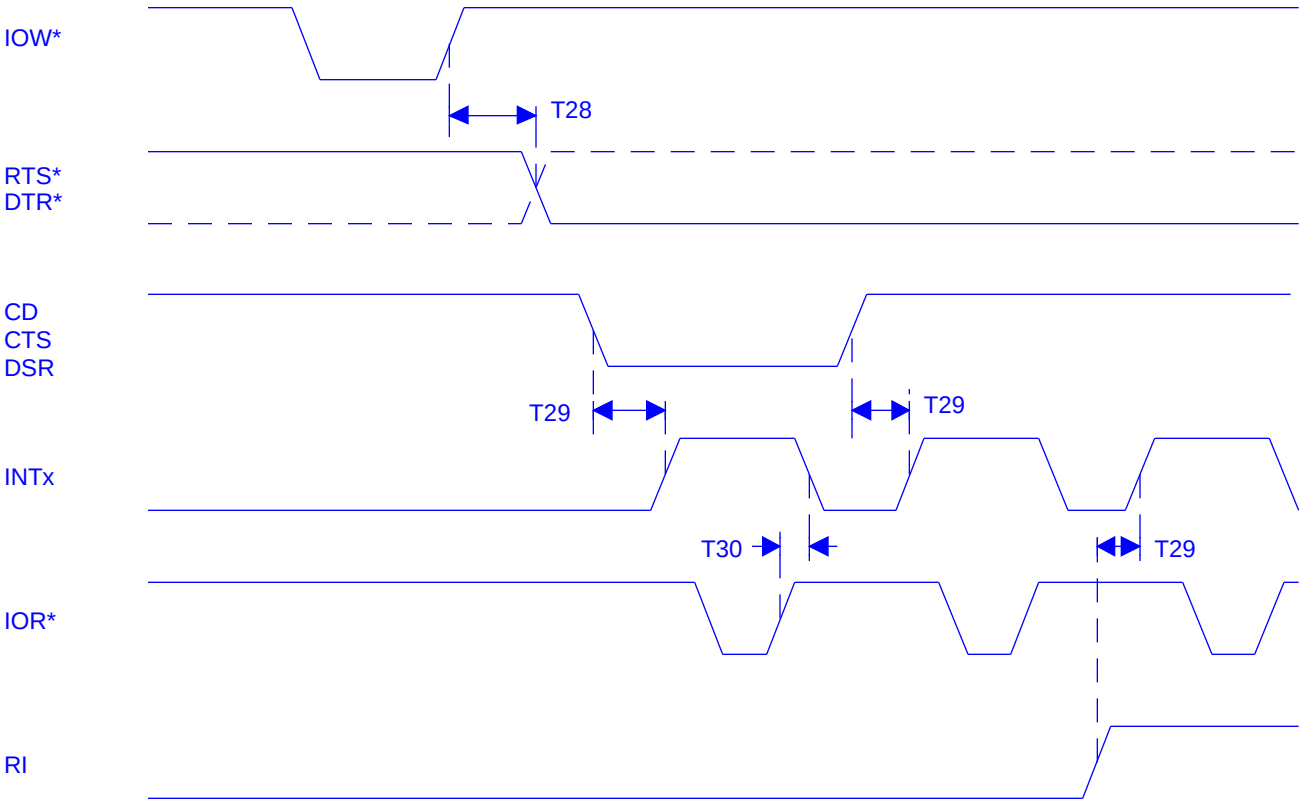
GENERAL WRITE TIMING



162450-WD-1

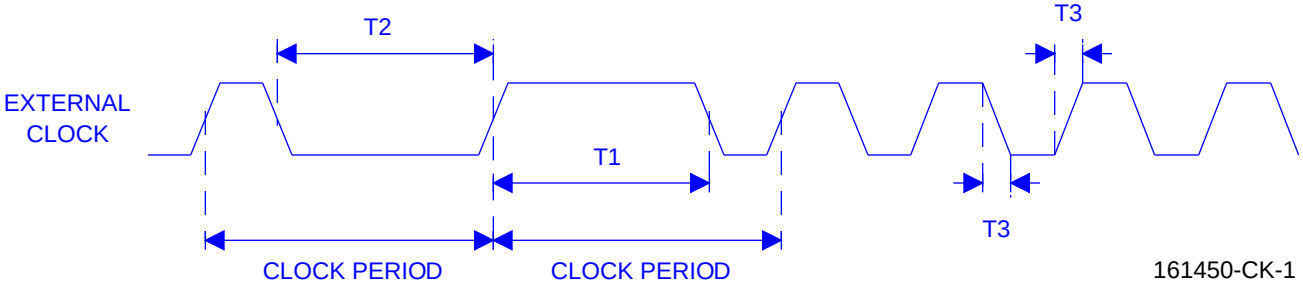
ST16C554D

