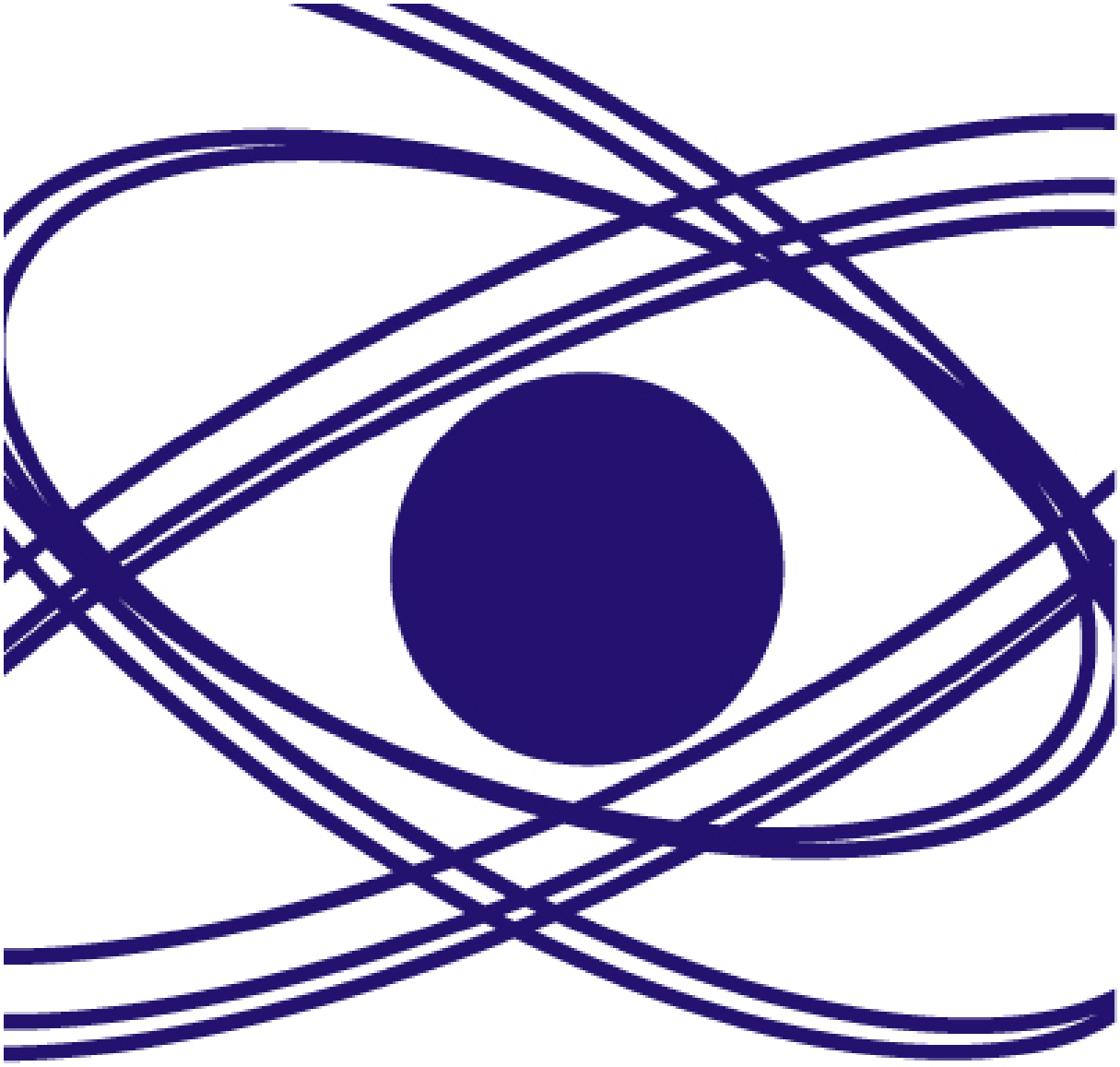


Tapestry

Graphical User Interface for the

Navigation Laboratories, Inc.
GPS Constellation Simulators

USERS GUIDE – Remote Control Interface

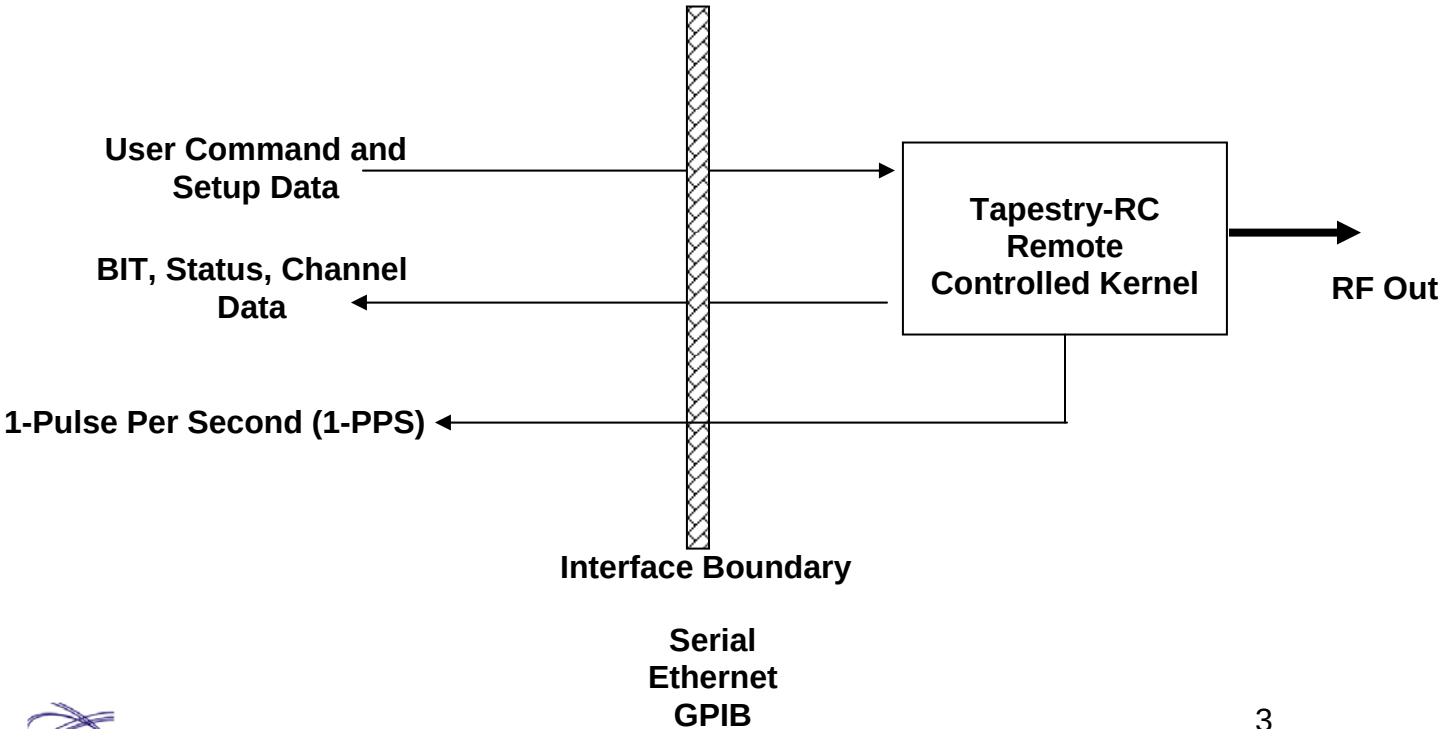
1 Scope

The Tapestry Constellation Simulator can be operated in one of three configurations: File Based, Closed Loop, and Remote Control.

In File Based configuration, Tapestry generates simulated vehicle motion truth data via the supplied ***Build Scenario*** application. ***Build Scenario*** provides all of the vehicle modeling required to construct line-of-sight GPS measurement data. See the Users Guide for the Build Scenario and Run Scenario Application for a description of this operational configuration.

In Closed Loop configuration, an external user provided process generates the simulated vehicle motion trajectory using a high speed data bus such as GPIB, Ethernet, or SCRAMNet. In this mode precise timing relationships between the Tapestry and external process must be maintained. See the Users Guide for the Closed Loop Operation for a description of this operational configuration.

In Remote Controlled configuration, The Tapestry system provides a framework for command and control of the simulator in real-time using an external data bus such as RS232, GPIB, or Ethernet. In Remote Controlled mode, the user loads basic parameters into the system and Tapestry computes the GPS related parameters automatically with little or no input from the user. This document describes the Remote Control Interface (RCI) for the Tapestry Constellation GPS Simulator. The interface between the Controller and the Tapestry Simulator can be configured to be GPIB, RS232, or Ethernet protocol. In all configurations, a one-pulse-per-second (1-PPS) output will be present and can be used for synchronization if desired.



