Appendix A

This Appendix includes the following supplemental material:

- Cabling Diagrams and Instructions

- Connectors (9-pin D-type)

- Data Transfer Protocols Usage/Handshaking

- Ultimax Dual Screen Console Keyboard Layout

- Ultimax Single Screen (SSM) Console Layout

- Ultimax Input Screen Hierarchy Chart for Conversational Programming Selections

- Coordinate System Diagram (Showing Spindle Movement Relative to Part Zero)

- Suggested Cutting Speeds in Feet-per-Minute for High Speed Steel End Mills

- Recommended Feed/Flute (Chip Loads)

- Feeds and Speeds for High Speed Drills

Cabling Diagrams and Instructions

There are several cabling options for Hurco machine tools and the equipment that can be connected to them.

Port Locations

RS–232–C connections to a Hurco control are located on the machine's electrical cabinet. There are two identically hardware-configured ports. Either of these ports can be connected to a computer, printer, or another Hurco machine with an Ultimax control.

Connector Types

Hurco machines have RS-232-C serial ports that use 9-pin D-type connectors (DB–9) for Ultimax 3 (single screen) and Ultimax 4 (dual screen) controls.

While the signals conform to the RS-232-C standard, it should be noted that not all standard RS-232-C signals are available at these ports. Some peripheral devices may have provisions for RS-232-C control signals not available at the ports described here. However, such devices can usually be adapted to these ports. In some cases, it may be necessary to add jumpers to the connector for the peripheral device.

Cable Recommendations

Hurco recommends the following for connecting cables:

- Use six-conductor shielded low capacitance cable designed for this purpose.

- Use 22 gauge stranded wire minimum size.

- Limit cable lengths to 100 feet or less. Cable lengths of 50 feet or less can be single shielded. Cables over 50 feet must be multi-shielded.

- Connect cable shield(s) only to the PG pin (protective/frame ground) at the Hurco machine. Do not connect shield(s) to the opposite end of the cable (peripheral device).

- Route cables away from fluorescent lighting fixtures or other possible sources of electrical noise (e.g., power cables, contactors, transformers, and arc welders).

Cable Configurations

The following cable configuration diagrams are a guide for constructing cables that will connect Hurco machines to communication devices.

Note

All cables must support TXD, RXD, and SG signaling. Cables used in hardware handshaking must also support RTS and CTS signaling.

9–pin D–type Connectors

The 9–pin ports are used on the electrical cabinets of Hurco machines equipped with the Ultimax 3 or Ultimax 4 control.

9–Pin Port

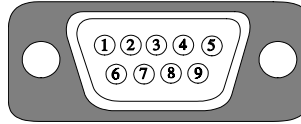The 9–pin port has this pin arrangement:

Figure A-1.   *Front View of 9–pin Connector*

The RS–232–C signals available on the pins are as follows:

Pin ID and Name	Description
1 - DCD	Not used.
2 – RXD (Receive Data)	Data received (by machine) in serial format from peripheral device on this line.
3 – TXD (Transmit Data)	Data transmitted (by machine) to peripheral device in serial format on this line.
4 – DTR (Data-Terminal-Ready)	Not used.
5 - SG (Signal Ground)	Line establishing the common ground reference potential for all interface lines.
6 - DSR (Data Set Ready)	Signal to notify the printer that the transmitter is ready for transmission.
7 - RTS (Request-to-send)	Line used by the machine control to instruct the peripheral device to get ready to receive data. Data can be transmitted after the Clear-to-send signal is received from the connected peripheral device.
8 - CTS (Clear-to-send)	Control line used by peripheral device to indicate that it is ready to receive data from the machine.
9 - RI (Ring Indicator)	Signal indicates receiving the ring of an incoming call.

9–Pin Cabling

There are two possible connection configurations for the 9–pin D–type connector:

- 9–pin to 9–pin
- 9–pin to 25–pin

This diagram shows connection of a Hurco machine equipped with an Ultimax 3 or Ultimax 4 control to a computer (9–pin port) or to another Hurco machine equipped with an Ultimax 3 or Ultimax 4 control.
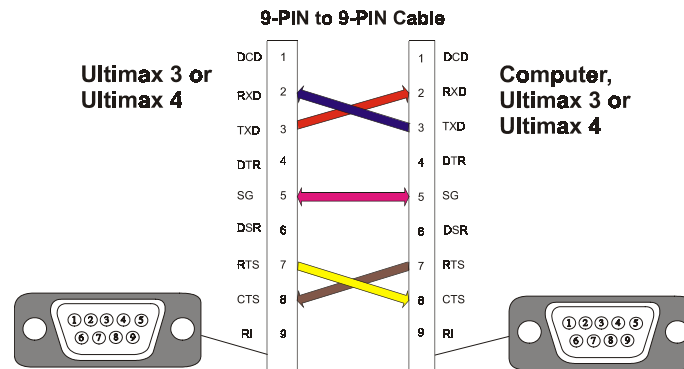

Figure A-2.   9–pin to 9–pin Cable Connections

This diagram shows connection of a Hurco machine equipped with an Ultimax 3 or Ultimax 4 control (9-pin port) to a computer (25–pin port).
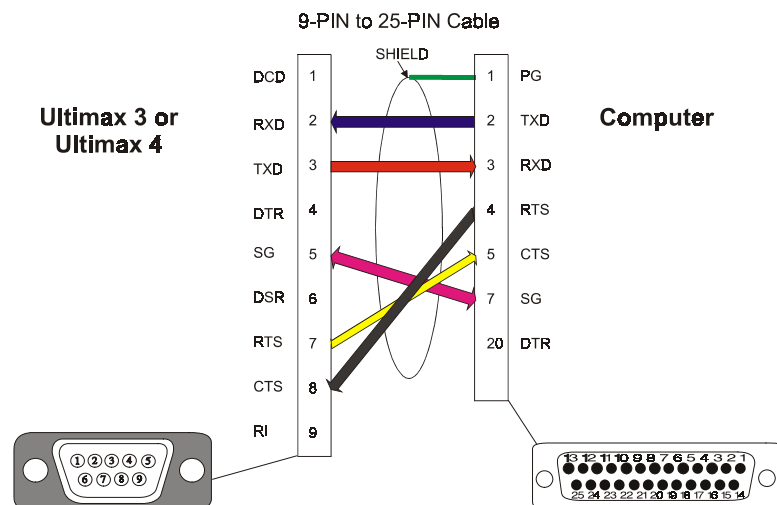

Figure A-3.   9–pin to 25–pin Cable Connections

Protocols Usage or Handshaking

The term protocol or *handshaking* refers to the method by which data is transferred between two devices. Without protocols, the sender may send characters faster than the receiver can handle them. The receiver's internal buffers will probably overflow causing characters to disappear. The Ultimax control uses two types of protocols:

- Hardware
- Software

Hardware handshaking physically connects the RTS and CTS pins of one device to the CTS and RTS pins of another device via a cable. With *hardware handshaking*, the receiver uses electronic signals to tell the sender when the sender can and cannot send characters. This method usually reacts quickly to the signals but requires that the cabling contain enough wires to accommodate the additional signals.

Software handshaking uses the software programmable XON and XOFF control characters embedded in the 7-bit ASCII code. Using *software handshaking*, the receiver sends a special character to the sender instructing the sender to stop until further notice. When the receiver is ready to receive characters, it sends another special character that instructs the sender to continue sending. This method requires fewer signal wires than the hardware handshaking, but reaction time is slower.

CTS/RTS Protocol

The *CTS/RTS protocol* is described from the Ultimax control side of the connection. The following Control As Sender section describes what happens when the control sends a program; the Control As Receiver section describes what happens when the control receives a program. With this information and the knowledge of the remote device, you can adjust the remote equipment to communicate with the control.

When the control's ports are initialized, the TD, RTS and DTR (if available) signals are held in the low state (from -3 to -15 volts or disasserted). When you start a load or save operation at the control console, the control brings the RTS and DTR lines into a high state (from +3 to +15 volts or asserted). This tells the remote device that the control is ready to communicate.

CTS is an input to the control. Its state depends on the state of the handshake lines attached to it from the remote device. If the remote device is ready to communicate, it should be high; if not, it should be low.