MODEM TIMING



162450-MD-1

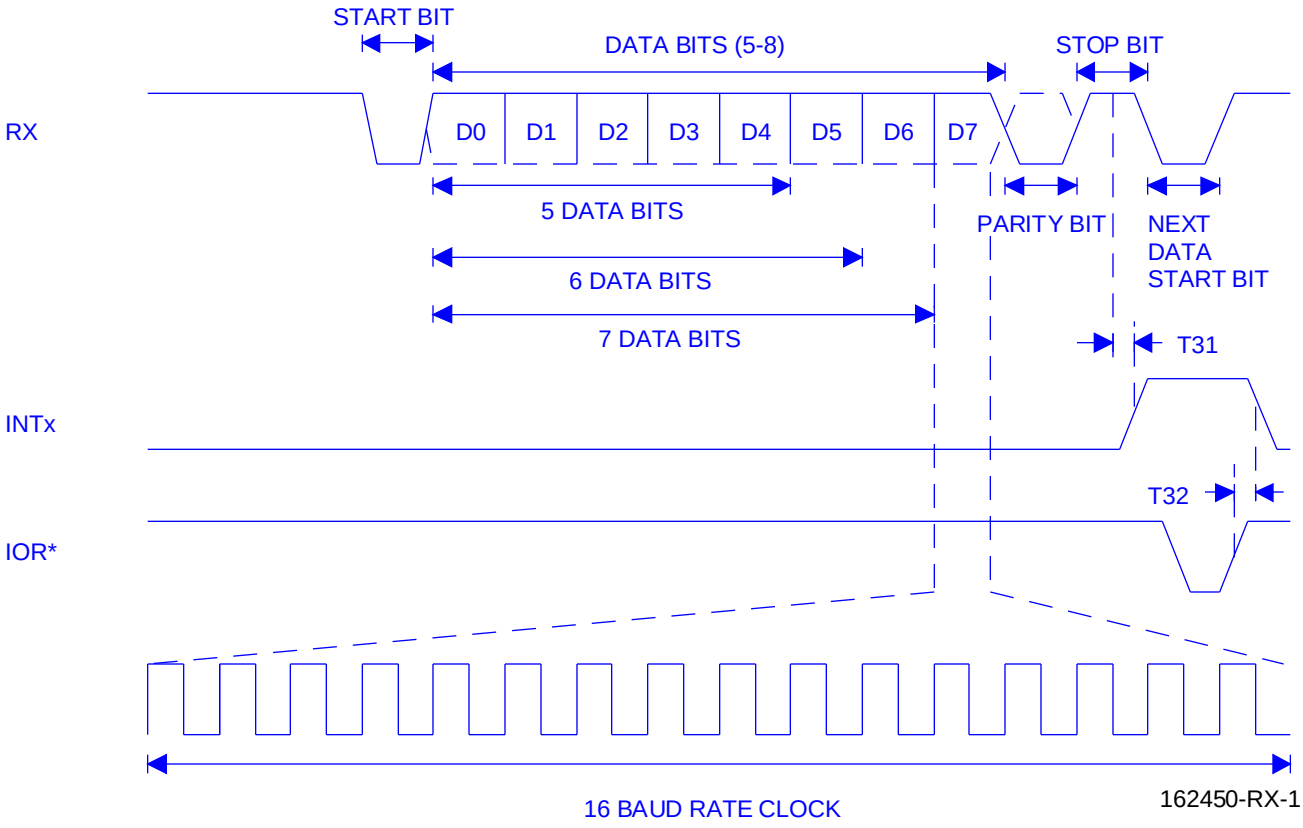
CLOCK TIMING



161450-CK-1

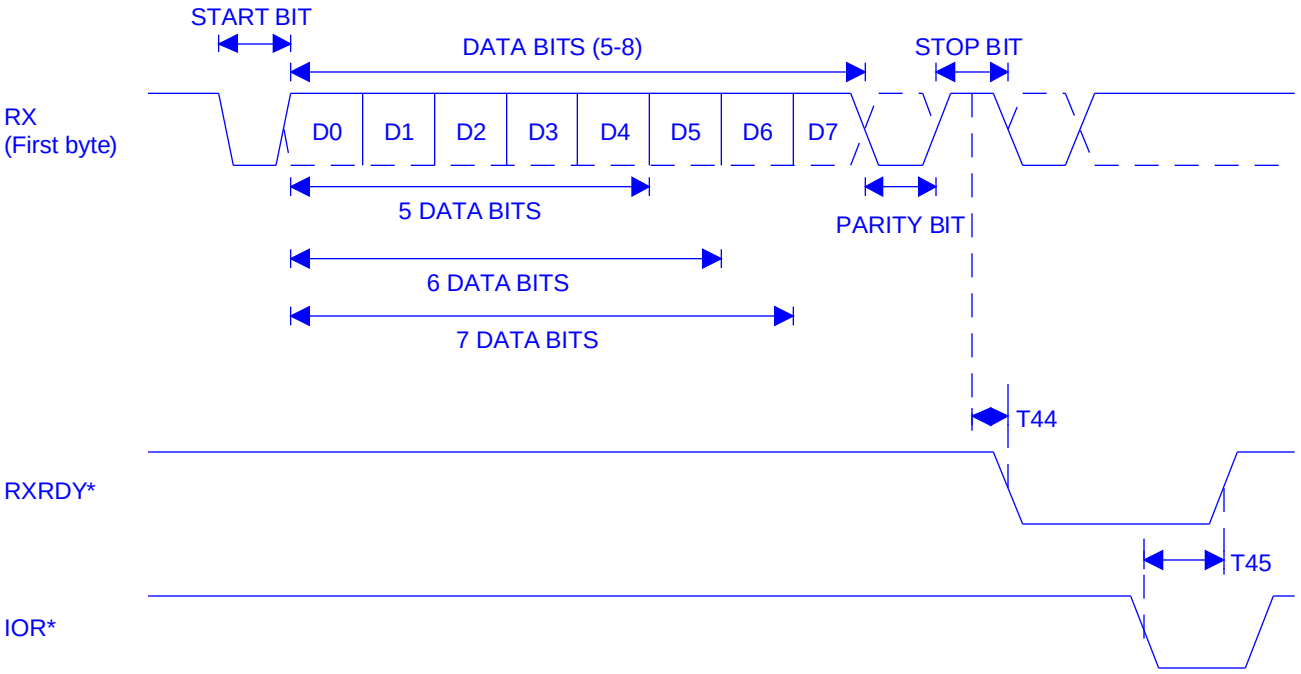
