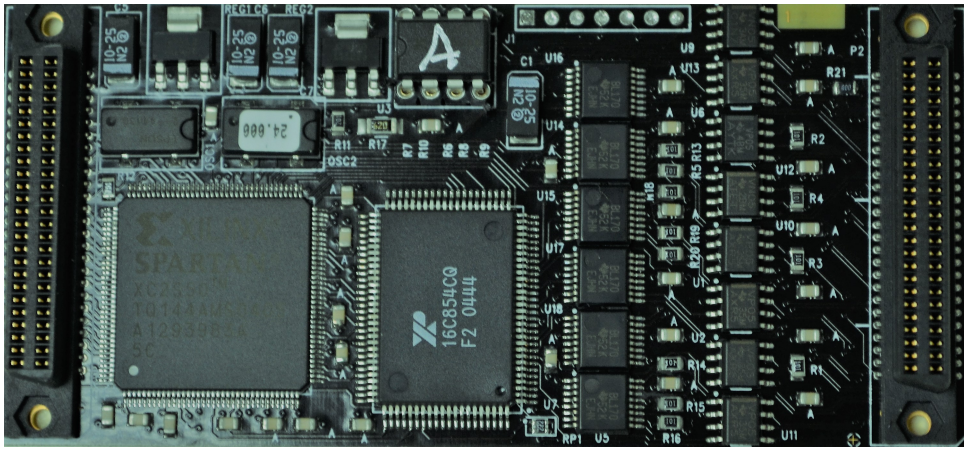
DYNAMIC ENGINEERING
150 DuBois St. Suite B/C Santa Cruz CA 95060
831-457-8891 https://www.dyneng.com
sales@dyneng.com
Est. 1988

User Manual

IP-QuadUART-485

Four Channel UART with Programmable Full/Half-Duplex RS-485 Interface IP Module



Revision 2p1
Corresponding Hardware: Revision 02
10-2002-1702

IP-QuadUART-485
Programmable Differential UART
IP Module

Dynamic Engineering
150 DuBois St. Suite B/C
Santa Cruz, CA 95060
831-457-8891
www.dyneng.com

Revised 4/18/22

Table of Contents

List of Figures

Product Description

IP-QuadUART is part of the IP family of modular I/O components from Dynamic Engineering. This module provides a flexible interface for asynchronous communication with special features that allow more efficient IP bus utilization both in and out of the board.

IP-QuadUART and IP-QuadUART-485 are very similar designs. The original model IP-QuadUART has programmable 485/232 IO. IP-QuadUART has fixed RX-485 IO. The differential speed of the -485 model is higher and the design has differential RTS, CTS & DSR/DTR signals available. The programming interfaces are nearly identical. 3 bits in the channel control registers have different meaning for the two designs.

XR16C854 implements the UART interface. This device is compatible with the industry standard 16550 UART, and is enhanced with 128 byte FIFOs, independent Tx and Rx FIFO counters with programmable trigger levels, automatic hardware/software flow control, and many other features.

The four UART ports on IP-QuadUART-485 can be programmed for full and half duplex operation individually. Each channel has three transmitters and three receivers. Although the transceivers can support higher speed operation the UART is limited to 2.0 Mbps. The 32 MHz IP clock source is required for the 16x clock input at this rate. Performance will depend on cable length and construction.

The RS-485 transmitter enables are controlled by software to allow for multiple drivers on a common differential signal pair. When in half-duplex mode the receiver can be selected to be on the transmitter signal pair. This prevents the need for an extra connection in the cable. In addition the termination resistors are on the Rx pairs, selection the Tx pair for the Rx function eliminates the termination allowing for a multi-drop topology with the termination in the cable rather than at each end point.

When in full duplex mode the on-board terminations and separated Tx and Rx signals mean higher speed point to point operation. Hardware flow control is built in and under software control [trigger levels and enabled or not].

IP-QuadUART supports both 8 and 32 MHz IP Bus operation. The IP Clock and two on-board oscillators are used to provide the reference clock for UART operation. Each UART port has a separate baud rate generator that divides the input clock by any value from one to 2^{16}-1. The output of this stage becomes the 16x clock used for the detection of received data and is divided by 16 to clock out transmit data.

The reference clock input to the UART is selected by software from two on-board oscillators or the IP clock. The standard values for the oscillators are 24 MHz and 18.432 MHz, which allows a wide range of baud rate selections. If a specific frequency is required that cannot be derived from any of the clock sources, a different frequency oscillator can be installed. The 24 MHz oscillator allows the UART to operate up to 1.5 Mbps while the 18.432 MHz clock rate provides the standard baud rates used in most applications. The reference clock for port C is selected using separate bits in the base control register and can therefore use a different clock source than the other three channels. This allows more flexibility in baud rate configuration.

