Intelligent Interface

Option Handbook

for

THE CHYRON

iNFiNiT!

Family of Products

An Infinite Freedom Component

Pub. No. 2A01989 ***February, 1999***
Revision C

CHYRON Corporation
5 Hub Drive
Melville, NY 11747
516-845-2000

CHYRON

FCC Rules and Regulations

WARNING

THIS EQUIPMENT GENERATES, USES AND CAN RADIATE RADIO FREQUENCY ENERGY AND IF NOT INSTALLED AND USED IN ACCORDANCE WITH THE INSTRUCTION MANUAL, MAY CAUSE INTERFERENCE TO RADIO COMMUNICATIONS. IT HAS BEEN TESTED AND FOUND TO COMPLY WITH THE LIMITS FOR A CLASS A COMPUTING DEVICE PURSUANT TO SUBPART B OF PART 15 OF FCC RULES, WHICH ARE DESIGNED TO PROVIDE REASONABLE PROTECTION AGAINST SUCH INTERFERENCE WHEN OPERATED IN A COMMERCIAL ENVIRONMENT. OPERATION OF THIS EQUIPMENT IN A RESIDENTIAL AREA IS LIKELY TO CAUSE INTERFERENCE IN WHICH CASE THE USER, AT HIS OWN EXPENSE, WILL BE REQUIRED TO TAKE WHATEVER MEASURES MAY BE REQUIRED TO CORRECT THE INTERFERENCE.

Canadian RFI Requirement

WARNING

THIS DIGITAL APPARATUS DOES NOT EXCEED THE CLASS A LIMITS FOR RADIO NOISE EMISSIONS FROM DIGITAL APPARATUS SET OUT IN THE RADIO INTERFERENCE REGULATIONS OF THE CANADIAN DEPARTMENT OF COMMUNICATIONS

LE PRESENT APPAREIL NUMERIQUE N'EMET PAS DE BRUITS RADIOELECTRIQUES DEPASSANT LES LIMITES APPLICABLES AUX APPAREILS NUMERIQUES DE LA CLASSE A PRESCRITES DENS LE REGLEMENT SUR LE BROUILLAGE RADIOELECTRIQUE EDICTE PAR LE MINISTERE DES COMMUNICATIONS DU CANADA.

Table of Contents

Section 1 - Introduction

Section 2 - Software Installation

Section 3 - Communications Setup

Section 4 - Operation

Section 7 - Telnet Operation

Appendix A - Error Responses

Appendix B - Reference Tables

Appendix C - Sample Commands

List of Figures

Section 1 - Introduction

Section 2 - Software Installation

Section 3 - Communications Setup

Section 4 - Operation

Section 5 - Serial Port Data Display

Section 6 - Embedded Commands

Section 7 - Telnet Operation

Appendix A - Error Responses

Appendix B - Reference Tables

Appendix C - Sample Commands

List of Tables

Section 1 - Introduction

Section 2 - Software Installation

Section 3 - Communications Setup

Section 4 - Operation

Section 5 - Serial Port Data Display

Section 6 - Embedded Commands

Section 7 - Telnet Operation

Appendix A - Error Responses

Appendix B - Reference Tables

Appendix C - Sample Commands

Section 1 - Introduction

1.1 Overview

Intelligent Interface is a program which allows a remote computer operator to send data such as election results or sports scores to an iNFiNiT! Family system for live display, via a serial data or telnet connection.

For example, information can be fed into Tab fields set in a Tab Description message. A Tab Description message is a page that contains Tab fields, laid out as they will be seen on air. For example, there can be two rows which contain the following: One Tab field to hold a team name, and nine more Tab fields to the right of it to hold scores for each inning of a baseball game. The information is displayed in the preset Tab fields, using fonts and colors already set on the system, or by sending font and color information from the remote computer.

Information can also be recorded by the remote operator using the Tab Description message, and recalled by the iNFiNiT! Family system operator for later display.

Intelligent Interface can also send commands to the system, which can set up and trigger effects execution for Transform, Read Effects, Mix Effects, and others.

Transform parameters can be remotely set for an effect, e.g. Rotation, Revolution, Warp, X and Y offset, etc. Intelligent Interface can then execute the Script can then be remotely executed.

1.2 Version Numbers

Intelligent Interface can be used on the following systems: iNFiNiT!/060 and /040; MAX!>/040; MAXINE!; MAX!> IMAGESTOR!; and MAXINE! IMAGESTOR!.

Message Compose version 10.0 must be installed in the system in order to use Intelligent Interface version 10.0.

1.3 Package Contents

All software for Intelligent Interface is contained on a ZIP™ disk, either supplied separately or with other programs (when supplied with an upgrade.)

Additional cables are necessary to connect your system to the host computer, if using the serial port control method. Since user applications vary widely, these cables are not supplied; however, they are easy to construct. Specifications for the cables are provided in Section 2 of this manual. *If using Telnet control, only a network connection is required.*

1.4 Software Description/Protection

The software is specially keyed to be used only with your system and no other. Once the software has been copied to your boot disk (typically drive C/), it is loaded with the Operating System during system boot. There is no program icon associated with Intelligent Interface in the Program loader menu.

1.5 CPU Transition Board (Serial Ports)

iNFiNiT! - The standard CPU Transition board of the system provides the required port for the Intelligent Interface system. **SERIAL PORT 4** is used for the iNFiNiT! - computer interface. The software allows raw data to be transmitted to the system as long as the system is powered up, and remains dormant in your unit until data commands are passed to it through the designated I/O port.

MAX!> - The software allows raw data to be transmitted to the system as long as the MAX!> is powered up, and remains dormant in your unit until data commands are passed to it through **SERIAL PORT 2.**

MAXINE! - The software allows raw data to be transmitted to the system through the **INTEL. INTF.** port as long as the MAXINE! is powered up.

Section 2 - Software Installation

2.1 General

This section describes installation of the option software, as well as related issues such as ZIP disk installation/ configuration, and backing up the system hard disk.

NOTE

It is not necessary to back up the system hard disk before installing the Option software - however, we recommend you back up any critical files before proceeding with software installation.

2.2 CHYRON Software Installation

The iNFiNiT! Family system software is on the enclosed bootable ZIP cartridge. *For information on installing/ configuring a ZIP disk drive, refer to* *§2.3 on page 2-6*. *For information on backing up the system hard disk, refer to* *§2.4 on page 2-10*.

[ ] 1. Turn the power on. Once the system has booted, the Program Loader menu (Figure 2-1) is displayed.

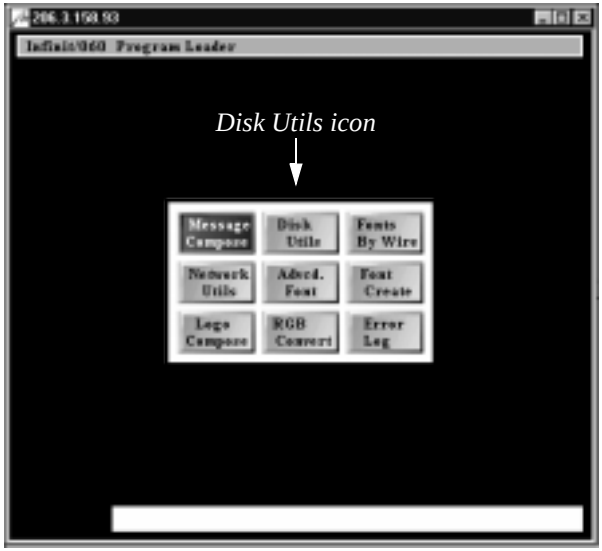


Figure 2-1. Program Loader Menu

[ ] 2. If not already highlighted, highlight and select the **Disk Utils** icon by either cursoring to the icon, then pressing ⏎; or by clicking on the icon. The Disk Utilities main menu (Figure 2-2) is displayed.

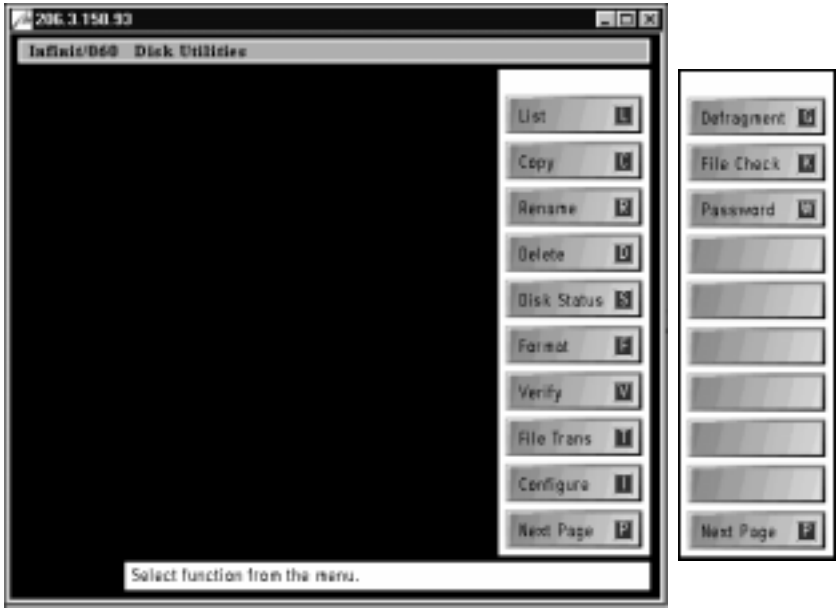


Figure 2-2. Disk Utilities Main Menu

[ ] 3. Press **C** or select **Copy** to display the COPY menu (Figure 2-3).

Figure 2-3. Disk Utilities COPY Menu

[ ] 4. In the fields at the top of the Disk Utilities COPY menu, enter the following information:

FROM: <Drive Letter A or **B>/**
TO: C/
FILE TYPE: ALL
CONFIRM: Y
VERIFY: Y
FILE NAME: *

It is recommended, though not necessary, that **CONFIRM:** be set to **Y**. If **Confirm** is set to **N**, **Copy** executes without confirmation. If **Confirm** is set to **Y**, any time **Copy** encounters a file with the same name on the **TO:** disk that has the same name as the file being copied, a pop-up window (such as in Figure 2-4) appears asking if the file on the **TO:** disk should be overwritten.

It is recommended, though not necessary, that **VERIFY:** be set to **Y**. Once a copy operation is

complete, **VERIFY** performs a bit-by-bit comparison of the original and copied files, to make sure that both versions are identical, thereby verifying the integrity of the copies. When **VERIFY** is set to **Y**, **Copy** execution takes slightly longer.

[ ] 5. Insert the first **Message Compose** disk into the appropriate drive. If software was supplied on ZIP, there is only one disk.

[ ] 6. Press **CNTL X** or select **Execute**. If a pop-up window (Figure 2-4) appears asking for overwrite permission, press **Y** to overwrite the file, or any other key to stop the overwrite. **Copy** execution will move to the next file.

To completely exit **Copy** execution, press **ESC**.

Figure 2-4. Overwrite Permission

[ ] 7. When **Copy complete.** appears on the Prompt Line, remove the disk.

[ ] 8. Press **CNTL Q** or select **Quit** to return to the Disk Utilities main menu.

[ ] 9. Press **MODE SLCT** to return to the Program Loader menu.

[ ] 10. Press **SET-UP** or click on the right side of the screen.

[ ] 11. Press **B or** select **Reboot**. *This step is mandatory!*

The software is now fully installed. Store all disks in a cool, dry place.

2.3 ZIP Disk Configuration

NOTE

A ZIP drive must be connected to the iNFiNiT! Family system's SCSI port using a 25 pin DSub female to 50 pin Centronics male cable. Or, an adapter can be used with the standard ZIP drive cable.

In order to install software from a ZIP drive it must be configured in Disk Utilities. Typically, ZIP drives are assigned drive **B** - this enables you to use the ZIP as a boot device if needed. If you have a Jaz, Bernoulli or external hard disk drive configured as drive **B**, you may install the ZIP using another drive letter, or delete the original drive using the Disk Utilities CONFIGURATION menu, then add the ZIP as drive **B**.

To add the ZIP drive:

[ ] 1. Once the system has been powered up as described above, enter Disk Utilities by selecting it from the Program Loader menu. The Disk Utilities main menu (Figure 2-7) is displayed.

[ ] 2. Press **I** or select **Configure**. The DISK CONFIGURATION menu (Figure 2-5) is displayed.

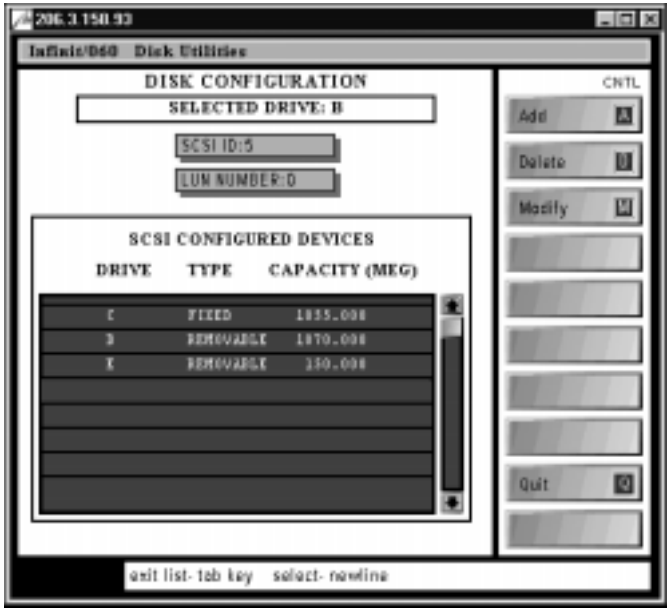


Figure 2-5. Disk Utilities DISK CONFIGURATION Menu

[ ] 3. If necessary, highlight the **SCSI ID:** field, then press the **SPACE BAR** or click on the field to toggle from **SCSI Express ID** to **SCSI ID**. *This step is necessary to avoid confusion, since SCSI Express uses a separate SCSI bus (path, channel. interface).*

[ ] 4. Enter the SCSI ID (**5** or **6**) of the ZIP drive in the **SCSI ID:** field. The **LUN NUMBER** should always be **0**.

[ ] 5. Press **CNTL A** or select **Add**. The system displays the following:

MOUNTING DRIVE - PLEASE WAIT

After the system has mounted the drive, a new entry appears in the drive configuration list.

To delete a drive:

[ ] 1. Turn the power on. Once the system has booted, the Program Loader menu is displayed:

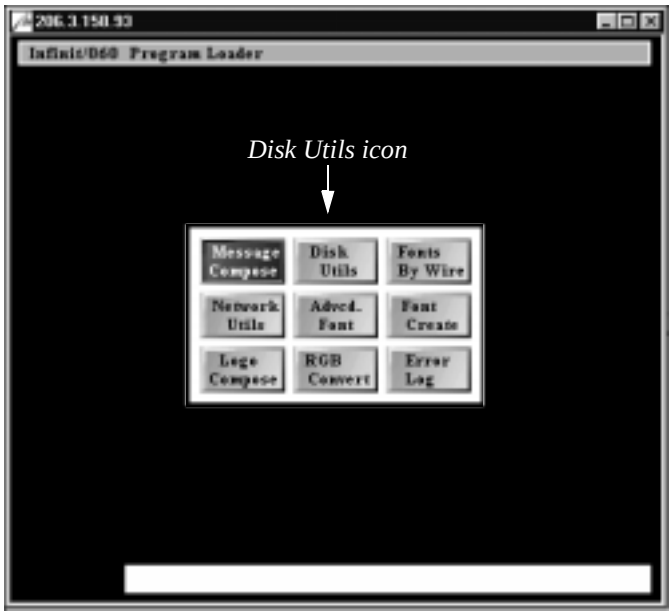


Figure 2-6. Program Loader Menu

[ ] 2. If not already highlighted, highlight and select the **Disk Utils** icon by either cursoring to the icon, then pressing ⏎; or by clicking on the icon. The Disk Utilities main menu is displayed:

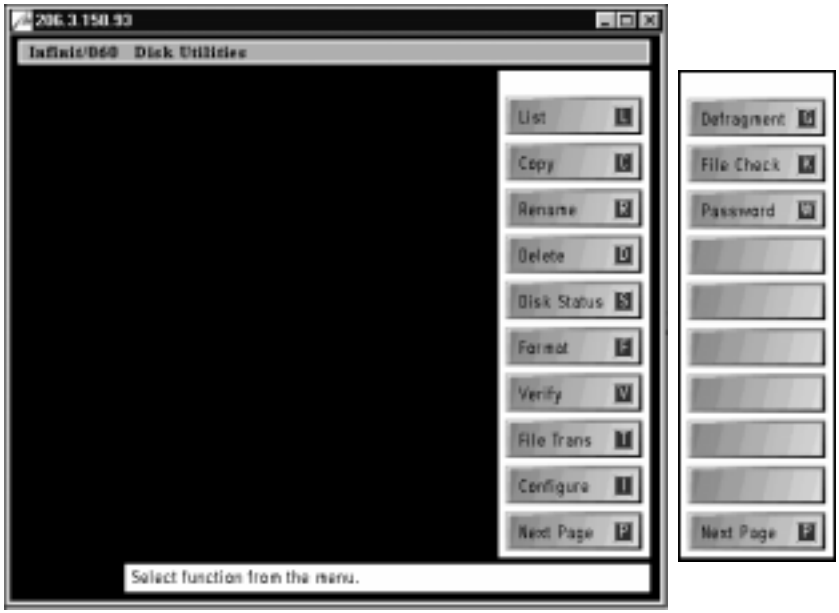


Figure 2-7. Disk Utilities Main Menu

[ ] 3. Press **I** or select **Configure**. The DISK CONFIGURATION menu is displayed:

Figure 2-8. DISK CONFIG. Menu - Drive Selected

[ ] 4. Place the cursor on the **B** drive entry, then press ⏎; or click on the **B** drive entry. An arrow appears to the left of the entry.

[ ] 5. Press **CNTL D** or select **Delete**. The drive is deleted from the disk configuration.

2.4 Backing Up the Hard Drive

Before copying the upgrade software to the hard drive, you can back up the hard drive to a removable medium such as a ZIP or Bernoulli disk.

[ ] 1. From the Disk Utilities main menu, press **C** or select **Copy**. The Disk Utilities COPY menu (Figure 2-9) is displayed.

Figure 2-9. Disk Utilities COPY Menu

[ ] 2. In the fields at the top of the Disk Utilities COPY menu, enter the following information:

FROM: C/
TO: <Drive Letter>/
FILE TYPE: ALL
CONFIRM: Y
VERIFY: Y
COPY NAMES: *

It is recommended, though not necessary, that **CONFIRM:** be set to **Y**. If **Confirm** is set to **N**, **Copy** executes without confirmation. If **Confirm** is set to **Y**, any time **Copy** encounters a file with the same name on the **TO:** disk that has the same name as the file being copied, a pop-up window (such as in Figure 2-10) appears asking if the file on the **TO:** disk should be overwritten.

