MACRO Network Specification

INTRODUCTION

MACRO (Motion And Control Ring Optical) is a communications standard for distributed machine control -- both motion and I/O. It is ring network connecting a master controller to slave nodes consisting of smart motor drives and I/O.

Overview

MACRO is organized as a ring of masters and slaves, with a high-frequency real-time communications system appropriate for precision motion control and associated tasks. It is not a peer-to-peer network, nor is it a token-ring network.

A MACRO network must be connected in a ring topology; stations in the ring communicate through each other. Therefore stations on the ring must be able to pass through data not addressed to them.

Communications across the ring take place at a user-definable fixed frequency. Each ring communications cycle, all masters on the ring sequentially send command data to their corresponding slaves, and receive feedback data from these slaves. The masters always originate the communications; slaves can only respond to master communications, by substituting their feedback data for their command data into the communications stream. (The only exception to this rule occurs if the slave detects a ring break directly upstream; in this case, it must turn itself into a master and send a message that declares a ring break.)

It is the intent of the MACRO network design that the actual data transmission across the ring be transparent to the software running at either end (master or slave). There must be a corresponding set of data registers, both input and output, at both master and slave stations for any active node on the ring. For a master to send command data to a slave, its software must simply write values to the proper output registers in its own address space; from this point, ring hardware takes over to shift this data across the ring to the matching input registers of the appropriate slave node, where the slave can read the values as if they had been locally generated. Similarly, for a slave to send feedback data to its master, it simply writes values to the proper output registers in its own address space, and ring hardware shifts this data across the ring to the matching input registers of the appropriate slave node.

Note: It is not an absolute requirement of the MACRO standard that all of the actual data transmission be handled directly in hardware, but the very high frequencies at which data transmission occur make software involvement in the actual transmission impractical.

This method of communications makes it as easy for a master to send command data, whether motor torque, motor velocity, or discrete outputs, across the ring to a remote node as it is to command this data locally, such as to an on-board digital-to-analog converter, or discrete output line. Similarly, it is as easy for the master to read feedback data, such as motor position or discrete inputs, from a remote node, as it is to read an on-board encoder counter or discrete input line.

At the bit and byte level, communications are based on an encoding scheme originally developed for the Fiber Distributed Data Interface (FDDI), which sends a self-clocking data stream with error checking encode/decode. This protocol has been implemented in the TAXIchip™ ICs from Advanced Micro Devices, and compatible ICs from other vendors. These aspects of the MACRO standard can be met by using the TAXIchip or equivalent ring transmitter and receiver ICs.

Glossary of MACRO Terms

Active Node: A node on the ring, master or slave, that is "turned on" to be able to transmit new data packets.

Baton Signal: The signal between master stations on the ring that permits coordination of the timing of communications. Each master station transmits the baton signal to the next master downstream when it is done sending all of its command data packets.

Broadcast: The act of sending one data packet to multiple nodes. All but the last node to receive the packet must be inactive nodes.

Data Packet: The set of data that is transmitted or received by a node each ring communications cycle.

Inactive Node: A node on the ring, master or slave, that is "turned off", so unable to transmit new data packets. An inactive node can still receive and re-transmit broadcast data packets.

Master: An entity on the ring that sends command data packets and receives feedback data packets.

Master Node: A logical unit and set of registers on the ring that can send command data packets to a corresponding slave node, and receive feedback data packets from the corresponding slave node.

Master Number: A value from 0-15 associated with each master node, slave node, and data packet for the matching of data packets with nodes.

Master Station: A station on the ring containing one or more master nodes.

Node: A logical unit and set of registers on the ring. A node can send and receive a data packet each ring communications cycle. A node must possess both a master number (0-15) and a slave number (0-15), whether the node is a master node or a slave node. For communications to occur, there must be corresponding master and slave nodes. There can be one or more nodes per station on the ring.

Slave: An entity on the ring that receives command data packets and sends feedback data packets.

Slave Node: A logical unit and set of registers on the ring that can receive command data packets from a corresponding master node, and transmit feedback data packets back to the corresponding master node

Slave Number: A value from 0-15 associated with each master node, slave node, and data packet for the matching of data packets with nodes.

Station: A physical unit on the ring with a ring receiving circuit, and ring transmission circuit, and the circuitry for one or more nodes.

Slave Station: A station on the ring containing one or more slave nodes.

Sync Byte: A byte sent across the ring when no real data is being sent, in order to keep the ring clocks properly synchronized.

Synchronizing Master: The single master station on the ring that starts a ring communications cycle based on its own internal timing circuitry. Any other master stations on the ring must await receipt of the baton signal from the upstream master before starting its communications.

NETWORK ORGANIZATION

MACRO Stations

A MACRO network consists of a group of stations on the ring, each with a single ring output and a single ring input. The output of one station is connected to the input of the next station on the station, forming a ring topology.

One or more of the stations are master stations, each containing one or more master nodes; one or more of the stations are slave stations, each containing one or more slave nodes. A single station may only contain master nodes or slave nodes, it may not contain both.