IP-QuadUART-485 conforms to the VITA standard for IP modules. This guarantees compatibility with multiple IP Carrier boards. Since the IP maintains plug and software compatibility while mounted on different form factors, system prototyping may be done on one IP Carrier board, with final system implementation done on a different one. In standard configuration it is a Type 2 mechanical with low profile components on the back of the board and one slot wide.

Interrupts are supported by the IP-QuadUART-485. Each UART channel has a separate interrupt signal input to the module interrupt generation circuit which causes an interrupt on IntReq0 if enabled. Direct and latched versions of the interrupt signals can be read from the module status register to determine which channel caused the interrupt and appropriate action taken. If the master interrupt enable is not set, the interrupt status is still available ⇔ operate in polled mode.

Please see the Software manuals for current offerings of device support. Each package includes a driver and reference software. Currently the Windows version is used along with a Dynamic Engineering IP carrier to perform the ATP on this device.

Theory of Operation

IP-QuadUART-485 is designed to implement four UART channels with programmable serial protocols in a single IP module.

IP-QuadUART-485 features a Xilinx FPGA. The FPGA contains the IP and UART interfaces as well as configuration and status registers for defining the interface between them and to the external serial connections.

IP-QuadUART-485 is a part of the IP family of modular I/O products. It meets the IP Module Vita Standard. In standard configuration it is a Type 2 mechanical one slot wide. Contact Dynamic Engineering for a copy of this specification. It is assumed the reader is at least casually familiar with this document and logic design.

A logic block within the Xilinx device contains the decoding and timing elements required for the host CPU to interface with the IP bus. The timing is referenced to the 8 or 32 MHz IP logic clock. The IP responds to the ID, INT, and IO selects. The DMA control lines are connected to the Xilinx for future revisions, but are not used at this time. The QuadUART design requires wait states for read or write cycles to any address. Hold cycles are supported as required by the host processor. Data remains enabled during a read until the host removes the SEL line. Data is latched and written at the beginning of a write cycle, but the acknowledge will be held until the SEL is de-asserted.

Most accesses require only one wait state, however normal UART reads require two wait states at 8 MHz bus speed and three at 32 MHz. Two features have been added for UART data accesses to increase bus bandwidth. The first is 16-bit data accesses with a single bus cycle and the second is auto pre-read of received data.

The UART data bus is eight bits wide. 16-bit accesses for Tx and Rx data can be enabled on a per-channel basis. On a write cycle 16 bits of data are latched and after one wait state acknowledge is asserted, releasing the bus. The data is then written into the UART Tx FIFO as a background process with two successive UART data writes. During this time the IP bus is free for accesses other than the UART, which will not be available until the writes are complete. The 16-bit read functions similarly, but the bus is not released until the two UART reads have completed and the data is enabled onto the bus. This requires five wait states at 8 MHz and seven at 32 MHz.

If the background pre-read function is enabled, the RXRDYA-D lines are used to determine when data is available to be read from the Rx FIFO for the respective channel. An eight-byte circular buffer is provided for each channel and when data is

available, it is automatically read and stored in this buffer. If more than one channel needs to be serviced simultaneously they will be read in a round-robin pattern as long as data is available and the storage is not full. The data in these buffers is read just like a normal data read, but only one wait state is required for the access. Either one or two bytes will be read in a cycle depending on whether 16-bit data reads are enabled.

It should be noted that these two features are intended for data accesses only, since all the internal registers in the UART are eight bits wide. Therefore the special accesses only occur when the lower address bits are zero, however, the UART has shadow registers at the same address as the data ports that are used for configuration. These shadow registers are enabled by writing certain bits in other registers e.g. the lower byte of the baud-rate setting is enabled at this address when the Line Control Register bit 7 is set to '1'. Therefore it is important that these special registers are disabled before the enhanced data access features are enabled. Please refer to figure 2 and the XR16C854 data sheet for information on enabling and disabling these registers.

If data is requested from the pre-read data store when there is not sufficient data stored to satisfy the request, the bus acknowledge will be held off until the data is present. If this delay is long enough an IP bus error will occur. To avoid this, a user programmable watchdog timer is supplied to complete the bus cycle and issue an interrupt if the timer expires. The timer count is programmed by writing a value to the TIMEOUT_COUNT register. If this value is zero, the counter never expires; otherwise the value is the number of IP clocks before the bus cycle is aborted. This value should be set to be something shorter than the bus error timeout on the host system in order to avoid this error condition. The data read when the timeout occurs comes from the TIMEOUT_DATA register and can be set to any value desired.

The four interrupts from the individual UART channels each have an interrupt enable in their respective control register. These, along with the timeout interrupt and master interrupt enables in the BASE_CONTROL register, control the IP bus interrupt generation. The channel(s) that cause an interrupt can be determined by reading the STATUS register. Numerous interrupt conditions can be setup and monitored using the UART internal registers (see the XR16C854 data sheet). When the UART interrupt conditions have been resolved, the corresponding bits in the STATUS register will be cleared. The timeout interrupt status bit is cleared when the STATUS register is read.