Control As Sender

When the control is Saving (sending) a program to the remote device, setting RTS and DTR to HIGH means that the control is READY to transmit. No transmission occurs until the remote device sets its handshake signal to HIGH (which should set the control's CTS to HIGH). As soon as the control sees a HIGH CTS signal, it begins to transmit as long as the CTS input stays HIGH or until all the data has been transmitted.
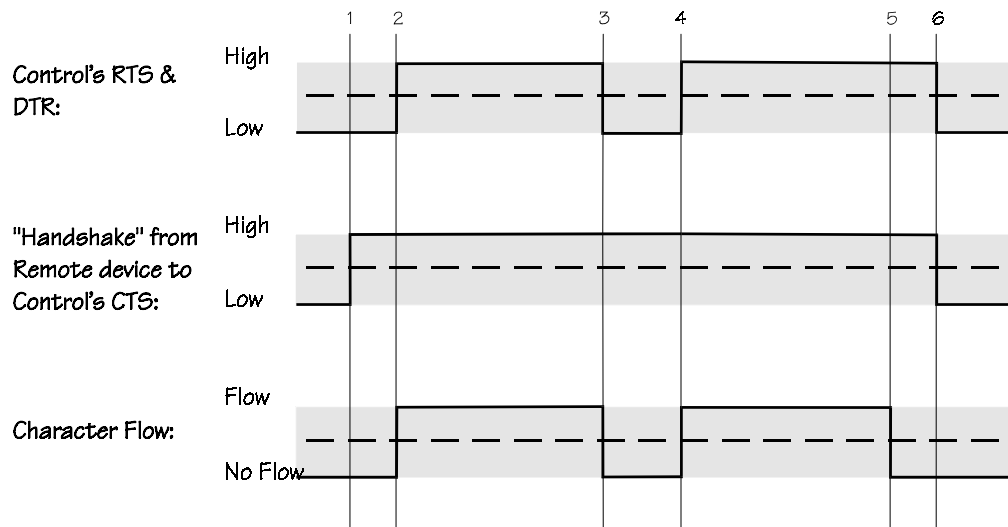

Figure A-4.   Order of Events when the Control Sends
These are the Control as Sender's order of events:

1. The Save operation is initiated at the Ultimax control console.

2. A program is started at the remote device that receives the characters sent by the control.

3. The Remote device's internal buffers fill up. The handshake line is reset LOW to stop the flow of characters.

4. The Remote device is able to receive characters again so it sets the handshake signal HIGH to resume character flow.

5. The control reaches the end of the file and has no more characters to transmit. The End Of File (E) character is transmitted.

6. The remote device acknowledges the E character and closes its port. The control ensures that the last character in the file has been sent, and then closes its port. Both the control and the remote device return to an idle state shown by their handshake signals returning LOW.

With this method, the remote device can regulate the control's character flow by periodically resetting its handshake output LOW causing a LOW control CTS input, thus suspending the flow of characters from the control. When the remote device is able to receive characters again, the device sets its handshake signals HIGH causing a HIGH control CTS, thus resuming the flow of characters.

Control As Receiver

When the control is receiving a program, it signals the remote device that it is ready to receive characters, by setting its RTS and DTR lines HIGH. No communication can occur until the remote device is ready to transmit. When the remote device is ready, it sets its handshake lines HIGH which should set the control's CTS to HIGH. The remote device will begin transmitting when it sees that both the control and the remote device are ready.
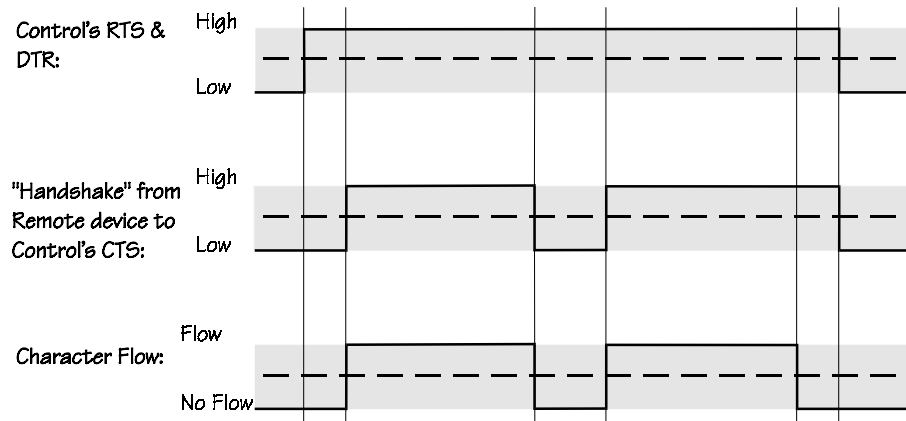
Figure A-5.   Order of Events when the Control Receives

The system follows these steps when receiving a transmitted file:

1. A program is started at the remote device that transmits the characters to the Ultimax control.

2. The load operation begins at the control console.

3. The control's internal buffers fill up. Its RTS and DTR lines are reset LOW to stop the flow of characters.

4. The control is able to receive characters again, so it sets RTS and DTR HIGH to resume character flow.

5. The remote device reaches the end of the file and has no more characters to transmit. The End Of File (E) character is transmitted.

6. The control acknowledges the E character and closes its port. The remote device ensures that the last character in the file has been sent and closes its port. Both the control and the remote device return into an idle state shown by their handshake signals returning LOW.

XON/XOFF Protocol

The XON/XOFF protocol is a software flow control method. It is the preferred transfer protocol for NC programs. The only wires that must be in the cable are TD, RD, SG, and the shield (PG). This method uses three special control characters to start and stop the flow of data characters. The characters that control the character flow follow:

Mnemonic	HEX	Char Name	CTRL
DC1	11H	XON	CTRL Q
DC2	12H	PUNCH ON	CTRL R
DC3	13H	XOFF	CTRL S

The XON/XOFF protocol begins with a synchronization phase. It consists of the sender announcing its presence by sending "DC2" one time every 250 msec for five seconds, until it receives "DC1" (XON), from the receiver or until the operator aborts the sending operation. If the sender receives XON or the five second timer expires, it sends "DC2" to the receiver and begins to send the file.

Meanwhile, the receiver advertises its presence by sending XON one time every 250 msec. The receiver continues to send XONs until it receives a character from the sender or until the operator aborts the receiving operation.

Once the sender begins to send the file, it continues until the receiver sends the XOFF signal, the end of file is reached, or the operator aborts the sending operation. If the sender receives XOFF, it stops transmission and waits for XON before it resumes. The receiver sends XOFF if its internal buffers are full. This buffer level is usually calculated so the receiver sends the XOFF signal early enough for the sender to process the message and stop transmission before the receiver's buffers overflow. After the receiver is able to handle more characters, it sends an XON signal.
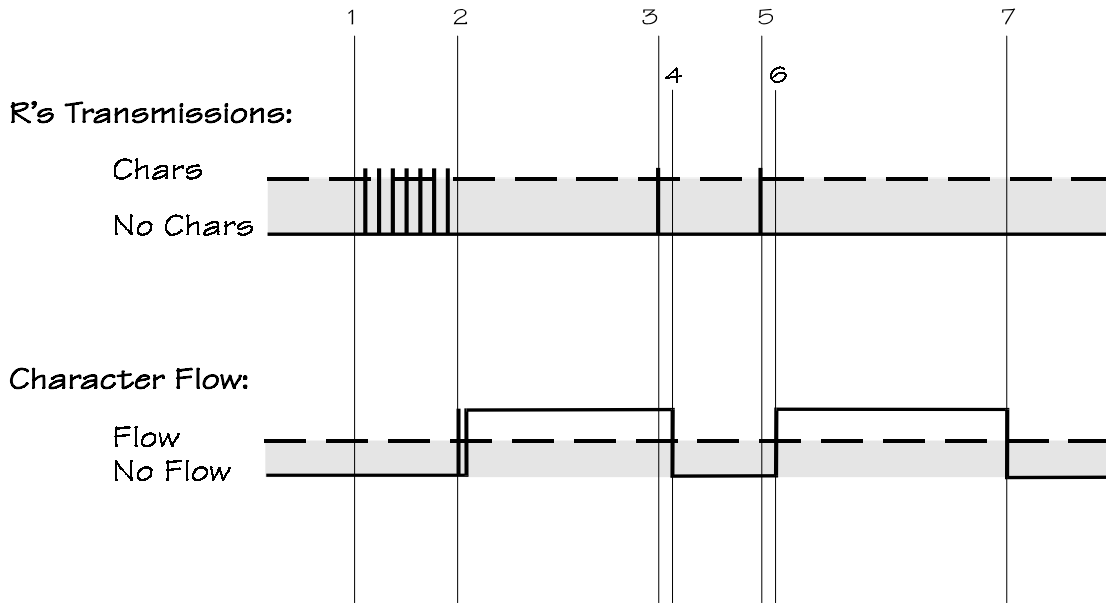
Figure A-6.   XON/XOFF Protocol Transmission Order

The XON/XOFF protocol follows these transmission steps:

1. The receive sequence is started at the receiver. The receiver sends XONs at 250 msec intervals.

2. The transmit sequence is started at the sender. The sender sends a DC2. It sees one of the receiver's XONs and then begins to send characters.

3. The receiver's internal buffer fills up. The receiver sends an XOFF to stop the sender.

4. The sender receives the XOFF and stops transmission.

5. The receiver is able to receive characters again so it sends the sender an XON.

6. The sender receives the XON and resumes transmission.

7. The sender reaches the end of the file, sends the End Of File (E) character and then has no more characters to transmit. It closes the port after it ensures that no more control characters are transmitted from the receiver. The receiver receives the E and then closes its port.

Note

Between steps 3–4 and 5–6, there are time lags as the receiver sends the control characters and the sender responds to them.

When using XON/XOFF protocol, initiate the receiver first so that it is immediately ready to receive. After it starts, you may start the sender. If you do not initiate the receiver and the sender in this order, the sender's five-second timer may expire and characters will be sent without a receiver.

Full Handshake

As with XON/XOFF, full handshake is a software protocol differing from XON/XOFF only in the beginning synchronization stage. Full handshake uses the same type of cabling as XON/XOFF.

With full handshake, the sender does not have a five-second time-out. Instead, DC2s are sent every 250 msec until the confirming that the DC1 is received, or the sender aborts the operation. Once the sender receives the DC1, it sends the confirming DC2 and begins the file transmission. This protocol makes it easy to start both the sender and the receiver at the operator's leisure and in whichever order is appropriate for the application.