It is recommended, though not necessary, that **VERIFY:** be set to **Y**. Once a copy operation is complete, **VERIFY** performs a bit-by-bit comparison of the original and copied files, to make sure that both versions are identical, thereby verifying the integrity of the copies. When **VERIFY** is set to **Y**, **Copy** execution takes slightly longer.

[ ] 3. Press **CNTL X** or select **Execute**. If a pop-up window (*Figure 2-10*) appears asking for overwrite permission, press **Y** to overwrite the file, or any other key to stop the overwrite. **Copy** execution will move to the next file.

To completely exit **Copy** execution, press **ESC**.

Figure 2-10. Overwrite Permission

Copy complete. appears on the Prompt Line when the backup is complete.

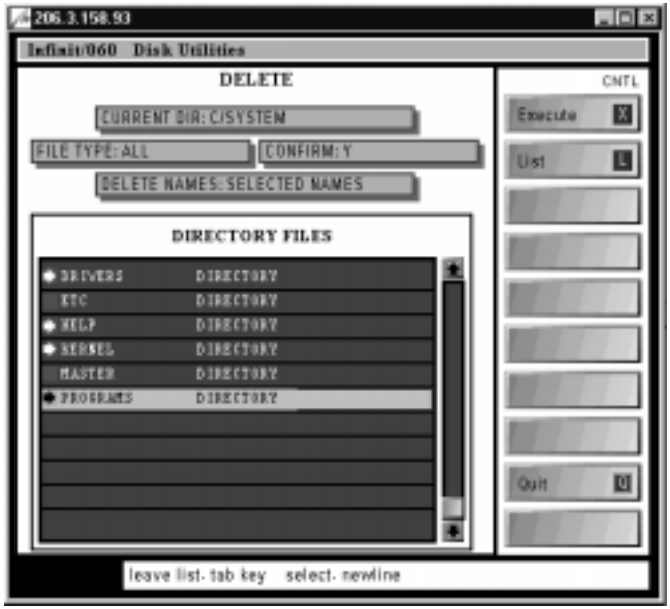


Figure 2-11. Directory Files List - Directories Selected

[ ] 4. Press **CNTL X** or select **Execute**. If **Confirm** is set to **N**, **Delete** executes without confirmation. If **Confirm** is set to **Y**, a pop-up window (Figure 2-12) appears asking if that directory should be deleted.

Figure 2-12. Delete Permission

Press **Y** to delete the directory, or any other key to stop the deletion. **Delete** execution will move to the next directory. To completely exit **Delete** execution, press **ESC**.

Section 3 - Communications Setup

3.1 Overview

This section describes the communications setup for the Intelligent Interface application, including instructions for cabling/setup for host-to-system communications via system serial port and also information about Telnet control.

NOTE

Intelligent Interface v10.0 (and later) supports Telnet control. Applications utilizing Telnet control do not require use of the iNFiNiT! Family system serial port.

3.2 Serial Port Installation/Setup

3.2.1 Selecting Serial Interface Protocol

The Intelligent Interface application may be controlled via either of two serial interface protocols. **RS-232** may be used for cable runs less than 50 feet. Use **RS-422** for long cable runs (up to 1000 feet.)

Before constructing the cable, you must decide which serial interface protocol you intend to use.

NOTE

The RS-232 interface standard does not support cable runs of over 50 feet. For longer runs, the RS-422 standard must be used. Cable runs using RS-422 cannot exceed 1000 feet.

The RS-232 and RS-422 cable pinouts are different! Carefully refer to the diagrams in this section to be sure you are constructing the correct cable!

3.2.2 Materials Required

The following material is needed to build either cable:

- Shielded, twisted-pair interface cable
- One interface connector as required by the host computer (usually a DB-25 male connector)
- One DB-9 (male) connector, for iNFiNiT! family system

3.2.3 RS232 Serial Interface Pinout

Refer to Table 3-1 and Table 3-2, and/or Figure 3-1 for signal names and pinouts for the RS232 protocol.

NOTE

The pinout for the RS232 protocol is the same for all iNFiNiT! family systems.

Table 3-1 provides a typical DTE hookup to an Intelligent Interface host with a DB-25 connector. Table 3-2 provides a typical DTE hookup for a DB-9 to DB-9 cable.

Please check the host computer's serial port pinout documentation for proper hookup!

NOTE - SERIAL PORT CONFIGURATION

Some interfaces require the RTS signal (pin 7) to be HIGH. Because pin 7 is LOW, use pin 4 (DTR) instead. This may require a modification to your serial interface cable. Refer to Table 3-1 **and** Table 3-2**, and** Figure 3-1**.**

Table 3-1: Typical RS-232 Interface Cable Pinout (DB-9 to DB-25)

iNFiNiT! System (DB-9)		Host Computer (DB-25)	
Pin	**Signal Name**	**Signal Name**	**Pin**
2	RXD	TXD	2
3	TXD	RXD	3
5	GND	GND	7
		RTS / CTS	Bridge 4 — 5
		DSR / DCD / DTR	Bridge 6 — 8 — 20

Table 3-2: Typical RS-232 Cable Pinout (DB-9 to DB-9)

iNFiNiT! System (DB9)		Host Computer (DB-9)	
Pin	**Signal Name**	**Signal Name**	**Pin***
2	RXD	TXD	System Specific

Table 3-2: Typical RS-232 Cable Pinout (DB-9 to DB-9)

iNFiNiT! System (DB9)		Host Computer (DB-9)	
Pin	**Signal Name**	**Signal Name**	**Pin***
3	TXD	RXD	System Specific
5	GND	GND	System Specific
		DSR / DTR	Bridge Signals
		RTS / CTS	Bridge Signals

** Consult the host computer's hardware documentation for pin assignments on its DB-9 serial port.*

RS-232 Serial Connector Pinout
All iNFiNiT! Family Systems
Rear View of Connector Shown

DB-9 Serial Port Male

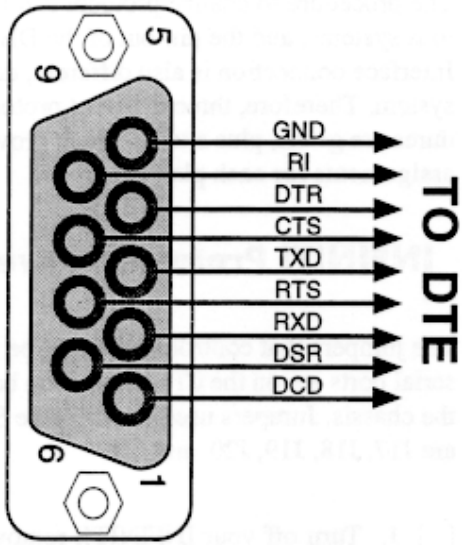


Figure 3-1. Serial Port Pinout (All systems, RS-232 only)

3.3 RS232 to RS422 Conversion

CAUTION

UNDER NO CIRCUMSTANCES SHOULD YOU CONNECT ANYTHING TO THE INFINIT! SERIAL PORT WITHOUT CONFIRMING THAT THE PORT SELECTION IS CORRECT!

EQUIPMENT DAMAGE MAY RESULT IF THE PORT SELECTION DOES NOT MATCH THE PROTOCOL IN USE!

The system can be set up to use either RS232 or RS422 protocol with the Intelligent Interface option. Factory configuration is RS232. If the cable to the host computer is less than 50 feet, RS232 can be used. However, if the distance between the system and the host computer is greater than 50 feet, and under *1000 feet*, the RS422 protocol must be used.

To change the system's Serial Port from RS232 to RS422, *the interface protocol on the system and host computer must match*.

The procedure to change protocol is different for the various systems, and the pin-out of the DB-9 Intelligent Interface connection is also different, depending on the system. Therefore, three different protocol change procedures are given, plus a table which provides signal pin assignments for each platform.

3.3.1 iNFiNiT! Protocol Conversion

The jumpers that control interface type for the iNFiNiT! serial ports are on the CPU Transition board at the rear of the chassis. Jumpers **JP17**, **JP18**, **JP19**, **JP20**, and **JP21** are used to configure SERIAL PORT 4, as shown in Figure 3-2.

[ ] 1. Turn off iNFiNiT!, then remove the power cord from the iNFiNiT! or from the outlet.

[ ] 2. Open the rear access door and locate the CPU Transition panel in the rear card cage of the iNFiNiT!.

[ ] 3. Loosen the four (4) captive screws which secure the transition panel to the iNFiNiT!.

[ ] 4. Label the cables connected to the Transition Board and carefully remove them.

Remove the CPU transition board from the iNFiNiT! and locate jumpers **JP17** through **JP21**:

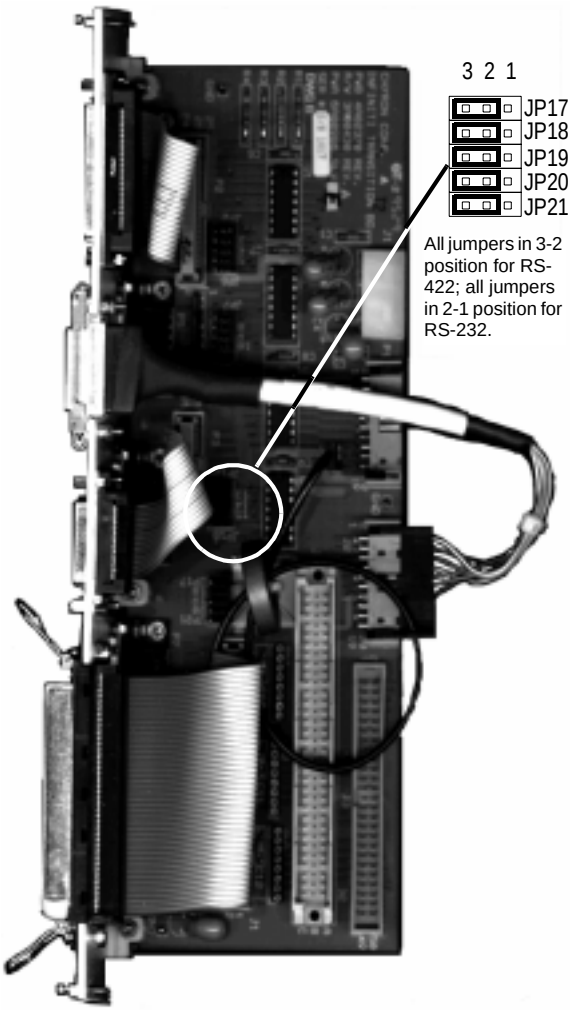


Figure 3-2. iNFiNiT! CPU Transition Board Jumpers

[ ] 5. Move the shorting blocks on jumpers **JP17** to **JP21** from pins one and two to pins two and three.

[ ] 6. Replace the transition board, then reconnect the cables removed in Step 4.

[ ] 7. Secure the transition board with the captive screws.

[ ] 8. Reconnect the power cord to the iNFiNiT! and the power outlet, then turn the iNFiNiT! back on.

Serial Port 4 on the iNFiNiT! is now set to operate with RS-422 signal levels. If you wish to change the port back to an RS-232 interface, simply move the **JP17** through **JP21** shorting blocks back to pins one and two.

3.3.2 MAX!>/020 Protocol Conversion

[ ] 1. **Turn off the MAX!>! Remove the power cord from the MAX!> and the outlet!**

[ ] 2. Remove the front panel.

[ ] 3. Remove the CPU board from the bottom slot (J1).

[ ] 4. Locate Jumper JP10 on the 6U side of the CPU board. Move the jumper from pin positions 2-3 to pin positions 1-2 (Figure 3-3).

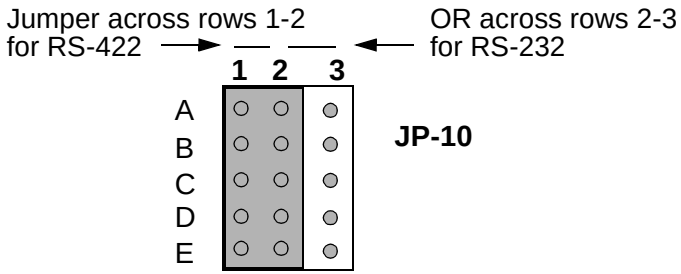


Figure 3-3. JP10, RS-422/RS-232 Selection

[ ] 5. Reinstall the CPU board, then replace the front cover.

Whenever MAX!> is powered up, serial port 2 will default to the values you last programmed into the Set Communication window.

Switch the jumper back to the "2-3" position to change MAX!> back to RS-232.

3.3.3 MAX!>/040, MAXINE! Protocol Conversion

These systems can be set to use either RS-232 or RS-422 protocol via a software configuration change.

To change the protocol, move the cursor to the Signal Level parameters on the Set Communications menu. Use the space bar to toggle through the choices.

These values must match the settings of your host computer serial port. Whenever the system is powered up, the J2 port will default to the values you last programmed into the Set Communications window.

3.3.4 RS-422 Signal Pinouts by System

Table 3-3 shows system Intelligent Interface connector pinouts. Refer to this table if you are configuring your setup using RS-422.

Table 3-3: RS-422 Signal Pinouts

Pin#	**iNFiNiT!** (Serial Port 4)	**MAX!>/020 or MAXINE!** (Serial Port 2 / INTEL.INTF)	**MAX!>/040** (Serial Port 2)
1	nc	nc	nc
2	TXD+	RXD+	RXD+
3	TXD-	TXD+	TXD+
4	nc	TXD-	RXD-
5	gnd	gnd	gnd
6	nc	RXD-	TXD-
7	RXD+	nc	nc
8	RXD-	nc	nc
9	nc	nc	nc

3.4 iNFiNiT! Family System/Host Hookup

3.4.1 iNFiNiT! Systems

Connect one end of the interface cable to **SERIAL PORT 4** (the bottom of the four ports) on the CPU Transition board, and connect the other connector to the computer's serial port. **Only Serial Port 4 can be used for Intelligent Interface on the iNFiNiT!**

3.4.2 MAX!> Systems

Connect one end of the interface cable to **J2** (the right port) on the rear of the MAX!>. Connect the other connector to the host computer's serial port. **Only Serial Port 2 can be used for Intelligent Interface on the MAX!>**.

3.4.3 MAXINE! Systems

Connect one end of the Intelligent Interface cable to the 9 pin D-SUB labelled **INTEL.INTF.** on the rear panel. Connect the other connector to the host computer's Serial Port. **Only the INTEL.INTF. port can be used for Intelligent Interface.**

3.5 Serial Port Set-up

Before using the Intelligent Interface application, you must set up the system Serial Port. And, before using the Intelligent Interface System, the configuration of the Serial Port must be checked.

3.5.1 Serial Port Data Parameters

3.5.1.1 iNFiNiT! Serial Data Parameters

Serial Port 4 has the following configuration set as a default at the factory:

Parameter	Value
Baud Rate	9600
Packet Size	8 bits
Parity	None
Stop bits	1

Whenever the iNFiNiT! is operating, Serial Port #4 will default to these values. For the Intelligent Interface to operate properly, the configuration of Serial Port 4 and the Serial Port on your host computer must match. Refer to the host computer's Operators Manual to set the configuration for the host Serial Port.

NOTE

Some interfaces require the RTS signal (pin 7 serial port 4) to be HIGH. Because pin 7 is LOW, use pin 4 (DTR) instead. This may require a modification to your serial interface cable. Refer to Figure 3-1.

3.5.1.2 MAX!> Serial Data Parameters

Serial Port 2 configuration must be checked before using Intelligent Interface.

[ ] 1. From the Loader menu, press **SET-UP**.

[ ] 2. Select the Set Com icon (or the C key), to open the Set Communication window (Figure 2-5).

The data parameters and machine ID parameters must be checked and altered if necessary before using the host computer. The following MAX!> Serial Port 2 values are factory-set in battery-backed RAM:

Parameter	Value
Baud Rate	9600
Packet Size	8 bits
Parity	None
Stop bits	1

These values must match the settings of your host computer Serial Port. Whenever MAX!> is powered up, Serial Port 2 will default to the values you last programmed into the Set Communication window.

NOTE

Some interfaces require the RTS signal (pin 7 serial port 2) to be HIGH. Because pin 7 is LOW, use pin 4 (DTR) instead. This may require a modification to your serial interface cable. Refer to Figure 2-2.

3.5.1.3 MAXINE! Serial Data Parameters

The INTEL.INTF. port configuration must be checked before using Intelligent Interface.

[ ] 1. From the Loader menu, press **SET-UP**.

[ ] 2. Select the Set Com icon (or the **C** key), to open the Set Communications window.

The data parameters and machine ID parameters must be checked and altered if necessary before using the host computer. The following INTEL.INTF. port values are preset in battery-backed RAM:

Parameter	Value
Baud Rate	9600
Packet Size	8 bits
Parity	None
Stop bits	1
Signal Level	RS-232

The MAXINE! can be set to use either RS-232 or RS-422 protocol with the Intelligent Interface option. Factory configuration is RS-232, which can be used as long as the interface cable is less than 50 feet long. If the distance between MAXINE! and the host computer is *greater* than 50 feet, and under 1000 feet, the MAXINE! and host must be configured to RS-422.

To change the protocol, move the cursor to the Signal Level parameters on the Set Communications menu. Use the space bar to toggle through the choices.

These values must match the settings of your host computer Serial Port. Whenever MAXINE! is powered up, the INTEL.INTF. port will default to the values you last programmed into the Set Communication window.

NOTE

Some interfaces require the RTS signal (pin 7 INTEL.INTF. Port) to be HIGH. Because pin 7 is LOW, use pin 4 (DTR) instead. This may require a modification to your serial interface cable. Refer to Figure 2-1.

3.6 Machine ID Code Setup

With Intelligent Interface, a single host computer can theoretically supply data to over 200 Systems on a single serial communication line at the same time. To send data to only one System, "directions" must be given to the host.

The Machine ID Number assigns a specific code number to each system. When a command that includes a machine ID is sent by the host computer, machine code, only the machine(s) specified by that code will be updated. Commands sent by the host computer consist of single letters followed by data. The following rules apply:

- An **UPPER CASE** global command affects all units.

- A **lower case** command uses a machine code to select the desired unit, which executes the command. When a lowercase command is used, it must be followed *immediately* by a valid machine code for the system to recognize the command.

The machine code is specified through the Machine ID icon in the Loader Set Up menu. To change the machine ID of the system:

[ ] 1. Display the Loader menu (press **MODE SLCT**, or the RESET button on the CPU board).