Address Map

Function	Offset	Width	Type
BASE_CONTROL	0x00	word	read/write
VECTOR	0x02	byte on word boundary	read/write
STATUS	0x04	word	read
U_PORT STATUS	0x06	byte on word boundary	read
A_CONTROL	0x08	word	read/write
B_CONTROL	0x0A	word	read/write
C_CONTROL	0x0C	word	read/write
D_CONTROL	0x0E	word	read/write
TIMEOUT_COUNT	0x10	word	read/write
TIMEOUT_DATA	0x12	word	read/write
IP_QUART_A	0x40-4E	byte on word boundary	read/write
IP_QUART_B	0x50-5E	byte on word boundary	read/write
IP_QUART_C	0x60-6E	byte on word boundary	read/write
IP_QUART_D	0x70-7E	byte on word boundary	read/write

FIGURE 1 IP-QUADUART-485 ADDRESS MAP

The address map provided is for the local decoding performed within the IP-QuadUART. The addresses are all offsets from the IO space base address. The IDPROM is mapped to the ID space. The carrier board that the IP is installed into provides these addresses.

The four UART channel address ranges access a number of UART internal registers. All of these registers are 8 bits wide, but the Rx and Tx data port accesses can appear to be 16 bits wide if the special data access features implemented in the Xilinx interface are enabled.

The address offsets for the UART registers and the conditions necessary to access them are listed in the following figure. These are offsets from the base address of the respective UART channels.

REGISTER	OFFSET	FUNCTION
IP_QUART_DATA	0X00	UART read/write data
IP_QUART_IEN	0X02	UART write interrupt enable
IP_QUART_ISTAT	0X04	UART read interrupt status
IP_QUART_FCNTL	0X04	UART write FIFO control
IP_QUART_LCNTL	0X06	UART write line control
IP_QUART_MCNTL	0X08	UART write modem control
IP_QUART_LSTAT	0X0A	UART read line status
IP_QUART_MSTAT	0X0C	UART read modem status
IP_QUART_SPAD	0X0E	UART read/write scratchpad
UART baud rate register defines (enabled when *LCNTL bit-7 = 1, ≠ 0xbf)		
IP_QUART_DLL	0X00	UART read/write LSB divisor
IP_QUART_DLM	0X02	UART read/write MSB divisor
UART enhanced register offsets (enabled when *LCNTL = 0xbf)		
IP_QUART_FTC	0X00	UART FIFO read count/write trigger level
IP_QUART_FEAT	0X02	UART write feature control
IP_QUART_ENF	0X04	UART read/write enhanced features
IP_QUART_XON1	0X08	UART read/write Xon-1 word
IP_QUART_XON2	0X0A	UART read/write Xon-2 word
IP_QUART_XOFF1	0X0C	UART read/write Xoff-1 word
IP_QUART_XOFF2	0X0E	UART read/write Xoff-2 word
UART EMS registers (enabled when *FEAT bit-6 = 1, *LCNTL bit-7 = 0)		
IP_QUART_EMS	0X0E	UART write enhanced mode select
IP_QUART_FLV	0X0E	UART read FIFO level counter

FIGURE 2 IP-QUADUART-485 UART REGISTER ADDRESS OFFSET MAP

Programming

Programming IP-QuadUART-485 requires only the ability to read and write data in the host's I/O space. The beginning address of the IO space for the slot that the IP is installed in is referred to, in this document, as the IP module base address. The IP Carrier board that the module is installed in determines the base address.

Before the UART is accessed, set the CLOCK SPEED bit in the BASE_CONTROL register to the appropriate value. This bit when '1' adds an extra wait state to the UART reads and writes to comply with the UART timing requirements when the IP bus is running at 32 MHz. When the IP bus is running at 8 MHz set the bit to '0' to speedup UART accesses.

The reference clock for the UART is common to all four channels and is selected with the REF CLOCK SEL bits in the BASE_CONTROL register. The baud rates are individually by the UART internal registers for that channel. Refer to the XR16C854 data sheet for details on configuring the UART FIFOs, interrupts, baud rate, etc. After the UART has been configured and the VECTOR register initialized (if needed) the UART channel interrupts can be enabled in the individual slot control registers A_CONTROL - D_CONTROL and the MASTER INT ENABLE set in the BASE_CONTROL register.

A typical sequence would be to first write to the vector register with the desired interrupt vector. Please note that some carrier boards do not use the interrupt vector. The interrupt service routine should be loaded and the appropriate interrupt enables set. When an interrupt is received, first read the STATUS register to determine the source. If it is a timeout interrupt appropriate action should be taken to resolve the lack of expected received data (e.g. UART reset) and try again.