ST16C554D

RECEIVE TIMING



ST16C554D

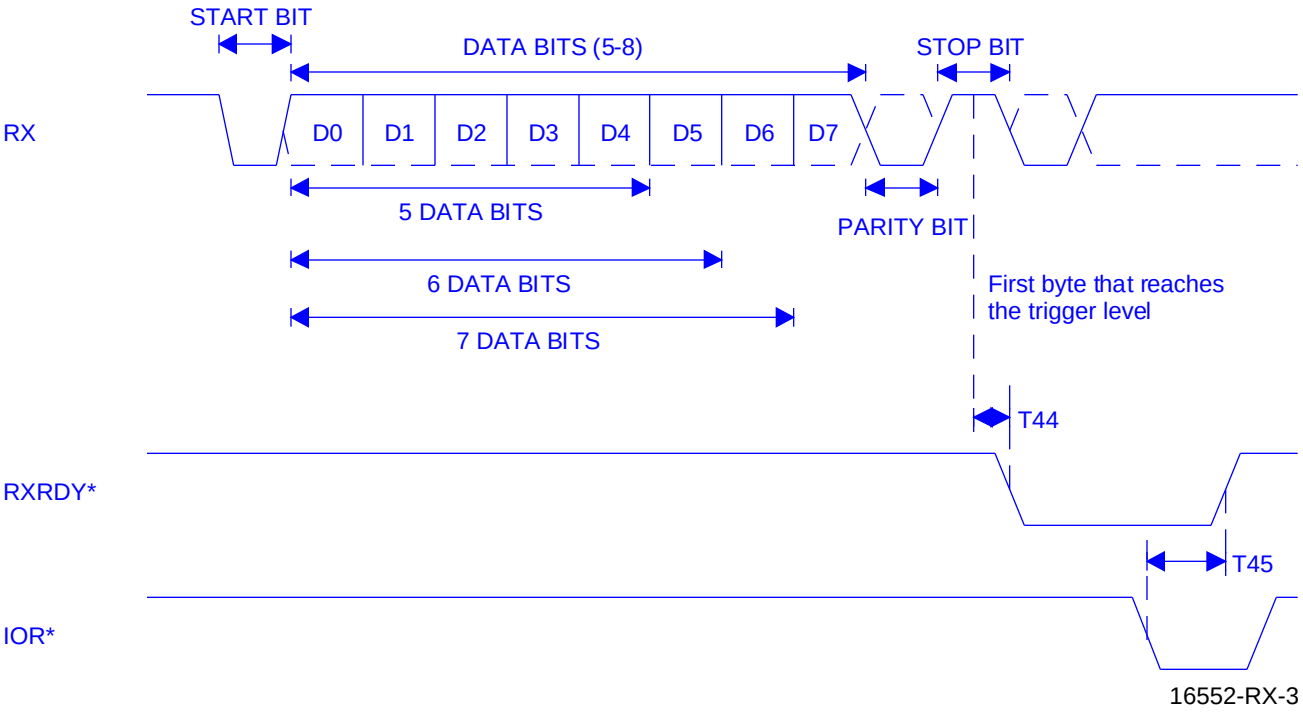
RXRDY TIMING FOR MODE "0"