MACRO Nodes

A node is the basic logical unit on the ring. A node can potentially send and receive a data packet each ring communications cycle. Each node must have 4 input registers and 4 output registers. In each set of 4 registers, 1 register is 3 bytes (24 bits) and 3 registers are 2 bytes (16 bits).

In a master node, the output registers contain command data written by the local station processor and transmitted over the ring as command data packets. The input registers contain feedback data received over the ring as feedback data packets, to be read by the local station processor.

In a slave node, the output registers contain feedback data written by the local station processor and transmitted over the ring as feedback data packets. The input registers contain command data received over the ring as command data packets, to be read by the local station processor.

Nodes on the station, master or slave, may be active or inactive. Inactive nodes may “listen” (latch in data packets), but they may not “talk” (their station may re-transmit incoming data packets with their address, but they may not originate or substitute data packets).

Node Addressing

Every node on the station must have an 8-bit address number; 4 bits represent the master number, and 4 bits represent the slave number. The primary ring communications takes place between active master nodes and active slave nodes having the same address number. There may be only one active master node and one active slave node on the ring with any given address number. Since there are 256 possible address numbers, there may be 256 active master nodes and 256 active slave nodes on the ring at once. There is no limit to the number of inactive nodes on a ring.

Note that *stations* on the ring do not possess address numbers, master numbers, or slave numbers; *nodes* within these stations possess these numbers. While it may be possible, even common, in the design of a master station or slave station to require that all of the nodes in that station must have the same master number, this limitation is a choice of the designer, not a requirement of the MACRO standard (but it is not a violation of the MACRO standard). In addition, multiple master stations may possess active master nodes with the same master number, as long as they do not share the same slave number as well. It is possible for a MACRO ring to have 256 separate master stations, each with one active master node, even though there are only 16 master numbers.

There are no requirements in the standard for any particular ordering of stations on the ring. Physical order does not have to match any numerical order of node numbers, and master and slave stations may be mixed at will. However, if there is a desire to organize the order of the data packets traveling the ring for the highest degree of synchronization, the ordering of stations may become relevant.

NETWORK OPERATIONAL FREQUENCIES

MACRO networks are designed to communicate at a very high, fixed frequency. Because the communications occurs at finite speed, there is of course a limitation to the frequency. The ring itself imposes three limitations on communications frequency, due to:

1.) The number of data packets sent per cycle (equals the number of active master nodes)
2.) The number of stations on the ring, each with a transmission delay
3.) The length of the conductors between stations, allowing for the speed of electromagnetic radiation.

1.) Number of data packets sent: The fundamental transmission rate of a MACRO network is 125 megabits per second (Mbps). Each actual byte is transmitted as 10 bits, including 2 redundant bits, for a rate of 12.5 megabytes per second. Each data packet consists of 12 bytes, resulting in a rate just over 1 million packets per second. Therefore, it takes just under 1 microsecond for a packet to pass a given point on the ring.

2.) Number of stations: Each packet encounters delays in passing through each station. First, the data is de-serialized, then the station must decide the appropriate action to take with the packet. Finally, the data must be reserialized for re-transmission. While the exact delaly at a station is dependent on the design of the station, typical delays are 0.5 to 0.6 microseconds per packet per station.

3.) Length of conductors: Data transmission rates are limited by the speed of electromagnetic radiation in the fiber or copper conductors. This speed is equal to 300 mm per nsec (1 ft per nsec) in a vacuum; about 250 mm per nsec in fiber or copper. Unless the conductors are very long, this is not a significant factor in limiting the ring update rate.

The minimum time for a ring communications cycle can be calculated as:

Min_time = 1.0 μsec * (Number of active master nodes)
+ 0.6 μsec * (Number of stations)
+0.004 μsec * (total meters of conductors)

Allowing a 10% safety margin, the maximum ring update frequency can then be calculated as:

Max_frequency = 0.90 * (1 / Min_time)

Example: A MACRO network has 36 active master nodes on 4 master stations. The matching slave nodes exist on 24 slave stations. The total cable length is 100 meters The minimum time for a ring communications cycle is

Min_time = 1.0μsec * 36 + 0.6μsec * (4+24) + 0.004μsec * 100 = 53.4μsec

The maximum possible ring frequency is then

Max_frequency = 0.90 * (1 / 53.4μsec) = 0.0169 Mhz = 16.9 kHz

PHYSICAL REQUIREMENTS OF TRANSMITTER/RECEIVER

Optical Fiber

For the optical-fiber interface, MACRO utilizes standard FDDI (Fiber Distributed Data Interface) optical transceiver pairs, compliant with the FDDI Physical Layer Medium Dependent (PMD) standard, ISO/IEC 9314-3, and ANSI X3.166-1990. These transceivers must interface with glass optical fibers. The maximum station-to-station distance with glass fiber is 3000 m (~2 miles).

This fiber optic interface standard uses SC-style connectors. Optical transceiver components that can be used include Hewlett-Packard's HFBR-5103 transceiver, and AMP's 269040-1 SC Duplex Transceiver.