If the interrupt is a UART interrupt, read the ISR (Interrupt Status Register) for that channel to determine the cause. The interrupts are prioritized in this register and only the highest priority interrupt currently pending is reported. When this has been resolved, lower priority interrupts may be read if they are in effect (Refer to the XR16C854 data sheet for details).

Special UART data access features have been implemented on IP-QuadUART to increase the data bandwidth and IP bus efficiency. Before these features are enabled be sure that the alternative UART register sets have been disabled so that the UART data port addresses are accessing the Tx and Rx data ports and not some configuration register. If 16-bit data accesses are enabled the user should ensure that sufficient Rx data is available or sufficient room exists in the Tx FIFO for the transfer to succeed, otherwise data can be lost or duplicated and errors will result. This can be accomplished

by setting FIFO trigger levels and only transferring data when these triggers occur, and only as much data as these levels allow.

When the data pre-read feature is enabled, data will be read from the UART whenever it is available, provided there is room in the pre-read store. This prevents rereading an empty Rx FIFO port, however, if insufficient data is present in the store when the data port is read the bus cycle will be suspended until either the store data is available or the timeout expires.

Refer to the XR16C854 data sheet, the Theory of Operation section above, and the Interrupts section below for more information regarding the exact sequencing of UART accesses and interrupt definitions.

REGISTER DEFINITIONS

BASE_CONTROL

0x00 IP-QuadUART-485 Base_Control Register Port [read/write]

CONTROL REGISTER 0

DATA BIT	DESCRIPTION
9, 8	CHAN C CLOCK SEL
7	TIMEOUT INT ENABLE
6	spare
5, 4	REF CLOCK SEL
3	CLOCK SPEED
2	INT FORCE
1	MASTER INT ENABLE
0	UART RESET

FIGURE 3 IP-QUADUART-485 CONTROL REGISTER 0 BIT MAP

UART RESET when ‘1’ sends a reset to the UART chip, when ‘0’ (default state) the UART becomes operational.

MASTER INT ENABLE when ‘1’ enables interrupts onto the IP bus intreq0n line if any enabled interrupt conditions exist. When ‘0’ (default state) no IP interrupt will occur regardless of any existing interrupt conditions.

INT FORCE is used to create an interrupt for test and software development purposes. Set the bit to cause the interrupt and clear the bit (default state) to remove the interrupt. It requires the master interrupt enable to be ‘1’ to have an effect.

CLOCK SPEED when ‘1’ indicates that a 32 MHz bus clock is being used. When ‘0’ (default state) it means that an 8 MHz bus clock is being used. This bit is used to adjust the timing of the UART control signals to match the clock timing.

REF CLOCK SEL, CHAN C CLOCK SEL controls the source of the reference clock to the UART. Channel C is controlled separately from A, B, and D; the same selections apply.

00 – IP clock (default state)
01 – Oscillator 1 (18.432 MHz)
10 – Oscillator 2 (24 MHz)
11 – No clock (logic low output)

TIMEOUT INT ENABLE when ‘1’ enables the bus time-out interrupt. This interrupt occurs when pre-read UART data is requested but not available.

VECTOR

0x02 IP-QuadUART Interrupt Vector Register Port [read/write]

The interrupt vector for the QuadUART is stored in this byte-wide register. This read/write register is initialized to 0XFF upon power-on reset or software reset. The vector is stored in the odd byte location [D7..0] and is driven onto the data bus during an interrupt acknowledge cycle. The vector should be initialized before the interrupt is enabled or the mask is lowered. The interrupt is automatically cleared when the CPU acknowledges it.

STATUS

0x04 IP-QuadUART-485 Status Port [read only]

Data Bit	Status	
7	INT_D	Interrupt from UART D
6	INT_C	Interrupt from UART C
5	INT_B	Interrupt from UART B
4	INT_A	Interrupt from UART A
3	INT_LAT_T	Latched Time-out Interrupt
2	INTR1	IP Bus Interrupt 1
1	INTR0	IP Bus Interrupt 0
0	IREQ	Interrupt Request

FIGURE 4 IP-QUADUART-485 STATUS REGISTER BIT MAP

IREQ when ‘1’ indicates an enabled interrupt condition exists. If MASTER INT ENABLE is ‘1’, IP Bus interrupt 0 will be asserted, otherwise this status bit allows polling of the interrupt condition.

INTR0 when ‘1’ indicates IP Bus interrupt 0 is active, when ‘0’ this interrupt is inactive.

INTR1 when ‘1’ indicates IP Bus interrupt 1 is active, when ‘0’ this interrupt is inactive. This interrupt is not currently used in the IP-QuadUART-485.

INT_LAT_T when ‘1’ indicates that a time-out interrupt has occurred. This bit is latched and is cleared when the status port is read.

INT_ A when ‘1’ indicates that an interrupt condition is present on UART channel A. This bit is direct from the UART and will clear when all enabled UART interrupt conditions for this channel has been resolved.