During the actual transmission of the file, XON/XOFF and full handshake operate identically.

System Communication and Memory

This section describes memory usage and transfer efficiency for transferring programs to the CNC over an RS-232-C remote device. There are several factors that affect the *efficiency* of this transfer:

- CPU speed of the computer used to transfer files to the CNC

- Transfer rates

- Error checking, block interpretation, and execution

- Loading the program in the control's memory

Transfer Rates

The *baud rate*, speed of the communication link, determines how fast a single character's bits (with synchronization bits) are sent across the link. The speed at which characters are sent to the control are measured in characters per second.

Clear distinctions exist between baud rate and average bit rate.

- *Bit rate* is the number of bits of useful information sent per second.

- The baud rate limits the maximum bit rate, thus limiting the number of characters per second. Additionally, if time gaps exist between the transmitted characters, the character rate is further limited and the bit rate decreases; however, the baud rate does <u>not</u> change.

For information about selecting the baud rate, refer to the "Setting up the Serial Port" section of this chapter.

Hints and Tricks

Always use your fastest computer for transferring programs to the CNC.

Program Interpretation and Execution Times

The program interpretation time is the length of time required to convert commands in the program to motion commands. The control must convert the program characters into machine executable form. The ASCII or EIA characters must be converted into binary coordinates and other machine controlling information.

The execution time is the minimum amount of time required to actually execute the interpreted block. The distance that the axes move in combination with the program feedrate can be used to calculate the execution time for the block. Most control systems set the minimum length of time for an interpreted block.

Memory Usage

In the NC System, the time required to load an NC program and check for errors is combined. If a program fits in the control's memory in its entirety, factors that affect the block transfer time are the program interpretation, execution time, and CNC processor speed. Such a program is executed with no delays or dwells.

Large Programs

Programs that are larger than the control's memory capacity must be handled differently. If the line segment length and the program feedrate are properly matched, the control will not dwell. Try to keep the communication speed as high as the cabling allows.

If the machine executes the blocks faster than they can be received and checked, the internal buffers are starved and the machine dwells because it runs out of data to execute. In this system, the control tries to minimize the number of these dwells. If this dwell situation occurs, the Z axis retracts from the part and the internal buffer fills to its maximum capacity. This message appears: "Reloading Buffer…Tool Has Been Raised From Part Surface." The Z axis then returns to the part surface and program execution continues. Change the Depletion Retract distance using the General Parameters screen.

A part program larger than the control's memory will cause problems when checking for errors. The beginning of the program will no longer be in memory when the check for errors function is completed. Therefore, for the program to run successfully it will need to be read in its entirety using either the serial port or hard drive.

If a syntax error appears during error checking and your program spans the control's memory capacity, the block number reported in the error will not match the block number displayed in the editor. The block number

reported in the error is based on the number of blocks from the program's beginning. Reload the part program and search for the block number referenced by the syntax error.

Consider upgrading the memory capacity for the control if you use a lot of programs that span the control's current memory capacity. Upgrading memory is inexpensive and eliminates the need for program reloading.

Allocation

By default, the system reserves 64KB of memory to serve as a buffer for display, temporary memory, and NC programs. Upon system initialization, the amount of bytes for program storage is displayed. After the system displays the "Loading executive - please wait" message, several lines of information appear as the CNC powers up. This information is eventually replaced by the Input screen.

The available memory appears in the Free: xxxxxxx field at the bottom of each screen, where xxxxxxx represents the amount of available memory, including the reserved 64KB.

Each NC block that is loaded has some associated memory overhead (21 bytes per each line of NC, regardless of line length) and is included in the total bytes allowed. The number of bytes for an NC program is displayed on the Current Directory screen for the program highlighted.

- For every line in the NC part program (including the % and E lines) an additional 21 bytes of program memory are used for formatting and displaying the program.

- Therefore, a 1000 line program will consume an additional 21000 bytes of program memory for overhead.

The amount of memory needed to load in a program is based on the sum of these values:

- Size of program

- Number of lines x 21

- 64KB

Upload and Download Programs

The Ultimax Upload/Download function allows you to transfer part programs between the control and various peripheral devices (e.g., a personal computer or an external disk drive). Programs are transferred from these devices over RS–232–C serial ports.

- Conversational programming can transfer programs between memory and peripheral devices.

- NC programming can transfer programs between memory and peripheral devices, and between the current drive and peripheral devices.

Transfer	Conversational	NC
Memory to Peripheral Device (Download Send)	Yes	Yes
Current Drive to Peripheral Device (Download Send)	No	Yes
Peripheral Device to Current Drive (Upload Receive)	No	Yes
Peripheral Device to Memory (Upload Receive)	Yes	Yes

⇨ **Important**

When sending programs to a remote device, the control sends only *ASCII* data.

The communications facilities of the Ultimax control are compatible with the Hurco's offline software packages and the Hurco NC Communications Utility. For descriptions of cabling requirements, refer to the Appendix.

Setup

Before attempting to communicate with any connected peripheral device, be certain that the correct cable is connected between the designated serial port and the device. Refer to the *Maintenance Manual* for more information about required cabling. Also, ensure that the Hurco serial port and the peripheral device are identically configured. Use the Serial Port Setup softkey, described in this section, to check and change the configuration.

⇨ **Important**

Improper cable connections and incompatible communications settings are the most common problems encountered when attempting to communicate between two electronic devices.

From the Auxiliary screen, press the Upload/Download (F3) softkey. The Upload/Download settings are accessed the same way for conversational part programming and NC part programming; however, the screens differ depending on the selected program editor.

The Upload/Download screen's softkeys are defined as follows:

- **Upload Receive (F1)** – allow the control to receive data from properly connected peripheral such as a personal computer.

- **Download Send (F3)** – send data to a properly connected peripheral.

- **Select Port (F5)** – switch between the ports. When this softkey is pressed, the screen changes to display the other port (and its settings). The Select Port softkey is available for conversational programs only.

- **Abort Operation (F5)** – stop the current operation. The Abort Operation (F5) softkey is available for NC programs only. Conversational provides an abort (F8) softkey after an upload or download is started.

- **Reset Port Parameters (F6)** – reset the displayed serial port to its default values.

- **Serial Port Setup (F7)** – select the communication settings for the displayed serial port. For NC, you can select communication settings for one or both of the serial ports.

- **Exit (F8)** – return to the Auxiliary screen's original softkey options.

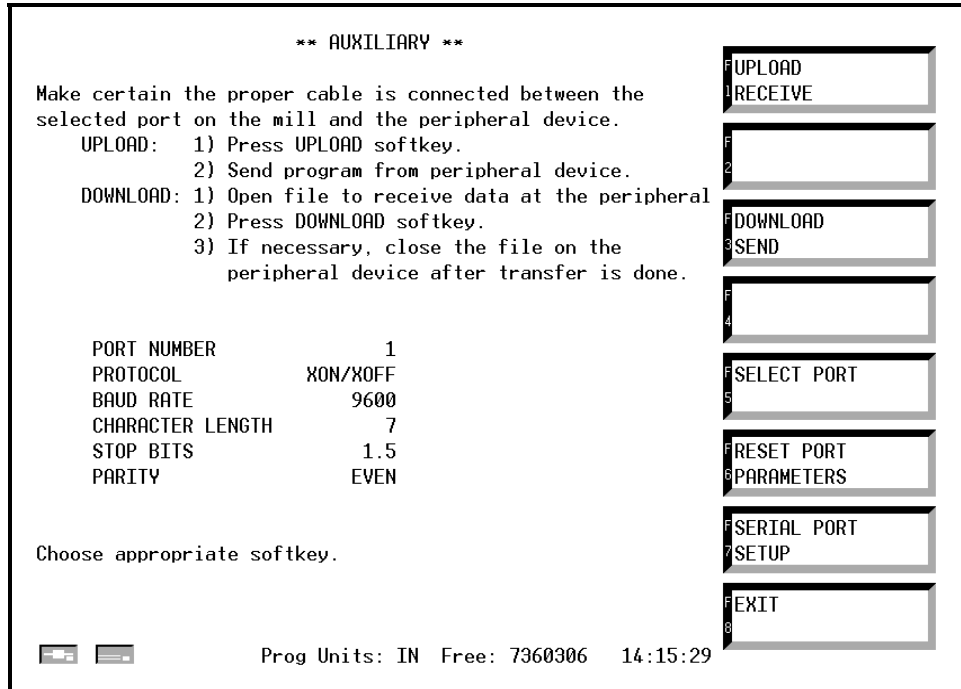Here is a conversational Upload/Download screen with serial port settings:

```
                    ** AUXILIARY **                    ┌─────────────┐
                                                       │ UPLOAD      │
  Make certain the proper cable is connected between the │ RECEIVE     │
  selected port on the mill and the peripheral device.   └─────────────┘
     UPLOAD:    1) Press UPLOAD softkey.                ┌─────────────┐
                2) Send program from peripheral device. │ F           │
     DOWNLOAD: 1) Open file to receive data at the peripheral│ 2       │
                2) Press DOWNLOAD softkey.              └─────────────┘
                3) If necessary, close the file on the ┌─────────────┐
                   peripheral device after transfer is done.│ DOWNLOAD │
                                                       │ SEND        │
                                                       └─────────────┘
                                                       ┌─────────────┐
                                                       │ F           │
        PORT NUMBER            1                       │ 4           │
        PROTOCOL           XON/XOFF                    └─────────────┘
        BAUD RATE             9600                     ┌─────────────┐
        CHARACTER LENGTH         7                     │ SELECT PORT │
        STOP BITS              1.5                     │             │
        PARITY              EVEN                       └─────────────┘
                                                       ┌─────────────┐
                                                       │ RESET PORT  │
  Choose appropriate softkey.                          │ PARAMETERS  │
                                                       └─────────────┘
                                                       ┌─────────────┐
                                                       │ SERIAL PORT │
                                                       │ SETUP       │
                                                       └─────────────┘
                                                       ┌─────────────┐
                                                       │ EXIT        │
                                                       │             │
  ▭  ▭         Prog Units: IN  Free: 7360306   14:15:29└─────────────┘
```