Co-axial Cable

For the coaxial-cable interface, MACRO uses standard BNC connectors. Two coaxial cables must be connected between adjacent stations on the ring to carry the differential signal. These cables must be within 150 mm (6 inches) of each other in length in order to carry the signal properly. The maximum station-to-station distance with coaxial cable is 30m (100 ft).

RJ-45 Twisted Pair

MACRO can support RJ-45 "phone cord" twisted-pair cables for interface between stations. The maximum station-to-station distance with RJ-45 cable is 3m (10 ft). This interface can be useful for compact but constrained wiring environments, such as the inside of robot arms.

SIGNAL CHARACERISTICS

Note: The specifications for signal characteristics listed in this section will be met automatically through use of Advanced Micro Devices TAXIchip™ set or equivalent running from a 12.5 MHz crystal, plus the proper fiber optic transceiver or electrical connector.

MACRO data transmission between stations is a 125 megabit-per-second serial data stream, with NRZI (non-return to zero, invert on ones) coding. NRZI represents a "1" by a transition and a "0" by the lack of transition. A station must be capable of handling a +/-0.1% tolerance in this data rate for the data that it receives, and not impose a greater than +/-0.1% tolerance in the rate of the data that it transmits.

Optical

The optical signals in the fiber are 1300 nm wavelength, as per the FDDI PMD standard.

Electrical

The electrical signals, whether transmitted over coaxial cable or RJ-45 cable, are differential, 100K ECL-compatible, referenced (shifted) to operate from a +5V supply.

DATA ORGANIZATION

Byte Format

Note: The specifications of the byte formatting will be met automatically through the proper use of the Advanced Micro Devices TAXIchip™ set or equivalent transmitter and receiver ICs.

Each byte transmitted between stations on a MACRO ring is sent as a 10-bit serial data stream, with 4B/5B coding, as per the ANSI X3T9.5 (FDDI) code. In this coding, each nibble of the byte is represented as a 5-bit quantity, with a redundant bit for error checking. The 10 bits representing the byte are shifted out with the most significant bit of the most significant nibble coming out first.

Data Bytes

The 4B/5B code encodes a byte of data as two nibbles, each represented as 5 bits, by the following patterns:

Hex Data	4-bit value	5-bit pattern	Hex Data	4-bit value	5-bit pattern
0	0000	11110	8	1000	10010
1	0001	01001	9	1001	10011
2	0010	10100	A	1010	10110
3	0011	10101	B	1011	10111
4	0100	01010	C	1100	11010
5	0101	01011	D	1101	11011
6	0110	01110	E	1110	11100
7	0111	01111	F	1111	11101

Command Bytes

Sixteen 10-bit patterns not used by data byte encoding in the 4B/5B code are reserved for “command bytes”, labeled CMD-0 through CMD-F (15). The MACRO standard presently uses only CMD-0 and CMD-1.

CMD-0, which is automatically transmitted when no other command or data byte is being sent, is called the “sync byte”, because its purpose is to maintain synchronization between stations when no data is being sent, so the first true byte sent can be received, and so a station knows the difference between a temporarily idle ring and a broken ring. The sync byte is represented by (11000 10001).

MACRO uses CMD-1 as a packet header, the first byte of the packet that notifies the receiving station that a data packet has arrived. It is represented by (11111 11111).

CMD-2 through CMD-F are reserved for future use in the MACRO standard.

Baton Signal Format

The “baton signal” consists of two consecutive CMD-1 command bytes. The baton signal is used to coordinate the sending of data packets by multiple masters on the ring. Each master transmits the baton signal at the end of its transmissions, passing it downstream to the next master in the ring, like runners in a relay race pass the baton.

Data Packet Format

The data transmitted over a MACRO network is organized into packets. Each communications cycle of the ring, an active master node transmits one command (output) data packet to its corresponding slave node, and receives one feedback (input) data packet from the slave node in return. Note that both master and slave stations may have multiple active nodes, and therefore send and receive multiple data packets in a single ring communications cycle.

A data packet consists of 12 bytes, 9 bytes of which hold actual data.

The first byte is the CMD-1 “command byte” which announces to any station on the ring that a packet has arrived.

The second byte is the address byte, which is organized as 2 4-bit nibbles. The more significant nibble carries the number of the master (0-15) from which a command packet came, or to which a feedback packet is headed. The less significant nibble carries the number of the slave (0-15) to the specified master to which a command packet is headed, or from which a feedback packet came.

The 3rd through 11th bytes contain the actual data of the packet, in one of the formats explained in the following section.

The 12th byte is a checksum byte. Each bit in the byte is set as the even parity of the corresponding bits of the 2nd through 11th bytes.

DATA MEANING

The 9 bytes of actual data in each data packet have a specific organization in the MACRO standard, detailed below.

Register Structure

Each packet of data for a node contains 4 registers: 0, 1, 2, and 3. Register 0 is a 24-bit (3-byte) auxiliary register for non-real-time data. Registers 1, 2, and 3 are 16-bit (2-byte) registers for real-time data. Each node has a command packet of 4 registers and a feedback packet of 4 registers.