INT_ B when ‘1’ indicates that an interrupt condition is present on UART channel B. This bit is direct from the UART and will clear when all enabled UART interrupt conditions for this channel has been resolved.

INT_ C when ‘1’ indicates that an interrupt condition is present on UART channel C. This bit is direct from the UART and will clear when all enabled UART interrupt conditions for this channel has been resolved.

INT_ D when ‘1’ indicates that an interrupt condition is present on UART channel D. This bit is direct from the UART and will clear when all enabled UART interrupt conditions for this channel has been resolved.

U_PORT STATUS

0x06 IP-QuadUART-485 UART A – D Data Status Port [read only]

Data Bit	Status	
7	TXRDY_D	Data can be written to UART D
6	TXRDY_C	Data can be written to UART C
5	TXRDY_B	Data can be written to UART B
4	TXRDY_A	Data can be written to UART A
3	RXRDY_D	Data is available to be read from UART D
2	RXRDY_C	Data is available to be read from UART C
1	RXRDY_B	Data is available to be read from UART B
0	RXRDY_A	Data is available to be read from UART A

FIGURE 5 IP-QUADUART-485 UART DATA PORT STATUS REGISTER BIT MAP

RXRDY_A - D when ‘1’ indicates that data is present in the Receive Holding Register for the corresponding UART channel. When ‘0’ indicates that no data is available to be read.

TXRDY_A - D when ‘1’ indicates that there is room for data to be written into the Transmit Holding Register for the corresponding UART channel. When ‘0’ indicates that no more data can be written.

A_CONTROL - D_CONTROL

0x08, 0x0A, 0x0C, 0x0E IP-QuadUART-485 UART A – D Control Register Port [read/write]

CONTROL REGISTER A - D

DATA BIT	DESCRIPTION
7	spare
6	ENABLE DATA PRE-READ
5	ENABLE 16-BIT WRITES
4	ENABLE 16-BIT READS
3	RS485 Mask for half duplex mode
2	RS485 RX Source Select half/full duplex
1	RS485 TX ENABLE
0	INT ENABLE

FIGURE 6 IP-QUADUART-485 UART CONTROL REGISTER BIT MAP

INT ENABLE when ‘1’ enables the interrupt for the corresponding UART channel, when ‘0’ (default) the interrupt is disabled. This bit must be set for the latched interrupt status to be asserted (see Status register description above).

RS485 TX ENABLE when ‘1’ enables the Tx and RTS output drivers when in RS-485 mode, when ‘0’ (default) these transmitters are tri-stated in RS-485 mode. All output drivers are always active in RS-232 mode. Can be used for half duplex control in 485 mode. RX and TX pairs would be interconnected for one differential pair in this configuration.

RS485 Source Select when ‘1’ select the TX transceiver for Half Duplex operation. When ‘0’ the separate RX receiver is selected for Full Duplex operation.

RS485 Mask when set [‘1’] and in half duplex mode and transmitting the received data is held hi to prevent a local echo of the characters transmitted.

ENABLE 16-BIT READS when ‘1’ executes a double read of the Receive Holding Register of the corresponding UART channel when a read is performed from that address. The results of the first read are placed in the lower byte (D0..7) and the results of the second read are placed in the upper byte (D8..15). The FIFO mode should be enabled and the user should ensure that sufficient data is present to satisfy the request when this mode is used, as the second read occurs automatically whether new data is present or not. When data pre-reads are enabled, this bit controls how much data is read from the pre-read store, in this case, if insufficient data is in the store to satisfy the request, a bus time-out interrupt will occur. When this bit is ‘0’ (default) all UART reads

are byte-wide.

ENABLE 16-BIT WRITES when ‘1’ executes a double write to the Transmit Holding Register of the corresponding UART channel when a write is performed to that address. The lower byte (D0..7) is written in the first write operation and the upper byte (D8..15) is written in the second write operation. The FIFO mode should be enabled with enough room left in the FIFO to hold the amount of data written when this mode is used. When this bit is ‘0’ (default) all UART writes are byte-wide.

ENABLE DATA PRE-READ when ‘1’ enables the data pre-read process for the corresponding UART channel, when ‘0’ (default) this process is disabled. This process consists of two parts: the background reading of data from the UART using the RXRDY line to determine when data is available to be read and loaded in the 8-byte store for the corresponding channel, and the retrieval of stored data onto the IP bus when a read request is made, one or two bytes at a time (depending on the ENABLE 16-BIT READS bit described above). If insufficient data is present when the read request is made, the acknowledge will be held off until either data becomes available, or a user-programmable watchdog timer expires. If the timer expires three things happen: data from the TIMEOUT_DATA register is enabled onto the IP bus, an acknowledge signal is asserted, and the time-out interrupt occurs, provided the proper enables are present.