[ ] 2. Press the **SET-UP** key to call up the set-up icons.

[ ] 3. Select the Machine ID icon (or press M), enter a machine ID number from 1 to 255, then press **CNTL**.

If more than one system is being supplied with host data on the same serial communication line, unique Machine IDs must be set on each system. Enter a value from 1 to 255. See the Tabs section in the operator manual.

The machine code needs to be set only once, and is retained even when the system is OFF. *If the battery-backed RAM is corrupted, or the CPU board is serviced or replaced, the Machine ID number may need to be re-entered!*

The Intelligent Interface system is now ready for use. Section 4 of this manual tells you how to use the application.

3.7 Intelligent Interface Telnet Server

The Intelligent Interface Telnet Server (IITS) provides network users with virtual control of Intelligent Interface operations that were previously limited to console operators, and entities connected through serial port links. The extensibility of control to network hosts opens a new dimension of flexibility for time-critical solutions, convenience of host location, multiple concurrent operators, and new opportunities for third-party automation tool development.

Connections established through serial ports or via Telnet operate identically, with the exception of the following:

- Polling of incoming messages/commands to the IITS is shared among all connections including Telnet clients; however, serial port connections maintain highest priority.
- External Update Requests are no longer limited to the serial ports. All valid connections will receive identical External Update Requests; the requests are disseminated to each connection.
- Incomplete command strings entered on a serial connection will be reset after a minimum of 15 seconds if a Telnet connection is active. This will not occur on a Telnet connection since its data is buffered differently than serial port data. Resetting of a dormant serial connection will prevent the exclusion of command data originating from other sources such as Telnet clients.

More information about Telnet control of the Intelligent Interface application appears in Section.

Section 4 - Operation

This section provides procedures for construction, Intelligent Interface control and command protocol used when entering data on the host computer.

4.1 Overview

The Intelligent Interface System allows you to update graphics and text through the system, and control most aspects of the system, using a remote host computer as the input device. Effects such as Fades, Wipes, Transform Scripts, etc. can be set up and triggered as well.

4.1.1 Tab Description Messages

When working with Intelligent Interface, you can create template pages called Tab Description messages, using Message Compose. These Tab Description messages can be conposed of static (non-changing) text, and/or images and variable fields where the host (i.e. remote) computer places data such as scores, tallys, etc.

4.1.2 Tab Data Messages

After creating a Tab Description message, data is sent to the iNFiNiT! Family system and stored in Tab Data messages (known as Template Data messages). The Intelligent Interface command references them to a specified Tab Description message. The Tab Data messages can then be displayed using the layout set in the Tab Description message. Note that the same Tab Description message can be referenced as many times as necessary. Because the Tab information is *not* stored with each Tab Data message, execution of Intelligent Interface commands is very fast.

4.2 Creating a Tab Description Message

The Tab Description messages used by the Intelligent Interface are created in Message Compose, and can contain any combination of text, background, fonts, or palette selections.

The Tab Description message is constructed in two steps: First, static information such as graphics and headers are placed on the page. Then, Tab fields, which act as the "windows" that the host computer fills with data are added.

4.3 Tab Description Message Tutorial

To create a Tab Description message:

[ ] 1. From the Program Loader menu, which is the first menu displayed when booting the system, load Message Compose. Create a static display consisting of a background and any text that will not be updated using Intelligent Interface.

In the sample message below (Figure 4-1), the static text includes **Election '92 Returns**, party names, row headers (**Total Votes**, etc.), and percent signs. The dashed boxes represent the position and size of the Tab fields that will be placed on the page.

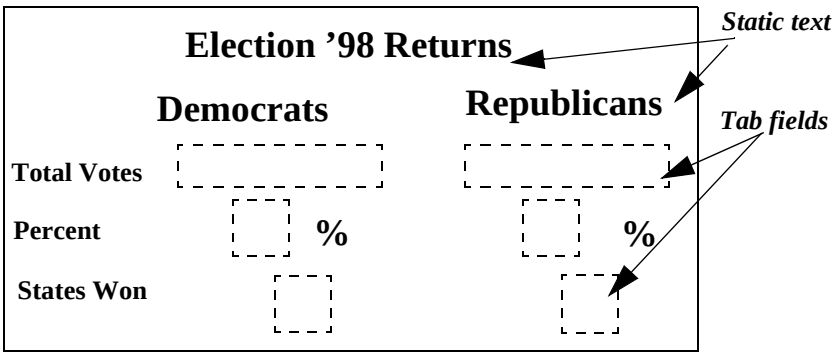


Figure 4-1. Sample Static Tab Message

[ ] 2. After the static material has been placed on the page, press **SET-UP TAB**. The Tab Set-Up Utility menu (Figure 4-2) is displayed.

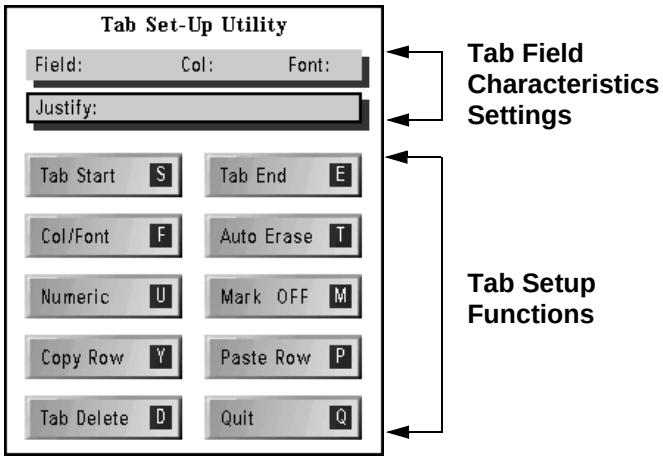


Figure 4-2. Tab Set-Up Utility Menu

[ ] 3. Position the cursor where you want the text supplied by the host computer to appear.

[ ] 4. Press **S** to set the left margin of the Tab Field. Use the ➔ key as follows to increase the size of the Tab field:

- Press ➔ once for each character that can appear in the field. In Figure 4-1, eight spaces to the right of the **Total Votes** text may be required, for a range of **0** to **99999999**, whereas States Won needs only two, for a range of **0** to **50**.

When setting the size of the Tab Field, be sure to provide enough room in the field for the maximum number characters that may be entered. The Tab field does not expand to accommodate additional characters received via Intelligent Interface.

[ ] 5. When the Tab is the proper size, press **E** to set the end of the Tab field.

NOTE

The order in which the Tab fields are created is important. The first Tab field created, regardless of screen position, is assigned Tab #0, the second tab created is made Tab #1, the third, Tab #2, and so on. The Tab "Write" and "Update" commands fill fields in the order of Tab creation.

- **To renumber a Tab field, press O, then enter the new number and press ⏎.**
- **To renumber all Tab fields from left-to-right and top-to- bottom, press J.**

[ ] 6. If desired, set Tab field characteristics as described in Table 4-1. If none are set, the current color and font are used. The characteristics, once set, appear in the Tab Field Characteristics Settings at the top of the Tab Set-Up Utility menu (Figure 4-2). Text typed in the field takes on the set characteristics.

A full description of each characteristic can be found in §7.4.3 of the iNFiNiT! Family operation Manual (Pub. No. 2A01976).

[ ] 7. If desired, repeat Steps 3 through 6 to set other Tab fields.

[ ] 8. Once all Tab fields have been set, press **CNTL** to exit the Tab Set-Up Utility menu.

[ ] 9. Press **CLR**, then enter an ID number for the Tab Description message on the Address Keypad. Press **RCD**. The Tab Description message is now recorded.

4.4 Tab Field Characteristic Commands

The following commands affect the Tab field in a message. They can be grouped into three types: **Field Characteristic Commands** (Table 4-1), which change the appearance or condition of a Tab field; **Tab Edit Commands** (Table 4-2); and **Mode/Store/Recall Commands** (Table 4-3).

Table 4-1: Tab Setup - Field Characteristic Commands

KEYSTROKE(S)	COMMAND
A	All data aligns vertically on the first decimal point
B	All data aligns vertically on the first comma
C	Center Justify - Centers all text in the Tab field
L	Left Justify - Aligns all text to left edge of Tab field
N	No Justify - Turns off text alignment in the Tab field
R	Right Justify - Aligns all text to right edge of Tab field
SPACE BAR	Toggles through available justifications
F	Col/Font - Selecting F allows the user to predetermine fonts or colors in the Tab field. Additional fonts and colors can be selected in Message Compose Mode
Color Keys	Press to select a character color for the Tab field
Font Keys **ALT Font Keys**	Press to select a font for the Tab field
COLOR FONT	Sets Tab field to current color and font of the character the cursor is on
I	Initializes (clears settings) Tab field
H	Allows you to assign a name to each Tab field
J	Renumbers all Tab fields in order (0, 1, 2, 3...) from left to right, top to bottom

Table 4-1: Tab Setup - Field Characteristic Commands

KEYSTROKE(S)	COMMAND
O	Renumbers the current Tab field number to a newly entered value
K	Forces upper case typing in the Tab field
U	**Numeric** - Compensates for the different widths of numbers by aligning them vertically within Tab fields in a uniformly spaced fashion
T	Auto Erase - Clears Tab field for new data when updating. When set, all data in the tab field is erased when the first new character is typed on re-entry into the tab field.
X	External Input - Enables Tab field update by Intelligent Interface
Z	Turns off external update for all Tab fields
M	**Mark On/Off** - If On, displays Tab field markers when you exit from Tab Set-up. This must be set for each frame buffer.

Table 4-2: Tab Setup - Tab Edit Commands

KEYSTROKE(S)	COMMAND
S	**Tab Start** - Sets the (start) of a new Tab field
E	**Tab End** - Sets the (end) of a new Tab field
G	Copy Row - Copies the Tab characteristics of the selected row to all empty (i.e. non-Tabbed) rows below
Y	**Cut Row** - Cuts all the Tab characteristics from the cursor-selected row of Tab fields and places them in a “paste” buffer
P	**Paste Row** - Deposits the Tab characteristics stored in the “paste” buffer to the cursor-selected row
D or **DEL CHAR**	**Tab Delete** - Deletes the Tab field that the cursor is on
DEL ROW	Deletes entire row of Tab fields the cursor is on
ERASE	Deletes all Tab fields from the Edit Monitor

Table 4-2: Tab Setup - Tab Edit Commands

KEYSTROKE(S)	COMMAND
CNTL ERASE	When in the Message Compose main menu, deletes all text, but leaves the Tabs intact
SHIFT CHAR + ←→	If the Tab Start or End encounters another Tab Start or End while moving, the moving Tab field overwrites the stationary Tab field boundary.
SHIFT PAGE + ←→	Moves all Tab fields left or right
SHIFT ROW + ←→	Allows you to align the Tab Starts of more than one field. Press **SHIFT ROW +** ←→ to move all Tab fields whose Tab Starts are vertically in line. If the Tab Starts of the moving fields pass the Tab Start(s) of any other field(s), those Tab field(s) are also picked up and carried along with the other Tab fields.
ALT SHIFT CHAR + ←→	Expands selected Tab field to the left or to the right

Table 4-3: Tab Setup - Mode/Record/Read Commands

KEYSTROKE(S)	COMMAND
SETUP or **CNTL TAB**	Accesses Tab Setup Mode from the Message Compose main menu
READ	Reads a message from disk from Tab Setup Mode or the Message Compose main menu
RCD	Records only the Tab settings as a message
DEL MSG	Deletes an entire message
ESC	Exits from Tab Setup Mode without saving changes
Q or **CNTL**	Exits from Tab Setup Mode - saves changes

4.5 Intelligent Interface Command Syntax

Commands consist of bidirectional data exchange between the host computer and the system. The protocol consists of a group of single-letter commands, which are followed by various modifiers and data as required.

Since several systems may be connected to one host computer, the host can address all units with one command (a global command), or address one unit at a time, using the individual system machine code.

Each major command is a single letter. If the command is in UPPER CASE, all units on the same serial communication line are addressed. If the command is in lower case and followed by a valid machine code, the individual system with that code is addressed, and executes the command.

Each command is always terminated with **Carriage Return** and **Line Feed** commands. They can be specified as follows: **CRLF**, **<CR><LF>**, or **<OD><OA>**. The command can also contain an optional **checksum** command which is entered directly before the **CRLF** commands. **Checksum** is used to verify the integrity of the command that is sent to the system, and appears as **[checksum]** in the commands shown in this manual. Note that the brackets (**[]**) are not entered - they simply alert the user that the command is optional. *See §4.9 for information on **checksum**.*

All ranges in the parameter descriptions are inclusive, e.g. **0** - **255** includes both **0** and **255**.

Note that if the keyboard buffer becomes full, additional commands are still processed - they are not lost.

The following must be observed when entering any Intelligent Interface commands.

- All listed parameters, except those specified as optional, must be entered.

- Parameters must be entered in the proper order.

The major Intelligent Interface commands are listed in Table 4-4. Detailed information on each command is found starting in §4.6, starting on page 4-13.

Table 4-4: Intelligent Interface Commands

Command	Description
C	Change color of specified field
F	Change font of specified field
M	Set disk drive and message directory
P	Change Time Slicing (Turbo Mode)
Q	Resend previous response
R	Response to External Update request
T	IMAGESTOR! commands
U	Update one field of a Tab Data message
V	Special Effects commands
W	Write to Tab Data message
Y	Send packet of keyboard codes to Primary user
Z	Send packet of keyboard codes to Secondary user.
The following special codes can be used in addition to the single letter commands described above:	
<01>	Subsequent codes are redirected to the keyboard buffer of the Primary User.
<02>	Command Terminator (for codes **01 and 02**).
<03>	Subsequent codes are redirected to the keyboard buffer of the Secondary user.
<05>	Request for Message Compose completion status. Used with **Y**, and **01** and 02 commands.

NOTE

A request for Message Compose completion status (used with the Y, Z, 01 and 03 commands) *cannot* be used to signal completion of a Roll, Crawl, or Slow Reveal.

In this manual, program code **<xx>** refers to an eight-bit decimal or hexadecimal number equal to the **xx** value, and is used to represent non-printable characters such as **Carriage Return** and **Line Feed**. For example, to represent a carriage return, the notation **<0D**> is sometimes used. An eight-bit number equal to **0D** must be sent. *DO NOT use the ASCII equivalents of* **<**, ***0***, ***D*** and **>**. Carriage Return and Line Feed are commonly used characters, and are also represented by **<CR>** and **<LF>** respectively, or as **CRLF**.

Printable characters can also be represented using decimal or hexadecimal code. Thus, a command such as **Y\ABC\\<0D><0A>** can also be represented as **<59><5C><41><42><43><5C><5C><0D><0A>**. Note that the backslashes have been converted to hexadecimal code as well. *Additionally, since the backslash is used as a field separator, any fonts used by Intelligent Interface should not contain a character stored under the* **** *key (hex* ***5C****), as it will be misinterpreted.*

NOTE

It is important to understand how to represent an iNFiNiT! key by using code. For example:

To send the code for Font 1, color green, the characters A, Space character, B, C, and then the code for HOME, it can be programmed as follows:

<A0><AD><41><20><42><43><CB>

Note that the letter characters in the code are not in turn represented by codes themselves - they *are* the codes. In other words, <AD>, the code for Font 1, is *not* represented by the code for A (<41>), then the code for D (<44>). Sending <41><44> sends the characters A and D; not the command to use Font 1.

A string can contain numerous fields; fields are separated by a backslash (****). The list is terminated by a double backslash (****), an optional two byte checksum, and a carriage return and/or line feed.

Example:

To send characters **ABC** to the keyboard buffer of the Primary user, send the following string:

Y\ABC\\33<0D><0A>

The **W**, **U**, **C**, and **F** commands can act on the field number assigned in the message. For example, if you have three fields numbered **0**, **2**, and **6**, then a **W** command (Write Tab Data Message) would look like the following:

W\100\10\AAA\ \BBB\ \ \ \CCC

Note the empty backslashes between **AAA** and **BBB** (skip from field #**0** to field #**2**), and between **BBB** and **CCC** (skip from field #**2** to field #**6**). These "empty" fields contain one space character each.

To directly address one field, enter the field number. For example, the **U** command to update field #**6** would be as follows:

U\100\6\DDD

To request a completion status from Message Compose, **<05>** can be included in the character string. When Message Compose receives the status request, **<05>**, it sends an acknowledgment of the form ***1** or ***2** (depending on whether the **Y** or **Z** command was issued) to the serial port. A completion status request might be sent as follows:

Y\ABC<05>\\38<0D><0A>

The Intelligent Interface response consists of two parts:

- ***<0D><0A>**indicates that the command has been accepted with no checksum error.
- ***1<0D><0A>**indicates that all characters have been processed. ***2** would be sent for the **Z** command.

Error Handling: If a checksum or other error was detected, an 8-byte ASCII error code is transmitted, followed by a carriage return and line feed. In this case, the second part of the response would not be transmitted. The second response is only transmitted after the first ***** has been sent and only if an **<05>** is included in the command string. *Appendix A contains a listing of all Intelligent Interface error codes.*

Host Acknowledgment: Once the computer has received an acknowledgment of command completion, it can send the next command/data stream to the unit.

String Size: The size of an individual command string cannot exceed 6000 characters. If at any time the buffer becomes loaded with 3072 characters, **XOFF** (decimal code **19**) is transmitted to the host computer. When the buffer empties to 1024 characters, **XON** (decimal code **17**) is transmitted, allowing the host computer to resume transmission.

NOTE

All of the system keyboard characters can be transmitted via the computer interface. This includes all alternate character values that are used for accents and special characters.

The data stream representation for these alternate characters must consist of the 7-bit ASCII ESC (Escape) character <1B> (hex) or <027> (decimal); or the 8-bit iNFiNiT! Family keyboard ALT character <CF> (hex) or <207> (decimal), followed by the character code. For example, to use the alternate form of upper case A (ALT A), the code sequence <1B><41> should be used. This applies to W, U and V commands.

4.6 Major Commands

A major command is always the first letter in the string. If it is in **UPPER CASE**, the command goes out to all systems. If in **lower case**, it only goes to the system specified by the machine code in the string (the second position in the string). The following must be observed when entering any Intelligent Interface commands.

- All listed parameters, except those specified as optional, must be entered.
- Parameters must be entered in the proper order.

4.6.1 Select Message Directory (M)

The **M** or **m** command is used to change the current directory. All subsequent commands only affect messages in this directory. This command is followed by the directory name.

If the command letter is in lower case, a machine code must be used as the first modifier. When executed, the command changes the active directory on the selected disk drive.