Register #	0 (24-bit)	1 (16-bit)	2 (16-bit)	3 (16-bit)
Command	Auxilary Register Aux. Data Ident.	Real-Time Register 1	Real-Time Register 2	Real-Time Register 3
Feedback	Auxilary Register Aux. Data Ident.	Real-Time Register 1	Real-Time Register 2	Real-Time Register 3

Register Transmission Byte Ordering

The data is transmitted with byte ordering as follows: least significant byte of register 0 first, most significant byte of register 3 last.

Real-Time Registers

The real-time registers are for data that is to be processed every ring cycle. The intent is that these registers contain numerical command and feedback data for use in real-time control algorithms. As such, there is basically no handshaking on the data in these registers. Each active node picks up the data sent to it in these registers every software cycle as if it were locally generated data.

The MACRO protocol supports several different modes as to the meanings of the real-time registers, both for motion control and I/O. This permits different modes of operation in both types of equipment.

1. Direct PWM: In this mode, the central controller performs all control tasks, including motor phase commutation and current-loop closure, and transmits the instantaneous voltage (usually PWM) commands

to each phase (up to 3) of the motor of the slave node. It receives back from the slave node actual current information from 2 of the phases and current information. In this mode the structure of the registers is:

Register #	0 (24-bit)	1 (16-bit)	2 (16-bit)	3 (16-bit)
Command	Auxilary Register Aux. Data Ident.	Phase C Voltage Cmd.	Phase B Voltage Cmd.	Phase A Voltage Cmd.
Feedback	Auxilary Register Aux. Data Ident.	Position	Phase B Current	Phase A Current

2. Phase Current: In this mode, the central controller performs all control tasks down to the motor phase commutation, and transmits 2 of the phase current commands over the ring to the slave node. It receives back from the slave node position information -- at minimum 16 bits; optionally 32 bits. In this mode the structure of the registers is:

Register #	0 (24-bit)	1 (16-bit)	2 (16-bit)	3 (16-bit)
Command	Auxilary Register Aux. Data Ident.	(Reserved)	Phase B Current Cmd.	Phase A Current Cmd.
Feedback	Auxilary Register Aux. Data Ident.	Supp. Real- Time Fdbk.	Position	Extended Position

3. Torque: In this mode, the master controller closes the position and velocity loops, and transmits a torque command to the slave node. The torque command is a 16-bit register; optionally, 16 fractional bits of the command can also be sent (the "residual"), but it is not required either to transmit or receive the residual. It receives back from the slave node position information -- at minimum 16 bits; optionally 32 bits. In this mode, the structure of the registers is:

Register #	0 (24-bit)	1 (16-bit)	2 (16-bit)	3 (16-bit)
Command	Auxilary Register Aux. Data Ident.	(Reserved)	Torque Residual	Torque Cmd.
Feedback	Auxilary Register Aux. Data Ident.	Supp. Real- Time Fdbk.	Position	Extended Position

4. Velocity: In this mode, the master controller closes the position loop, and transmits a velocity command to the slave node. The velocity command is a 16-bit register; optionally, 16 fractional bits of the command can also be sent (the "residual"), but it is not required either to transmit or receive the residual. It receives back from the slave node position information -- at minimum 16 bits; optionally 32 bits. In this mode, the structure of the registers is:

Register #	0 (24-bit)	1 (16-bit)	2 (16-bit)	3 (16-bit)
Command	Auxilary Register Aux. Data Ident.	(Reserved)	Velocity Residual	Velocity Cmd.
Feedback	Auxilary Register Aux. Data Ident.	Supp. Real- Time Fdbk.	Position	Extended Position

Note: This commanded velocity value is the commanded velocity inside the position loop (assuming that a position loop is being closed), reacting to position errors, among other factors. As such, 16 bits is sufficient resolution for almost all applications. This value is distinct from the commanded velocity value that is part of the trajectory generation algorithm, which is integrated to create successive commanded position values. For long-term position accuracy, that value is often 32-bits or more in a typical positioning system.

5. Position: In this mode, the master controller closes no loops, and transmits a position command to the slave node. The position command is a 16-bit register; optionally, 16 extended bits of the command can also be sent. It receives back from the slave node position information -- at minimum 16 bits; optionally 32 bits. In this mode, the structure of the registers is:

Register #	0 (24-bit)	1 (16-bit)	2 (16-bit)	3 (16-bit)
Command	Auxilary Register Aux. Data Ident.	(Reserved)	Position Cmd.	Ext. Position Cmd.
Feedback	Auxilary Register Aux. Data Ident.	Supp. Real- Time Fdbk.	Position	Extended Position

6. General Input/Output

If the slave node is for general input and output, and not motion control, the 48 bits of the real-time registers are for fast, direct inputs and outputs. The auxiliary registers can support vast amounts of relatively slower I/O, plus set-up and diagnostic information.