Note: These three special access modes are intended for data accesses only and therefore only function when the UART data ports are being addressed. However the XR16C854 has a special and an enhanced register set that are enabled by writing bit combinations to the UART Line Control Register. These register sets have alternative registers that are used for initial setup, at the same addresses as the data ports. Therefore care should be taken to only enable the special access modes when the alternative register sets are disabled.

TIMEOUT_COUNT

0x10 IP-QuadUART-485 Time-out Count Register Port [read/write]

The 16-bit count used by the bus time-out watchdog timer is stored in this register. This count is loaded into the watchdog timer and when a request is made to retrieve data from the pre-read store, the timer begins to count down one count for each IP bus count. If the timer reaches zero before a bus acknowledge signal occurs, the contents of the TIMEOUT_DATA register is enabled onto the IP bus, a bus acknowledge signal is asserted, and a time-out interrupt is signaled. If the time-out count is 0x0000 (default), the time-out process will be disabled and a bus error will be generated at the IP bus level.

TIMEOUT_DATA

0x12 IP-QuadUART-485 Time-out Data Register Port [read/write]

The 16-bit data word enabled onto the bus when a time-out occurs is stored in this register. The default value is 0x0000.

INTERRUPTS

All IP Module interrupts are vectored. The vector from the IP-QuadUART-485 comes from a vector register loaded as part of the initialization process. The vector register can be programmed to any 8 bit value. The default value is $FF which is sometimes not a valid user vector. The software is responsible for choosing a valid user vector. Some carriers and systems treat the interrupts as Auto Vectored and do not use the vector. For example in PCIe based systems.

IP-QuadUART-485 has five basic interrupt sources, the four UART ports and the bus timeout interrupt. Each of the UART ports can respond to numerous conditions by generating an interrupt. These conditions are setup in the UART internal registers for each port.

The interrupt level seen by the CPU is determined by the IP Carrier board being used. The master interrupt and timeout interrupt can be disabled or enabled through the BASE_CONTROL register. The individual UART channel interrupts are enabled through the A_CONTROL - D_CONTROL registers. Once the interrupt request is set, the way to clear the request is to service the request, or disable the interrupt.

If operating in a polled mode and making use of the interrupts for status, the master interrupt should be disabled and the individual interrupts of interest enabled. When the STATUS register shows an interrupt pending the appropriate action can take place to clear the interrupt request.

Power on initialization will provide a cleared interrupt request, interrupts disabled, and interrupt vector of $FF.

ID PROM

Every IP contains an ID PROM, whose size is at least 32 x 8 bits. The ID PROM aids in software auto configuration and configuration management. The user's software, or a supplied driver, may verify that the device it expects is actually installed at the location it expects, and is nominally functional. The ID PROM contains the manufacturing revision level of the IP. If a driver requires that a particular revision be present, it may check for it directly.

The location of the ID PROM in the host's address space is dependent on which carrier is used.

Standard data in the ID PROM for IP-QuadUART-485 is shown in the figure below. For more information on IP ID PROMs refer to the IP Module Logic Interface Specification, available from Dynamic Engineering.

Any modifications to the IP-QuadUART-485 board will be recorded with a new code in the DRIVER ID location. The model number is set to 0x07 with a customer number of 0x00.

Address	Data	
01	ASCII "I"	(0x49)
03	ASCII "P"	(0x50)
05	ASCII "A"	(0x41)
07	ASCII "H"	(0x48)
09	Manufacturer ID	(0x1E)
0B	Model Number	(0x07)
0D	Revision	(0xA0)
0F	Customer	(0x00)
11	Driver ID, low byte	(0x00)
13	Driver ID, high byte	(0x00)
15	No of extra bytes used	(0x0C)
17	CRC	(0xAE)

FIGURE 7 IP-QUADUART-485 ID PROM

Loop-back

The Engineering kit has reference software, which includes an external loop-back test for each interface specification. Each test requires an external cable with the following pins connected.

RS-485 Loop-back

Signal A	Signal B	From	To
RXA+	TXA+	1	3
RXA-	TXA-	2	4
CTSA+	RTSA+	5	7
CTSA-	RTSA-	6	8
DSRA+	DTRA+	9	11
DSRA-	DTRA-	10	12
RXB+	TXB+	13	15
RXB-	TXB-	14	16
CTSB+	RTSB+	17	19
CTSB-	RTSB-	18	20
DSRB+	DTRB+	21	22
DSRB-	DTRB-	23	24
RXC+	TXC+	25	27
RXC-	TXC-	26	28
CTSC+	RTSC+	29	31
CTSC-	RTSC-	30	32
DSRC+	DTRC+	33	35
DSRC-	DTRC-	34	36
RXD+	TXD+	37	39
RXD-	TXD-	38	40
CTSD+	RTSD+	41	43
CTSD-	RTSD-	32	44
DSRD+	DTRD+	45	47
DSRD-	DTRD-	46	48

IP Module Logic Interface Pin Assignment

The figure below gives the pin assignments for the IP Module Logic Interface on the IP-QuadUART-485. Pins marked n/c below are defined by the specification, but not used on the IP-QuadUART. Also see the User Manual for your carrier board for more information.