Figure 2-37.   Conversational Upload/Download Screen

Here is an NC Upload/Download screen with serial port settings:

```
                    ** AUXILIARY **                    ┌─────────────┐
                                                       │ UPLOAD      │
  Make certain the proper cable is connected between the │ RECEIVE     │
  selected port on the mill and the peripheral device.   └─────────────┘
     UPLOAD:    1) Press UPLOAD softkey.                ┌─────────────┐
                2) Select the desired serial port.     │ F           │
                3) Send program from peripheral device.│ 2           │
     DOWNLOAD: 1) Open file to receive data at the peripheral└─────────┘
                2) Select the desired serial port.     ┌─────────────┐
                3) Press DOWNLOAD softkey.             │ DOWNLOAD    │
                4) If necessary, close the file on the │ SEND        │
                   peripheral device after transfer is done.└─────────┘
  SERIAL PORT 1:            (NOT BUSY)                  ┌─────────────┐
       PROTOCOL          LEVEL 2 XON/XOFF               │ F           │
       BAUD RATE              9600                      │ 4           │
       SOURCE/DEST           MEMORY                     └─────────────┘
  SERIAL PORT 2:            (NOT BUSY)                  ┌─────────────┐
       PROTOCOL          LEVEL 2 XON/XOFF               │ ABORT       │
       BAUD RATE              9600                      │ OPERATION   │
       SOURCE/DEST           MEMORY                     └─────────────┘
  Choose appropriate softkey.                          ┌─────────────┐
                                                       │ RESET PORT  │
                                                       │ PARAMETERS  │
                                                       └─────────────┘
                                                       ┌─────────────┐
                                                       │ SERIAL PORT │
                                                       │ SETUP       │
                                                       └─────────────┘
                                                       ┌─────────────┐
                                                       │ EXIT        │
  ▭  ▭         Prog Units: IN  Free: 7022549   10:55:02│             │
                                                       └─────────────┘
```
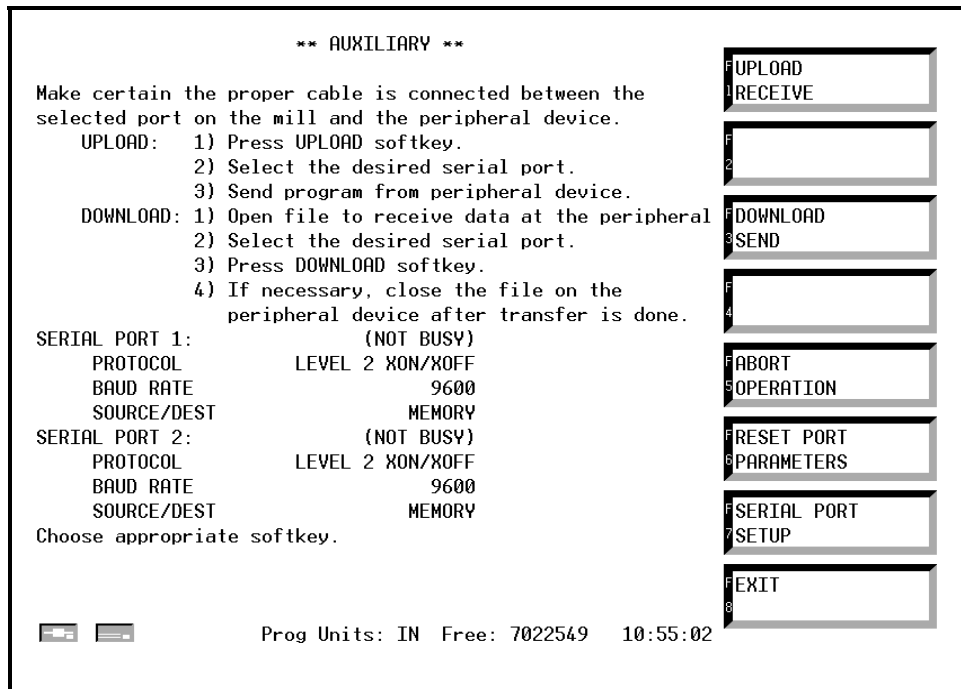
Figure 2-38.   NC Upload/Download Screen

Defaults

The Reset Port Parameters (F6) softkey resets the communication settings of the displayed serial port to the following initial default values for both ports:

Communication Setting (Field)	Conversational	NC
Protocol	XON/XOFF	XON/XOFF
Baud Rate	9600	9600
Character Length	8	NA
Stop Bits	1	NA
Parity	None	NA
Source/Dest	NA	Memory

Important

Before attempting to transmit data, be certain that the connected serial port at the control and peripheral port are identically configured.

Status Messages

The CNC identifies the status of the Upload/Download process on the Auxiliary screen.

- For Conversational programs, messages appear in a pop-up window.

- For NC programs, messages appear in the appropriate Serial Port field.

Conversational Message	NC Message	Description
No message	(Not Busy)	The Serial Port is inactive and has not been requested to transmit any data.
Waiting to Receive…	(Busy Reading Leader)	The Serial Port is active and is ready to receive data from the peripheral device; begin sending data from the peripheral device.
Receiving Data…	(Busy Loading)	The Serial Port is active and receiving data from the peripheral device.
Sending Data…	(Busy Sending)	The Serial Port is active and is sending data from the CNC to the peripheral device.
Waiting to Send…	(Waiting To Send)	The Serial Port is active and is waiting on the peripheral device to be ready to receive data before beginning to send data.

Protocol

The term protocol or *handshaking* refers to the method that data is transferred between two devices. Without protocols, the sender is permitted to send characters faster than the receiver can handle them. If so, the receiver's internal buffers would probably overflow causing characters to disappear. The Ultimax control uses two types of protocols:

- Hardware
- Software

Refer to the Appendix for descriptions of these protocols and the reasons to select each. An operator who wants to write communications utilities will find this information useful. Here are the Protocol Option softkeys:

- RS491 Level 1 CTS/RTS (F1)
- RS491 Level 2 XON/XOFF (F2)
- RS491 Level 3 Full Handshake (F3)

Baud Rates

You may choose from any of seven baud rates (2400, 4800, 9600, 19200, 38400, 56000, and 115200). Select the proper baud rate for the connected equipment using the Baud Rate field's softkey menu.

Important

The 9600 baud is the default setting for the control and software. For most applications, this is an adequate speed.

The baud rates for the remote device and the control must match.

Source/Destination

For NC Upload/Downloads, select the control program source (for sending/downloading) or destination (for receiving/uploading) by advancing the cursor to the Source/Dest field of the port description and choosing Memory or Disk File.

Changes to Serial Port Settings

Follow these steps to change Serial Port settings for each port:

1. From the Upload/Download menu, press the Serial Port Setup (F7) softkey.

2. Use the Arrow keys or press Enter to move the cursor through the fields. Enter data by selecting the appropriate softkey for each field.

3. Move the cursor to the Protocol field and select the appropriate communication protocol softkey.

4. Move the cursor to the Baud Rate field and select the baud rate softkey for the port.

5. For conversational Upload/Download, select the Character Length, Stop Bits, and Parity by entering the appropriate value in each field.

   - Follow the directions on the screen to perform an Upload or Download operation.
   - Refer to the appropriate section that follows for additional details.

   In addition, for NC Upload/Download, select the control program source (for sending/downloading) or destination (for receiving/uploading) by advancing the cursor to the Source/Dest field of the port description. Select the appropriate softkey: Memory (F1) or Disk File (F2).

6. Press the Exit (F8) softkey to redisplay the previous Auxiliary screen softkey options.

Receive or Send a Part Program

The Upload Receive (F1) and the Download Send (F3) softkeys initiate the appropriate Upload/Download operation. Follow the instructions on the screen corresponding to the enabled softkey to enable the transfer. Be certain to identify the correct serial port number. A message is displayed indicating the status of the selected operation.

To stop the transfer process, press the Abort Operation (F8) softkey.

To determine the current mode (NC or Conversational), press the console Auxiliary key and check the Editor Type field. If that field currently indicates that the control is in Conversational mode, press the Change Editor (F6) softkey. Then the Editor Type field shows that the control is in NC mode.

The program being loaded into the control's memory buffer may be machined while it is loading (*drip feed*). The file may be received from a remote device through a serial port and saved to the control's hard disk.

While an NC program is being read or written in memory, an "R" (reading) or a "W" (writing) icon appears in the screen's status line. Additionally, the screens contain icons of storage devices, which light when they are being accessed. Refer to the blown-up section of a sample screen below:

Figure 2-39.   Disk Activity Indicators and Read Icon for NC Programs

When a read or write activity is completed, a confirmation message will appear.