The syntax for the **Select Message Directory** command is as follows:

M\Drive Letter/Directory Name
[checksum]<CR><LF>

Or, if addressing a particular system:

m\Machine Code\Drive Letter/Directory Name
[checksum]<CR><LF>

Examples:

M\C/ELECTION96\\<CR><LF>

OR

m\8\C/ELECTION96\\<CR><LF>

The second example only addresses machine #**8** on the serial line, whereas the first is global to all systems (note the uppercase **M**).

Note that the slash character (*/*) in the directory name is a *forward* slash, unlike the *backslash* characters (****) used in the Intelligent Interface commands.

The only response to a successful disk drive selection and message directory change is an asterisk (*****). If there is an error, the system sends back an error code.

4.6.2 Write to Tab Data Message (W/w)

The **W** or **w** command is used to write new data to a Tab Data file that uses an existing Tab Description message as a template for display. **W** uses the following modifiers:

- **A "w" command requires a machine code as the first modifier.**
- **Tab Data message ID # (up to 4 digits)**
- **Tab Description message ID # (up to 4 digits)**
- **Data that will be sent and recorded to the Tab Data message(s).**

If the data overflows the Tab Data message recorded at the specified ID number, the system records additional Tab Data message(s) at the next available ID number(s), using the same specified Tab Description message as a template.

The syntax for the **Write to Tab Data message** command is as follows:

W\Tab Data message ID #\Tab Description message ID #\Data1\Data2\. . . \\ [checksum]<CR><LF>

Or, if addressing a particular system in a multi unit system:

w\Machine Code\Tab Data message ID #\ Tab Description message ID #\ Data1\Data2\ . . . \\[checksum]<CR><LF>

Examples:

W\150\2050\JACKSON\51%\BUSH\49%
<CR><LF>

OR

w\16\150\2050\JACKSON\51%\BUSH
49%\\<CR><LF>

In the first example, message file ID **0150** is written, according to the Tab descriptor file at ID **2050**, with four fields: **JACKSON**, **51%**, **BUSH**, and **49%**. **JACKSON** is written to the field that is numbered **0** in the Tab message; **51%** is written to Field #**1**, and so forth.

The first example above writes to all systems (note uppercase **W**). The second example writes only to system #**16**. The only response to a successful write is an asterisk (*****). If there is an error, the unit sends back an error code.

All of the system keyboard characters can be transmitted via the computer interface. This includes all Alternate Character values that are used for accents and special characters.

The data stream representation for these Alternate Characters consists of the ASCII **ESC** character code **<1B>** (hex) or code **<027>** (decimal), followed by the 8-bit iNFiNiT! Family keyboard character code of the Alternate Character **<CF**> (hex) or **<207>** (decimal).

4.6.3 Update One Field (U/u)

The **U** or **u** command is used to update one field in a Tab Data message, (i.e. a message created by a **W** command), without disturbing the remaining fields in the message. **U** uses the following modifiers:

- **A "u" command requires a machine code as the first modifier.**
- **Tab Data message ID # (up to 4 digits)**
- **Field #**
- **Data**

The syntax for the **Update One Field** command is as follows:

U\Tab Data message ID #\Field #\ Data\\[checksum]<CR><LF>

Or, if addressing a particular system in a multi-unit system:

u\Machine Code\Tab Data message ID #\ Field #\Data\\[checksum]<CR><LF>

Examples:

U\1050 \17\1,714\\<CR><LF>

OR...

u\16\1050\17\1,714\\<CR><LF>

In both examples, Tab Data message ID **1050**, Field #**17** is updated with new data (in this case, **1,714**). Any printable ASCII characters may be used in the updated field *except the backslash (******)*. The first example writes to all systems on the serial line. The second writes only to system #**16**.

The only response to a successful write is an asterisk (*****). If there is an error, the unit sends back an error code.

4.6.4 Change Field Color (C/c)

The **C** or **c** command is used to change the color of one field in a Tab message. **C** uses the following modifiers:

- **A "c" command requires a machine code as the first modifier.**
- **Tab Description message ID # (up to 4 digits)**
- **Field #**
- **Color Index (1 to 8)**

The color index is a number from **1** and **8** inclusive, corresponding to the Color Keys on the system keyboard as follows:

1 - Red	**3 - Blue**	**5 - Yellow**	**7 - White**
2 - Magenta	**4 - Cyan**	**6 - Green**	**8 - Black**

The syntax for the **Change Color** command is as follows:

C\Tab Description message ID #\Field #\ Color Index\\[checksum]<CR><LF>

Or, if addressing a particular system in a multi-unit system:

c\Machine Code\Tab Description message ID #\ Field #\Color Index\\[checksum]<CR><LF>

Examples:

C\1050\17\3\\<CR><LF>

OR...

c\16\1050\17\3\\<CR><LF>

In this example, Tab Description message ID **1050**, Field #**17** is changed to **Blue** (Color Index **3**). The first example writes to all systems, while the second addresses only system #**16** (note lower-case **c**).

The only response to a successful color change is an asterisk (*****). If there is an error, the unit sends back an error code.

4.6.5 Change Field Font (F/f)

The **F** or **f** command is used to change the font style of one field in a Tab message. **F** uses the following modifiers:

- **An "f" command requires a machine code as the first modifier.**
- **Tab Description message ID # (up to 4 digits)**
- **Field #**
- **Font Index (1 to 16)**

The Font Index is a number from **1** to **16** inclusive. Index **1** through **8** correspond directly to the Font Key number on the iNFiNiT! Family keyboard. Index **9** through **16** correspond directly to the Font Keys used with **ALT**.

The syntax for the **Font Change** command is as follows:

F\Tab Description Message ID #\Field #\ Font Index\\[checksum]<CR><LF>

Or, if addressing a particular system in a multi-unit system:

f\Machine Code\Tab Description Message ID #\ Field #\ Font Index\\[checksum]<CR><LF>

Examples:

F\1050\15\7\\<CR><LF>

OR...

f\16\1050\15\7\\<CR><LF>

In both examples, Tab Description message ID **1050**, Field #**15** is changed to Font Index **7** (corresponding to font position **7** on the system keyboard). The first example addresses all systems, the second only system #**16**.

The only response to a color change is an asterisk (*****). If an error occurs, the unit returns an error code.

4.6.6 External Update Request (X)

The **X** command is automatically sent to the host computer from the iNFiNiT! Family system when a Tab Data or a Tab Description message is read from within Message Compose, which has one or more Tab fields marked for external update (**X** in Tab Set-Up Utility). Note that when such a message is read, the Prompt Line in the Message Compose menu displays the following:

Waiting for response from INTELLINT - ESC to exit

The syntax for the **External Update** command is as follows:

X\User ID #\Tab Data message ID #\Tab Description message ID #\Field #[Field Name]\Field #[Field Name]\ . . . \\ [checksum]CRLF

Note that if a Tab Description message is read, the **Tab Data message ID #** field in the External Update request will contain four asterisks (********), as there was no Tab Data message tied to the Tab Description message. However, if a Tab Data message is read, both the **Tab**

Data message ID # field and the **Tab Description message ID #** field will have ID numbers in them, as a Tab Data message always has a Tab Description message referenced to it.

X uses the following modifiers:

- **User ID # specifies the Primary or Secondary user of the system and must be returned in the reply message.**
- **Tab Data message ID #**
- **Tab Description message ID #**
- **Field #[Field Name] represents the number or name of the field which needs to be updated. All of the field numbers have two ASCII digits (00 to 99), zero padded. If the field has been named, the Field Name immediately follows the corresponding Field #. The Field Name contains up to 15 ASCII characters and is displayed only if the field has been named previously.**
- **Optional checksum contains two hex digits, zero padded.**

Example:

```
X\1\0300\0350\00Team 1\01score 1\
03score 2\\[checksum]CRLF
```

After the **X** request is sent to the host computer, the system waits for a response, described next in §4.6.7.

Roll/Crawl Applications w/ External Update

You can Roll or Crawl Tab Data messages (created using the "W" command) that contain External Update fields. However, you cannot Roll or Crawl Tab Description messages which contain External Update fields - the fields will be ignored!

4.6.7 Response to External Update Request (R)

The **R** command is the host computer's reply to the iNFiNiT! **X** command, and must be sent to the iNFiNiT! system for the update to execute. The data sent by the host computer to the iNFiNIT! Family system fills the Tab fields sequentially in ascending order of the marked Tab fields. In the external update example above, Tab fields **00**, **01** then **03** would be filled.

R uses the following modifiers:

- **User ID # specifies the Primary or Secondary user of the system and must be returned in the reply message.**
- **Data**

The syntax for the **Reply** command is as follows:

R\User ID #\Data\Data\ . . . \\<CR><LF>

Example:

R\1\Red Sox\3\5\\<CR><LF>

Tab field **00** would update to **Red Sox**; Tab field **01** would update to **3**; and Tab field **03** would update to **5**.

4.6.8 Request Previous Transmission (Q/q)

The host computer sends a **Q** command to the iNFiNiT! Family system for a resend of the last transmission. This transmission can be an acknowledgment (*****), error codes, or the External Update Request response itself.

The syntax for the Resend Request command is as follows:

q\Machine Code

OR...

Q

The computer then resends the last transmission.

4.6.9 Alternate Method for Changing Font and Color

The font and color index can be changed inside any data field (i.e. the space between two backslashes that contains the data that is sent to be displayed) of a **W**, **U**, **V**, or External Update Intelligent Interface commands.

To change the font number or color index as a command within the data field:

- Embed the keyboard value for a Color Key **<168>** - **<175>** or Font Key **<160>** - **<167>** within the data field.

Example:

abc is displayed in black, and **def** is displayed in red. The Tab field's color remains red until a new command is read which would change the color.

W\6000\6001\<175>abc<168>def

Note that for fonts **9**-**16**, you must issue the keyboard value for the **ALT** key (**<207>**) immediately before the Font Key value.

4.6.10 Change Time Slicing Command - Turbo Mode (P/p)

By changing how the Intelligent Interface program time slices with Message Compose, Intelligent Interface can run more often, resulting in greater throughput, and therefore increased overall performance.

To turn on **Time Slicing**, enter the following:

P\0\\[checksum]CRLF

To turn off **Time Slicing**, enter the following:

P\1\\[checksum]CRLF

The default setting is **Off**.

Or, if addressing a particular unit in a multi-unit system, enter the following commands to turn **Time slicing** on and off respectively:

p\Machine Code\0\\[checksum]CRLF

p\Machine Code\1\\[checksum]CRLF

This change improves the performance of each individual command when other tasks (such as Roll, Networking, Clock, etc.) are not running. If another task is running, Intelligent Interface operation can be slower than it would normally be if Time Slicing were not activated. This command needs to be issued only once at the beginning of the Intelligent Interface session.

4.7 IMAGESTOR! (T) Commands

This section describes the Intelligent Interface command set for remote control of the Chyron IMAGESTOR! option.

NOTE

For the system to accept IMAGESTOR! commands, the file RK.X must be present in the SYSTEM/ PROGRAMS directory. *This program is present on the Recall Keyboard disk.*

YOU MUST OBTAIN THIS FILE EVEN IF YOUR SYSTEM IS NOT CONFIGURED WITH THE RECALL KEYBOARD OPTION!

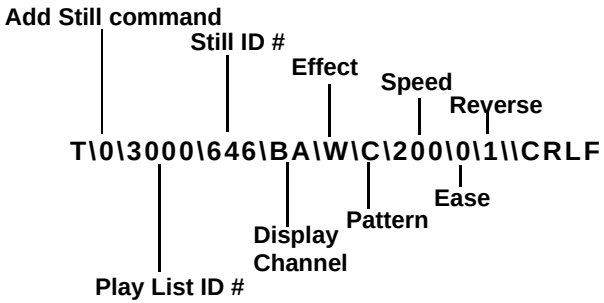
An IMAGESTOR! Play List can be either active or stored as a file. The active Play List is the same list used by the IMAGESTOR! Recall Keyboard. If a change is made through the Intelligent Interface to the active Play List it is immediately reflected in the Play List display on the Recall Keyboard. A system does not have to be configured with a Recall Keyboard for IMAGESTOR! Intelligent Interface control.

The general syntax for IMAGESTOR! commands is as follows. Note that not all IMAGESTOR! commands contain parameters.

T\Command #\Parameter 1\...
Parameter n\\[checksum]CRLF

Example:

To add a Still to a Play List, enter the following:



The command codes are as follows.

Table 4-5: IMAGESTOR! (T) Commands

Command	Description	Parameters
0	Add Still A new Play List entry is added to the current active list or to a previously recorded Play List. All entries in the Play List starting at the current line number are shifted down and renumbered to make room for the new entry. If the Play List does not exist it is created.	Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled C - Currently active Play List Entry # in Play List Range: 0 - 999 -1 - To add Still to end of Play List ID # of previously recorded Still that will be added to Play List Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled Channel A - Displays in Channel A B - Displays in Channel B AB - Effect displays from Channel A to Channel B BA - Effect displays from Channel B to Channel A NO - Machine Control [DDR] Only Effect (also known as Effect Type) W - Wipe P - Push C - Cut H - Hide R - Reveal D Dissolve M - Machine Control (DDR) Pattern applied to specific Effect (also known as Transition Pattern) A - Horizontal Slide Wipe B - Vertical Slide Wipe C - Horizontal Split Wipe D - Vertical Split Wipe E - Box Wipe NOTE: A through E can be applied to Wipe (W); only A and B can be applied to Push (P), Cut (C), Hide (H), Reveal (R) and Dissolve (D.) Speed (Duration) Range (fields): 0 - 300 Ease 0 - None 1 - Ease In 2 - Ease Out 3 - Ease In/Out Reverse 0 - Off 1 - On

Table 4-5: IMAGESTOR! (T) Commands

Command	Description	Parameters
1	Replace Still The current entry number at the line specified is replaced by the new entry. No renumbering of other entries in the Play List occurs.	Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled C - Currently active Play List Entry # in Play List Range: 0 - 999 ID # of previously recorded Still that will replace Still currently at specified Entry # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled Channel A - Displays in Channel A B - Displays in Channel B AB - Effect displays from Channel A to Channel B BA - Effect displays from Channel B to Channel A NO - Machine Control [DDR] Only Effect (also known as Effect Type) W - Wipe P - Push C - Cut H - Hide R - Reveal D Dissolve M - Machine Control (DDR) Pattern applied to specific Effect (also known as Transition Pattern) A - Horizontal Slide Wipe B - Vertical Slide Wipe C - Horizontal Split Wipe D - Vertical Split Wipe E - Box Wipe NOTE: A through E can be applied to Wipe (W); only A and B can be applied to Push (P), Cut (C), Hide (H), Reveal (R) and Dissolve (D.) Speed (Duration) Range (fields): 0 - 300 Ease 0 - None 1 - Ease In 2 - Ease Out 3 - Ease In/Out Reverse 0 - Off 1 - On

Table 4-5: IMAGESTOR! (T) Commands

Command	Description	Parameters
2	Move Entry All Play List entries between the Start and End number inclusive are moved to a new position in the Play List starting at the Destination number. The Play List is automatically renumbered after the Move command is executed.	Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled C - Currently active Play List Start Entry # in Play List Range: 0 - 999 End Entry # in Play List Range: 0 - 999 Destination Entry # in Play List Range: 0 - 999
3	Delete Entry The specified entry number is deleted from the Play List, and the Play List is automatically renumbered.	Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled C - Currently active Play List Entry # of Still that will be deleted from Play List Range: 0 - 999
4	Delete Play List	Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled C - Currently active Play List
5	Change Current Entry The current entry in the Play List is reset to the specified entry #. The Read Next buffer is cleared. *Therefore, two Read Next commands must be sent before the specified Still is displayed on screen*	New Current Entry # Range: 0 - 999
6	Read Next The Still specified as the current entry is displayed, using the specified Read Effect.	None

Table 4-5: IMAGESTOR! (T) Commands

Command	Description	Parameters
7	Read Still The Still is read directly into the channel specified. No effect is used and current Play List position is *not* affected.	Still ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled Channel A - Displays in Channel A B - Displays in Channel B
8	Read Play List The specified Play List is read from disk and made active. The first entry in the Play List is set as the current entry. The Read Next buffer is cleared. *Therefore, two Read Next commands must be issued before the first Still is displayed on screen.*	Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled
9	Query Play List The specified Play List is read from disk, and information about each entry (System Reply) is sent back in the following syntax: R\9\Play List ID #\Entry #1\Still #\Channel\Effect\ Pattern\Speed\ Ease\Reverse\.....\ Entry #n\Still #\Channel\Effect\ Pattern\Speed\ Ease\Reverse\\ [checksum]CRLF	HOST COMMAND PARAMETERS Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled C - Currently active Play List SYSTEM REPLY PARAMETERS For parameter choices and ranges, see the parameter list for the Add Still command (T\0). Play List ID # Entry # Still ID # at that Entry # Channel Effect Pattern Speed Ease Reverse In the reply, all parameters except for the Play List ID# are repeated in order as each entry is queried.

Table 4-5: IMAGESTOR! (T) Commands

Command	Description	Parameters
10	Query Entry The entry in the Play List specified is read from disk and data is sent back in the following syntax: R\10\Play List ID #\ Entry #\Still #\ Channel\Effect\ Pattern\Speed\ Ease\Reverse\\ checksum CRLF	HOST COMMAND PARAMETERS Play List ID # Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled C - Currently active Play List Entry # in Play List Range: 0 - 999 SYSTEM REPLY PARAMETERS For parameter choices and ranges, see the parameter list for the Add Still command (T\0). Play List ID # Entry # Still ID # at that Entry # Channel Effect Pattern Speed Ease Reverse
11	Keyboard Lock The Keyboard Lock command is used to restrict access to the IMAGESTOR! Recall Keyboard (if configured). The default setting is Unlocked.	Status 0 - Unlock 1 - Lock
12	Save Play List The current Play List is saved to disk at the specified file number, provided it is not already occupied.	ID # to which Play List will be recorded Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled

Table 4-5: IMAGESTOR! (T) Commands

Command	Description	Parameters
13	Query Status The current status of IMAGESTOR! is sent back according to the following syntax: R\13\Entry #\ Still ID # Channel A\ Still ID # Channel B\\ [checksum]CRLF	HOST COMMAND PARAMETERS None SYSTEM REPLY PARAMETERS Entry # in Play List Range: 0 - 999 ID # of Still displayed in Channel A Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled ' ' (space) - Not Active ID # of Still displayed in Channel B Range: 0 - 9999 if 8 Digit Ids not enabled 0 - 99999999 if 8 Digit Ids enabled ' '(space) - Not Active
14	Erase Channel The frame buffer assigned to the specified channel is erased, and the Read Next buffer is cleared.	Channel A - Channel A B - Channel B

4.8 Special Effects (V) Commands

Special Effects (**V**) commands support addressing individual systems. If addressing an individual system instead of all systems, the command must start as follows: **v\Machine code\.....**

NOTE

Intelligent Interface 4.11 and later support V commands executed on a MAXINE! system. Only iNFiNiT! and MAX!> systems can execute V commands using earlier versions of Intelligent Interface.