GND	GND	1	26
CLK	+5V	2	27
Reset*	R/W*	3	28
D0	IDSEL*	4	29
D1	n/c	5	30
D2	MEMSEL*	6	31
D3	n/c	7	32
D4	INTSEL*	8	33
D5	n/c	9	34
D6	IOSEL*	10	35
D7	n/c	11	36
D8	A1	12	37
D9	n/c	13	38
D10	A2	14	39
D11	n/c	15	40
D12	A3	16	41
D13	INTREQ0*	17	42
D14	A4	18	43
D15	n/c	19	44
BS0*	A5	20	45
BS1*	n/c	21	46
-12V	A6	22	47
+12V	Ack*	23	48
+5V	n/c	24	49
GND	GND	25	50

NOTE 1: The no-connect signals above are defined by the IP Module Logic Interface Specification, but not used by this IP. See the Specification for more information.

NOTE 2: The layout of the pin numbers in this table corresponds to the physical placement of pins on the IP connector. Thus this table may be used to easily locate the physical pin corresponding to a desired signal. Pin 1 is marked on the IP Module.

FIGURE 8 IP-QUADUART-485 LOGIC INTERFACE

IP Module IO Interface Pin Assignment

The figure below gives the pin assignments for the IP Module IO Interface on the IP-QuadUART-485. Also see the User Manual for your carrier board for more information.

RXA+	RXC-	1	26
RXA-	TXC+	2	27
TXA+	TXC-	3	28
TXA-	CTSC+	4	29
CTSA+	CTSC-	5	30
CTSA -	RTSC+	6	31
RTSA+	RTSC-	7	32
RTSA-	DSRC+	8	33
DSRA+	DSRC-	9	34
DSRA-	DTRC+	10	35
DTRA+	DTRC-	11	36
DTRA-	RXD+	12	37
RXB+	RXD-	13	38
RXB–	TXD+	14	39
TXB+	TXD-	15	40
TXB–	CTSD+	16	41
CTSB+	CTSD-	17	42
CTSB-	RTSD+	18	43
RTSB+	RTSD-	19	44
RTSB-	DSRD+	20	45
DSRB+	DSRD-	21	46
DSRB-	DTRD+	22	47
DTRB+	DTRD-	23	48
DTRB-	GND*	24	49
RXC+	GND*	25	50

NOTE 1: The layout of the pin numbers in this table corresponds to the physical placement of pins on the IP connector. Thus this table may be used to easily locate the physical pin corresponding to a desired signal. Pin 1 is marked with a square pad on the IP Module.

FIGURE 9 IP-QUADUART-485 IO INTERFACE

The pin definition for the IP-QuadUART-485 was chosen for compatibility with existing products to minimize the cost of migration.

GND* The grounds are tied to the local ground plane via 0Ω resistor. Optionally a capacitor or open can be installed to create DC, AC, or isolated references depending on your interfacing needs. DC coupled is the default. –AC will install a .1 uF cap. –OPN will leave the position open. Please note that the GNDs are tied together on IP-QuadUART-485.

Applications Guide

Interfacing

Some general interfacing guidelines are presented below. Do not hesitate to contact the factory if you need more assistance.

Watch the system grounds. All electrically connected equipment should have a fail-safe common ground that is large enough to handle all current loads without affecting noise immunity. Power supplies and power-consuming loads should all have their own ground wires back to a common point.

Power all system power supplies from one switch. Connecting external voltage to the IP-QuadUART when it is not powered can damage it, as well as the rest of the host system. This problem may be avoided by turning all power supplies on and off at the same time.

Keep cables short. Flat cables, even with alternate ground lines, are not suitable for long distances. IP-QuadUART does not contain special input protection.

We provide the components. You provide the system. Safety and reliability can be achieved only by careful planning and practice. Inputs can be damaged by static discharge, by applying voltage less than ground or more than +5 volts with the IP powered. With the IP unpowered, driven input voltages should be kept within .7 volts of ground potential.

Terminal Block. We offer a high quality 50-screw terminal block that directly connects to the flat cable. The terminal block mounts on standard DIN rails.
[https://www.dyneng.com/HDRterm50.html]

Ribbon Cable. Custom lengths built to order.
https://www.dyneng.com/HDRribn50.html

Many flat cable interface products are available from third party vendors to assist you in your system integration and debugging. These include connectors, cables, test points, 'Y's, 50 pin in-line switches, breakout boxes, etc.

Construction and Reliability

IP Modules were conceived and engineered for rugged industrial environments. IP-QuadUART is constructed out of 0.062 inch thick high temperature ROHS compliant FR4 material.