Transfer from Memory to Peripheral Device

Follow these steps to send a copy of a conversational or NC program in memory to a remote device via a serial port:

1. Prepare the remote equipment to receive the program sent by the control. Start the remote equipment's communication software so that it is ready to receive the program when the control sends it.

2. Make sure the desired program is currently in memory.

3. Press the console Auxiliary key.

4. Press the Upload/Download (F3) softkey.

5. Set up the port parameters.

6. For NC, set the Source/Dest parameter to Memory.

7. Start the transfer by pressing the Download/Send (F3) softkey.

8. Press the appropriate softkey for the correct sending port. The control automatically starts sending the program.

   - The Status Message appears.

   - Upon completion, the system displays the "NC Part Program Written" message at the bottom of the screen and the Status Message changes upon completion.

Transfer from Current Drive to Peripheral Device

To send an NC program from the current drive to a peripheral device through the serial port, follow these steps:

1. Prepare the remote device to receive the program sent to the control.

2. Display the Download/Upload facility from the Auxiliary screen.

3. Setup the port parameters. Refer to the "Setup" section of this chapter.

4. For NC, set the Source/Dest parameter to Disk File.

5. Start the program save by pressing the Download/Send (F3) softkey.

6. Press the appropriate softkey to select the correct port to send the file.

7. Select the disk drive if the current device does not contain the program to be sent.

    a. Press the Select Current Storage Drive (F6) softkey to display the drive selection softkeys.

    b. Select the appropriate device by pressing its softkey.

8. Select the appropriate directory if the current directory does not contain the program to be sent.

    a. Use the Arrow keys to position the cursor on the "Parent" (root) directory.

    b. Press the Change to Directory (F1) softkey to select the root directory and display the directory contents.

    c. Use the Arrow keys to position the cursor on the proper directory.

    d. Press the Change to Directory (F1) softkey to select the directory and display the directory contents.

9. Select the file to be sent by highlighting it with the Arrow keys and pressing Enter.

    - The program automatically downloads at this point.
    - The message next to the sending port number changes to "Busy Sending" and then to "Not Busy" when complete.

Transfer from Peripheral Device to Current Drive

Follow these steps to load NC programs from a peripheral device through the serial port to a disk file:

1. On the Auxiliary screen, press the Download/Upload (F3) softkey.

2. Set up the port parameters. Refer to the "Setup" section of this chapter.

3. Make sure the Source/Dest parameter is set to Disk File.

4. Press the Upload Receive (F1) softkey.

5. Press the appropriate softkey to select the correct port to receive the program.

6. If needed, select the disk drive to receive the program.

    a. Press the Select Current Storage Drive (F6) softkey to display the drive selection softkeys.

    b. Select the appropriate device by pressing the softkey.

7. If needed, select the directory to receive the program.

    a. If needed, use the Arrow keys to position the cursor on the "Parent" (root) directory.

    b. Press the Change to Directory (F1) softkey to select the root directory and display the directory contents.

    c. Use the Arrow keys to position the cursor on the proper directory.

    d. Press the Change to Directory (F1) softkey to select the directory and display the directory contents.

    e. Press Enter to display the Auxiliary screen.

8. Input the filename using one of the following methods:

    - Type the filename using the numeric keypad.
    - Use the Letter Selection screen, when activated.
    - Type the filename using the computer keyboard.

9. Press the Enter key to create the filename.

10. The Auxiliary screen redisplays.

11. The message next to the receiving port number changes from "Not Busy" to "Busy Reading Leader."

12. Send the file from the remote device.

13. The program is sent to the disk drive at this point. The message next to the receiving port number changes to "Busy Loading" and then to "Not Busy" upon completion.

Transfer from Peripheral Device to Memory

Follow these steps to load Conversational or NC programs into the control's memory buffer from a peripheral device via the serial port:

1. Press the Download/Upload (F3) softkey.

2. Set up the port parameters for the appropriate port.

3. Set the Source/Dest parameter to Memory.

4. Start the program loading operation by pressing the Upload Receive (F1) softkey.

5. Press the appropriate softkey to select the correct port to receive the program.

   a. If there is currently a program in memory, the system prompts "Should the current NC file in memory be deleted?"

   b. Press the Yes (F2) softkey to continue. Replying Yes to this message will delete the current NC file from memory only; it will not delete it from a hard drive or diskette if it has been saved previously.

   c. The Status Message appears.

6. Start sending the program to the control by initiating data transfer from the peripheral device.

   - The control takes the program information as it becomes available and assembles it in the NC memory buffer.

   - The Status Message changes upon completion.

RS232

RS232, or RS232C as it is by now, is a standard for serial data communications. There are other serial data standards other than RS232, but 232 has been around so long that an understanding of it is required when considering serial data communications. Special cable and precautions must be used for effective transfer. Hurco recommends the following:

1. SIX CONDUCTOR SHIELDED LOW CAPACITANCE CABLE. (buy cable designed for this purpose) THE HURCO PART NUMBER FOR SIX WIRE RAW CABLE IS **406-1244-002.**

2. 22 GAUGE STRANDED WIRE, AS THE MINIMAL WIRE SIZE.

3. 50 FEET OR LESS CAN BE SINGLE SHIELDED, OVER 50 FEET MUST BE MULTI- SHIELDED.

4. A MAXIMUM OF 100 FEET BUT 80 FEET OR LESS IS BEST. If the cable is made longer than 100 feet, expect to have to reduce the Baud rate.
   If the customer needs to communicate over greater distances, use RS422 converters.

5. DO NOT RUN THE CABLE NEAR FLUORESCENT LIGHTING FIXTURES OR OTHER POSSIBLE ELECTRICAL NOISE SOURCES SUCH AS: POWER CABLES, CONTACT- ORS, TRANSFORMERS, ETC.

6. THE CABLE SHIELD(S) SHOULD BE CONNECTED AT ONLY ONE CONNECTOR, <u>DO NOT CONNECT CABLE SHIELDS TO BOTH</u> <u>CONNECTORS.</u> CONNECT SHIELD(S) TO FRAME GROUND OF THE HURCO MACHINE ONLY.

TERMS AND CONNECTIONS:

The speed of serial data transfer is referred to as BAUD RATE. For purposes of this paper, baud rate is same as bits-per-second. Commonly used baud rates are: 300, 600, 1200, 2400, 4800, 9600, and 19,200.
Divide the baud rate into the number one to get bit time.

RS232 is asynchronous (not synchronous) serial data transfer meaning that framing information is sent along with each character transmitted to tell the receiver how to reconstruct the character. This framing information is overhead that costs time but allows for a simpler system than synchronous serial data transfer. The format for RS232 requires that every character transmitted be preceded by one low START BIT. After the start bit comes the DATA WORD, typically seven or eight bits of information. Although any mutually acceptable seven or eight bit code might be used here, seven bit ASCII is a widely used standard code for information interchange. Use seven bits unless the hardware requires eight. An ASCII chart is given later. ISO code is ASCII with an even parity bit appended. EIA is ASCII with an odd parity bit appended.

Following the seven or eight bits of character information, an optional PARITY BIT may be used. The parity bit is a system of error detection. If the parity bit option is used, it operates for each character transferred. So if a "parity" error occurs, it means that a character was not transferred properly. Most devices allow choices of "No Parity", "Even" or "Odd".

Following the optional parity bit, and the last part of the complete word, are HIGH STOP BITS. Most machines allow the operator to choose either 1, 1 1/2 or 2 stop bits to be used.

If a FRAMING ERROR occurs, it means that the receiver was not able to properly "frame" the data word with the one low start bit and the preselected number of high stop bits.

All of this, and more, is designed into and taken care of by the integrated circuit used as the serial transmitter/receiver port. USART is the acronym for Universal Synchronous Asynchronous Receiver Transmitter, which is one such IC used as a serial port.

DTE and DCE are designations that seem to confuse more than help. DTE stands for data terminal equipment. DCE stands for data communications equipment, typically a modem. Hurco machines are DTE devices. Computers are DTE devices. Peripheral equipment, such as printers and tape readers are usually DCE but might be either. Before interfacing a Hurco machine or any computer with another piece of equipment, find out if the equipment is DCE or DTE because different connections are used between DTE and DTE than are used between DTE and DCE.

DTE devices output signals on TXD and RTS and receive signals on RXD and CTS. DCE devices output signals on RXD and CTS and receive signals on TXD and RTS. DTE devices are interconnected with a cable that has TXD at one end connected to RXD at the other end and RTS at one end connected to CTS at the other end. DTE to DCE requires a two-to-two, three-to-three, etc., cable. Both cables should have the shield(s) connected to frame ground at the machine only and not connected to anything at the other end.

Following is a definition of RS232 signals for both DTE and DCE:

  Frame Ground is self explanatory and should be connected to the shield(s) of the cable and left unconnected at the peripheral end. Frame ground is connected to the machine frame ground which should also be connected to earth or power bus ground.

  TXD or Transmit Data is the line on which a DTE device transmits data. However, a DCE device receives data on this line. TXD of a DTE device should be connected to RXD of a DTE device but it should be connected to TXD of a DCE device.

  RXD or receive Data is the line on which a DTE device receives data. However, a DCE device transmits data on this line. RXD of a DTE should be connected to TXD of a DTE device and RXD of a DCE device.

  RTS is the line on which a DTE device instructs the other device to get ready to receive data. (Data can then be transmitted after the clear-to-send signal has been received from the peripheral device provided the equipment has been set up for CTS/RTS handshaking)
  RTS of a DTE should be connected to CTS of a DTE device and RTS of a DCE device.