Register #	0 (24-bit)	1 (16-bit)	2 (16-bit)	3 (16-bit)
Command	Auxilary Register Aux. Data Ident.	Output bits 0-15	Output bits 16-31	Output bits 32-47
Feedback	Auxilary Register Aux. Data Ident.	Input bits 0-15	Input bits 16-31	Input bits 32-47

Auxiliary registers

The auxiliary MACRO registers are for data that do not have to be processed every ring cycle. A command from the master in the auxiliary register does not have to be processed immediately by the slave node, and a response from the slave node does not have to be processed immediately by the master. Therefore, low-priority background tasks in both master and slave can service these registers.

Because these registers do not have to be processed immediately, a handshaking scheme must be used to show when the auxiliary data registers have been processed, showing that the registers are free for the next use. This protocol is described below.

Structure

The auxiliary data registers are a pair of 24-bit registers for each node. One register is written to by the master and read by the slave; the other is written to by the slave and read by the master. As with the real-time data registers, the slave node's hardware automatically substitutes the data register value it has written into the ring data stream, replacing the data register value the master had written, which is diverted into a read-only register on the slave.

The high 16 bits of the 24-bit word contain values; the low 8 bits contain identifiers.

Aux. Register Part	High 16 bits	Low 8 bits
Command	Value	Identifier
Feedback	Value	Identifier

Procedure

A command sequence for the auxiliary registers on MACRO is a 4-step process. All four steps must be completed before the next command sequence can begin. In summary, the four steps are:

1.) Master command: This is either a command to write, or a command to read.

2.) Slave response: This is a simple acknowledgement of the write command, or an acknowledgement with response of the read command.

3.) Master “idle” command: This command lets the slave node know that the master has accepted its acknowledgement, and permits the slave to "reset" in preparation for the next command.

4.) Slave “idle” response: This response lets the master know that the slave is ready for the next command.

1. Master Command Format

Write commands: To create a write command, the master puts an identifier value of 2 to 253 in the low 8 bits of its auxiliary write register. This specifies which parameter is being written to. The master also puts the value that is to be written into this parameter in the high 16 bits of the word. The meaning of this value can vary depending on what is being written to. It can represent a numerical value, as in the case of a gain; it can specify a particular action from a list of action values; or it can represent 2 characters of an ASCII data stream.

The identifier value of 253 has been reserved for the command value to represent commands from a list. The identifier value of 252 has been reserved for the command value to represent two characters of an ASCII string. The first of the pair of characters must be in the lower 8 bits of the 16-bit value (Intel ordering).

Aux. Register Part	High 16 bits	Low 8 bits
Command	Parameter Value	Parameter Number
Feedback (from before)	Node status word	0

Read commands: To create a read command, the master puts an identifier value of 1 in the low 8 bits of its auxiliary write registers. This signifies to the slave node that a data response is required. The master also puts the number of the parameter whose value is requested into the high 16 bits. Parameter numbers less than 256 (actually 2 to 252) represent values that can be written to by the host, such as gains. A parameter number of 252 represents a request for the next two characters of an ASCII text response string. Parameter numbers 256 or greater represent read-only values from the slave, such as bus voltage. A

Aux. Register Part	High 16 bits	Low 8 bits
Command	Parameter Number	1
Feedback (from before)	Node status word	0

2. Command Response Format

Response to write command: In response to a valid write command from the master, the slave puts a value of 255 into the low 8 bits (identifier byte) of *its* auxiliary write (feedback) registers. It can also put the 8-bit parameter number in the middle 8 bits.

Aux. Register Part	High 16 bits	Low 8 bits
Command	Parameter Value	Parameter Number
Feedback	Parameter Number	255

In response to an invalid write (or read) command from the master, the slave puts a value of 254 into the low 8 bits of its auxiliary write registers. It also puts the error code into the middle 8 bits.

Aux. Register Part	High 16 bits	Low 8 bits
Command	Parameter Value	Parameter Number
Feedback (from before)	Error Code	254

Response to read command: In response to a valid read command from the master, the slave puts a value of 255 into the low 8 bits (identifier byte) of its auxiliary write registers. It also puts the value of the requested parameter into the high 16 bits. This response can represent, among other things, a numerical value, a set of status bits, or (for "parameter" 252) a pair of ASCII characters.

Aux. Register Part	High 16 bits	Low 8 bits
Command	Parameter Number	1
Feedback (from before)	Parameter Number	255

In response to an invalid read (or write) command from the master, the slave puts a value of 254 into the low 8 bits of its auxiliary write registers. It also puts the error code into the middle 8 bits.

Aux. Register Part	High 16 bits	Low 8 bits
Command	Parameter Number	1
Feedback (from before)	Error Code	254

For either a read or write command, the slave only responds once to a given command. That is, it only accepts the action of a write command once, whether to put a value in a register, put two characters in the command string, or execute a command from a list. For a read command, it only responds once with the requested data, whether numerical, bit field, or ASCII characters, even if its software sees this command several times. In either case, the slave must recognize and respond to the master's "idle" command (see below) before it will respond to another auxiliary command from the host.