16552-RX-2

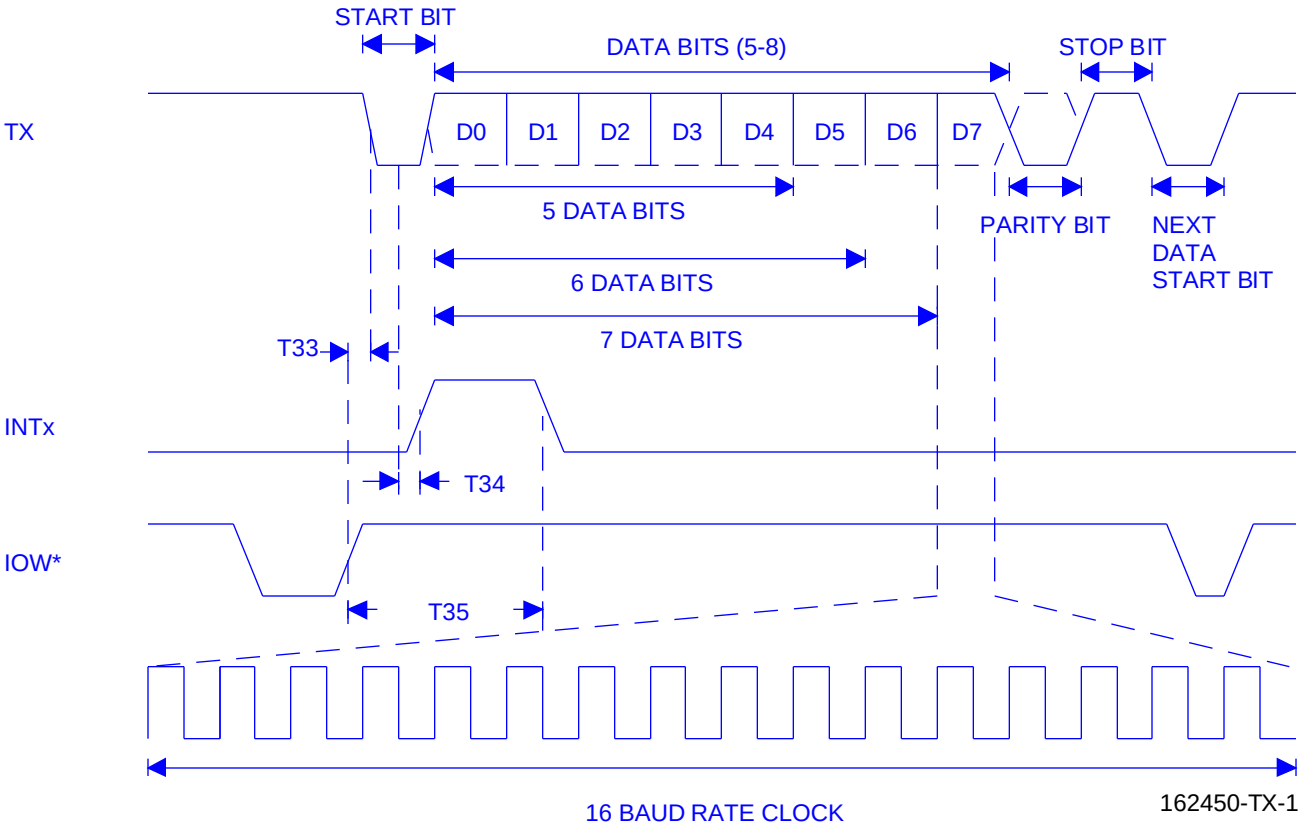
ST16C554D

RXRDY TIMING FOR MODE "1"