  CTS is the line on which a DTE device receives a "clear-to-send" signal from another device. This input line is used by the other device to indicate when it is ready to receive data. CTS of a DTE should be connected to RTS of a DTE device and CTS of a DCE device.

Signal Ground establishes the common ground reference potential for all the
interface lines and should always be connected between devices regard|ess of
whether they are DTE or DCE.

DTR, not used on Hurco BX controls, is Data-Terminal-Ready. This output line is
used by the machine control to indicate that it is ready for communications. If
used, connect it to DSR of a DTE device and DTR of a DCE device. DTR can be used
to switch on a modem.

DCD, used only on Max 3 controls, is Data-Carrier-Detect. This line is an output
from modems to tell the control that the modem has received a signal that it
recognizes as a valid modem signal. Expect to use this signal only if you are
connecting a modem to the control.
It's an input to the control and all DTE devices.

DSR, used only on Max 3 controls, is Data-Set-Ready. This line is an output from
modems (DCE devices) t• tell the control that the modem is ready to go. It
doesn't mean that the modem is wanting to send data but that the modem is
functionally ready. It is an input to the control and all DTE devices.

RI, used only on Max 3 controls, is Ring Indicator. This line is an output from
modems to tell the control that the phone is ringing, there is an incoming call.
It's an input to the control and all DTE devices.

The RS232 signals used by Hurco conform to the RS-232-C standard. Not all standard
RS-232-C signals are used on all controls. See those listed below. Some peripheral
devices may have provisions for RS-232-C control signals that are not available from
the control, however, such devices can usually be adapted to these ports.

If hardwired handshaking isn't required, it will not be necessary to use 4, 5, 6, 8,
& 20. If you are having difficulty transferring data and suspect handshaking
problems, try jumping CTS to RTS and DTR to DSR to DCD (4 to 5, also jumper 6 to 8
to 20) at each connector to disable hardware handshaking. If you can now transfer
data, the problem was in the hardware handshaking using these signals. Try low baud
rates first, increasing only after successful data transfer occurs.

<div align="center">

RS232 PINOUTS:

</div>

Pinouts are shown in the following chart, earlier Hurco machines used the Mil Circle
connector, later mAX 2 machines use the D25 connecter. Max 3 & 32 machines use a
D9 connector. Most computers will use either the D9 or D25 connector.

<div align="center">

PINOUTS

</div>

FUNCTION	MIL CIRCLE	D9	D25	DTE	DCE
Frame ground	A		1	Frame gnd	Frame gnd
Transmit Data, TXD	C	3	2	Output	Input
Receive Data, RXD	B	2	3	Input	Output
Request To Send, RTS	E	7	4	Output	Input
Clear To Send, CTS	D	8	5	Input	Output
Signal ground	F	5	7	Sig ground	Sig ground
Data Terminal Ready, DTR	G*	4	20	Output	Input
**Data Carrier Detect, DCD		1		Input	Output
**Data Set Ready, DSR		6		Input	Output
**Ring Indicator, RI		9		Input	Output

*Not in BX machines

<div align="center">

**Only in Max 3 machines

</div>

<p style="text-align:center">**CABLES**</p>

```
        MACHINE                                          COMPUTER
MIL     DTE/FUNCTION                                 D25      DTE
 A Frame ground, shield(s) machine end only
 C TXD, Transmit Data, output     --------------->   3 RXD, Input
 B RXD, Receive Data, input       <-------------     2 TXD, Output
 E RTS, Request To Send, output   --------------->   5 CTS, Input
 D CTS, Clear To Send, input      <-------------     4 RTS, Output
 F Sig ground  -------------------------------- 7    Sig ground


        MACHINE                                          COMPUTER
 D25     DTE/FUNCTION                                 D9       DTE
1 Frame ground, shield(s) machine end only
2 TXD, Transmit Data, output      --------------->   2 RXD, Input
3 RXD, Receive Data, input        <-------------     3 TXD, Output
4 RTS, Request To Send, output    --------------->   8 CTS, Input
5 CTS, Clear To Send, input       <-------------     7 RTS, Output
7 Sig ground  -------------------------------- 5     Sig ground


        MACHINE                                          COMPUTER
D25     DTE/FUNCTION                                 D25      DTE
 1 Frame ground, shield(s) machine end only
 2 TXD, Transmit Data, output     --------------->   3 RXD, Input
 3 RXD, Receive Data, input       <-------------     2 TXD, Output
 4 RTS, Request To Send, output   --------------->   5 CTS, Input
 5 CTS, Clear To Send, input      <-------------     4 RTS, Output
 7 Sig ground  ------------------------------- 7     Sig ground


        MACHINE                                          COMPUTER
D9      DTE/FUNCTION                                 D9       DTE
 3 TXD, Transmit Data, output     --------------->   2 RXD, Input
 2 RXD, Receive Data, input       <-------------     3 TXD, Output
 7 RTS, Request To Send, output   --------------->   8 CTS, Input
 8 CTS, Clear To Send, input      <-------------     7 RTS, Output
 5 Sig ground  -------------------------------- 5     Sig ground Connect to
        shield(s)machine end only
```

 A Null-Modem adapter must be used between computer and machine with a purchased
 cable that is wired 2-2, 3-3, 4-4, etc. Such a cable is designed for machine or
 computer to DCE devices, usually a printer or modem.

B or BX Control Tape Drive Emulator
Cable's

Note: On the Tape Emulator, the DB9 Male Plug is not wired as a Standard 9 pin
Communications Port.   Therefore follow the cabling scheme below to fabricate a cable
from the Machine (DB9 Male) to your Computer (DB9 or DB25)

Machine **DB9** and your Computer with a **DB9**, use the cable diagram below!


```
        MACHINE                                                    COMPUTER
D9  Pin (M)                                               D9   Pin (F)
 2 TXD, Transmit Data ------------------------------>   2 RXD
 6 RXD, Receive Data  <-----------------------------    3 TXD
 7 DTR, Data Terminal Ready ----------------------->    4 DTR
 3 Sig ground  ------------------------------------    5 Sig ground  Connect to
shield(s)machine end only
```


Machine **DB9** and your Computer with a **DB25**, use the cable diagram below!


```
        MACHINE                                                    COMPUTER
D9  Pin (M)                                               D25  Pin (F)
 2 TXD, Transmit Data, output    ------------------>   2 RXD
 6 RXD, Receive Data, input      <-----------------    3 TXD
 7 DTR, Data Terminal Ready ----------------------> 20 DTR
 3 Sig ground  ------------------------------------    5 Sig ground Connect to
      shield(s)machine end only
```

USING RS232 WITH THE HURCO:

Both pieces of equipment must be set up exactly alike regarding: baud rate, number of data bits, parity, and number of stop bits. If handshaking is to be used, both should be set up for the same mode of handshaking. If hardware handshaking is to be used, then RTS and CTS as mentioned in the previous paragraph will be used. Software handshaking does not use any signal lines other than the data lines. Data transfer might still occur even if the two pieces of equipment are not set up alike regarding handshaking, however, if you intend to use handshaking, then both must be set up alike for that mode of operation.

Hardware handshaking uses RTS and CTS. It might be partial handshaking where a computer monitors only CTS and sends data only when a "clear to send" is received or it might be complete handshaking where one device sends an RTS, "request to send" and waits until a CTS is received to send data.

Software handshaking does not use RTS and CTS but involves automatically monitoring the data lines for special control codes that
regulate the transfer of data. The BX NC setup allows use of RTS/CTS or combinations of four software control codes; DC1, DC2, DC3 and DC4. Ultimax allows use of RTS/CTS or DC1 and DC3 as desired. Commonly used Software Handshaking codes are:

DC1 = Control Q, ^Q = XON = Reader On = 11 as hexadecimal ASCII
DC2 = Control R, ^R = Tape or Punch On = 12 as hexadecimal ASCII
DC3 = Control S, ^S = XOFF = Reader Off = 13 as hexadecimal ASCII
DC4 = Control T, ^T = Tape or Punch Off = 14 as hexadecimal ASCII
Some people use the expression "Mode 1, 2 or 3" when referring to various forms of software handshaking, others may say "Level 1, 2 or 3. It is best to find out how the particular manufacturer defines "Mode X" or "Level X" before trying to use software handshaking. Mode 1, or Level 1, is typically hardwired handshaking using RTS and CTS. Modes 2 & 3, or Levels 2 & 3, use software handshaking.

WITH THE HURCO BX CONTROL:

The BX NC manual defines Mode 1 or Level 1 as using RTS to signal when the control can receive data. For sending data, it monitors CTS and sends when the other unit signals a clear to send. It waits while RTS is off.

Mode 2 or Level 2 is defined as a partial handshake as follows: Wait five seconds for DC2, ^R, if not received, send DC1, ^Q, and wait for data. When the buffer is filled less 20 characters, send DC3, ^S, to halt transmission. Send DC1, ^Q, when once again ready to receive. **For transmitting:** Wait five seconds for DC1, ^Q, if not received, send DC2, ^R, <u>and</u> data. Transmission stops when DC3, ^S, is received and resumes when DC1, ^Q, is received.
Mode 3 or Level 3 is defined as full handshake as follows: Wait five seconds for DC2, ^R,if not received, send DC1, ^Q, every three seconds and wait for data. When the buffer is filled less 20 characters, send DC3, ^S, to halt transmission. Send DC1, ^Q, when once again ready to receive. For transmitting: Wait five seconds for DC1, ^Q, if not received, send DC2, ^R, every three seconds until a DC1, ^Q, is received. Transmission is stopped when DC1, ^Q, is received.