3. Master "Idle" Command

To send the "idle" command, the master puts a 255 in the low 8 bits of its auxiliary write (command) word, and a 0 in the high 16 bits. This command word must be sent to terminate any other command. Therefore, this is what is left in the auxiliary command register when it is not being used.

Aux. Register Part	High 16 bits	Low 8 bits
Command	0	255
Feedback (from before)	Code / Value	254/255

4. Slave "Idle" Response

When the slave node receives the idle command, it responds by putting a 0 in the low 8 bits of its auxiliary write (response) register. It also puts the node's general status word in the high 16 bits. This status word is to be updated regularly as long as the master's idle command is present. That is, every time that the slave's software looks at the auxiliary command register and sees the idle command, it should put the current general status word in the auxiliary response register (remember that this does not have to be at ring cycle rates). This permits fast and efficient monitoring of slave node status by the master, both when auxiliary commands are being given, and when they are not.

Aux. Register Part	High 16 bits	Low 8 bits
Command	0	255
Feedback (from before)	Node status word	0

This action of responding to the idle command should prepare the slave software to accept the next auxiliary command. That is, when the command identifier byte changes from a value of 255, the slave must recognize this as a new auxiliary command to which it must respond.

Note on ASCII communications: The above section has described the handshaking procedure for each pair of characters sent or received by the master controller. The content of the ASCII commands is not part of the standard, nor is the exact line-by-line handshaking, although it is expected that both command strings and response strings will be terminated by the carriage-return character.

MASTER STATION RESPONSIBILITIES

A master station on a MACRO ring has several responsibilities for handling ring data:

1.) Transmit command data packets for its active nodes in the proper sequence
2.) Latch in feedback data packets for its active nodes as they come in
3.) Latch in and re-transmit broadcast data packets as they come in
4.) Re-transmit any data packets not addressed to it
5.) Transmit the baton signal when finished transmitting command data packets

Synchronizing Master Station Special Responsibilities

There must be one and only one synchronizing master station on a MACRO ring. The synchronizing master must start the ring communications cycle based on its own internal timing circuitry. As such, it is the synchronizing master station that determines the frequency of ring communications cycles for the entire ring.

The synchronizing master station, as with non-synchronizing master stations, must transmit the baton signal when it is finished transmitting all of its command data packets. When the synchronizing master receives the baton signal, it does not start sending command data packets (unlike non-synchronizing master stations); this marks the end of the ring communications cycle.

If there is only a single master station on the ring, it must be the synchronizing master. The baton signal that it sends passes through all of the slave stations and is received by its input circuits, marking the end of the cycle.

Non-Synchronizing Master Station Special Responsibilities

A non-synchronizing master station must await receipt of the baton signal before transmitting its command data packets. When the baton signal is received, the station must immediately start transmitting

the command data packets for all of its active nodes. As with the synchronizing master, when it is done sending its command data packets, it must transmit the baton signal.

Responsibilities For All Master Stations

Transmitting Data Packets

Any master station, at its appropriate time, must transmit the command data packets for all of its active nodes. It must not transmit command data packets for nodes which are present in the station but inactive.

The data packets must be transmitted in numerical order of packet address byte, from lowest to highest. Because the packet address byte contains both the master number and the slave number, and the master number is the more significant nibble, if packets for multiple master numbers are to be transmitted from a single station, all the packets for the lowest master number are to be transmitted first, from lowest slave number to highest slave number, followed by the packets from the next highest master number, and so on.

In between each data packet, the master must transmit a single sync byte. This sync byte provides increased tolerance for differing station frequencies, and by its unique pattern, is likely to allow decoders in the receivers to recover from erroneous data.

Immediately after transmitting command data packets for all of its active nodes, the master station must transmit the baton signal, to let the next master downstream in the ring (if there is one) know that it should start transmitting its data packets.

Receiving Data Packets

Master stations must be ready to receive data packets at any time, asynchronously to the transmission of packets. Depending on the contents of the address byte of the packet, it must take one of the following actions:

1. Packet address byte matches address of active node on station: In this case, the station latches the data packet into its appropriate receiving registers, and does not re-transmit the packet. “Address matching” can mean either the matching of the full address byte, master number and slave number (this is the normal case for receiving feedback), or just the matching of the slave number if the node has been told to ignore the master number.

2. Packet address byte matches address of node on station that is present but not active: In this case, the station latches the data packet into its appropriate receiving registers, and re-transmits the packet. “Address matching” can mean either the matching of the full address byte, master number and slave number (this is the normal case for single master broadcast), or just the matching of the slave number if the node has been told to ignore the master number (this is the normal case for multi-master broadcast).

3. Packet address byte does not match address of any node on station: In this case, the station simply re-transmits the data packet, which should be the command packet from another master station, or the feedback packet for another master.