Also note that commands involving multi-buffer operations cannot be executed by MAXINE! or one buffer MAX!> systems.

Special Effects commands are grouped into seven categories: Read Effects (by buffer), Mix Effects, Transform, Miscellaneous and Trigger. Each category is assigned an identifier, which is placed in the field following the **V** command. *See Appendix C for examples.*

The identifiers are as follows:

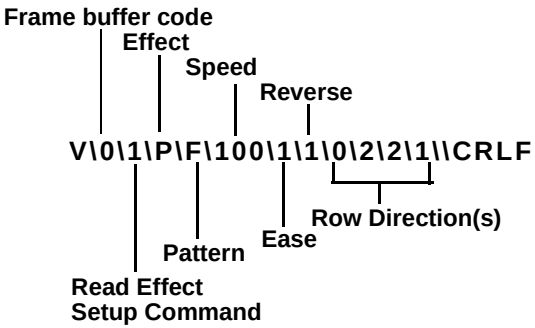
Table 4-6: Special Effects (V) Commands

ID#	Function
0	Read Effects (Buffer 1)
1	Read Effects (Buffer 2) - iNFiNiT! and MAX!>, if so equipped
2	Read Effects (Buffer 3) - iNFiNiT! only, if so equipped
3	Mix Effects
4	Transform
5	Miscellaneous
6	Trigger

Before a Read Effect is triggered, the message must be prebuilt in Message Compose. Do this using Miscellaneous Command **V\5\3** (prebuild). After issuing the command, do not use that frame buffer until after the effect has been triggered.

Example:

To set up a Read Effect in frame buffer 2, enter the following:



This command sets up the Read Effect as follows: **Row Slide** for frame buffer **1**; Speed = **100**; Ease **In**; Reverse **On**; Row 0 slides **Up**, Row 1 slides **Left**, Row 2 slides **Left**, and Row 3 slides **Down**.

The following tables list parameters that can be entered using Special Effects commands.

4.8.1 Read Effects (V\0 - V\2)

There are three types of Read Effects setup commands: Standard; Roll/Crawl; and Total Motion Control, which includes CMF (Chyron Motion File) and JPEG (CLYPS). The command syntax for each is detailed in §4.8.1.1 to §4.8.1.3. To execute Read Effects, a trigger command (**V\6**) must be sent. *Please refer to §4.8.6 for information on triggering effects.*

The general syntax for Read Effects setup commands are as follows:

V\Read Effects Frame buffer code
Parameter 1\...\Parameter n
[checksum]CRLF

The Read Effects frame buffer codes are as follows:

0 - Read Effect is set up in frame buffer 1
1 - Read Effect is set up in frame buffer 2
2 - Read Effect is set up in frame buffer 3

4.8.1.1 Read Effects - Standard (V\0 - V\2)

The general syntax for Read Effects setup commands is as follows:

V\Read Effects Frame Buffer code\1
Effect\Pattern\Speed\Ease\Reverse
Row 0 Direction\...\Row n Direction
[checksum]CRLF

Table 4-7: Read Effects Setup (V\0 - V\2) Command

Command	Description	Parameters
1	Set Up Read Effect	Effect W - Wipe P - Push H - Hide R - Reveal D - Dissolve C - Cut* Pattern A - Horizontal Slide Wipe B - Vertical Slide Wipe C - Horizontal Split Wipe D - Vertical Split Wipe E - Box Wipe F - Row Slide/Wipe G - Scanline Wipe H - Row-by-Row Scanline Wipe I - Screen Scanline Wipe J - Pixel Wipe Speed (Duration) Range (fields): 1 - 300 Ease 0 - No Ease 1 - Ease In 2 - Ease Out 3 - Ease In and Out Reverse 0 - Off 1 - On Row Direction 0 - Up 1 - Down 2 - Left 3 - Right 4 - Static

* *A valid syntax for the **Cut** command is as follows:*

V\0\1\C\A\60\0\0

*All fields following **C** are ignored, but must be valid.*

4.8.1.2 Read Effects - Roll & Crawl (V\0 - V\2)

NOTE

For Roll and Crawl operations, Message Compose must be active, and the fonts used in the message(s) must be loaded.

The general syntax for Read Effects Roll and Crawl setup commands are as follows:

V\Frame buffer code\1\Effect\Roll or Crawl ID #\Duration\\CRLF

The general syntax for aborting a Read Effects Roll or Crawl is as follows:

V\Read Effects Frame buffer code\1\A\\CRLF

Do NOT use the iNFiNiT! Family keyboard to abort the Roll or Crawl operation!

Table 4-8: Read Effects/Roll & Crawl (V\0 - V-2) Commands

Command	Description	Parameters
1	Set Up Effect - Roll or Crawl	Effect X - Roll Z - Crawl Roll or Crawl Message ID # Duration Range (fields): If Duration is 0, the duration set in the Roll or Crawl message is used.
1	Abort Roll or Crawl	Effect A - Abort Roll or Crawl

4.8.1.3 Read Effects - CMF & JPEG (V\0 - V\2)

A CMF animation or JPEG file can be set up via an Intelligent Interface Read Effects command. It can then be executed using an Effects trigger command.

The general syntax for CMF setup is as follows:

V\Read Effects Frame buffer code\1\V\EEFB Foreground or Background buffer\Start Frame #\End Frame #\Absolute Path\\[checksum]CRLF

Foreground or Background Buffer, Start Frame and End Frame *cannot* be specified when setting up a JPEG file. Type a space where the data would normally be entered. Thus, the general syntax for JPEG setup would be typed as follows:

V\Read Effects Frame buffer code \1\V\ \ \ \Absolute Path\\[checksum]CRLF

Do not leave the space characters out, as any two consecutive slashes terminate the command.

Example 1:

V\1\V\A\0\600\A\Z/INFINIT/STILL/355\\\[checksum]CRLF

The CMF stored at **Z/INFINIT/STILL/355** is set to excute in the Foreground buffer (**A**) of frame buffer 2 (**1**), starting at frame **0**, and ending at **600**.

NOTE

The frame that was displayed when a CMF was recorded becomes an identifier for the CMF. For example, it is used as a Thumbnail in IMAGESTOR!. When a CMF is set up

for execution via Intelligent Interface, this frame is displayed until the CMF is triggered. When the Trigger command is received, the CMF begins execution from the specified Start Frame. CMF's recorded at a different frame from the Start Frame should be triggered in Preview, then switched to Air after the identifying frame has been displayed.

Example 2:

V\2\V\ \ \ \A\Z/INFINIT/STILL/356\\\[checksum]CRLF

The JPEG stored at **Z/INFINIT/STILL/356** is set to excute in frame buffer 3 (**2**).

Table 3-1: CMF/JPEG Setup (V\0 - V\2) Command

Command	Description	Parameters
1	**Set Up Effect - CMF or JPEG**	**Effect** **V - CMF or JPEG** **EEFB Foreground or Background Buffer (CMF's Only)** **A - Foreground** **B - Background** **Start Frame # (CMF's Only)** **End Frame # (CMF's Only)** **NOTE: If both the Start Frame # and End Frame # are set to 0, the entire CMF executes.** **Absolute Path** **Full path and ID #. Only drives W, X, Y and Z are valid source drives.**

To execute the CMF or JPEG, a trigger command must be sent. *Please see §4.8.6 for information on trigger commands.*

4.8.2 Mix Effects Commands (V\3)

The general syntax for Mix Effects setup commands is as follows.

V\3\Command #\Parameter 1\...\ Parameter n\\[checksum]CRLF

To execute Mix Effects, a Trigger Effects command (**V\6**) must be sent. *Please refer to §4.8.6 for information on triggering effects.*

Table 4-9: Mix Effects (V\3) Commands

Command	Description	Parameters
1	Set up Effect	Effect W - Wipe F - Fade Pattern A - Horizontal Slide Wipe B - Vertical Slide Wipe C - Horizontal Split Wipe D - Vertical Split Wipe E - Box Wipe F - Row Slide/Wipe G - Scanline Wipe H - Row-by-Row Scanline wipe Speed Range (fields): 1 - 300 Ease 0 - No Ease 1 - Ease In 2 - Ease Out 3 - Ease In And Out Reverse 0 - Off 1 - On Percent - Channel A Range: 0 - 100 Percent - Channel B Range: 0 - 100 Soft Mask Range: 0 - 60

Table 4-9: Mix Effects (V\3) Commands

Command	Description	Parameters
2	Execute Flash Sequence	Channel 1 - Channel A Air 2 - Channel B Air 3 - Preview 1st Buffer 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3 4 - Blank screen Duration 1 Range (fields): 1 - 1,000,000 2nd Buffer 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3 4 - Blank screen Duration 2 Range (fields): 1 - 1,000,000
3	Stop Flash Sequence	None

4.8.3 Transform Commands (V\4)

The general syntax for Transform setup commands is as follows.

V\4\Command #\Parameter 1\...\ Parameter n\\[checksum]CRLF

Note that not all Transform commands contain parameters. Additionally, Script, Object and Effect names are case sensitive, e.g. the Effect name **SPIN** is different from the Effect name **Spin**.

To execute a Transform Script, a Trigger Effects command (**V\6**) must be sent. *Please refer to §4.8.6 for information on triggering effects.*

Table 4-10: Transform (V\4) Commands

Command	Description	Parameters
1	Reset Hardware	None
2	Change DIR	Directory name (example: C/MAX)
3	Load message NOTE: This command only works with "old style" Transform messages (Transform version 1.4 or earlier).	Message ID #
4	Load Script	Script ID # or name
5	Set Transform Priority	Transform Priority 0 - Frame Buffer only 1 - Transform only 2 - Frame Buffer priority 3 - Transform priority
6	Displays specified Script frame	Script Frame Number Range: 0 - [Script length - 1]
7	Escape at Pause Point	None
8	Bind Object and Effect to Event. Recompiles Script automatically.	Event Number Object Effect
10	Clear Script from Memory	None
11	Modify Script Length	Time in frames
12	Erase Transform display	None
13	Binds Object and Effect to Event. Does NOT recompile the Script. This command is faster than command 8, but must be used in conjunction with 8. The Script must be recompiled at least once prior to execution.	Event Number Object Effect

4.8.4 Advanced Transform (V\4) Commands

Advanced Transform commands allow remote modification of Transform Scripts, Events and Effects. These commands can be used to create dynamically

changing Transform animations, based on real-time data for events such as sports and elections. For example, graphs can grow or shrink to reflect updated information, or an animated golf ball can follow a path that reflects the actual shot at a golf match.

All advanced Transform commands check for errors in parameter range, command syntax, etc.

These setup commands use the same general syntax as the Transform setup commands described in §4.8.3:

V\4\Command\Parameter 1\...
Parameter n\\[checksum]CRLF

The following applies to the commands listed in Table 4-11:

- Any parameter field containing an asterisk (*****) maintains its previous value.
- **<** or **>** can be placed in front of an entry in a numerical parameter field to decrement or increment, by the specified amount, respectively, the current value of the field.
- Script, Object and Effect names are case sensitive, e.g. the Effect name **SPIN** cannot be sent as **Spin**.
- Some of the commands contain a **Mode** field which allows a query option which returns, rather than modifies the same data for the currently loaded Script.

  Any data fields resulting from a query command containing **\?** indicate that the parameter was an input parameter and not relevant as an output from the query command.

  Also, to increase throughput, query commands need not pass in all parameters; only enough to distinguish the object (i.e. Effect Name, Frame Number, etc.).

The **Mode** field is not case sensitive. Data can be entered in either upper or lower case letters.

- To improve overall performance, some commands allow the option of **No Compile** (**\N**). this allows multiple modification commands to be issued in succession, with one operator issued compile command at the end. *However, note that triggering a Transform Script without a proper compile can yield unpredictable results.*
- Transform Intelligent Interface commands cannot be used concurrently with Message Compose Transform menu operation.

Table 4-11: Advanced Transform (V\4) Commands

Command	Description	Parameters
14	Displays specified Script frame. This command performs a compile.	Script Frame Number Range: 0 - [Script length - 1]
15	Compiles current Script. To improve performance, the individual modification commands can be issued with the **No Compile** option set. This command is then issued once at the end.	Mode C - Compile only D - Compile, then display Frame 0
16	Modifies length of specified Script	Mode N - Update parameters with no compile C - Update parameters and compile D - Update parameters, compile, then display Frame 0 Q - Query based on Script name. Parameters following Effect name are ignored on input, and the current values are returned in reply. Script Length Range (fields): 2 - 10,000 Proportional N - No Y - Yes Extend Events N - No Y - Yes

Table 4-11: Advanced Transform (V\4) Commands

Command	Description	Parameters
17	Modifies length of specified Effect	Mode **N - Update parameters with no compile** **C - Update parameters and compile** **D - Update parameters, compile, then display Frame 0** **Q - Query based on Script name. Parameters following Effect name are ignored on input, and the current values are returned in reply.** Effect Length **Range (fields): 2 - 3,600** Proportional **N - No** **Y - Yes**
18	Modifies all event specific parameters. These are the same parameters that are found in the Event window in the Transform main menu. The following can be modified: • Start and End frames • Overall X and Y screen positioning • The assignment of objects and/or effects *See §2.1.5 in the Transform Option Handbook (Pub. No. 2A01993) for more information on the Event window.*	Mode **N - Update parameters with no compile** **C - Update parameters and compile** **D - Update parameters, compile, then display Frame 0** **Q - Query based on Script name. Parameters following effect name are ignored on input, and the current values are returned in reply.** Event Number **Range: 1 - Total number of events** Start Frame **Range (fields): -10,000 - Script Length** End Frame **Range (fields): -10,000 - Script Length** X Offset **Range: -999.9 - 999.9** *Can contain a floating decimal point* Y Offset **Range: -999.9 - 999.9** *Can contain a floating decimal point* Object Name Effect Name

Table 4-11: Advanced Transform (V\4) Commands

Command	Description	Parameters
19	Modifies or adds a Key frame to the specified Effect. If the Key frame already exists, the frame is modified. If the frame exists as a Tween frame, a new Key frame is added.	Mode N - Update parameters with no compile C - Update parameters and compile D - Update parameters, compile, then display Frame 0 Q - Query based on Script name. Parameters following effect name are ignored on input, and the current values are returned in reply. Effect Name Effect Key Frame Range: 0 - [Effect length - 1] Frame Type S - Start E - End C - Control U - cUsp L - Locked T - Tween

Table 4-11: Advanced Transform (V\4) Commands

Command	Description	Parameters
20	Modifies the Warp values of the specified Effect at the specified Key frame. Note that a Tween frame cannot be modified.	Mode **N -** Update parameters with no compile **C -** Update parameters and compile **D -** Update parameters, compile, then display Frame 0 **Q -** Query based on script name. Parameters following effect name are ignored on input, and the current values are returned in reply. Effect Name Effect Key Frame **Range: 0 - [Effect Length - 1]** **The following eight parameters have the same range: -20.0 - 20.0** *They can contain a floating decimal point.* X Upper Left Y Upper Left X Upper Right Y Upper Right X Lower Left Y Lower Left X Lower Right Y Lower Right

Table 4-11: Advanced Transform (V\4) Commands

Command	Description	Parameters
21	Modifies the Position, Rotation Axis or Revolution values of the specified Effect at the specified Key frame. A Tween frame cannot be modified.	Mode N - Update parameters with no compile C - Update parameters and compile D - Update parameters, compile, then display Frame 0 Q - Query based on Script name. Parameters following effect name are ignored on input, and the current values are returned in reply. Effect Name Effect Key Frame Range: 0 - [Effect Length - 1] Type P - Position A - Rotation Axis R - Revolutions X Position, Rotation or Revolution Y Position, Rotation or Revolution Z Position, Rotation or Revolution The ranges of the above three parameters are as follows: X, Y, Z Position Range: -1000.0 - 1000.0 X, Y, Z Rotation Range: -1000.0 - 1000.0 X, Y, Z Revolution Range: -50.0 - 50.0

Table 4-11: Advanced Transform (V\4) Commands

Command	Description	Parameters
22	Modifies the Transparency of the specified Effect at the specified Key frame. A Tween frame cannot be modified.	Mode N - Update parameters with no compile C - Update parameters and compile D - Update parameters, compile, then display Frame 0 Q - Query based on Script name. Parameters following effect name are ignored on input, and the current values are returned in reply. Effect Name Effect Key Frame Range: 0 - [Effect Length - 1] Transparency Range: 0.0 - 255.0 *Can contain a floating decimal point.*
23	Copies the attributes of the source Key frame to the destination Key frame. A Tween frame cannot be copied.	Mode N - Update parameters with no compile C - Update parameters and compile D - Update parameters, compile, then display Frame 0 Q - Query based on Script name. Parameters following effect name are ignored on input, and the current values are returned in reply. Effect Name Source Key Frame Range: 0 - [Effect Length - 1] Destination Key frame Range: 0 - [Effect Length - 1]

4.8.5 Miscellaneous (V\5) Commands

The general syntax for Miscellaneous Special Effects commands is as follows.