If set up for XON/XOFF, the machine will send data until the other device responds with a DC3, ^S, then data will stop. If the other device later sends a DC1, ^Q, data transfer resumes. XON/XOFF also works when the Hurco is set up in the FULL HANDSHAKE mode.

The following information is from the Hurco Special Executive NC manual, 704-0001-230, beginning on page 30:

From the menu shown below, any device can be connected to either port and set to either mode 1, 2 or 3. The cursor is advanced through both columns using the ENTER/ADV key.

DEVICE	CONNECTOR	MODE
READER	2	1
PUNCH	1	1
PRINTER	1	1
REMOTE	2	1

   Read modes are defined as:

Mode 1 (RS-491, Level 1) The control sets RTS on when it can accept data and resets RTS off when it cannot. This is a hardwired function only. Mode 1 works with RS-241 (EIA odd parity) and with RS-358 (ASCII even parity).

Mode 2 (RS-491, Level 2, Limited Handshake) The control waits five seconds for DC2, ^R. If no DC2, ^R, is received, the control sends DC1, ^Q, and then waits for data. When the control can handle only a few more characters (20), it sends DC3, ^S, to suspend transmission. The control will send DC1, ^Q, to resume transmission when appropriate. This mode operates with RS-358 only and is modem compatible.

Mode 3 (RS-491, Level 2, Full Handshake) The control waits five seconds for DC2, ^R. If no DC2, ^R, is received, the control then sends DC1, ^Q, every three seconds until data is received. The control will send a DC3, ^S, to suspend transmission and a DC1, ^Q, to resume. Mode 3 operates with RS-358 only and is modem compatible. This mode is preferred for some remote computers.

   Punch/Print Mode:

Mode 1 (RS-491, Level 1) The control sends data while the CTS line from the peripheral is on. It waits while RTS is off. This is a hardwired connection only.

Mode 2 (RS-491, Level 2, Limited Handshake) The control waits five seconds for DC1, ^Q. If no DC1, ^Q, is received, the control sends DC2, ^R, and then immediately sends data. Transmission is suspended by the control if a DC3, ^S, is received. The control resumes transmission when a DC1, ^Q, is received. This mode operates with RS-358 only and is modem compatible. Mode 2 is suitable for many readers, punches, printers, and remote units.

Mode 3 (RS-491, Level 2, Full Handshake) The control waits five seconds for a DC1, ^Q. If no DC1, ^Q, is received, the control sends DC2, ^R, every three seconds until a DC1, ^Q, response is received. The control then sends data. Transmission is suspended by the control if a DC3, ^S, is received.

The control resumes transmission when a DC1, ^Q, is received. This mode operates
with RS-358 only and is modem compatible.

When printing, 600 baud, 8 data bits, no parity, and 1 stop bit is recommended,
however, be sure the Hurco and the printer are set up alike. 600 baud, 7 data bits,
even parity, and 1 stop bit are recommended for punch. When interfacing with a
computer, use the same setup as for a punch except adjust the baud rate upward.

The following information is from Vic Baker's memo June 29, l990 titled **"Downloading
From Max PC to BX Control". Setup with 7 data bits, 1 stop bit and even parity**.

If you have pins 4 and 5 connected to D and E respectively, you can use hardwired
handshake. Use Mode 1 on the BX and Mode 1 on the PC. I tested this with a file
bigger than the buffer and this also works fine.

> Although the BX manual infers that Modes 2 and 3 are the same as 2 and 3 on Max and
> Max PC, this is not the case. To send XON/XOFF mode, you must setup the mill first
> to receive using Mode 3, then send from the PC using Mode 2. Again, I used a file
> bigger than the buffer and XON/XOFF worked fine.

I could not get Mode 2 on the mill to work at all; neither could I download using
Mode 3 on the PC.

Another point worth knowing is, although the BX is setup by default to receive data
in Port 2 and send data from Port 1, you can use Port 1 to receive data. There is,
in effect, no difference between the ports.

   Test 1:

Mill set to Mode 1 (RTS/CTS) and PC set to Mode 1 (RTS/CTS). Set up mill first,
i.e., press "Next Block". PC says "Sending" and Mill fills buffer to about 21000
characters. PC display does not change when buffer is full. Clear buffer at mill
and PC starts sending again. PC says "Download complete". Do an "Item Search" for
M2 and find it at the end of the program. You can use RTS/CTS and setup either mill
or PC first.

   Test 2:

Mill set to Mode 3 and PC set to Mode 2 (XON/XOFF). Set up mill first, i.e., press
"Next Block". Now send tape file from PC. Mill fills buffer to about 21000
characters. PC displays "XOFF received. Waiting for XON...." Clear buffer at mill
and PC says "XON received. Sending..." and starts sending again. PC says "Download
complete". Do an "Item Search" for M2 and find it at end of program.

Any other tests I ran failed. I could not get BX Mode 2 to work at all. It
generally said "Transmission Error." I could not get PC Mode 3 to work with BX Mode
3 either.

   END OF MEMO

INTERFACING WITH A HURCO ULTIMAX CONTROL:

REGARDING 8.0 NC: (Reference: Beginning on page 3-11 of 8.0 NC Instruction Manual)

Level 1; when receiving a program from a computer using RTS/CTS handshaking and before the computer is connected, the machines CTS line is inactive. Once the computer is connected, its RTS line (tied to machines CTS line) goes active. When the control is ready to receive, it signals the computer by making its RTS line (tied to the computers CTS line) active. The computer sees this and sends data. When transmission is over, the machine carries its RTS line, connected to the computers CTS, back to inactive. When transmitting a program to a computer using RTS/CTS handshaking, the machines RTS line is made active. The machine will then send data provided the computers RTS line (tied to machines CTS line) is active. When transmission is over, the machine carries its RTS line, connected to the computers CTS, back to inactive.
Level 2, when receiving a program from a computer using XON/XOFF, the machine signals with XON, ^Q, every 250 ms. until it receives data. Once transmission begins, the machine can stop transmission using XOFF, ^S, and restart using XON, ^Q,. Notice that the computer doesn't have to send any control information just data. When transmitting a program to a computer, the machine sends DC2, ^R, every 250 ms. for five seconds, expecting XON, ^Q, from the computer. If the machine receives an XON, ^Q, or the five second period passes, it sends DC2, ^R, followed by data.

Level 3 differs from Level 2 only in that there is no five second time period when transmitting a program, the machine sends DC2, ^R, every 250 ms. until it receives XON, ^Q, from the computer. When DC1, ^Q, is received, DC2, ^R, is sent followed by data.

INTERFACING THE HURCO WITH A GRECO MINIFILE:

When using the Greco with a machine having BX control:

   Set up the Greco this way:

     49#   Selected channel   0#   Channel 0 active
     47#   Channel 0 protocol   7#   Punch/reader
     43#   Channel 0 baud rate   9600# 9600 baud
     42#   Store Parameters   Wait till display clears

   Set up the Hurco this way:

   ON THE CPU PCB, SET THE SHUNT FOR TRACK #1 TO 9600 BAUD

   1. Press ENTER to display ENTER MODE MENU
   2. Press "6" to select OTHER
   3. Press "2" to select IN/OUT CONNECTOR SELECT
4. Position cursor on REMOTE and CONNECTOR column and change to value to a "1" (top connector #1)
5. Position cursor on REMOTE and MODE column and change the value to a "2" (RS491, Level 2, Limited Handshake)

To send a file from the Greco to the BX control:
1. On the Greco enter 1#, followed by a label, followed by a #.
2. FILE OPEN LED will flash.
3. On the Hurco:
   Press ENTER to display ENTER MODE MENU.
   Press "4" to select SAVE PROGRAM
   Press "3" to select REMOTE
   Press "0" to select PUNCH COMPLETE TAPE
4. ACTIVITY LED lamp on the Greco will flash.
5. Data transfer is complete when the ACTIVITY LED quits flashing.
6. Enter "3" at the Greco to close the file.

To send a file from the BX control to the Greco:
1. On the Greco enter 7#, followed by a label, followed by a #.
2. FILE OPEN LED will flash.
3. On the Hurco:
   Press ENTER to display ENTER MODE MENU.
   Press "3" to select LOAD PROGRAM
   Press "3" to select REMOTE
   To start loading, press NEXT BLOCK
4. ACTIVITY LED lamp on the Greco will flash.
5. Data transfer is complete when the FILE OPEN and ACTIVITY LEDs quit flashing.
6. Enter "3" at the Greco to close the file.