2 Remote Controlled Simulator Modes

If configured with the Remote Controlled feature, the simulator, when powered up and ready, will be in one of three operational modes:

- IDLE
- RUNSCEN
- RUNLOCAL.

In IDLE, the simulator is in a ready state with no GPS signals being generated and the 1PPS is not transmitted. This is the default mode on power up. RUNSCEN mode is active when the simulator is executing a pre-built scenario resident on the computer hard disk. The 1-PPS will be active in this mode. RUNLOCAL mode is active when the simulator is running real-time based upon user initialization commands. In this mode the 1-PPS is output.

In any of these modes, the simulator will accept input messages to transition between modes. Not all messages can be processed in all modes, with certain messages resulting in an error if the simulator receives them in an incorrect configuration. For example, if the SETPOWER message is received while in IDLE, the simulator will respond with an ACK message of *SIMMODEINVALID*.

The message set and associated allowable configurations will be discussed subsequently.

3 Hardware Protocol

There are three allowable message hardware interface types: ***GPIB***, ***Ethernet***, ***RS232***. Each implements the same message set and format with device settings peculiar to each hardware interface type.

3.1 GPIB Interface

The Tapestry-RC system will be delivered with a PCI GPIB card installed and configured with the primary address set to 15.

The simulator is setup as a GPIB slave device with a configurable primary address.

- The interface is configured within the text file TapRci.ini. This file provides access to the base address and other parameters and is located in the Windows directory. You should not edit this file unless directed to by Navigation Laboratories or one of its representatives. A description of the file and parameters is in the TapRci section of this document

- Tapestry-RC will accept messages from the GPIB controller at any time.

- When transmitting data, Tapestry-RC will raise the Service Request line (SRQ) and waits to be addressed to talk.

3.2 RS232 Interface

The serial port parameters can be changed by the user be editing the TapRci.ini file. The default parameters are

- COM Port 1
- Baud = 19200
- No Parity
- 8 Bits (this parameter cannot be changed)
- 1 Stop Bit

It is important to remember that a NULL modem must be used if a PC is being used to communicate with the simulator.

3.3 Ethernet Interface

The Ethernet parameters are also stored in the TapRci.ini file. Communication between the external user and the simulator is done through sockets. The simulator performs a socket listen while the external user performs a socket connect. Note that the network needs to be working properly before any simulations can be run. The Network Interface Card (NIC) is already installed, but the network parameters must be set for your particular network (addresses, masks, etc.).

4 Message Protocol

4.1 General Message Format

The following message formats apply irrespective of the interface protocol implemented. All communication has an optional 2 byte hex ASCII checksum appended to the message.

The message structure is constrained to the following format.

$aaaaaa,xxxxx,….,xxx*hh<CR><LF>

ASCII	HEX	Description
“$”	24	Start of message
aaaaaa		Message identifier
xxxx		Message specific data. The format of this data is determined by the

		message identifier. Data fields are of variable length and are preceded by the delimiter "," (a comma).
","	2C	Field Delimiter. Starts each field except Message Identifier and checksum fields. If it is followed by a null field (two commas in a row) this indicates that there is no data in this field.
"*"	2A	Checksum Delimiter. Follows the last data field of the sentence. It indicates that the following two alphanumeric characters show the HEX value of the checksum.
hh		ChecksumField. The absolute value calculated by exclusive or'ing the 8 data bits of each character in the message, between, but excluding "$" and "*". The hexadecimal value of the most significant and least significant 4 bits of the result are converted to two ASCII characters (0-9, A-F (upper case)) for transmission. The most significant character is transmitted first.

4.2 Controller to Simulator Messages

4.2.1 Start Scenario (STARTSCEN)

This message immediately starts a pre-built scenario resident on the simulator hard disk. It is only valid in IDLE mode.

Field No	Symbol	Field Description	Field Type	Example
	$STARTSCEN	Start of message		$STARTSCEN
1	SCENNAME	Scenario Name	A	ENDOFWEEK
2	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$STARTSCEN,IBMThroughput*1B

A scenario is a pre-built file set resident on the Tapestry computer system. The scenario must be located within a folder off of the main scenario run directory:

C:\voyager\runs

When delivered, your system will contain several pre-built scenarios that include as a minimum, c:\voyager\runs\default – referenced as DEFAULT in the start scenario command, c:\voyager\TapRemote – referenced as TapRemote, and c:\voyager\runs\static - referenced as STATIC for the Start Scenario command. Other scenarios may be present on your system. When executing a scenario, the input data is pre-defined and repeatable run to run.

4.2.2 Initialize Geodetic (INITGEO)

The Tapestry system, as a minimum, requires an initial start position and time. In a scenario this data is contained within the file set. In a real-time configuration this data is supplied by this message. This message initializes the simulator with the given position and time. It does not start the simulator. It is used to change the position simulated by the hardware. It is accepted in any mode.

Field No	Symbol	Field Description	Field Type	Example
	$INITGEO	Start of message		$INITGEO
1	LAT	Latitude (deg)	ll.llll	34.1234
2	LAT_REF	Latitude Direction (N, S)	A	N
3	LON	Longitude (deg)	yyy.yyyy	118.001
4	LON_REF	Longitude Direction (E,W)	A	W
5	ALT	Altitude (m)	xx.xxx	12.23
6	TIME	UTC Time (Hours, Minutes, Seconds)	hhmmss	130405
7	DATE	UTC Date	ddmmyyyy	25122002
8	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$INITGEO,34.0000000,N,118.0000000, 0.00000,133821,12032002*45

4.2.3 Start Local (STARTLOCAL)

This message starts the simulator with the initialization data received. This message is valid only in IDLE mode.

Field No	Symbol	Field Description	Field Type	Example
	$STARTLOCAL	Start of message		$STARTLOCAL
1	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$STARTLOCAL*0D

4.2.4 Set SVID (SETSVID)

Nominally, in either **Local** (RUNLOCAL) or **Scenario** (RUNSCEN) mode, Tapestry will determine and output the GPS satellites based upon the elevation mask angle and current user position autonomously. If desired, the remote user may command the assigned satellites on a per channels basis. Nominally there are 12-14 available channels.

This message sets specific channels to commanded satellites (SVIDs). Channels not specified will transmit satellites based on the normal satellite selection algorithm of the simulator contained within the TapGPIB scenario. This message is valid only in RUNLOCAL mode.

Field No	Symbol	Field Description	Field Type	Example
	$SETSVID	Start of message		$SETSVID
1	CHAN	Channel	C	2
2	SVID	Svid to transmit on preceding fields channel	cc	24
3	…	…	…	…
4	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$SETSVID,1,0,2,1,3,2,4,3,5,4,6,5,7,6,8,7,9,8,10,9,11,10,12,11*79

Setting a channel to a specific satellite will cause the simulator to override it's normal satellite selection algorithm for that channel and use the override SVID until commanded otherwise. An SVID of 0 will turn off the specified channel. An SVID of –1 will turn off the override and allow the simulator to assign the channel with it's normal satellite selection algorithm. Note, the commanded satellite will be transmitted – and continually transmitted – rather visible or not until overridden or de-assigned by the remote user.

4.2.5 Set Power (SETPOWER)

Nominally, in either **Local** (RUNLOCAL) or **Scenario** (RUNSCEN) mode, Tapestry will determine and output the GPS satellites with an output power determined by the settings within the TapRemote configuration scenario. If desired, the remote user may command the assigned satellites on a per channels basis to another output power. Nominally there are 12-14 available channels.

This message sets specific channels to specific power levels. These power levels remain in effect only as long as the currently transmitted satellite is being transmitted. When a new satellite is assigned, the default power is applied which is the maximum power output of the simulator (the internal attenuator values are set to 0). This is an optional message and if not received the simulator will automatically set the power based to the maximum power level. This message is valid only in RUNLOCAL mode.

Field No	Symbol	Field Description	Field Type	Example
	$SETPOWER	Start of message		$SETPOWER
1	CHAN	Channel	C	2
2	POWER	Power to assign this channel (dBm)	-ccc	-132
3	…	…	…	…
4	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$SETPOWER,1,-116*37

4.2.6 Stop Simulator (STOP)

This message stops the current simulation, whether scenario based or local. The simulator transitions to IDLE mode and the 1PPS terminates.

This message is valid in RUNLOCAL, RUNSCEN mode.

Field No	Symbol	Field Description	Field Type	Example
	$STOP	Start of message		$STOP
1	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:

$STOP*18

4.2.7 Shutdown (SHUTDOWN)

This message will cause the simulator to power off. This command by-passes the nominal shutdown of Windows2000 and is an alternative to turning the system off remotely. It is valid in any simulation mode.

Field No	Symbol	Field Description	Field Type	Example
	$SHUTDOWN	Start of message		$SHUTDOWN
1	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$SHUTDOWN*08

4.2.8 Query (QUERY)

This message allows the GPIB controller to obtain information and status from the Tapestry-RC system. Messages can be output from Tapestry as either query/ response couplets or periodic data streams. Messages from the Tapestry-RC can be turned on/off if desired.

Message	Description	Default Output Rate
SIMMODE	Current simulation mode (IDLE, RUNLOCAL, RUNSCEN)	On Change *(whenever Tapestry transitions from one to another mode)*
CHANSTAT	Current channel status including svid generated and power level	On Change
BITRESULTS	Results of built in test	Never (Note that even if queried at a rate, BIT will not be run unless simulator is in IDLE
SYSCFG	System configuration	Never

Field No	Symbol	Field Description	Field Type	Example
	$QUERY	Start of message		$QUERY
1	MSGTYPE	Message type	A	CHANSTAT
2	RATE	Rate for simulator to transmit message	C	Seconds between output times.

				0 = On Change, -1 = Once, -2 = Never
3	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

The RATE controls how often the requested message will be transmitted by Tapestry-RC.

Example:
$QUERY,SIMMODE,0*2E

4.2.9 Regenerate Scenario (REGENSCEN)

This message immediately rebuilds a pre-built scenario resident on the simulator hard disk. It is only valid in IDLE mode.

Field No	Symbol	Field Description	Field Type	Example
	$REGENSCEN	Start of message		$REGENSCEN
1	SCENNAME	Scenario Name	A	RCVRTEST
2	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$REGENSCEN,RCVRTEST*6F
A scenario is a pre-built file set resident on the Tapestry computer system. The scenario must be located within a folder off of the main scenario run directory:

C:\voyager\runs

When a scenario is in the process of regenerating the simulation mode is set to REGENSCEN. Upon completion the simulation mode is to IDLE.

4.2.10 Time Set Pulse (TIMESETPULSE)

This message synchronizes the simulator with an external pulse. This message is only valid in IDLE mode.

Field No	Symbol	Field Description	Field Type	Example
	$TIMESETPULSE	Start of message		$TIMESETPULSE
1	LAT	Latitude (deg)	ll.llll	34.1234

2	LAT_REF	Latitude Direction (N, S)	A	N
3	LON	Longitude (deg)	yyy.yyyy	118.001
4	LON_REF	Longitude Direction (E,W)	A	W
5	ALT	Altitude (m)	xx.xxx	12.23
6	TIME	UTC Time (Hours, Minutes, Seconds)	hhmmss	130405
7	DATE	UTC Date	ddmmyyyy	25122002
8	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$TIMESETPULSE,34.0000000,N,118.0000000, 0.00000,133821,12032002*45

The given time and date in the message reflect the time of the pulse. The simulator enters the PULSEWAIT simulation mode and will wait for the external pulse until one of the 3 following conditions occur:

1. A valid pulse is received.
2. A STOP message is received. The simulator enters the IDLE mode if a STOP message is received.
3. Any message is received that is invalid either due to simulator mode or the message itself being invalid. In either case, the simulator enters IDLE mode. It is recommended that the ACK every message is enabled if the user intends to use the Time Set Pulse message.

The user should note that the external reset pulse actually resets the hardware. It will take several seconds before valid RF is generated, but the RF will be consistent with the information in the Time Set Pulse message.

The timing of the message and the pulse is crucial. The pulse can be generated in any manner the user desires, including a continuous 1PPS from a receiver or other timing source. However, there are certain timing constraints due to message transport delay and processing time. The Time Set Pulse message must be received no later than 300 milliseconds before the active edge of the pulse. If there is a continuous stream of pulses it is essential that the Time Set Pulse message is sent after the previous pulse and at least 300 milliseconds before the pulse.

4.3 Simulator to Controller Messages

These messages define the format of the output from Tapestry-RC to the remote command user.

4.3.1 Simulator Mode (SIMMODE)

Field No	Symbol	Field Description	Field Type	Example
	$SIMMODE	Start of message		$SIMMODE
1	MODE	Current Simulator Mode (IDLE, RUNLOCAL, RUNSCEN, REGENSCEN, PULSEWAIT)	a	RUNLOCAL
1	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$SIMMODE,IDLE*7C

4.3.2 Channel Status (CHANSTAT)

This message contains the currently transmitted satellites and power levels for all channels in the simulator. You should allow for up to 14 channels of data returned. Note that the power value is based on the calibrated maximum power output of the simulator and includes the user entered value for external attenuation.

Field No	Symbol	Field Description	Field Type	Example
	$CHANSTAT	Start of message		$CHANSTAT
1	CHAN	Channel	c	2
2	SVID	SVID	Cc	12
3	POWER	Power to assign this channel (dBm)	-ccc	-132
4	…	…	…	…
7	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$CHANSTAT,1,1,-110,2,0,-110,3,7,-120,4,8,-146,5,10,-110,6,11,-110,7,13,-146,8,27,-110,9,2,-110,10,31,-110,11,28,-110,12,26,-110*10

4.3.3 BIT Results (BITRES)

This message contains the results of the Built In Test (BIT). BIT is run at system startup and when QUERY'd if the simulator mode is IDLE.

Field No	Symbol	Field Description	Field Type	Example
	$BITRESULTS	Start of message		$BITRES
1	DATE	Date of the BIT results	A	mm/dd/yyyy
2	TIME	Time of the BIT results	A	hh:mm:ss
3	X1HWPROG	X1 Hardware programmed correctly (0 = valid, 1 = invalid)	c	0
4	X2HWPROG	X2 Hardware programmed correctly (0 = valid, 1 = invalid)	c	0
5	X3HWPROG	X3 Hardware programmed correctly (0 = valid, 1 = invalid)	c	0
6	INTERRUPT	Interrupts received properly (0 = valid, 1 = invalid)	c	0
7	OCXO	Clock valid (0 = valid, 1 = invalid)	c	0
8	PLL	Phase Lock Loop valid (0 = valid, 1 = invalid)	c	0
4	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$BITRESULTS,3/12/2002,4:23:08 PM,1,0,1,0,0,0*35

All values, 0 indicates good, 1 indicates failure.

4.3.4 Acknowledge (ACK)

This message is sent by the simulator if the previous message received was in error or if the "ACKEVERY" flag is set in the TapRci.ini file.

Field No	Symbol	Field Description	Field Type	Example
	$ACK	Start of message		$ACK
1	MSGID	Message ID of the message being	A	STARTSCEN

		acknowledged		
2	MSGVALID	Indicates if the message was valid/invalid (See below)	A	MSGINVALID
3	SIMMODEVALID	Indicates if the Simulation Mode was valid/invalid for the message type	A	SIMMODEINVALID
4	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example:
$ACK,STOP,MSGVALID,SIMMODEVALID*70

The MSGVALID field is set to invalid in two cases: the message checksum was incorrect or some part of the message was invalid. For example, if the INITGEO message was sent to the simulator with the time field set to 250000, an ACK message would be sent back with the MSGINVALID indicator since 25 is an invalid hour (valid values are from 0-24).

4.3.5 System Configuration (SYSCFG)

This message is sent by the simulator when QUERY'd.

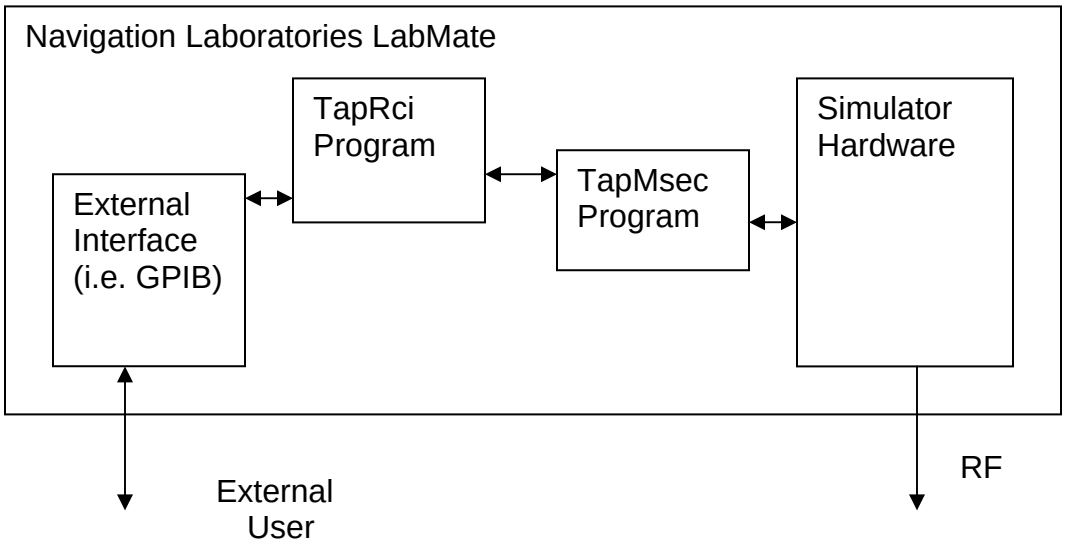
Field No	Symbol	Field Description	Field Type	Example
	$SYSCFG	Start of message		$SYSCFG
1	NUMCHANS	Number of channels in the system	ii	12
2	FREQ	The frequencies supported by the simulator. Either L1 or L1/L2	A	L1
3	MINPWR	Minimum power the simulator will output in dBm. (See below)	-xxx.xx	-161.00
4	MAXPWR	Maximum power the simulator will output in dBm. (See below)	-xxx.xx	-125.00
5	ChkSum	Checksum (Optional)	*hh	
	<CR><LF>	Message Terminator		<CR><LF>

Example
$SYSCFG,12,L1,-146,-110*66

The Max/Min power values are based on the calibration values entered into the simulator. See the calibration documentation for the calibration procedures. All power values include the calibrated value for maximum power and the value entered for external attenuation.

5 Tapestry Remote Control Interface Program (TapRCI)

This program communicates with the user through an external interface and controls the simulator. The simulator is configured to start this program automatically when the simulator is powered up. Certain settings can be set by the user by editing the TAPRCI.INI file located in the Windows directory. Before editing this file, the TapRCI program must be terminated and then restarted for the new settings to take effect. TapRci.exe does not communicate with the hardware directly (other than performing BIT) but instead starts another program that actually controls the simulator (TapMsec.exe).



5.1 TapRci.Ini

The user should not have to edit any values in this file. They are described here for completeness. It is located in the Windows main directory. Double clicking the file will bring up notepad allowing the user to edit values.

5.1.1 DISPLAY

This section of the TAPRCI.INI file has no impact on the operation of the simulator, but only affects the display.

DispRelativePower

This parameter controls the power display. If set to 1 (DispRelativePower=1) then the power values displayed on the TapRci screen are in dB and reflect the attenuator settings. If set to 0, then the display values are in dBm and reflect the calibration values, attenuator settings and the user entered external attenuation.

Example:
[DISPLAY]
DispRelativePower=0

5.1.2 CONTROL

This section of the TAPRCI.INI file sets parameters that determine how TapRci interacts with the simulator program (TapMsec).

TapMsecPath

This parameter determines the location of the simulator control program (TapMsec.exe).

Example:
[CONTROL]
TapMsecPath=c:\voyager\tapmsec\tapmsec.exe

RemoteScn

This parameter determines the base location of the remote scenario. The remote scenario is used as the basis for the RUNLOCAL mode. GPS ephemeris/almanac data, atmospheric parameters, etc. for local mode are located in this directory.

Example:
[CONTROL]
RemoteScn=c:\voyager\runs\TapRemote

VerifyOperationAtStart

This parameter determines if the program verifies proper operation of the TapRci program at startup. If there are any errors, a form is presented to the user showing the operational status, including any errors that caused the Verify Operation to fail. These errors include any failure in BIT, an inability to find either the remote scenario or TapMsec, a failure in the GPIB, and other possible errors. If this flag is set, then external communication via GPIB or any other method is not attempted until the errors are corrected.

Example:
[CONTROL]
VerifyOperationAtStart=1

5.1.3 INTERFACE

This section of the TAPRCI.INI file sets parameters that determine how TapRci communicates with the external user (i.e. GPIB, Ethernet, Serial)

GPIB

This parameter, when set, tells the TapRci program to use the GPIB device to communicate with the external user. Valid values are 0 and 1. Please note that if this value is 1, then both Serial and Ethernet parameters must be set to 0.

Example:
[INTERFACE]
GPIB=1

Serial

This parameter, when set, tells the TapRci program to use a serial port to communicate with the external user. Valid values are 0 and 1. Please note that if this value is 1, then both GPIB and Ethernet parameters must be set to 0.

Example:
[INTERFACE]
Serial=1

Ethernet

This parameter, when set, tells the TapRci program to use Ethernet to communicate with the external user. Valid values are 0 and 1. Please note that if this value is 1, then both GPIB and SERIAL parameters must be set to 0.

Example:
[INTERFACE]
Ethernet=1

5.1.4 GPIB

This section of the TAPRCI.INI file sets parameters that determine how the GPIB device is configured.

GPIBPrimaryAddress

This parameter is the primary address for the simulator.

Example:
[GPIB]
GPIBPrimaryAddress=15

GPIBBoardIndex

This parameter is the board number that is used to communicate with the external user. This value should always be zero, except in the event that there are multiple GPIB cards present in the system.

Example:
[GPIB]
GPIBBoardIndex=0

5.1.5 SERIAL

This section of the TAPRCI.INI file sets parameters that determine how the serial port device is configured.

COMPORT

This parameter tells the TapRci program which serial port to use to communicate with the external user.

Example:
[SERIAL]
COMPORT=1

BAUD

This parameter tells the TapRci program what baud rate to use. Valid values are normal baud rates (1200 – 115200).

Example:
[SERIAL]
Baud=19200

Parity

This parameter tells the TapRci program what parity to use. Valid values are None, Odd, Even.

Example:
[SERIAL]
Parity=None

StopBits

This parameter tells the TapRci program how many stop bits to set the serial port to. Valid values are 1 and 2.

Example:
[SERIAL]
StopBits=1

5.1.6 ETHERNET

This section of the TAPRCI.INI file sets parameters that determine how the Ethernet device is configured.

HostAddress

This parameter tells the TapRci program what host address to connect to. Valid values are valid Ethernet addresses for your network (i.e. 204.123.21.2)

Example:
[ETHERNET]
HostAddress=204.123.21.1

Port

This parameter tells the TapRci program what port number to connect to. Valid values are valid Ethernet port numbers for your network (i.e. 5307)

Example:
[ETHERNET]
Port=5307

5.1.7 MSGCONTROL

This section of the TAPRCI.INI file sets parameters that determine messaging behavior.

AckEvery

This parameter is a debug parameter. It can be set to 1, telling the TapRci program to send an acknowledge message in response to every message received. If it is 0, TapRci only sends an acknowledge message when an error has occurred. This can be useful during integration.

Example:
[MSGCONTROL]
AckEvery=1

5.2 VOYAGER.INI

This file controls the operation of the simulator and contains many parameters that are not described here. Only parameters that effect the operation of the simulator are described here.

5.2.1 TAPPTINIT

This section contains parameters that are used in Local mode. They are set automatically by the TapRci program in response to user commands.

Date

This is the start date of the simulation in Local mode.

Example:
[TAPPTINIT]
Date=02/14/2002

Time

This is the start time of the simulation in Local mode.

Example:
[TAPPTINIT]
Time=22:42:00

Lat

This is the latitude of the simulation in Local mode in degrees. Negative values indicate South.

Example:
[TAPPTINIT]
Lat=34.000

Lon

This is the longitude of the simulation in Local mode in degrees. Negative values indicate West.

Example:
[TAPPTINIT]
Lon=-118.000

Alt

This is the altitude of the simulation in Local mode in meters.

Example:
[TAPPTINIT]
Alt=10.23

5.2.2 PPS

This section controls the output of the 1 pulse per second (1PPS).

PPSAlwaysOn

The simulator can be configured such that the 1PPS always comes out, as opposed to only when GPS RF is being transmitted.

Example:
[PPS]
PPSAlwaysOn=0

PPSWidth

The width of the 1PPS can be adjusted with this parameter. The lsb of the pulse width is 20 microseconds. Therefore a value of 5 is 100 microseconds.

Example:
[PPS]
PPSWidth=1

5.2.3 EXTERNALRESET

This section controls the External Reset Pulse.

PulsePolarityUp

The External Reset Pulse is not used by all users. If used, the simulator can be synchronized with an external timing source. The pulse sense is edge detected and needs to be a minimum of 1 microsecond and a maximum of 100 milliseconds wide. This parameter controls whether the simulator will reset on the up or down edge of the pulse. A value of '1' indicates that the simulator will be reset when the external pulse goes from low to high (TTL level).

Example:
[EXTERNALRESET]
PulsePolarityUp=1

6 Hardware Interface

6.1 External Reset Pulse

Pulse information is obtained through the DB25 connector on the simulator. The relevant pins are shown below.

DB25 Pin	Signal Description	Notes	Signal Configuration
14	Simulator	The GPS Simulator	The Output 1 PPS is

	1PPS Outpus	can output a 1PPS. Please see the PPS section described in the Voyager.ini file for configuration of this signal.	configured through the Voyager.ini file. Please see the Voyager.ini PPS section for details.
20	External Pulse Reset	This pulse will cause the simulator to synchronize it's time with the time received in the Time Set Pulse message. NOTE: This pulse is not active unless and until the Time Set Pulse is received and the simulator enters the Pulse Wait mode. See the Time Set Pulse section for details.	This is an input pulse, TTL level. It is edge sensitive. The edge type (up or down) is defined in the PPS section of the Voyager.ini. The pulse should be a minimum of 1 microsecond and a maximum of 100 milliseconds.