V\5\Command\Parameter 1\...
Parameter n\\[checksum]CRLF

Table 4-12: Miscellaneous (V\5) Commands

Command	Description	Parameters
1	Turn Keyboard Input Off NOTE: If the Keyboard Input is turned off via the Intelligent Interface command V\5\1\Keyboard #\\CRLF, and is not turned back on before the end of the Intelligent Interface session, it can be turned back on by rebooting the iNFiNiT! Family system.	Keyboard # 1 - Primary 2 - Secondary
2	Turn Keyboard Input On	Keyboard # 1 - Primary 2 - Secondary
3	Prebuild frame buffer If in Display mode, message is displayed immediately. If in Non-Display mode, a trigger command (such as Trigger Read Effects) is needed to display message.	Keyboard # 1 - Primary 2 - Secondary Buffer # 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3 Message # Mode 0 - Non-displaying 1 - Displaying
4	Set Message Compose active buffer	Keyboard # 1 - Primary 2 - Secondary Buffer # 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3

Table 4-12: Miscellaneous (V\5) Commands

Command	Description	Parameters
5	Set Palette (Palette index and color, variable size, six fields per entry). NOTE: The Color Key #, Palette Index #, Red, Green, Blue and Key parameters of the Set Palette command can be repeated in that order to load multiple palette entries using one command. The command is terminated with a "\\".	Buffer # 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3 Color key # 1 - Red 2 - Magenta 3 - Blue 4 - Cyan 5 - Yellow 6 - Green 7 - White 8 - Black Palette Index # Range: 1 - 31 Red Range: 0 - 255 Green Range: 0 - 255 Blue Range: 0 - 255 Key 0 - On 1 - Off
6	Quick Erase for displaying frame buffer NOTE: Buffer # can be any combination of 1, 2, and 3. For example, V\5\6\1\2\3\\ erases the displaying buffers of frame buffers 2 and 3 accessed by Keyboard 1.	Keyboard # 1 - Primary 2 - Secondary Buffer # 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3
7	Quick Erase for the non-displaying frame buffer NOTE: Buffer # can be any combination of 1, 2, and 3.	Keyboard # 1 - Primary 2 - Secondary Buffer # 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3

Table 4-12: Miscellaneous (V\5) Commands

Command	Description	Parameters
11	Disable Prompts Monitor Updates This speeds up Intelligent Interface operation. The iNFiNiT! Family keyboard is still active, but any keystrokes entered are not reflected on the Message Compose menu. For example, if CNTL R is pressed, the Read Effects menu is activated, however the menu is not displayed on the Prompts Monitor.	Keyboard # 1 - Primary 2 - Secondary
12	Enable Prompts Monitor Updates Keystrokes entered on the iNFiNiT! Family keyboard are reflected on the Message Compose menu.	Keyboard # 1 - Primary 2 - Secondary
13	Prebuild using raw data formatted to a Tab Description message in either displaying or non-displaying buffer of the specified frame buffer. Example: V\5\13\1\1\9876\1\ABC\DEF\\CRLF From keyboard 1, a prebuild is done in frame buffer 1, using Tab Description message #9876. It is built in the displaying buffer. ABC is displayed in the first Tab field. DEF is displayed in the second tab field.	Keyboard # 1 - Primary 2 - Secondary Buffer # 1 - Frame buffer 1 2 - Frame buffer 2 3 - Frame buffer 3 Message # Display Mode 0 - Non-displaying 1 - Displaying Data Separate each field with a "\".

4.8.6 Triggering Effects (V\6)

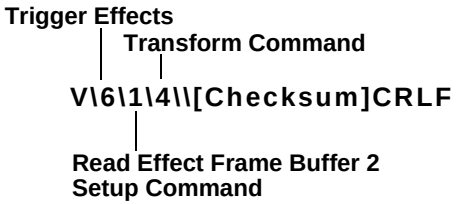
One or more events can be simultaneously triggered using this command. Each event to be triggered is listed by its command number in a separate field after the **6** command.

The general syntax for Triggering Effects commands is as follows.

V\6\Special Effects Command 1\...\Special Effects Command n\\[checksum]CRLF

Example:

To trigger Read Effects for frame buffer **2** and Transform, send the following command string:



To continue execution of a Transform Script after a Pause point, send the following command:

V\6\4\\[checksum]CRLF

To trigger execution of a Read Effect for just frame buffer 1, send the following command:

V\6\0\\[checksum]CRLF

Other examples are included in Appendix C.

4.8.7 Completion Status and Error Handling (V Commands Only)

Command acknowledgment and error responses/ handling are treated differently for the **V** command than for other major commands.

If a checksum is included in the command, it is tested first. If an error exists, an error code is transmitted to the host computer by Intelligent Interface, and the command is not sent to the designated Effects task. Otherwise, a "*****" followed by a carriage return (**CR**) and line feed (**LF**) is transmitted indicating that the command has been accepted and sent to the Effects task.

Intelligent Interface processes a new command only if the previous command has been interpreted, executed, and a response has been sent.

Examples:

If an error is generated from a Read Effects task and there is no checksum error, the following responses are transmitted by Intelligent Interface in response to a **V\1** command:

***<0D><0A>**
V\1\00004001\\<0D><0A>

If no errors are generated, and the entire command is processed successfully, the following responses are sent:

***<0D><0A>**
V\1\\<0D><0A>

Sending an effects trigger command results in a response being sent back for each task that was triggered. If the following trigger command is sent:

V\6\1\4\\<0D><0A>

the Intelligent Interface response, if no errors are detected, is as follows:

***<0D><0A>**
***V\1\\<0D><0A>**
***V\ 4\\<0D><0A>**

4.9 Checksum Calculation

Checksum is used to verify the integrity of the command that is sent to the system, and appears as **[checksum]** in the commands shown in this manual. Note that the brackets (**[]**) are not entered - they simply alert the user that the command is optional.

Checksum is calculated by adding the ASCII value of all characters in the command string starting at the first character and ending at the second backslash. The total is computed as modulo 256 and displayed as a 2 byte hex value in ASCII.

For example:

Command: **M\C/SCRIBE\\CRLF**
Checksum: **907 MOD 256 = 139 decimal = 8B hex**

Command with checksum:
M\C/SCRIBE\\8BCRLF

4.10 Restrictions

- Read Effects cannot be triggered in a frame buffer that is currently being used by Message Compose.
- Transform should not be entered from Message Compose while performing Transform operations with Intelligent Interface.
- No Transform operation can be performed if the current Transform script is at a Pause point. An Escape Pause command (**V\4\7**) should be sent.
- If a **V** command is issued, no other Intelligent Interface commands are processed until the **V** command has completed execution.
- A request for Message Compose completion status, used with **Y**, **Z**, **1** and **3** commands, cannot be used to signal completion of a Roll, Crawl, or Slow Reveal.
- **iNFiNiT! Only:** Miscellaneous commands **3**, **4**, **6**, **7**, and **13** (prebuild and frame buffer change) execute only if the Primary user is running Message Compose.

Section 5 - Serial Port Data Display

This section describes the Serial Port Data Display feature for the iNFiNiT!, MAX!> and MAXINE!

5.1 Introduction

The Serial Port Data Display feature for the iNFiNiT! family of systems supports fast, on-screen display of ASCII data received from the Intelligent Interface serial port. This software is available in all systems configured with the Intelligent Interface option and Message Compose version 3.4 or later. The screen position, font, and color can be easily controlled and modified. Any Machine Font character, including full color RGB icons, can be displayed. Up to 256 independent data fields can be displayed simultaneously, and each data field can consist of up to 63 characters.

5.2 Communication Protocol

Data is input through the Intelligent Interface serial port. This port can be configured as RS422 or RS232. The baud rate, number of bits, parity, and number of stop bits are user selectable in the system Set-up menu. Please *refer to §3.6.6 Set Com (Communications) (SET-UP C) of the iNFiNiT! Family Operation Manual (Pub. No. 2A01976) for detailed information on communications setup.* Software flow control using **XON** and **XOFF** must be used.

5.3 Message Compose Access

To enable Serial Port Data Display:

- From the Message Compose main menu, press **CNTL 6**.

Once enabled, the current frame buffer is dedicated to serial data display.

NOTE

The Real time Data Display software requires all system resources and therefore prevents the user from running other Message Compose or Intelligent Interface functions.

To exit Data Display mode and return to the Message Compose main menu:

- Press **ESC** on the iNFiNiT! Family keyboard, or send the **D\2** command from the host computer.

5.4 Serial Data Display (D) Commands

General command syntax is as follows:

D\Command\Parameter 1\...\Parameter n\\ [checksum]CRLF

Among the parameters that can be set is the data field number. A data field is an area of the screen dedicated to a particular display of information. For example, one data field can be set to display a clock; another to display a station logo; and a third to accept another type of information. The size, location, font and font color in each data field can all be set, as well as whether the data field is displayed in the foreground or background buffer of an Extended Effects Frame Buffer (EEFB). A total of 256 data fields can be set to display on one screen. The following table outlines the **D** commands.

The command codes are as follows.

Table 5-1: Serial Data (D) Commands

Command	Description	Parameters
0	Data Setup	Data Field # - The following parameters affect the specified data field. Font Index - Machine Font to use for display of data (must correspond to a preloaded Message Compose Machine Font) Range: 1 - 16 Color Index - Color to use for display of data (ignored for RGB characters) 0 - Black 1 - Green 2 - Blue 3 - Cyan 4 - Red 5 - Yellow 6 - Magenta 7 - White x - Starting x coordinate (upper left corner) Range: 0 - 720 y - Starting y coordinate (upper left corner) Range: 0 - 486 (NTSC) 0 - 576 (PAL) x - Ending x coordinate (lower right corner) Range: 0 - 720 y - Ending y coordinate lower right corner) Range: 0 - 486 (NTSC) 0 - 576 (PAL)
1	Data Input Setup	Data field to receive input Range: 0 - 255 ASCII Data to be displayed The ASCII data must correspond to valid characters in the specified Machine Font. The entire area occupied by the data field is always cleared before new data is displayed. Send a single space character to clear the entire field
2	Exit Data Display Exits Data Display mode and returns to Message Compose main menu	None

Table 5-1: Serial Data (D) Commands

Command	Description	Parameters
3	Data Set-Up Command for fastest display of dynamic data (requires Extended Effects Frame buffers). This set-up command is issued instead of the regular D\0 command and can be issued only once per session. All characters in the specified font are preloaded into a frame buffer cache for fast display in the active frame buffer when required. An error message is generated on the Prompts monitor if the frame buffer cache memory is exceeded.	Data Field # - The following parameters affect the specified data field. Range: 0 - 255 Font Index - Machine Font to use for display of data (must correspond to a preloaded Message Compose Machine Font) Range: 1 - 16 Color Index - Color to use for display of data (ignored for RGB characters) 0 - Black 1 - Green 2 - Blue 3 - Cyan 4 - Red 5 - Yellow 6 - Magenta 7 - White x - Starting x coordinate (upper left corner) Range: 0 - 720 y - Starting y coordinate (upper left corner) Range: 0 - 486 (NTSC) 0 - 576 (PAL) x - Not used, set to 0 y - Not used, set to 0

Table 5-1: Serial Data (D) Commands

Command	Description	Parameters
4	Data Display setup command for display of serial data in Background buffer of Extended Effects Frame Buffer (EEFB).	Data Field # - The following parameters affect the specified data field. Font Index - Machine Font to use for display of data (must correspond to a preloaded Message Compose Machine Font) Range: 1 - 16 Color Index - Color to use for display of data (ignored for RGB characters) 0 - Black 1 - Green 2 - Blue 3 - Cyan 4 - Red 5 - Yellow 6 - Magenta 7 - White x - Starting x coordinate (upper left corner) Range: 0 - 720 y - Starting y coordinate (upper left corner) Range: 0 - 486 (NTSC) 0 - 576 (PAL) x - Ending x coordinate (lower right corner) Range: 0 - 720 y - Ending y coordinate lower right corner) Range: 0 - 486 (NTSC) 0 - 576 (PAL)
5\0	Resumes display:	None
5\1	Suspends display: Stops serial data display, but updates global information. For example, if three data fields are sent information, they are updated on screen as the data is received. This may not be visually effective. In order for the updates to occur simultaneously, a Suspend Display command can be sent, followed by the data updates. When the Resume Display command is sent, all data fields update simultaneously.	None

Example 1:

D\0\1\6\2\100\200\250\350\\CRLF

Data field **1** is set up to use Machine Font **6**, color **BLUE** and is displayed on screen starting at pixel x = **100**, y = **200**, and ending at x = **250**, y = **350**.

Example 2:

D\1\1\00:23.45\\CRLF

00:23.45 is displayed on screen according to the last valid setup command for data field 1.

For example, if the command in Example 2 followed the setup command in Example 1, **00:23:45** would be displayed in data field **1**, using Machine Font **6**, color **BLUE** and displayed on screen in an area starting at pixel x = **100**, y = **200**, and ending at x = **250**, y = **350**.

Example 3:

D\4\3\11\6\200\250\300\400\\CRLF

Data field **3** is set up to display in the background buffer of an Extended Effects Frame Buffer (EEFB), using Machine Font **11**, color **MAGENTA**, and displayed on screen starting at pixel x = **200**, y = **250**, and ending at x = **300**, y = **400**.

5.5 Programming Notes

It is strongly suggested that a baud rate of **19200** be used. In addition, the user should test the throughput of the system using "worst case" data to ensure that the iNFiNiT!, MAX!>, or MAXINE! can keep up with the data rate supplied. The number of characters displayed and the complexity and size of the characters are important variables in determining the time required by the system to display the data. Since many factors

contribute to the data throughput, it is left up to the user to determine if the responsiveness of the system is adequate for the specific application.

If the machine font can fit into the memory cache (293 x 768 pixels) and Extended Effects Frame buffers are available, the setup command for fast data display (**D\3**) should be used. This command simply copies the character data from one part of the frame buffer to another without requiring the usual erase and rebuild steps associated with the basic set-up command (**D\1**).

The **x** and **y** coordinates defined in the **D\0** setup command must accommodate the maximum extents of all characters that will be displayed within this field. Character data will not be erased properly if the starting or ending coordinates are not correct.

Section 6 - Embedded Commands

NOTE

This section applies to WiNFiNiT! users only.

Intelligent Interface Commands can be embedded in a text file created using a Windows application, and imported to the iNFiNiT! Family system.

These commands allow imported text files to specify the message number(s) at which to record each page, as well as the Tab Description message(s) used to format them. Color and font can also be set for each Tab field.using these commands. Additionally, the directory where the Tab Description message is found and where the imported files will be recorded can be specified.

6.1 General Command Syntax

The general syntax for these commands is as follows:

\X\parameter\[optional parameter]

- **X** is the letter that identifies the command.
- **parameter** specifies the setting.
- **[Optional parameter]** specifies an optional setting. This setting is not required for command execution.

NOTE

It is essential that the leading backslash and the ending double backslashes are included. Otherwise, the command is treated as text, and is typed on the screen with the rest of the imported text. Additionally, the command is not executed.

If the syntax *within* the command is wrong, the entire command is ignored. If an extra unnecessary parameter(s) is present, the extra parameter(s) is ignored when the command is executed. Note that the syntax of the embedded commands differs from their Intelligent Interface counterparts.

6.2 Write to Message File (W)

The general syntax for the Write to Message File command is as follows:

\W\Message ID #\[Tab Message ID #]

- This command specifies that the text file be written to the specified message ID number. Additionally, a Tab message can be applied to the incoming text.
- If only the message ID number is supplied, the text is written to the specified ID number. If there is too much information to fit on one page, messages are created as needed at the next available ID numbers. Existing messages are not overwritten.
- If the current page is not blank when the **W** command is encountered, the page displayed on screen is recorded at the current message ID number. Then, the Tab Description message specified in the **W** command is read. Finally, the current message ID number is changed to the message ID number specified in the **W** command. When end-of-page is reached, or when another **W** command is encountered, the page will be recorded at the specified message number. As mentioned above, if there is too much information to fit on one page, messages are created as needed at the next available ID numbers. Existing messages are not overwritten.

- If a Tab Description message ID number is specified in the command, it overrides any Tab Description message ID number specified in the **Tab Message Number** field in the Import Windows File as iNFiNIT! Message menu. The Tab Description message ID number specified in the command is used for all pages that are imported following the command, unless it is itself overridden by another **W** command containing a **Tab Description Message ID #**.
- If no Tab Description message ID number is specified, then future pages are built without Tab fields. Therefore, all Tab Field information on the current page is discarded.

6.3 About Tab Field Numbering

Before entering commands that change font or color of Tab fields, make sure that each Tab field on the page has a unique field number. After some editing operations, more than one Tab field can have the same field number.

To renumber a specified Tab field:

[ ] 1. From the Message Compose main menu, press **SET-UP TAB**.

[ ] 2. Place the cursor on the Tab field you want to renumber, then press **O**.

[ ] 3. A prompt will request a new field number. Enter the new field number, then press ⏎.

[ ] 4. Press **Q** or **CNTL** or select **Quit** to exit the Tab Set-Up Utility.

To renumber all Tab fields:

[ ] 1. From the Message Compose main menu, press **SET-UP TAB**.

[ ] 2. Place the cursor on the Tab field you want to renumber, then press **J**.

[ ] 3. Press **Q** or **CNTL** to exit the Tab Set-Up Utility.

The Tab fields renumber from left to right, and top to bottom, in numerical order starting with **0**.

For more information on Tab field renumbering, see §7.4.3.9 and §7.4.3.10 of the iNFiNiT! Family Operation manual.

6.4 Set Text Color (C) and/or Font (F)

The general command syntax for setting text and color is as follows:

\C\Color Index\[Tab Field #]
\F\Font Index \[Tab Field #]

- The color index is a number from 1 and 8 inclusive, corresponding to the Color Keys on the system keyboard as follows:

  1 - Red 3 - Blue 5 - Yellow 7 - White
  2 - Magenta 4 - Cyan 6 - Green 8 - Black

- The Font Index is a number from 1 to 16 inclusive. Index 1 through 8 correspond directly to the Font Key number on the iNFiNiT! Family keyboard. Index 9 through 16 correspond directly to the Font Keys used with ALT.

These commands specify the color and font respectively to use when building the next page of the imported text file. When a Tab field number is not specified, this command affects all fields on all pages following the command.

If a Tab field number is specified, it affects *only* the specified Tab field on the first page to which the command applies, not following pages. The Tab field number must be set for each command separately if changing the color and font of a particular Tab field.

The exceptions are as follows:

- If a Tab field in the specified Tab message has **Col/Font** set, it can only be overridden by a **C** or **F** command which:

  [ ] A. Specifies a particular Tab field as explained above.

  [ ] B. Is issued without the optional Tab field number parameter, but while the cursor is within the desired Tab field.

  *For more information on color and/or font assignment, or setting **Col/Font**, see §7.4.3.3 and §7.4.3.4 respectively of the iNFiNiT! Family Operation manual.*

6.5 Select Message Directory (M)

The syntax for the Select Message Directory command is as follows:

M\Absolute Path

This command specifies the Message Directory used for recording future pages (messages), as well as reading Tab Description messages.

6.6 Example

The following illustrates a typical procedure to create several Tab Description messages using the WiNFiNiT! Import feature:

[ ] 1. In the Message Compose main menu, load fonts into Font Keys **1** through **5**. Make **Font Key 1** and **Color 6** (green) current. Using Block Message Utility (**CNTL L D**), delete files **201** - **225**.

[ ] 2. Using Message Compose, create, then record at ID **250** a Tab message containing no text, that consists of four Tab fields. These fields can contain any field attributes (**Color**, **Font**, **Justify**, **Col/Font**, etc.) that can be set in the Tab Set-Up Utility. *See §7.4 of the iNFiNiT! Family Operation manual for more information on setting Tab fields.* Arrange the fields as follows:

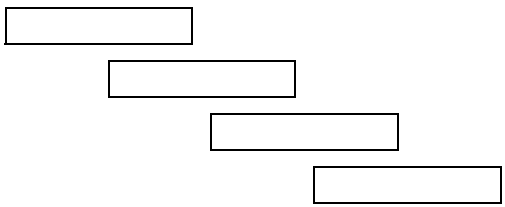


Figure 6-1. Tab Message

[ ] 3. On the PC, open a word processor and create a text document as shown. It is important that **TAB** characters separate each of the text entries, i.e. between **one** and **two**, **two** and **three**, **three** and **four**, etc. Otherwise, the text will not be placed in the correct Tab fields upon importation. A **TAB** character need not be typed directly after a **W** command, as the first text encountered is automatically placed in the first **TAB** field.