SLAVE STATION RESPONSIBILITIES

A slave station on a MACRO ring has several responsibilities for handling ring data:

1.) Latch in command data packets for its active nodes as they come in, substituting its feedback data packets

2.) Latch in and re-transmit broadcast data packets as they come in
3.) Re-transmit any data packets not addressed to it
4.) Immediately re-transmit the baton signal

Slave stations must be ready to receive data packets at any time. Depending on the contents of the address byte of the packet, it must take one of the following actions:

1. Packet address byte matches address of active node on station: In this case, the station latches the data packet into its appropriate receiving registers, and substitutes its feedback data packet into the data stream. "Address matching" can mean either the matching of the full address byte, master number and slave number (this is the normal case for receiving commands), or just the matching of the slave number if the node has been told to ignore the master number.

2. Packet address byte matches address of node on station that is present but not active: In this case, the station latches the data packet into its appropriate receiving registers, and re-transmits the packet. "Address matching" can mean either the matching of the full address byte, master number and slave number (this is the normal case for single master broadcast), or just the matching of the slave number if the node has been told to ignore the master number (this is the normal case for multi-master broadcast).

3. Packet address byte does not match address of any node on station: In this case, the station simply re-transmits the data packet, which should be the command packet from another master station, or the feedback packet for another master.

SYNCHRONIZATION

MACRO provides a simple but powerful method of synchronization between stations on the ring. Each station must have the capability to specify the value of the data packet address byte whose receipt will generate a synchronizing signal on the station. The packet address byte value does not have to match the address of any active, or even inactive, node on the station.

It shall be sufficient for a station to support synchronization on packets of either one master number only (not independent of on-board node address master numbers) or on a "don't care" basis for the master number of the address byte. For example, if a slave station contained nodes only for master number 2, it shall be sufficient for it to have the capability to specify just the slave number of the synchronizing address packet, implying the master number is two, or to specify "don't care" for the master number, if it is desired to use a packet with a different master number.

The synchronization serves several purposes. First, it ensures the tightest possible coordination between axes. Second, it can be used to keep the clocks of different stations from "walking" with respect to each other, which could possibly introduce a beat frequency. Third, it can prevent contention between register access from the ring and from the station's processor, eliminating the need for double buffering of the registers.

Typically the synchronization signal will be used to reset a counter that generates a local station interrupt, keeping this interrupt in phase with other stations. The occurrence of the local interrupt is not dependent on receipt of the packet, but receipt of the packet can adjust the exact time at which the interrupt occurs.

If all stations on the ring are told to synchronize on the same packet address byte, the synchronization signal is delayed between stations only by the packet delay per station of about 0.5 μsec per station, and typically negligible speed-of-light delays.

ERROR DETECTION

There are several levels of error detection built in to the MACRO communications protocol.

1.) *Byte violation errors*: Since each byte is transmitted serially with two redundant bits, the receiving circuit can detect most errors in the data stream. Such a byte error is known as a "violation error", and the receving circuit that converts the input bit stream to an 8-bit parallel data stream also puts out a "violation" bit telling whether it has received an error.

Each station must be looking for violation errors on every byte it receives, whether or not these bytes are part of a packet addressed to a node on the station. If the error occurs as part of a packet addressed to a node on the station, active or inactive, the station should either not latch in the data packet, which leaves the previous cycle's data in place, or latch in the data flagged with the error. A slave station should still substitute its feedback data packet for the errant command data packet.

If the error occurs as part of a packet not addressed to a node on the station, the station should halt re-transmission of the data packet, even if the re-transmission has already started. The station containing the node to which the bad packet is addressed will reject it with a Packet Underflow Error, as described below.

2.) *Packet checksum errors*: As described under *Data Organization*, above, the last byte of the data packet is a checksum byte, with each bit being the odd parity bit for the corresponding bits of each previous byte in the packet.

If the error occurs as part of a packet addressed to a node on the station, active or inactive, the station should either not latch in the data packet, which leaves the previous cycle's data in place, or latch in the data flagged with the error. A slave station should still substitute its feedback data packet for the errant command data packet.

A station does not need to calculate the checksum for a packet not addressed to one of its nodes. On a packet that is re-transmitted, whether latched in to one of its registers or not, the station should re-transmit the checksum byte that it received, not one that it computes itself.

3.) *Packet Underflow Errors*: If there are less than 11 data bytes following a command byte before the next command byte is received, the station should report a packet underflow error. The sync byte that the originating master sends after each packet is a command byte; the header for the next packet is also a command byte.

If the error occurs as part of a packet addressed to a node on the station, active or inactive, the station should either not latch in the data packet, which leaves the previous cycle's data in place, or latch in the data flagged with the error. A slave station should still substitute its feedback data packet for the errant command data packet.

Packet underflow errors can occur when an upstream station has detected a byte violation error in part of the packet, and stopped re-transmitting the rest of the packet.

4.) *Packet Overflow Errors*: If there are more than 11 data bytes following a command byte before the next command byte is received, the station should report a packet overflow error.

If the error occurs as part of a packet addressed to a node on the station, active or inactive, the station should either not latch in the data packet, which leaves the previous cycle's data in place, or latch in the data flagged with the error. A slave station should still substitute its feedback data packet for the errant command data packet.