When using the Greco with a machine having an ULTIMAX control:

GRECO	GRECO SYNTAX
Channel 1 selected	"49", "#", "1", "#"
XON/XOFF protocol	"48", "#", "1", "#"
9600 baud	"44", "#", "9600", "#"
7 data bits	"66", "#", "7", "#"
1 stop bit	"62", "#", "1", "#"
Even parity	"46", "#", "2", "#"
Send file #1	"2", "#", "1", "#"
Create and open file #1	"1", "#", "2", "#"
Close file #1	"3", "#", "1", "#"
Delete file #1	"98", "#", "2", "#"
Store parameters	"42", "#"
Examine parameters	"40", "#"
Erase one entry	*
Display directory	"5", "#", # advances
Text file type enabled	"51", "#"
Line feed output disabled	"52", "#"
Line feed converted to C return	"55", "#"

Set up the Greco this way:

```
49#  Selected channel        1#  Channel 1 active
48#  Channel 1 protocol      1#  XON/XOFF
44#  Channel 0 baud rate     4800#  4800 baud
42#  Store Parameters        Wait till display clears
```

Set up the Hurco this way:

Get in the AUXILIARY mode and set serial port to 4800 baud, XON/XOFF.

To send a file from the Ultimax to the Greco:

1. On the Greco enter 1#, followed by a label, followed by a #.
2. FILE OPEN LED will flash.
3. On the Hurco:
    Press SAVE PROGRAM under AUXILIARY then SAVE TO PORT 1 or 2 (whichever you're
     using) ACTIVITY LED lamp on the Greco will flash.
5. Data transfer is complete when the ACTIVITY LED quits flashing.
6. Enter 3# at the Greco to close the file.

To send a file from the Greco to the Ultimax:

1. On the Greco enter 2#, followed by a label, followed by a #.
2. FILE OPEN LED will flash.
3. On the Hurco:
     Press LOAD PROGRAM under AUXILIARY screen.
     Press LOAD FROM PORT 1 or 2, whichever you're using.
4. ACTIVITY LED lamp on the Greco will flash.
5. Data transfer is complete when the FILE OPEN and ACTIVITY LEDs quit flashing.
6. Enter 3# at the Greco to close the file.

Notes from the appendix of the Greco Mini-file manual regarding handshaking
protocols:

Argument 0 is no protocol which means that the Greco will transfer data without any
software handshaking.

Argument 1 is operation of the Greco using XON/XOFF. No ASCII character is required
to condition the Greco to receive data. It will receive data as soon as the DEFINE
FILE command has been given it.

Once the data transfer is completed, a CLOSE FILE command must be given the Greco.
When transmitting a file, the Greco will begin transmitting as soon as the file is
opened with a SELECT FILE command. No ASCII character is required to condition the
Greco to transmit data. The receiver may interrupt data transfer by sending DC3 and
restart data transfer by sending DC1.

Argument 5, RS491, Level 2, Slave, is defined as follows: When the Greco is told to
receive data following the DEFINE FILE command, it will wait for DC2. When it
receives a DC2, it will transmit a DC1 indicating "ready". The transmitter may
indicate the end of data transfer to the Greco by sending EOT followed by DC4. The
Greco will respond with DC3 and automatically close the file. If the transmitter
cannot transmit EOT, etc., the file must be closed manually with a CLOSE FILE
command. When the Greco is to transmit a file as indicated by the SELECT FILE
command, it will wait for DC1. When it receives a DC1, it will start transmitting.
The receiver may interrupt by sending DC3 and restart transfer by sending DC1.

Argument 9, RS491, Level 1, is defined as follows: When the Greco is told to
receive data following the DEFINE FILE command, it will wait to receive data from
the transmitter if CHANNEL 0 MODEM interface is used. If CHANNEL 1 TERMINAL
interface is used, DATA CARRIER DETECT must be active. After data transfer is
complete, a CLOSE FILE command must be performed. When the Greco is to transmit a
file as indicated by the SELECT FILE command, it will wait for the receiver to make

RTS active if CHANNEL 0 MODEM is used or for CTS to be made active if CHANNEL 1 TERMINAL is used. The Greco will transfer data based on what the listener does with RTS/CTS depending on whether channel 0 or channel 1 is used.

Argument 10, Fanuc, is defined as follows: When the Greco is told to receive data following the DEFINE FILE command, it will wait for DC2. The Greco will begin receiving data as soon as DC2 is received. When a DC4 is received, the Greco will close the file. If the transmitter cannot transmit a DC4, the file must be closed manually. When the Greco is to transmit a file as indicated by the SELECT FILE command, it will wait for DC1 from the receiver. When the Greco receives a DC1, it begins transmitting. The listener may use DC3 to stop and DC1 to restart data transfer. The minifile ends the file with NULL characters. OR PERHAPS ETX?????

Argument 7, Punch/Reader, is defined as follows: When the Greco is told to receive data following the DEFINE FILE command, it will wait for DC2. The Greco will begin receiving data as soon as DC2 is received. When a DC4 is received, the Greco will stop receiving data until it once again receives a DC2. In other words, DC2 and DC4 start and stop transfer. The file must be closed manually. When the Greco is to transmit a file as indicated by the SELECT FILE command, it will wait for DC1 from the receiver. When the Greco receives a DC1, it begins transmitting. The listener may use DC3 to stop and DC1 to restart data transfer. The minifile ends the file with NULL characters. OR PERHAPS ETX?????

Notes from the appendix of Ultimax II PC:

When sending programs to the Hurco machines, the transfer will be in ASCII format. Configure the computer serial port for seven data bits, one stop bit and even parity (these are default values). Default values for protocol are Level 2, XON/XOFF. Default value for baud rate is 9600. The first character of an NC file must be a "%" sign and the last character must be an "E". These characters, respectively, activate and close the file loading process.

Protocols available are: Level 1, RS491 CTS/RTS; Level 2, RS491 XON/XOFF; and Level 3 full handshake.

THE SEVEN BIT ASCII CODE IN HEXADECIMAL

NUL	00	6	36	k	6B	
SOH	01	7	37	l	6C	
STX	02	8	38	m	6D	
ETX	03	9	39	n	6E	
EOT	04	:	3A	o	6F	
ENQ	05	;	3B	p	70	
ACK	06	<	3C	q	71	
BEL	07	=	3D	r	72	
BS	08	>	3E	s	73	
HT	09	?	3F	t	74	
LF	0A	@	40	u	75	
VT	0B	A	41	v	76	
FF	0C	B	42	w	77	
CR	0D	C	43	x	78	
SO	0E	D	44	y	79	
SI	0F	E	45	z	7A	
DLE	10	F	46	{	7B	
DC1	11	G	47		7C	should be a vertical bar
DC2	12	H	48	}	7D	
DC3	13	I	49		7E	horiz S, means approximately
DC4	14	J	4A	DEL	7F	
NAK	15	K	4B			
SYN	16	L	4C			
ETB	17	M	4D			
CAN	18	N	4E			
EM	19	O	4F			
SUB	1A	P	50			
ESC	1B	Q	51			
FS	1C	R	52			
GS	1D	S	53			
RS	1E	T	54			
US	1F	U	55			
SP	20	V	56			
!	21	W	57			
"	22	X	58			
#	23	Y	59			
$	24	Z	5A			
%	25	[	5B			
&	26	\	5C			
'	27	]	5D			
(	28	^	5E			
)	29	_	5F			
*	2A	'	60	should be an ' turned backwards		
+	2B	a	61			
,	2C	b	62			
-	2D	c	63			
.	2E	d	64			
/	2F	e	65			
0	30	f	66			
1	31	g	67			
2	32	h	68			
3	33	j	69			
4	34					
5	35					

CONVERSION TO BINARY: Convert the two Hexadecimal digits to their binary equivalents, examples:

A = 41 = 100 0001   Bit zero is the right most digit

B = 42 = 100 0010

C = 43 = 100 0011

a = 61 = 110 0001   Bit seven is the left most digit

J = 4A = 100 1010

K = 4B = 100 1011

L = 4C = 100 1100   Bit eight is [3]

M = 4D = 100 1101   assumed low [3]

N = 4E = 100 1110 [3]

O = 4F = 100 1111 [3]

TROUBLESHOOTING:

When a simple Radio Shack RS232 line monitor is connected to a Hurco, TXD and RTS are inactive **(LEDs are red indicating a logic high with a_negative voltage between 3 and 25)** and all other LEDs are off. When sending from a Hurco, RTS becomes active **(LED is green indicating a_logic low with a positive voltage between 3 and 25)** and TXD flickers red and green until data transfer is completed. After data transfer ends, both RTS and TXD will become inactive (red) again. You should get the same results with your line monitor connected at the far end of a DTE to DCE cable. **However, if the cable is DTE to DTE, then CTS and** RXD will be the two LEDs that will light if the monitor is at the far **end of the cable.** It is possible, of course, for a breakout box to be built with any color of LED in any position, however, inactive is a negative voltage and active is a positive voltage. The transmitter should always indicate inactive as a voltage between negative five and negative twenty-five while active (we are talking about "active low") should always be between positive five and positive twenty-five volts. The receiver is designed to honor anything more positi~e than three as "low" and anything more negative than three as "high".

Here is a description of a problem encountered when running an 8.0 NC program that spans memory. The cable is minimum configuration three conductor cable with XON/XOFF. level 2 protocol had worked earlier using software version 7.24. In this setup, however, the program would randomly stop. The short term solution is to use five conductor cable and short pin 3 to pin 4 inside the cable connector. The problem is more susceptible using a Greco Box for uploading and at higher baud rates. The cause of the problem is that each of the software protocols (levels 2 and 3) were **incorrectly** monitoring the RTS/CTS signal. Levels 2 and 3 are not supposed to use these signals. However, since these signals were not connected because of the three conductor cable configuration, the cable leads inside the cabinet acted as antennae and generated enough voltage to overcome the signal threshold and generate a system interrupt which in turn stopped the axis controllers.