\W\201\250\\one two three four five six seven eight\W\205\250\\12 Fahvre QB Packers\W\220\250\\15 Jones \F\5\2\\RB Packers 16 xxx xxx xxx xxx 17 \C\2\\\F\4\\Lofton WR Packers

[ ] 4. Save this as a text (**.txt**) file named **Football** on the PC.

[ ] 5. From the Message Compose main menu, select **Edit/Impor**t from the WiNFiNiT! pull-down menu. The Import Windows File as iNFiNiT! Message Menu (Figure 6-2) is displayed.

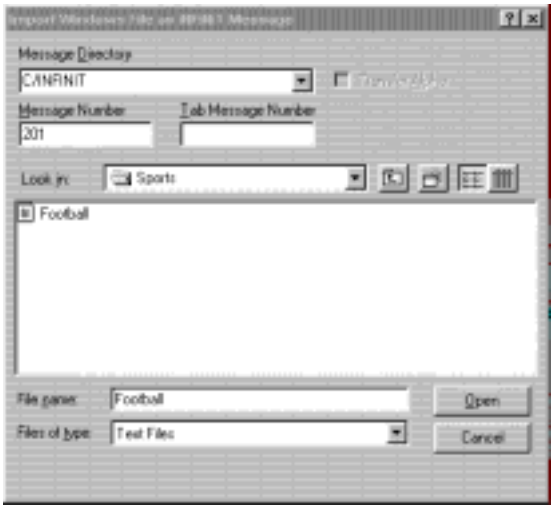


Figure 6-2. Import Windows as iNFiNiT! Message Menu

[ ] 6. Enter the **Message Directory**, **Message Number** and, if desired, the **Tab Message Number**. Note that commands that are written into the text can override these settings. For example, an **M** command overrides a **Message Directory** setting in

the menu; a **W** command overrides a **Message Number** setting in the menu; and a **W** command containing a **Tab Message ID#** also overrides a **Tab Message Number** in the menu.

Even if all three fields are being overwritten by embedded commands, it is still necessary to enter an unused **Message Number** in the menu in order for the import to execute.

[ ] 7. Select **Files of type: Text Files** in the menu. Highlight the desired file, in this case, **Football**, then select **Open**.

Message Compose now begins to import each file, displaying the text as it executes. The following files result from the above example. Note that since no color or font were specified, the current color (**6** - green) and font (**1**) are used, unless otherwise assigned, set by **Col/Font**; or changed by an embedded command.

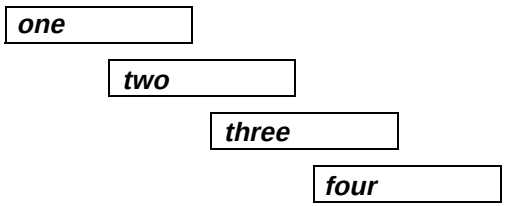


Figure 6-3. Message Number 201

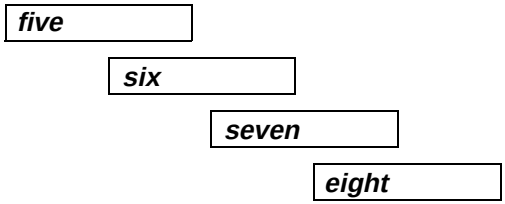


Figure 6-4. Message Number 202

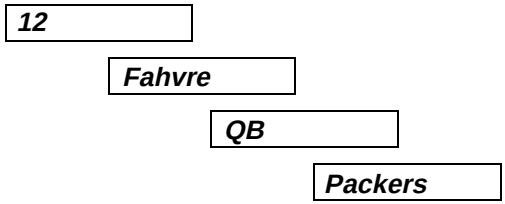


Figure 6-5. Message Number 205

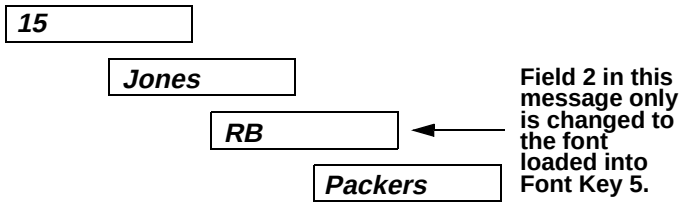


Figure 6-6. Message Number 220

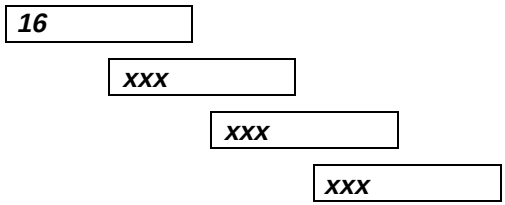


Figure 6-7. Message Number 221

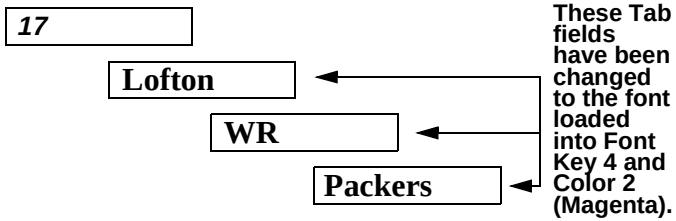


Figure 6-8. Message Number 222

[ ] 8. To display the list of newly created files, press **Y** when the following prompt appears.

Figure 6-9. List Imported Messages?

The list is displayed as follows:

Figure 6-10. List of Imported Files

[ ] 9. If the import executed again without removing these newly created files, the system generates an overwrite (**Overwrite? Y/N**) prompt for each of the files numbers (**201**, **205** and **220**) specified in the embedded commands. If **Y** is selected for each of those flies, the files that are created beyond the specified numbers (**202**, **221** and **222**) would copy to **203**, **223** and **224** respectively. If **N** is selected, the specified files and the added files are copied to the next available ID numbers.

Section 7 - Telnet Operation

7.1 Telnet Introduction

The Intelligent Interface Telnet Server (IITS) provides network users with virtual control of Intelligent Interface operations that were previously limited to console operators, and entities connected through serial port links. The extensibility of control to network hosts opens a new dimension of flexibility for time-critical solutions, convenience of host location, multiple concurrent operators, and new opportunities for third-party automation tool development.

7.2 Similarities/Differences between Telnet and Serial Control Methods

Behavioral differences between serial port connections and Telnet connections are minimal. The Telnet server was designed to be compatible with topologies implemented on serial port connections.

Connections established through serial ports or via Telnet operate identically, with the exception of the following:

- Polling of incoming messages/commands to the IITS is shared among all connections including Telnet clients; however, serial port connections maintain highest priority.
- External Update Requests are no longer limited to the serial ports. All valid connections will receive identical External Update Requests; the requests are disseminated to each connection.

NOTE

Serial port connections are half duplex, forcing the client software to take responsibility for displaying user input data. Telnet clients, on the other hand, are typical full duplex relying on IITS to echo input data back to the client's display. By virtue of the full duplex mode, Telnet clients in Remote Console Mode will appreciate the absence of unnecessary system input data on the client's screen.

7.3 Network Security

Network security is implemented by restricting clients to the same network and requiring appropriate user name and password (a maximum of 5 concurrent sessions is permitted.) Upon successful login, client will receive a "connected" message, signaling system readiness.

7.3.1 Password Protection Setup

NOTE

Complete instructions for Password Protection appear in the Networking Utilities handbook (Pub. No. 2A01991).

For network security purposes, an optional name and password may be entered for any iNFiNiT! Family system on the network. ***The Password Protection setup is available through the Network Utilities program, NOT Intelligent Interface!***

In operation, the name and password of the client system is tested automatically by the host system when a connection is attempted; if password protection is

enabled and the password does not match for the desired operation, a message is sent stating that access to that system has been denied.

NOTE

Access to the Password Protection Setup menu (Figure 7-1) is permitted only if the file <system_name>/ETC/ PASS.DEF is found on Drive A or Drive C. This enables the Network Administrator to control password and protection level changes, by keeping the PASS.DEF file only on the Network Utility disk.

To enable password protection, select **CNTL W** from the Network Utilities menu. The system will respond with Figure 7-1:

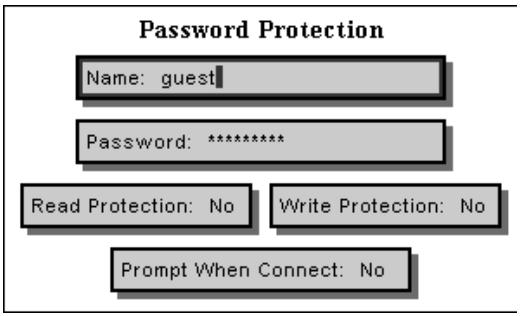


Figure 7-1. Password Protection Menu

Name: Enter the Host Name for this system into this field. This can be of any length up to 20 characters, and may include spaces.

Password: Enter a password for this system, also up to 20 characters including spaces. A "Clear Text" area of asterisks appears in the field. ***You will not see the text!***

Read Protection: If set to **Yes**, the remote user must enter the correct password in order to read from this system. If set to **No**, no password protection is enforced for read operations.

Write Protection: If set to **Yes**, the remote user must enter the correct password in order to write to this system. If set to **No**, password protection is not enforced for write operations.

Prompt When Connect: If set to **Yes**, the system will prompt you for the correct password on connection to the system (or upon an attempt to perform a protected operation.) If this field is set to **No**, you will not be prompted for the password upon connection.

NOTES

The "Prompt When Connect" field exists as a time-saver when you only wish to perform non-protected operations on an otherwise protected system. However, if this field is set to NO, no mechanism for attempting the protected operation(s) on the remote system is available to your system.

If "Prompt When Connect" is set to NO, the system automatically sends the default host name "GUEST" and the password "ANONYMOUS". The default parameters are accepted by CHYRON systems running without Password Protection.

7.3.2 IP Network Types/Masks

The Intelligent Interface Telnet Server (IITS) permits connections to hosts on the same network as the server. Most iNFiNiT! systems are installed on Class B or Class C networks, so these examples are provided.

For complete information regarding IP Network Types, consult your facility's Network Administrator.

7.3.2.1 Class B Network Connectivity

If the host is on a Class B network (i.e. the first three digits of the host Internet address are in the range of **128-191**), then the host permits connections as follows:

Host Address	Valid Client Addresses
188.14.152.21	**188.14.*<don't care>.<don't care>***

If a client on a different network segment (i.e. address **188.13.104.17**) attempts a connection, it will be refused.

7.3.2.2 Class C Network Connectivity

If the host is on a Class C network (i.e. the first three digits of the host Internet address are in the range of **192-223**), then the host permits connections as follows:

Host Address	Valid Client Addresses
207.14.152.21	**207.14.152.*<don't care>***

If a client on a different network segment (i.e. address **207.14.150.17**) attempts a connection, it will be refused.

7.4 Telnet Protocol

The Intelligent Interface Telnet Service utilizes the Telnet Protocol specified in R.F.C 854 to negotiate and communicate with clients. Due to the specialized purpose of the service within the context of the Intelligent Interface, many features are unnecessary and therefore not implemented. Telnet options are basically limited to Binary Data Mode. Enabling Binary mode will enable transmission of all iNFiNiT! Family Keyboard Codes; however, it's not necessary to enter Binary mode for a majority of operations. Binary Data Mode is a feature that may or may not be available on Telnet client software.

Client software will most likely need to map host keyboard codes to the desired system keycodes. Many Telnet client applications are available that provide keyboard mapping. Such off-the-shelf applications will allow system operators to utilize Remote Console Mode over the network immediately. In addition to the convenience of basic RCM operations, many Telnet client applications provide multi-code keyboard mapping; a single keystroke at the client will issue a sequence of commands to the IITS. In essence, the operator potentially has a set of macros that could be used to facilitate navigation and operation of the system.

7.5 Telnet Operation

To begin a session, start a Telnet client application and open a connection to the desired system:

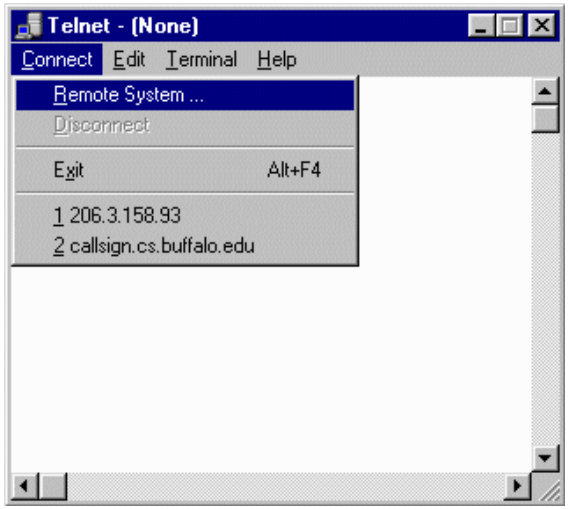


Figure 7-2. Initiating Telnet Connection

Next, provide the IP address of the host system you wish to connect to:

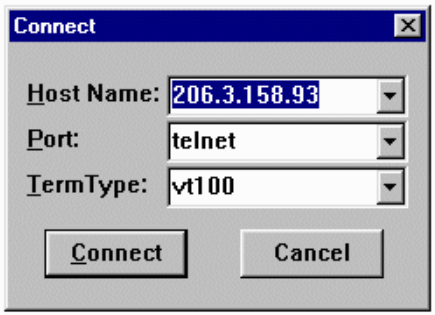


Figure 7-3. Specifying Host Name

When a connection is initiated, the message "Welcome to Chyron Intelligent Interface" is issued followed by a login prompt (Figure 7-4).

Type the user name followed by **ENTER**. A password prompt is returned; provide the password, then press **ENTER** (password characters sent by the user are not echoed back to the client.)

Upon successful validation, the host system issues a "connected" message.

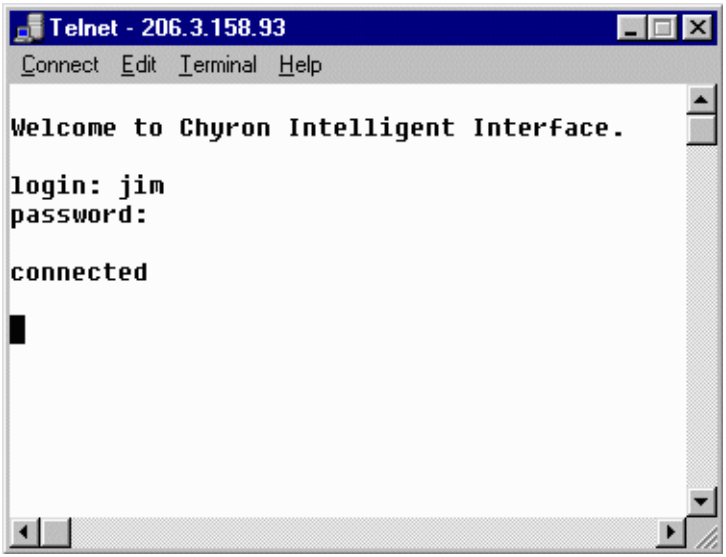


Figure 7-4. Successful Telnet Login

Users will be provided with three opportunities to supply valid identity. After the third failure, the connection will be closed preceded by the message "Telnet access denied, please reconnect."

The next sample pictured shows the result codes, 00004190, of 3 consecutive carriage returns. This simple test of connectivity should be familiar to users who have previously connected through the serial port.

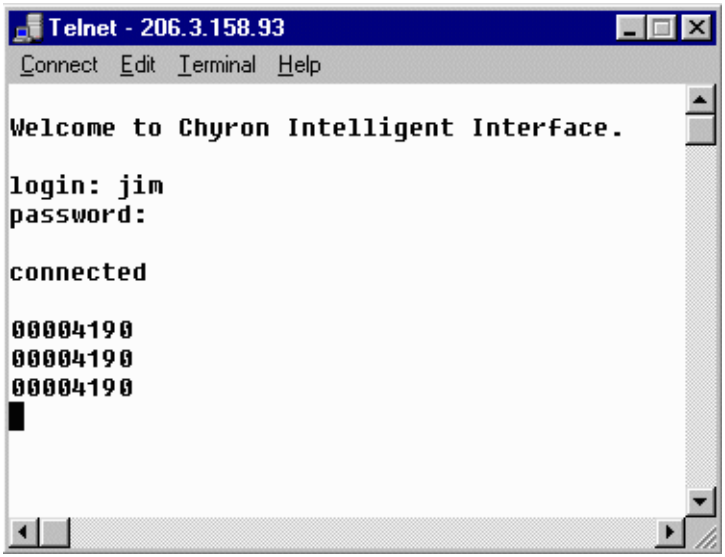


Figure 7-5. Initial Connectivity Test

7.6 User Interaction Examples

Processing of complex or frequently issued commands can be facilitated through the use of scripts. This would require a Telnet client application that features script functionality. For example, a user could create a script that does the following:

```
transmit "V\5\3\1\1\2003\1\\^M^J"
waitfor  "^J"
transmit "V\4\4\SPEEDOMETER1\\^M^J"
waitfor  "^J"
transmit "V\6\4\\^M^J"
```

The waitfor expression positioned between each transmission of a system command, which promotes pacing of data between IITS and the client. At least one response to a command is guaranteed; however, it's not uncommon for a command to initiate multiple responses

from the system. The Telnet server will accept all possible byte values between 0 and 255. When a Telnet session is negotiating in non-binary mode, end-of-line translation will occur. As of the current version, CR-LF, CR-NULL, and CR will be interpreted as CR. LF is generally ignored.

The system requires that commands must be terminated with a CR byte. CR represents byte value of 13 and LF represents byte value of 10. For the majority of cases, non-binary mode negotiation will suffice; the default mode for most Telnet client applications. For display purposes, clients will terminate commands with CR-LF.

Remote Console Mode sessions may at times require that a CR-LF pair not get converted to CR. To accomplish this, enable Telnet binary-mode negotiation. RCM over a typical Telnet session is sufficiently operational without the need for binary-mode data transfer.

As stated earlier, many Telnet client applications provide keyboard mapping. Mapping is designed to transmit data that is not normally associated with a particular key. For example, if the **ENTER** key under normal conditions generates a byte value of 13, keyboard mapping can result in the transmission of a byte value of 202. In order to emulate the system keyboard, mapping a Telnet client keyboard will be necessary.