5.) Ring Break Detection: A station can easily detect a ring break between it and the upstream station because it will receive continuous byte violation errors. As fast as it can read and clear its violation status bit, it will receive another one (12.5 MHz rate). If a station detects such a ring break, it should turn itself into a master if necessary and transmit a special "ring break" signal.

RING OPERATIONAL CYCLE

Synchronizing Master

A ring communications cycle is started by an internal timer-based signal on the synchronizing master station. The synchronizing master transmits command data packets for all of its active nodes, followed by the baton signal (two consecutive sync bytes).

Non-synchronizing Masters

Any non-synchronizing master station in the ring, when it receives the baton signal, starts transmitting command data packets for all of its active nodes. When it is done transmitting these, it sends the baton signal.

Slaves

Slave stations on the ring latch in data packets or not, re-transmit data packets, or substitute feedback data packets, depending on whether the packet address matches node address, and whether the node is active or inactive. Slave stations immediately re-transmit the baton signal (see *Slave Station Responsibilities*).

Master stations on the ring latch in data packets or not, and re-transmit data packets or not, depending on whether the packet address matches node address, and whether the ndoe is active or inactive (see *Master Station Responsibilities*).

When the synchronizing master station receives the baton signal, it does not re-transmit it or start sending data packets. This is the end of the ring communications cycle. The ring is in a quiescent state until the beginning of the next cycle.

REQUIREMENTS FOR A MACRO ASIC

Note: This section assumes that the MACRO ASIC will interface to a TAXIchip™ transmitter and receiver pair, or equivalent, and is driven by a 12.5 MHz (+/-0.1%) clock signal.

The design of an application-specific IC (ASIC) to perform all of the MACRO interface tasks except the byte-to-bit and bit-to-byte conversions typically performed by TAXIchip™ or equivalent ICs is a straightforward task. Such an ASIC consists solely of standard digital circuitry, with no clock frequencies higher than 12.5 MHz required.

Fundamentally, the ASIC together with TAXIchip™ ICs or equivalent must meet all of the specifications for the MACRO ring as explained above, in addition providing the processor on the station access to the ring data. The ASIC must transmit byte-wide data to the ring transmitter IC with proper handshaking and timing, and it must receive byte-wide data from the ring receiver IC with proper handshaking and timing.

It is possible to design an ASIC that can only work in master mode or only in slave mode (although a slave station must be able to turn itself into a master station if it detects a ring break), but very little extra circuitry is required to allow the station with the ASIC to function in either mode.

Number of Nodes

The number of nodes on a MACRO ASIC is up to the discretion of the designer, but it is suggested that a minimum of 4 nodes be present, even on a device intended for a single-axis slave station, so that extra nodes can be used in inactive mode to receive broadcast data.

Some local method of establishing node addresses, master number and slave number, on the station must be provided. Node addresses cannot be established across the ring.

ASIC Interface to Ring Transmitter IC

The physical interface of the MACRO ASIC to the ring transmitter IC contains the following lines, all at TTL levels:

8 data output lines
1 strobe output line
1 command output line
1 acknowledge input line (optional)

A data byte is sent to the ring transmitter IC by putting the byte on the 8 data output lines, holding the command output line low, then toggling the strobe output high, then low.

The command byte used for packet header and the baton signal is sent to the ring transmitter by setting the command output line high, then toggling the strobe output high, then low. The state of the data output lines does not matter in this case.

The command byte used for the sync byte to separate packets and during the ring idle state is sent by holding the command output line low (CMD-0), and *not* toggling the strobe output.

The acknowledge line can be used to confirm that the ring transmitter IC is ready, but would really only be needed if a separate device were sending the command bytes.

For the actual timing requirements of these signals, refer to a databook for the ring transmitter IC, such as the Am7968/Am7969 TAXIchip™ Handbook, from Advanced Micro Devices.

ASIC Interface to Ring Receiver IC

The physical interface of the MACRO ASIC to the ring receiver IC contains the following lines, all at TTL levels:

8 data input lines
1 data strobe input line
1 command strobe input line
1 command value output line
1 violation error input line

When the data strobe input goes high, the ASIC must latch in the 8 data input lines as a data byte, and latch in the violation error input line. It must be prepared to do this at the rate of 12.5 MHz.

When the command strobe input goes high, the ASIC must latch in the 1 command input line, and latch in the violation error input line. With no violation, a high value on the command input line means a packet header/baton signal command byte has been received; a low value means a sync byte has been received.

For the actual timing requirements of these signals, refer to a databook for the ring receiver IC, such as the Am7968/Am7969 TAXIchip™ Handbook, from Advanced Micro Devices.

ASIC Interface to Station Processor

While the nature of the interface of the ASIC to the station processor is not part of the MACRO standard, it is strongly suggested that the node input and output registers simply be mapped into the address space of the station processor, so these registers can be read from and written to as easily as local memory and I/O. Any control and status registers in the ASIC should be mapped in the same manner.