Through hole and surface mounting of components are used. IC sockets use gold plated screw machine pins. High insertion and removal forces are required, which assists in the retention of components. If the application requires unusually high reliability or is in an environment subject to high vibration, the user may solder the corner pins of each socketed IC into the socket, using a grounded soldering iron.

The IP Module connectors are keyed and shrouded with Gold plated pins on both plugs and receptacles. They are rated at 1 Amp per pin, 200 insertion cycles minimum. These connectors make consistent, correct insertion easy and reliable.

The IP is secured against the carrier with four metric M2 stainless steel screws. The heads of the screws are countersunk into the IP. The four screws provide significant protection against shock, vibration, and incomplete insertion. For most applications they are not required.

The IP Module provides a low temperature coefficient of 0.89 W/^{o}C for uniform heat dissipation. This is based upon the temperature coefficient of the base FR4 material of 0.31 W/m-^{o}C, and taking into account the thickness and area of the IP. The coefficient means that if 0.89 Watts are applied uniformly on the component side, then the temperature difference between the component side and solder side is one degree Celsius.

Thermal Considerations

The IP-QuadUART design consists of CMOS circuits. The power dissipation due to internal circuitry is very low. It is possible to create higher power dissipation with the externally connected logic. If more than one Watt is required to be dissipated due to external loading then forced air-cooling is recommended. With the one degree differential temperature to the solder side of the board external cooling is easily accomplished.

Warranty and Repair

Please refer to the warranty page on our website for the current warranty offered and options. https://www.dyneng.com/warranty.html

Service Policy

Before returning a product for repair, verify as well as possible that the suspected unit is at fault. Then call the Customer Service Department for a RETURN MATERIAL AUTHORIZATION (RMA) number. Carefully package the unit, in the original shipping carton if this is available, and ship prepaid and insured with the RMA number clearly written on the outside of the package. Include a return address and the telephone number of a technical contact. For out-of-warranty repairs, a purchase order for repair charges must accompany the return. Dynamic Engineering will not be responsible for damages due to improper packaging of returned items. For service on Dynamic Engineering Products not purchased directly from Dynamic Engineering contact your reseller. Products returned to Dynamic Engineering for repair by other than the original customer will be treated as out-of-warranty.

Out of Warranty Repairs

Out of warranty repairs will be billed on a material and labor basis. Customer approval will be obtained before repairing any item if the repair charges will exceed one half of the quantity one list price for that unit. Return transportation and insurance will be billed as part of the repair and is in addition to the minimum charge.

For Service Contact:

Customer Service Department
Dynamic Engineering
150 DuBois St. Suite B/C
Santa Cruz, CA 95060
831-457-8891
Internet Address support@dyneng.com

Specifications

Logic Interface:	IP Module Logic Interface
Serial Interface:	RS-485: Tx, Rx, RTS, CTS, DTR, DSR for each channel
Ref Clock sources:	IP CLK, 18.432 MHz Oscillator, 24 MHz Oscillator
Software Interface:	Control Registers, ID PROM, Vector Register, Status Ports, FIFOs
Initialization:	Hardware Reset forces all registers to 0 except the Vector Register which resets to 0xFF and channel control registers which reset to 0x02.
Access Modes:	Word in IO Space (see memory map) Word in ID Space Vectored interrupt
Access Time:	back-to-back cycles in 375ns (8MHz.) or 125ns (32 MHz.) to/from UART
Wait States:	1 to ID space, INT, or IO space except for UART reads
Interrupt:	UART interrupt for each channel Bus Timeout interrupt
DMA:	No Logic Interface DMA Support implemented at this time
Onboard Options:	All Options are Software Programmable
Interface Options:	50 conductor flat cable 50 screw terminal block interface User cable
Dimensions:	Standard Single IP Module. 1.8 x 3.9 x 0.344 (max.) inches
Construction:	FR4 Multi-Layer Printed Circuit, Through Hole and Surface Mount Components.
Temperature Coefficient:	0.89 W/^{O}C for uniform heat across IP
Power:	Max. **220** mA @ 5V

Order Information

IP-QuadUART-485	IP Module with 4 UART channels, RS-485 drivers and receivers, RS-485 receivers always terminated 16-bit IP interface
-ROHS	Manufacture with ROHS compliance. Standard is non-ROHS.
-CC	Add Conformal Coating.
Tools for IP-QuadUART	IP-Debug-Bus - IP Bus interface extender https://www.dyneng.com/ipdbgbus.html IP--Debug-IO - IO connector breakout https://www.dyneng.com/ipdbgio.html
Eng Kit–IP-QuadUART	IP-Debug-IO - IO connector breakout IP-Debug-Bus - IP Bus interface extender IP-QuadUART Reference test software Data sheet reprints are available from the manufacturer's web site
IP-QuadUART	IP Module with 4 UART channels, Programmable RS-485/RS-232 drivers and receivers Programmable terminations on RS-485 receivers 16-bit IP interface. See separate Manual