7.7 Terminating a Telnet Session

To terminate a Telnet session, simply use the client software's Disconnect command:

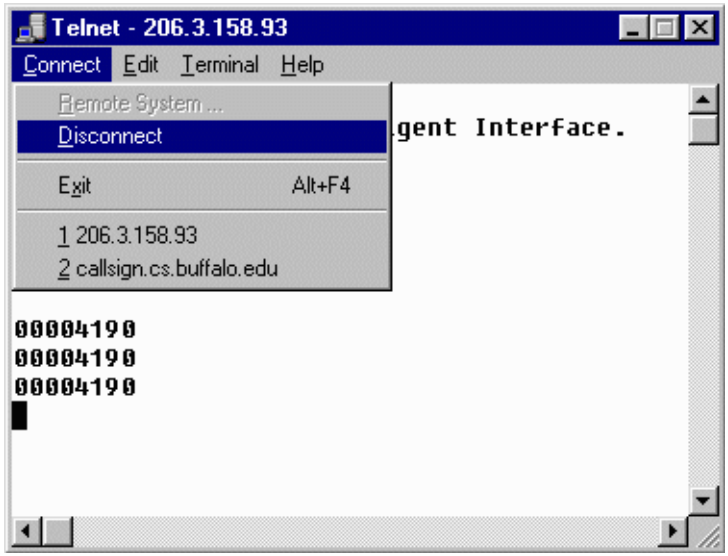


Figure 7-6. Terminating Telnet Session

Appendix A - Error Responses

If a checksum error or other error is detected, a 4-byte ASCII error code is transmitted, followed by a carriage return and line feed. If no error is detected and the command was processed successfully, an asterisk (ASCII 42) followed by a carriage return and line feed is transmitted.

Table A-1: Return Status and Error Codes

Code	Explanation
0001	Task timed out
0002	Unimplemented system service
0003	Illegal service
0004	Illegal node
0005	Object has been deleted
0006	Invalid object ID
0007	Incorrect object ID
0008	Incorrect object type
0009	Object not found
0014	Cannot create message - No task control blocks.
0015	Cannot create message - No stack space available.
0016	Cannot create the message -The stack space is too small.
0017	Cannot create the message -Priority out of range.
0018	Cannot start the process - Already active.
0019	Cannot restart the process - The process was never started.
0020	Cannot suspend process - Process is already suspended.
0021	Cannot resume process - It was never suspended.
0022	Cannot change priority - Out of range.
0023	Illegal task register number.
0024	Cannot delete - File is open.

Table A-1: Return Status and Error Codes

Code	Explanation
0051	Cannot create - No more queue control blocks.
0052	Cannot create or send - No message buffers.
0053	Cannot send - The message queue is full.
0054	Cannot receive - The queue has been deleted.
0055	Cannot Receive - There are no pending messages.
0056	Informative - There were tasks waiting when the queue was deleted.
0057	Informative - There were messages waiting when the queue was deleted.
0065	Cannot create - No more SCBs.
0066	Cannot acquire - Semaphore not available.
0067	Cannot acquire - Semaphore deleted while waiting.
0068	Informative - There were tasks waiting when the Semaphore was deleted.
2001	An invalid function number was called.
2002	pHILE failure was encountered.
2003	An inconsistent structure was found on the volume.
2004	No more TCB extensions.
2005	Init volume parameter error.
2006	No more volume table entries.
2007	Alien volume.
2008	Volume already mounted.
2009	Cannot unmount - there are files open.
200A	Device not mounted.
200B	File not found.
200C	Illegal filename.
200D	No default directory found.
200E	Directory file expected.
200F	Illegal annex size to create.
2010	Null filename.
2011	File already exists.
2012	Flist full.

Table A-1: Return Status and Error Codes

Code	Explanation
2013	File open.
2014	Directory is not empty.
2015	System or directory file.
2016	Different devices (MOVE_F).
2017	MOVE_F would result in non tree structure.
2018	No more open file entries.
2019	No more FCBs.
201A	Invalid FID, too large.
201B	Invalid FID, File closed.
201C	Indirect block full.
201D	Volume full.
201E	Bad position to lseek.
201F	Seek past end of file.
2021	Illegal device (exceeded maximum.)
2022	Data is locked.
2023	Bad fn in OPEN_VOL.
2024	Illegal flush mode to MOUNT_VOL.
2025	Illegal device name.
2026	Bad MS/DOS call.
2027	Illegal MS disk type.
2028	More than one type of MS disk type.
2040	Used Tab page
2F01	Insufficient data area.
2F0E	Check sum in pHILE.
2F0F	Check sum in pVERIFY+.
400B	Checksum doesn’t match computed checksum.
40B2	Requested message directory doesn’t exist.
40B3	Requested message file does not exist
412C	Requested description file isn’t a tab description file.
412D	Requested data file isn’t a tab data file.
412E	Requested field isn’t a valid field.

Table A-1: Return Status and Error Codes

Code	Explanation
412F	Requested attribute (color or font) isn't valid.
4130	Field contains too much data (limit 64 characters).
4131	Incorrect number of fields, R command doesn't match X command.
415E	Description message was created with old tabs.
415F	Number of tabs exceeds limit (99).
4190	Desired operation isn't terminated with '\\'.
4191	The command format is incorrect.
4192	An invalid operation has been requested.
4193	An invalid Machine ID has been used.
4194	An invalid Effects ID has been used.
41C2	An invalid user ID has been used (either 1 or 2).
41C3	An invalid keyboard ID has been used (either 1 or 2).
41C4	The command to set the active message compose frame buffer failed.
41C5	An invalid frame buffer ID was used (either 1, 2, or 3).
41C6	Error building into a non-displaying buffer (a bad message type was used or no read next command was issued.)
41C7	Command to set palette failed.
41C8	Color Key out of range (must be 1-8).
41C9	Palette index out of range (must be 1-31).
41CA	Color index out of range (must be 0-255).
41CB	Key out of range (must be 0-1).
41CC	Attempted to perform a Dual-User operation in Single-User mode.
41CD	Timed out waiting for response.
41CE	Error building Message Compose palette.
41CF	Error erasing displayed frame buffer.
41D0	Display/Non-Display flag out of range (i.e. 0-1).
41D1	No Intelligent Interface option.
5000	New Play list Created

Table A-1: Return Status and Error Codes

Code	Explanation
5001	Invalid Play List Number
5002	No Play List Loaded
5003	File Not Play List
5004	Old Version Play List
5005	Play List Already Exists
5006	Play List Finished Executing
5010	Invalid Play List Entry Number
5011	Error Allocating Play List Entry
5012	Error Adding Play List Entry
5020	Invalid Channel Allocated
5021	No Channel Allocated
5022	No Frame Buffer Allocated
5030	Invalid Still Message Number
5031	Still File Not Found
5050	Invalid IMAGESTOR! Command
5051	Invalid Effect
5052	Invalid Pattern
5053	Invalid Speed
5054	Invalid Ease Parameter
5055	Invalid Reverse Parameter
5056	Invalid Keyboard Lock Status
8001	Wrong Read Effect task.
8002	Bad Read Effect command.
8003	Wrong Read Effect command.
8004	Bad Read Effect mode.
8005	Bad Read Effect ease.
8006	Bad Read Effect speed.
8007	Bad Read Effect type.
8008	Bad Read Effect reverse.
8009	Bad Read Effect row direction.
800A	Bad Mix Effect Percent.

Table A-1: Return Status and Error Codes

Code	Explanation
800B	Bad Mix Effect Soft Edge.
800C	Rows cannot be wiped left and right on old frame buffer.
800D	Must specify one row up or down.
800E	Invalid effect type.
800F	No memory available for row type.
8010	Frame buffer error occurred.
8011	Frame buffer in use by another task (i.e. clock, flash)
805001	Bad Transform syntax.
805002	Bad Transform script.
805003	Bad Transform directory.
805004	Bad Transform message.
805005	Bad Transform value.
805006	Ignored.
805007	Bad Transform command.
805008	Transform not available.

Appendix B - Reference Tables

Table B-1 of this appendix provides a listing of all ASCII and hexadecimal codes for system characters and commands. Table B-2 contains system keyboard ID numbers and hexadecimal codes.

Table B-1: ASCII Codes

CHAR	DECIMAL	HEX	CHAR	DECIMAL	HEX
NUL	000	00	EM	025	19
SOH	001	01	SUB	026	1A
STX	002	02	ESC	027	1B
ETX	003	03	FS	028	1C
EOT	004	04	GS	029	1D
ENQ	005	05	RS	030	1E
ACK	006	06	US	031	1F
BEL	007	07	SP	032	20
BS	008	08	!	033	21
HT	009	09	"	034	22
LF	010	0A	#	035	23
VT	011	0B	¤	036	24
FF	012	0C	%	037	25
CR	013	0D	&	038	26
SO	014	0E	'	039	27
SI	015	0F	(	040	28
DLE	016	10	)	041	29
DC1(XON)	017	11	*	042	2A
DC2	018	12	+	043	2B
DC3 (XOFF)	019	13	,	044	2C
DC4	020	14	-	045	2D
NAK	021	15	.	046	2E
SYN	022	16	/	047	2F
ETB	023	17	0	048	30
CAN	024	18	1	049	31

Table B-1: ASCII Codes

CHAR	DECIMAL	HEX	CHAR	DECIMAL	HEX
2	050	32	S	083	53
3	051	33	T	084	54
4	052	34	U	085	55
5	053	35	V	086	56
6	054	36	W	087	57
7	055	37	X	088	58
8	056	38	Y	089	59
9	057	39	Z	090	5A
:	058	3A	[	091	5B
;	059	3B	\	092	5C
<	060	3C	]	093	5D
=	061	3D	^	094	5E
>	062	3E	-	095	5F
?	063	3F	‘	096	60
@	064	40	a	097	61
A	065	41	b	098	62
B	066	42	c	099	63
C	067	43	d	100	64
D	068	44	e	101	65
E	069	45	f	102	66
F	070	46	g	103	67
G	071	47	h	104	68
H	072	48	i	105	69
I	073	49	j	106	6A
J	074	4A	k	107	6B
K	075	4B	l	108	6C
L	076	4C	m	109	6D
M	077	4D	n	110	6E
N	078	4E	o	111	6F
O	079	4F	p	112	70
P	080	50	q	113	71
Q	081	51	r	114	72
R	082	52	s	115	73

Table B-1: ASCII Codes

CHAR	DECIMAL	HEX	CHAR	DECIMAL	HEX
t	116	74	z	122	7A
u	117	75	{	123	7B
v	118	76	|	124	7C
w	119	77	}	125	7D
x	120	78	-	126	7E
y	121	79	DEL	127	7F

Table B-2: iNFiNiT! Family Keyboard Codes

KEY	DECIMAL	HEX	KEY	DECIMAL	HEX
MODE SLCT	255	FF	BLK	175	AF
ESC	254	FE	Keypad 0	192	C0
FONT 1	160	A0	Keypad 1	193	C1
FONT 2	161	A1	Keypad 2	194	C2
FONT 3	162	A2	Keypad 3	195	C3
FONT 4	163	A3	Keypad 4	196	C4
FONT 5	164	A4	Keypad 5	197	C5
FONT 6	165	A5	Keypad 6	198	C6
FONT 7	166	A6	Keypad 7	199	C7
FONT 8	167	A7	Keypad 8	200	C8
RED	168	A8	Keypad 9	201	C9
MGTA	169	A9	Keypad CLR	202	CA
BLUE	170	AA	DEL ROW	218	DA
CYAN	171	AB	INS ROW	219	DB
YEL	172	AC	CNTR ROW	220	DC
GRN	173	AD	DEL CHAR	221	DD
WHT	174	AE	INS CHAR	222	DE
CNTR PAGE	223	DF	⏎ (NEWLINE)	206	CE
ALT	207	CF	←	140	8C
ERASE	205	CD	→	141	8D
DEL MSG	240	F0	↓	142	8E
RCD	241	F1	↑	143	8F

Table B-2: iNFiNiT! Family Keyboard Codes

KEY	DECIMAL	HEX
F1	208	D0
F2	209	D1
F3	210	D2
F4	211	D3
F5	212	D4
F6 (SHIFT + F1)	224	E0
F7 (SHIFT + F2)	225	E1
F8 (SHIFT + F3)	226	E2
F9 (SHIFT + F4)	227	E3
F10 (SHIFT + F5)	228	E4
CNTL	213	D5
CHNG	247	F7
XFER	249	F9
HOME	203	CB
COLOR FONT	204	CC
SHIFT CHAR + ←	136	88
SHIFT CHAR + →	137	89
SHIFT CHAR + ↓	138	8A
SHIFT CHAR + ↑	139	8B
SHIFT + SHIFT CHAR + ←	152	98
SHIFT + SHIFT CHAR + →	153	99
SHIFT + SHIFT CHAR + ↓	154	9A
SHIFT + SHIFT CHAR + ↑	155	9B
SHIFT ROW + ←	128	80
SHIFT ROW + →	129	81
SHIFT ROW + ↓	130	82
SHIFT ROW + ↑	131	83
SHIFT + SHIFT ROW + ←	144	90
SHIFT + SHIFT ROW + →	145	91
SHIFT + SHIFT ROW + ↓	146	92
SHIFT + SHIFT ROW + ↑	147	93
SHIFT PAGE + ←	132	84

Table B-2: iNFiNiT! Family Keyboard Codes

KEY	DECIMAL	HEX
SHIFT PAGE + →	133	85
SHIFT PAGE + ↓	134	86
SHIFT PAGE + ↑	135	87
SHIFT + SHIFT PAGE + ←	148	94
SHIFT + SHIFT PAGE + →	149	95
SHIFT + SHIFT PAGE + ↑	151	97
SHIFT + SHIFT PAGE + ↓	150	96
CNVRT (FLASH)	214	D6
MIX	233	E9
SET-UP	234	EA

KEY	DECIMAL	HEX
DO	243	F3
SLOW REVL	245	F5
READ NEXT	246	F6
CRAWL	216	D8
READ	248	F8
ROLL	217	D9
IMAGESTOR!* (Key left of SPACE BAR)	235	EB
SCSI Express† (key right of SPACE BAR)	236	EC
MAXINE! Preview‡ (key left of alphanu-meric 1)	231	E7

** Available only on an iNFiNiT! Family system equipped with the IMAGESTOR! option.*

† Available only on a MAX!> or MAXINE! system equipped with the SCSI Express option.

‡ Available only on a MAXINE! system equipped with the MAXINE! Preview option.

Appendix C - Sample Commands

The following shows how to set up and execute Multiple Effects via Intelligent Interface. In this sample, a Transform Script and a Read Effect are triggered. §C.1 explains the Transform setup; §C.2 covers the Read effect setup; and §C.3 explains how to trigger these effects.

Paragraph §C.4 describes how to execute a Mix Effects Flash sequence (on iNFiNiT! or MAX!> systems only.).

C.1 Transform Effect Setup

The following commands set up the Transform. The Transform display is erased; the Transform Script is loaded, then an object and effect are bound (assigned) to an event.

```
V\4\12\\CRLF
V\4\4\WEATHER\\CRLF
V\4\8\5\SUN\SPIN\\CRLF
```

Each line of the command is explained in Table C-1.

Table C-1: Transform Setup Commands

Command	Explanation
LINE 1	
V\4	Special Effects Transform command
12	Erase Transform display
LINE 2	
V\4	Special Effects Transform command
4	Load Script
WEATHER	Name of Script to be loaded
LINE 3	
V\4	Special Effects Transform command
8	Binds Object (SUN) and Effect (SPIN) to Event (5)
5	Event #5

Table C-1: Transform Setup Commands

Command	Explanation
SUN	Name of object to be assigned to Event #5
SPIN	Name of Effect to be assigned to Event #5

C.2 Read Effect Setup

This set of commands sets up a Read Effect in Frame Buffer 2; then changes active buffer in Message Compose to Frame Buffer 1.

```
V\1\1\W\G\30\0\0\1\1\1\1\\
V\5\3\1\2\4000\\
V\5\4\1\1\\
```

Table C-2: Read Effect Setup Commands

Command	Explanation
LINE 1	
V\1	Special Effects Read Effects command for frame buffer 2
1	Set up the Read Effect
W	Effect - Wipe
G	Pattern - G (Scanline Wipe)
30	Speed (Duration) of Effect execution = 30 fields
0	No Ease In or Out
0	Reverse On
1	First row Down
1	Second row Down
1	Third row Down
1	Fourth row Down
LINE 2	
V\5	Special Effects Miscellaneous command
3	Prebuild the non-displaying buffer
1	Keyboard (User) #1 prebuilds
2	Frame buffer 2 with...
4000	Message number 4000.
LINE 3	
V\5	Special Effects Miscellaneous command

Table C-2: Read Effect Setup Commands

Command	Explanation
4	Set Message Compose active buffer
1	Keyboard (user) #1 for...
1	Frame buffer #1.

C.3 Execute Multiple Effects

The next set of commands send text to frame buffer **1**; and the Read Effect and the Transform Script are triggered. All three operations run simultaneously.

```
Y\Sunny weather today!\\CRLF
V\6\0\4\\CRLF
```

Table C-3: Execute Setup Effects Commands

Command	Explanation
LINE 1	
Y	Sends text "Sunny weather today!" to the active frame buffer
LINE 2	
V\6	Special Effect Multiple Effects command
0	Trigger Read Effect set above
4	Trigger loaded Transform Script

Frame Buffer 1 remains accessible by Keyboard 1 during effect execution.

C.4 Execute Mix Flash Sequence iNFiNiT! and MAX!> Only

Executing the Mix Effects Flash command causes Mix Effects to alternate the channel's display back and forth between the two frame buffers or a frame buffer and a blank screen, each one displayed for a specified duration. The flashing continues until the host computer sends a stop command.

Although Mix Effects command (**V\3\....**) stops the flashing, it is recommended that the following stop command be used:

V\3\3

Once executing, the Flash effect remains active even on exit from Message Compose to the Program Loader menu. When Flash is running, the user should not attempt to perform operations in the Mix Settings window in the Multiple Effects menu, nor should the **Mix** line in the Multiple Effects window be activated.

The syntax of the Mix Effects Flash setup command is as follows:

V\3\2\Channel\frame buffer 1\duration 1\ frame buffer 2\duration 2

To execute Flash, send the following command:

V\3\2\1\1\120\4\60

This causes the contents of frame buffer **1** to display in Channel **A AIR** for two seconds, then to flash to a blank screen for **1** second. This sequence repeats until a Stop Flash command is sent.

Executing the command:

V\3\2\2\2\120\3\60

Causes the contents of frame buffer 2 to display in Channel B air for two seconds, then to flash to frame buffer 3 for 1 second, and so on until a Stop command is received.