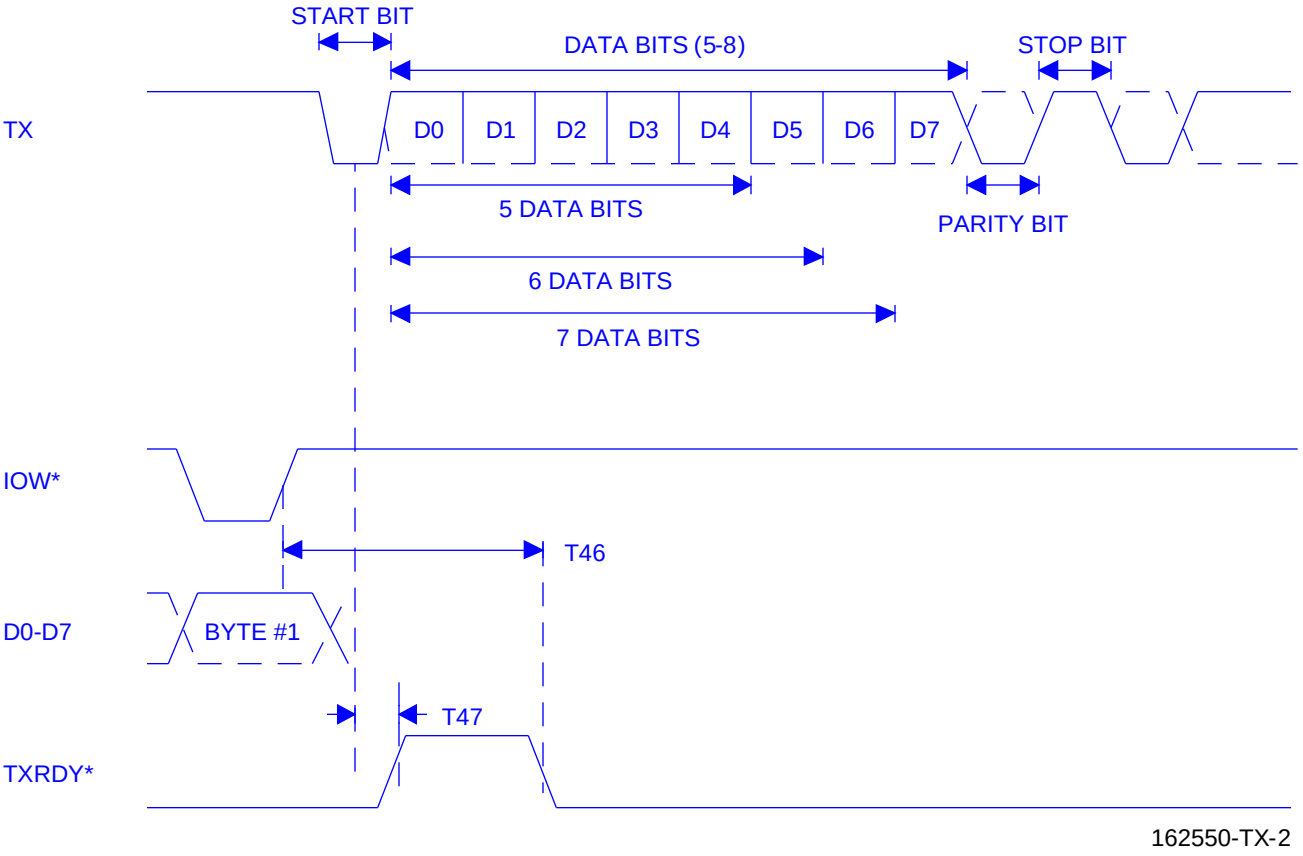
ST16C554D

TRANSMIT TIMING



ST16C554D

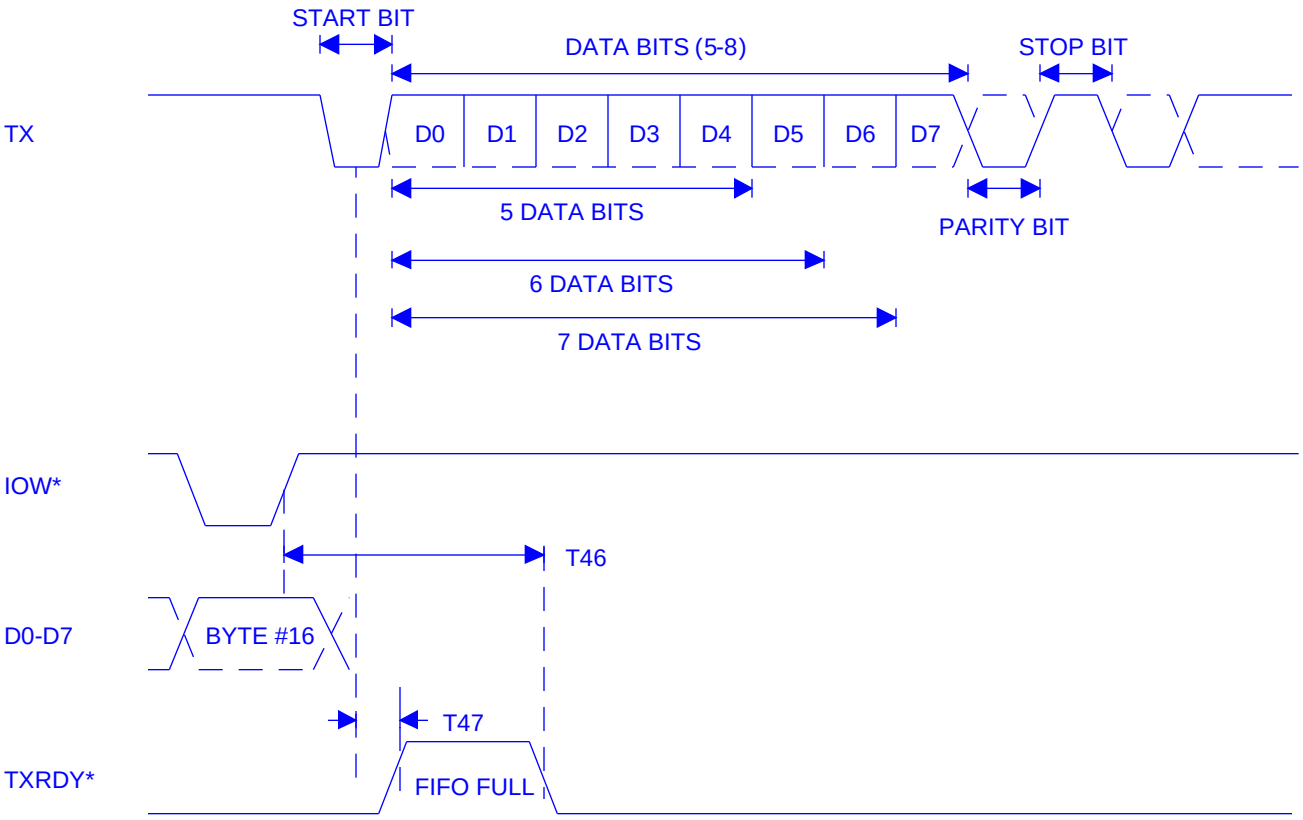
TXRDY TIMING FOR MODE "0"



162550-TX-2

ST16C554D

TXRDY TIMING FOR MODE "1"



162550-TX-3

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP