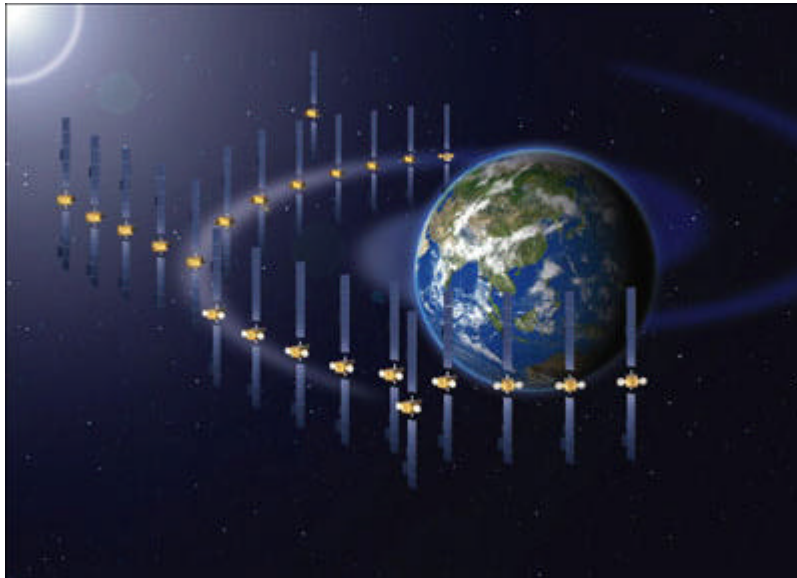




Digital Satellite Equipment Control (DiSEqC™)



TEST TOOLS USER MANUAL

June 30, 2006

This document is provided for information purposes. Whilst every effort has been made to provide accurate information, no responsibility is taken for errors or omissions. Eutelsat reserves the right to change this information without notice.

CONTENTS

1. Introduction.....	5
2. System Overview.....	5
3. The Test Tools Kit.....	6
3.1 Setting up the Test Tool.....	6
4. Bus-Monitor Reports (via RS 232).....	8
5. RS 232 Commands	10
Table 1 : Summary of RS232 Commands	11
5.1 DiSEqC™ Message Data	11
5.2 The “Gaps” Command (‘G n nn’)	13
Figure 1 : Full DiSEqC™ Command, Reply and Compatible Signalling.....	13
Table 2 : “Gap” Command Functions.....	13
5.3 DiSEqC™ Transmit Amplitude Command (‘H n’)	14
Table 3 : Transmitted Carrier Amplitude Commands	14
5.4 Backwards-Compatible Signalling Mode Command (‘M n’)	15
Table 4 : Backwards-compatible Signalling Commands.....	15
5.5 ToneBurst Duration Command (‘T n {n}’).....	15
5.6 Carrier Frequency Command (‘J n’).....	16
Table 5 : Carrier Frequency Command Parameters.....	16
5.7 Data-bit Keying Command (‘K n o p’).....	17
Figure 2 : Data-bit Keying	17
Table 6 : Data-bit Keying Command	18
5.8 Low-pass Filtering Command (‘L n’).....	18
6. User Flag Commands (‘U n’, ‘V n’).....	18
Table 7 : User Flags Commands.....	18
6.1 Numeric Timing Reports (User Flags 3 & 4).....	19
6.2 DiSEqC™ Transmission Reports (User Flags 5 & 6).....	19
6.3 RS232 Command Buffering (User Flag 9).....	19
6.4 Trigger output Pulses (User Flags A, B, C & F).....	20
6.5 Bus d.c. Detection (User Flag D).....	20

6.6	Bus Termination and Emulation (User Flag E).....	20
6.7	Byte control of Flags or External Ports.....	21
7.	Example Commands to the Test Tool.....	21
8.	Description of the Test Tools Hardware.....	24
	Figure 3 : Front Panel Layout	24
	Table 8 : Test Connector Pinning	24
	Table 9 : Serial Communications Connector Pinning	24
8.1	Access to the Printed Circuit Board	25
	Figure 4 : Basic component layout of the Printed Circuit Board.....	25
8.2	Functions of the Adjustment Potentiometers	25
	Table 10 : Potentiometer Adjustment Ranges	26
8.3	Brief Details of the Hardware Design	27
	Figure 5 : DiSEqC™ Test Tool Circuit Diagram	28
	Table 11 : Microcontroller Ports Pins Allocations	29
	Table 12 : Rear Connector Pin Allocations.....	30
9.	Slave Microcontroller Evaluation PCB.....	30
9.1	General Hardware Description.....	31
9.2	Setting up the Slave board	32
9.3	Pin-Link Functions	32
	Table 13 : Pin-Link Functions.....	34
	Figure 6 : Slave Circuit Board Pin-Links	35
9.4	Circuit Description.....	35
	Figure 7 : DiSEqC™ Slave PCB Circuit Diagram.....	38
APPENDIX.....		39
	Installation of User Software on a PC-Compatible Computer.....	39

Reference documents that define the DiSEqC™ system:

DiSEqC™ Bus Specification Version 4.2 (February 25, 1998)

DiSEqC™ Slave Microcontroller Version 1.0 (February 25, 1998)

DiSEqC™ Logos and Their Conditions of Use (February 25, 1998)

Associated Documents:

Update and Recommendations for Implementation Version 2.1 (February 25, 1998)

Application Information for Tuner-Receiver/IRDs (April 12, 1996)

Application Information for LNBS and Switchers (February 25, 1998)

Reset Circuits for the Slave Microcontroller (August 12, 1996)

Simple “ToneBurst” Detection Circuit (August 12, 1996)

Positioner Application Note Version 1.0 (March 15, 1998)

1. Introduction

This manual describes the setting up and operation of the DiSEqC™ “Test Tools” kit which has been developed for the evaluation and testing of DiSEqC™ “Masters” and “Slaves”. DiSEqC™ Masters are usually Satellite Tuner-Receivers or Integrated Receiver-Decoders (IRDs), and DiSEqC™ Slaves can be LNBs (Low Noise Blocks), “Switchers” or “Installation Aids”, etc.

The DiSEqC™ Test Tools hardware is derived from, and replaces, the original “DiSEqC™ Master Development Board” (APG 146B). It also incorporates some of the circuitry from the “Slave Evaluation Board” (APG 145B). However the Slave Evaluation Board remains relevant to testing and evaluating the functions of the DiSEqC™ Slave Microcontroller which is available for OEM use.

This updated manual includes a few additional facilities (described in italics) which were introduced with the “Version 0.8” software release, and used in Test Tools with serial numbers from 200 upwards. There is no necessity to upgrade Test Tools with lower serial numbers, but the additional functions can be obtained just by changing the software in the (socketed) EPROM. The “DiSEqC™ Test Tool Version 0.8” Object Code Program file may be obtained on request from EUTELSAT, and is planned to be posted on the Internet Site in due course.

2. System Overview

The DiSEqC™ Bus uses only a single conductor (co-axial with an earth return) for both power supply and signalling. It supports communication in only one direction at a time (half-duplex), but the direction may be reversed within about 10 milliseconds. However, the use of “Reply” messages is optional, and “DiSEqC™ 1.0” systems employ signalling just from a Master to one or more Slaves.

Only one Master is permitted on the Bus, and it is required to supply d.c. power onto the Bus. Slaves may be powered either from the Bus or locally, and the number of Slaves which can be operated on the Bus is primarily limited by the current available from the Master.

The function of the DiSEqC™ Test Tool is to convert the data carried on a DiSEqC™ Bus (primarily the 22 kHz amplitude-modulated signalling) to and from a convenient user-format (RS232 serial data). Usually, this interface will be to a Personal Computer, either running a general purpose Terminal Emulator Utility (for direct user access), or a custom Program (e.g. written in Basic) for specialised testing, such as on a Production Line.

The primary function of the Test Tool is as a “Bus Monitor” reporting not only the actual DiSEqC™ Protocol messages, but also (optionally) the timing of the data and the “steady state” conditions on the cable. However, it can also transmit messages on the Bus, using either “nominal” conditions, or over the full range of tolerances defined in the DiSEqC™ Bus Specification. The Tool can both transmit and monitor the Bus at the same time, so it can detect if its messages are being corrupted. Corruption might occur if an impedance on the Bus is incorrect (e.g. too high capacitance loading) or if a “collision” of messages occurs (e.g. due to an error by a Slave).

The types of variable parameter which may affect the performance of DiSEqC™ communications (and thus need to be generated and/or measured by a test tool) are :

- The voltage amplitude of the signals on the Bus (nominally 650 mV pk-pk),
- The carrier frequency (nominally 22 kHz),
- The bit-modulation period (nominally 1.5 ms),
- The modulation duty cycle (mark-space ratio) of the data bits (nominally 1:2 and 2:1),
- The data format (framing, address, command data, parity, validity, etc.),
- The duration of the "quiet" gaps between DiSEqC™ messages,
- The timing of the "backwards compatible" signals (Voltage, Tone and ToneBurst), and
- The static and dynamic characteristics of the d.c. voltage superimposed on the Bus (power present, start-up risetime, and backwards-compatible voltage-signalling).

The Test Tool can be powered from the Bus or from an external 24 volt d.c. power supply. When emulating a "Master", it must have an external power supply so that it can feed the required d.c. voltage onto the Bus. For use as a "Bus Monitor" or as a "Slave Emulator" (including the transmission of "Reply" messages) it can be powered solely from the Bus. However, it is generally better to use an external supply because there are occasions when the d.c voltage on the Bus may fall briefly to zero, (for example when a switch operates, or as an intentional "hardware reset" from a Master). In this case, the Test Tool will reset and have to re-initialise, which takes considerably longer than the Reset time specified for simple Slave microcontrollers.

3. The Test Tools Kit

The Test Tools Kit typically contains the following items:

- + The DiSEqC™ Test Tool in a metal enclosure approximately 220 x 150 x 40 mm (L x W x H)
- + A 24 volt d.c. "Plug Top" or "Desk Top" Power Supply, or a Power Supply cable
- + An RS232 serial communications cable terminated in 9 pin male and female 'D' connectors
- + A short length of satellite download cable, terminated in 'F' connector plugs
- + A 6-pin DIN plug for the front panel test points
- + This User Manual

The user will need a simple RS 232 Terminal, or preferably a Personal Computer running Windows™ or a Terminal Emulator Program. The kit should be installed as described in the following section.

An oscilloscope, a.c./d.c. voltmeter and/or a logic analyser, etc. may be useful for more specialised measurements.

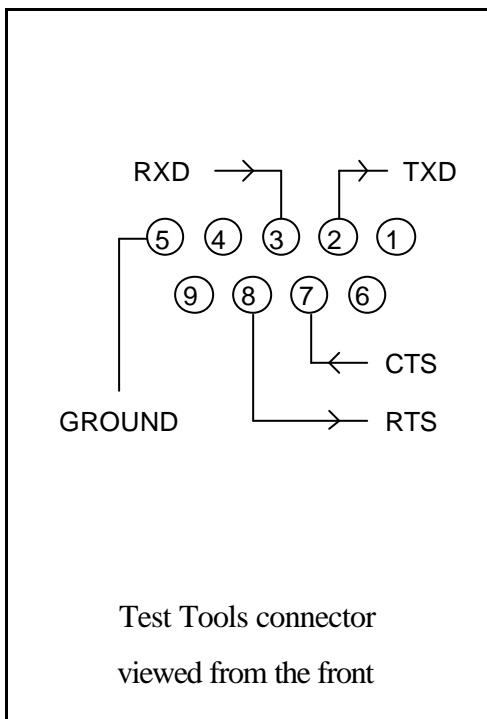
3.1 Setting up the Test Tool

Connect the 2.1 mm coaxial plug from a suitable 24 volt supply to the socket on the left hand side of the front panel. The voltage polarity is not important (provided the supply is "floating" or "negative earth") because the Tool has an internal diode bridge arrangement. However note that this 24 volt power supply

must not be used with the original DiSEqC™ Evaluation Kits (APG 145B and APG 146B) because the voltage is too high and may have incorrect polarity. The Test Tool consumes a quiescent current of about 125 mA. Thus, for use as a Bus Monitor, Slave Emulator or to test simple Switchers, a power supply of typically 300 mA rating is sufficient. However, a full DiSEqC™ Master is expected to be able to feed up to 500 mA onto the Bus, so a power supply of at least 650 mA should be used.

Connect the 9-pin 'D' connector via the cable to a suitable Terminal or Personal Computer. The 9 pins map directly to a typical 9-pin serial connector on IBM-compatible type PCs, or via a standard adapter to a 25-pin serial connector. The control software in the Test Tool requires RXD (pin 3), TXD (pin 2), RTS (pin 8), CTS (pin 7) and Ground (pin 5) to be connected. The PC or Terminal hardware or software may also require other pins to be held high or low. Here, RXD refers to the line Receiving data from the PC or terminal, and TXD refers to the line Transmitting data from the Test Tool. If difficulties are encountered, it is possible to bypass the handshake from the P.C., by instead connecting pin 7 (CTS) to pin 6 (+ve voltage) on the Test Tool connector.

If a PC is being used, suitable terminal-emulator software may need to be installed. Most utilities, such as "Windows™ Terminal" (usually found in the "Accessories" menu of Windows™ 3.1) or "Procom Plus", should be satisfactory because only standard ASCII alpha-numeric and control characters are employed. See the Appendix for details of setting up "HyperTerminal" usually found in "Windows™ 95/98/NT4". Set up the terminal (or emulator) to 9600 baud, 8 data bits, no parity and one Stop bit. The Test Tool uses hardware "handshaking", supported by internal software. Like the previous Master Development Board, the Test Tool can echo back most of the characters it receives. However this may be disabled by a "User-flag" command 'U 8', and re-enabled by command 'V 8' (see section 6). It also returns a "Bell" (Bleep) ASCII Character (07), if an incorrect command format is received via the RS 232 communications.



Thus, for Windows Terminal select :

"Settings": "Communications":

Baud Rate:	9600
Data Bits:	8
Stop Bits:	1
Parity:	None
Flow Control:	Hardware
Parity Check:	Off
Carrier Detect:	Off
Connector to suit the Port in use.	

For "Settings": "Terminal Preferences":

Selecting "Line Wrap" and "Sound" check-boxes is recommended.

The setup may then be stored using File : Save As...

The Tool can be tested without making any other connections, or it can be connected to any DiSEqC™ Master and/or Slave device. The two ‘F’ connectors on the front panel of the Test Tool are connected directly together, so a Bus cable can be “looped through” from a Tuner-Receiver/IRD to a Switcher or LNB. These connectors are not intended to emulate the path-switching obtainable with a “Switcher” or “Loop-through LNB”.

With the terminal emulator running, switch on, or plug in, the power supply to the Test Tool, or just connect a Bus cable to a Tuner-receiver/IRD. For verification, the Test Tool Software issues a “sign on” message of about 40 text characters via the RS232 port. If the Test Tool is being operated from a Tuner-receiver/IRD then any commands sent from this Master should be reported to the terminal without any further actions. The format of reports sent over the Bus (whether originated by the Test Tool, or by an external Master or Slave) is described in Section 4.

If the Test Tool is to be operated as a Master, then when the first <Enter> command is sent over the RS 232 link, the Tool should automatically apply power to the Bus (a click may be heard from the relay) and transmit the appropriate DiSEqC™ message from its buffer. The commands to make the Test Tool transmit DiSEqC™ messages over the Bus are described in Section 5. and some examples of command sequences are given in Section 7.

4. Bus-Monitor Reports (via RS 232)

The DiSEqC™ Test Tool can operate continuously as a Bus Monitor, except for occasional periods when another processing task is necessary, such as interpreting a command sequence from the RS232 communications channel. The Bus Monitor remains active when the Test Tool is transmitting a DiSEqC™ message and reports the actual data signalled on the Bus. Thus the format of the data presented at the user-terminal (via the RS232 communications) is very similar, whether the DiSEqC™ messages originate in the Test Tool or from an external Master or Slave device.

All normal DiSEqC™ messages on the Bus are reported as “ASCII-Hex” pairs of alphanumeric characters (zero - 9 and A - F). Each valid byte of the message is reported as two characters where the first character represents the “high” nibble, and the second character the “low” nibble. The character pairs are spaced to ease readability. If any DiSEqC™ byte contains a parity error (i.e. each 9th bit does not complete odd parity for the previous 8) then the reported byte is followed by a character ‘p’. Thus a typical message (but with invalid parity in the last byte) would appear as:

E0 10 38 F4p

All DiSEqC™ messages should be an integer multiple of 9 bits (i.e. an 8-bit byte plus a parity bit), but if an incorrect number of bits is detected then any “spare” bits are reported individually with the (lower case) letter ‘o’ representing a ‘0’ data-bit and letter ‘i’ representing a ‘1’ data-bit. Thus, if the previous message had “lost” its final parity bit then it would be reported as:

E0 10 38 iiiioioo

However, if the first data bit had been lost then the byte boundaries would be incorrect and the message becomes “scrambled”, for example:

C0p 20 70p iiiiooi

It is important to be able to identify individual message boundaries, sequences of messages, and sometimes to measure the time intervals between them. At the end of each reported DiSEqC™ message either a '-' or a '=' character is inserted. The '-' is the normal "delimiter", but the '=' is used if more than about +15 volts is detected on the Bus (i.e. the backwards-compatible Horizontal Polarisation signalling).

If the time between messages is more than about 250 milliseconds then the Bus Monitor transmits "Carriage Return" and "Line Feed" characters (at the end of the pause) to display the subsequent message in a new row. For shorter delays, the Monitor optionally sends the time delay in milliseconds between suitable delimiter characters. These time intervals can be displayed as two ASCII-Hex characters representing one byte (i.e. periods from 0 to 255 ms), as three decimal digits, or not at all, by altering some of the "User Flags" as described in section 6.

The time-delimiter characters used for a "quiet" period on the Bus are '<' and '>' so a typical message (from a Master) and reply (from a Slave) on the Bus might be reported as follows:

E2 10 38 F4 = < 032 > E4 =

where the '=' indicates that there is more than 15 volts d.c. on the Bus, <032> represents an approximately 32 millisecond delay and 'E4' is the "OK" reply from the Slave. Reporting of the time interval before a DiSEqC™ message is given only if there is a suitable time-reference, i.e. the termination of a previous continuous tone, a change in the Bus d.c. voltage, or if the message originates within the Test Tool.

If there is a 22 kHz tone on the Bus for much longer than a DiSEqC™ bit-period, then this is reported as a time interval between "quotation mark" delimiter characters. Thus the unmodulated "ToneBurst", which has a nominal duration of about 12 milliseconds, may be reported with either Decimal or Hexadecimal values, or as a "fixed" marker (depending on the User-Flags described in section 6.) as follows:

" 012 " or " 0C " or " TT "

The modulated "ToneBurst" is a valid DiSEqC™ message and thus normally appears as a Bus Monitor string **FF**, if it has exactly 9 cycles of modulation, or with individual data bits such as **iiiiiii** or **FF i**, for 8 and 10 cycles respectively.

Because Slave peripherals are often powered from the Bus, it may be necessary to know if the supply is interrupted. This is particularly relevant to Slave microcontrollers since they will normally reset after only a short (a few milliseconds) interruption to the supply. Tuner-Receiver/IRDs may actually use a brief power-down to force a hardware reset to any Slaves on the Bus. Therefore, the DiSEqC™ Test Tool monitors the d.c. voltage on the Bus and reports if it falls below 5 volts (nominal). The time below this voltage is reported as a Decimal or Hexadecimal value in milliseconds, between the delimiters '\' and '/' to represent the falling and rising voltage. Thus a complete "transaction" on the Bus, with a power down of 40 ms, then a pause of 100 ms before the message starts, and a gap of 20 ms before a modulated ToneBurst might appear as:

\ 040 / < 100 > E0 10 38 F4 = < 020 > FF =

Sometimes, the actual duration of DiSEqC™ messages, or the risetime of the d.c. voltage after a powerdown, may be of interest. These two parameters are not normally reported but may be selected by

User-flag command ‘V 2’ (see section 6.), and removed again with command ‘U 2’. The DiSEqC™ message time is reported in milliseconds between square delimiters ‘[]’ and the risetime as a single (hexadecimal) digit between ‘/’ delimiters. Note that the Test Tool only has d.c. measurement detectors for +5 and +15 volts so the risetime can only be determined between these levels. Therefore, the Bus voltage needs to be set to the backwards-compatible Horizontal Polarisation level (i.e. > 16 volts), or the risetime is reported as / - / to indicate it is not measurable. Thus a full transaction report (with 5 milliseconds risetime) could be:

\ 040 / 5 / < 100 > E0 10 38 F4 [052] = < 020 > FF [012] =

As the DiSEqC™ System is compatible with the established Voltage and Tone signalling methods, it is sometimes necessary to know the state of these continuous parameters. The Test Tool can report these (if selected by the associated User-flags) by sending a suitable “status” character approximately once per second. The continuous tone is of particular relevance because, if present, it can block the transmission of any other messages on the Bus. Therefore, the default User-flags setup is with the “Bus blocked” reports active (User-flag ‘0’) but the “Quiet Bus” reports disabled (User-flag ‘1’). Similarly, a d.c. voltage of less than 5 volts will prevent communication over the Bus so this marker is also enabled by the “Bus blocked” User-flag. These continuous markers, when enabled, are as follows:

- ~ Continuous tone on Bus, d.c. voltage less than 15 volts (High Band, Vertical Polarisation)
- ^ Continuous tone on Bus, d.c. voltage more than 15 volts (High Band, Horizontal Polarisation)
- No tone on Bus, d.c. voltage less than 15 volts (Low Band, Vertical Polarisation)
- = No tone on Bus, d.c. voltage more than 15 volts (Low Band, Horizontal Polarisation)
- _ Less than 5 volts d.c. on Bus (Slaves unpowered or reset)

5. RS 232 Commands

Commands to the Test Tool are sent from a terminal or PC in the form of single or short sequences of ASCII characters. Thus they can be entered directly onto the keyboard or may be generated by a custom Program. There are various types of command, some building up DiSEqC™ messages (in the Test Tool’s transmit buffer) to send over the Bus, and others controlling the mode of operation of the Tool. Some more specialised commands are primarily concerned with the evaluation and testing of the software within Slave microcontrollers.

All of the commands have initial “default” values loaded at power-up (reset), so it is only necessary to use most of them when testing system tolerances (e.g. carrier frequency, inter-message gaps, etc.). Some operating conditions of the Test Tool, such as the Bus d.c. voltage risetime and trimming of d.c. and signal levels, etc., are determined by hardware adjustments (pre-set potentiometers). These hardware functions are described in section 8.

The commands are summarised in Table 1 and described in the following subsections. Note that in the examples given, some spaces have been included to improve readability. However, the ASCII “space” character (i.e. the “space bar” on the keyboard) should only be used where specified explicitly as <space>. *Commands in italics apply to Version 0.8 software only.*

Command	Function	Section	Default
0 - 9	Data nibble to DiSEqC™ Transmit buffer	5.1	
A - F	Data nibble to DiSEqC™ Transmit buffer	5.1	
G n nn	Set Gap 'n' to byte value 'nn'	5.2	Table 2
H n	Select Transmit Amplitude (0 - 7 , for "background" add 8)	5.3	'H 5'
I	Increment DiSEqC™ message-byte in buffer (same as '+')	5.1	
J n	Select Tone frequency (signed, 3% steps, '0' = 22 kHz)	5.6	'J 0'
K n o p	Select keying tone cycles n:o:p (signed, '0' = eleven)	5.7	'K 0 0 0'
L n	Select Low-pass filtering threshold ('0' - '6')	5.8	'L 3'
M n	Select Backwards Compatible signalling Mode 'n'	5.4	'M D'
N nn	NOT byte 'nn' to Port 0 / User-flags 0-7	6.6	
O nn	OR byte 'nn' to Port 0 / User-flags 0-7	6.6	
P	Introduce Parity error to DiSEqC™ byte in Transmit buffer	5.1	
Q	Read byte from Port 0 / User-flags 0-7	6.6	
R	Read byte from Port 2 / User-flags 8-F	6.6	
S n { }	Reserved (Slave Simulation Mode)	N / A	
T n {n}	Set ToneBurst length to 'n' cycles (02 - 0F, 10 - 1F, 2 - F)	5.5	'T 9'
U n	Clear User flag 'n'	6.	Table 7
V n	Set User flag 'n'	6.	Table 7
W nn	Write byte 'nn' to Port 2 / User-flags 8-F	6.6	
X nn	XOR byte 'nn' to Port 0 / User-flags 0-7	6.6	
Y nn	Write byte 'nn' to Port 0 / User-flags 0-7	6.6	
Z	<i>Restart Test Tool</i>	5.1	
< space >	Interpret command(s) in RS232 receive buffer	6.3	
< Enter >	Interpret command(s) and Transmit DiSEqC™ message	5.1	
=	Go to next DiSEqC™ message-byte in transmit buffer	5.1	
+	Increment DiSEqC™ message-byte in buffer (same as 'I')	5.1	
- (dash)	Decrement DiSEqC™ message-byte in buffer	5.1	
. (dot)	<i>Separator for two Messages / Null-message</i>	5.1	
/	Power-down and restore Bus d.c. voltage before message	5.1	

Table 1 : Summary of RS232 Commands

5.1 DiSEqC™ Message Data

Each DiSEqC™ message byte is defined by a pair of ASCII-Hex characters, and the associated parity flag may be made not-valid (for testing purposes) by adding a 'P' between the two characters. Thus a sequence of the numerals 0 - 9 and the letters A - F and P build up a message in the transmit buffer. All letters may be either upper-case (CAPITALS) or lower-case. If the Test Tool has been set up to "Echo" the characters it receives (Sections 3.1 and 6.) then it automatically returns a <space> between each character-pair. *With version 0.8 software, a '.' (dot) may be used to separate two messages, or <dot> <Enter> to give a 'Null message', i.e. 'Backwards-compatible' signalling only.*

When there is a message in the buffer it can be transmitted (or re-transmitted) by using the <Enter> key. To avoid the need to re-enter complete commands there are simple "editing" functions. Instead of keying in ASCII-Hex character pairs, each existing data byte (in the DiSEqC™ transmit data buffer) can be "skipped over" with the '=' key. Then subsequent bytes can be changed by keying in new ASCII-Hex pairs, or they may be incremented (+1) or decremented (-1) with the '+' (or 'i') and '-' keys respectively. Any required

bytes after those edited must also be skipped (with the '=' key) before using <Enter> or the message will be truncated back to the edit point. Thus the number of bytes transmitted may be selected simply by using the '=' key a corresponding number of times, followed by <Enter>. *If used, the '.' (dot) message separator must be re-entered each time the buffer is edited.* Generally, the main keyboard keys must be used, not those in the "numeric" keypad.

The '/' key used at any time before <Enter> produces a d.c. "power-down" (Slave Reset) before the DiSEqC™ message is transmitted. 'Z' <Enter> *restarts the Test Tool processor (Version 0.8 only).*

5.2 The “Gaps” Command (‘G n nn’)

The timing of the “quiet” pauses on the Bus, before any DiSEqC™ (command) message, ToneBurst and Continuous tone are determined by the “Gaps” command ‘G’. Each ‘G’ is followed by 3 ASCII-Hex nibbles, the first to select which gap is to be changed, and the other two form a byte (not BCD) defining a time approximately in milliseconds (i.e. up to 255 decimal). Figure 1 and Table 2 show how the Gaps are numbered.

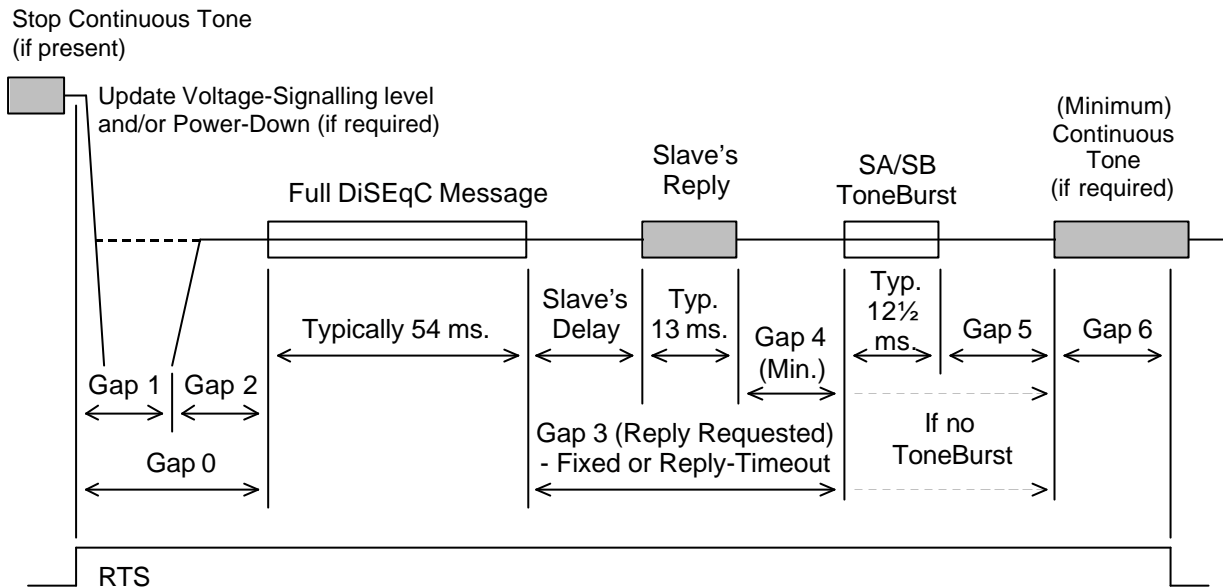


Figure 1 : Full DiSEqC™ Command, Reply and Compatible Signalling

For “two-way” DiSEqC™, the time-gap between the end of a command and the ToneBurst (or the continuous tone, if there is no ToneBurst) must be long enough to leave space for the Reply from the Slave. Since Slaves initially have a random delay of up to 150 ms (to reduce “collisions” if more than one Slave replies to the same command), the Gap needs to be set to about 200 ms (to leave space for the delay, 2 bytes of reply, and the gap after the reply). However, if the Slave is instructed to use its minimum delay of about 10 ms (e.g. by command E2 00 01) then the gap can be set smaller. In practice, two-way “Masters” should detect the reply and then transmit the ToneBurst and/or continuous tone after the minimum specified delay of 15 ms. To emulate this, the Test Tool has a “minimise delay” mode selected by the “User-flag” command ‘V 7’, and disabled by command ‘U 7’ (see section 6.).

Timing Period	Command Format	Nominal Default	Description
Gap 0	G0 nn	20 ms	Voltage-change to Message start (no power-down)
Gap 1	G1 nn	40 ms	Power-Down (If selected)
Gap 2	G2 nn	100 ms	Power-Up to Message start
Gap 3	G3 nn	200 ms	Fixed or Maximum Pause after Message (Reply Requested)
Gap 4	G4 nn	16 ms	Minimum (or No-Reply) Pause Before ToneBurst or Tone
Gap 5	G5 nn	16 ms	Minimum Pause After ToneBurst
Gap 6	G6 nn	50 ms	Minimum Continuous Tone Period (before RTS enabled)

Table 2 : “Gap” Command Functions

If the message run-in byte does not request a reply (e.g. if it is 'E0' or 'E1', then the "Gap 4" time delay is used before the ToneBurst (or continuous tone) is applied to the Bus. This allows appropriate time delays to be used automatically with a mixture of "No Reply" (short time delay) and "Reply Required" messages (long time delay).

The Test Tool software does not request another RS 232 command (i.e. it does not validate RTS) until the minimum continuous tone period has elapsed and the RS 232 transmit buffer is empty (to ensure that there should be adequate buffer-space for the next Bus-monitor report).

5.3 DiSEqC™ Transmit Amplitude Command ('H n')

The amplitude of the 22 kHz carrier tone transmitted by the Test Tool over the Bus is determined by a combination of software commands and hardware (preset potentiometer) adjustments. The hardware has three Bus Transmit stages (nominally for "low", "medium" and "high" amplitudes) which can be selected in any combination. The adjustment of the hardware is described in section 8.3. It will be set either to specific "pass" and "fail" levels for production testing, or more generally to "binary weighted" levels giving an approximately uniform sequence of 7 transmit levels in steps of just over 100 mV each. These 7 levels (plus zero) are selected by the three least-significant bits of the nibble-parameter used in the command.

To permit validation of the "carrier rejection level" of units under test, it is also possible to activate the low-amplitude transmit stage between DiSEqC™ message-bits and between messages. This is enabled by the most-significant bit in the nibble-parameter. Thus most of the 16 possible parameters have a useful function as shown in Table 3.

The command 'H 9' sets up a background level of about 260 mV and a DiSEqC™ message level of about 390 mV which is a severe test primarily intended for checking that the Test Tool receive level is set to a suitable level (i.e. between these values).

Command	Typical Carrier Amplitude, mV pk-pk	Typical "Background" Amplitude, mV pk-pk	Notes
H 0	0	0	No Data
H 1	130	0	
H 2	260	0	
H 3	390	0	Typical Minimum
H 4	520	0	
H 5	650	0	Nominal
H 6	780	0	
H 7	910	0	Typical Maximum
H 8	0	130	Inverse Data
H 9	390	260	Alignment Levels
H A	260	130	
H B	390	130	"Worst Case"
H C	520	130	
H D	650	130	Nominal (+ 'noise')
H E	780	130	
H F	910	130	

Table 3 : Transmitted Carrier Amplitude Commands

5.4 Backwards-Compatible Signalling Mode Command ('M n')

The Test Tool supports the ToneBurst and Backwards-compatible Voltage and Continuous tone signalling. The command controlling these facilities is an ASCII 'M' followed by a single hex nibble, n, which is valid for all 16 values. When n takes the value '0' (zero) the Backwards compatible signals are switched off so there is no ToneBurst, Continuous tone or d.c. voltage switching. In mode 'F', the Test Tool makes the Backwards-compatible signals follow the data sent from its transmit buffer, assuming they are relevant to a suitable Slave on the Bus (i.e. valid Framing byte, Address 00h or 10h - 1Fh, and Command bytes 20h - 22h, 24h - 26h or 38h nn). *With version 0.8 software only, it is possible to send just Backwards-compatible signals (i.e. no DiSEqC™ messages) by keying: '.' (dot) <Enter> or 'M n.(dot)' <Enter>.*

Command	ToneBurst	Voltage	Continuous Tone	Notes
M 0	Off	Low	Off	
M 1	Unmodulated (SA)	Low	Off	Group 8
M 2	Unmodulated (SA)	Low	On	
M 3	Unmodulated (SA)	High	Off	
M 4	Unmodulated (SA)	High	On	
M 5	Modulated (SB)	Low	Off	
M 6	Modulated (SB)	Low	On	
M 7	Modulated (SB)	High	Off	
M 8	Modulated (SB)	High	On	
M 9	Off	Low	Off	Group 4
M A	Off	Low	On	
M B	Off	High	Off	
M C	Off	High	On	
M D	SA / SB	Low / High	Off / On	Cycle through '8'
M E	Off	Low / High	Off / On	Cycle through '4'
M F	SA / SB	Low / High	Off / On	Follow DiSEqC™

Table 4 : Backwards-compatible Signalling Commands

Sometimes it is appropriate to send specific backwards-compatible signals different to the data in the DiSEqC™ message (e.g. to check which format a Slave is responding to), so the nibble values '1' - '8' select the 8 possible compatible signals, as shown in Table 4. Similarly, nibbles '9' - 'C' give the 4 possible backwards-compatible versions, but without any ToneBurst. Parameter nibbles 'D' and 'E' automatically increment the Backwards-compatible signals (i.e. to the next value through '1' to '8' or '9' to 'C' respectively) each time a DiSEqC™ command is transmitted.

5.5 ToneBurst Duration Command ('T n {n}')

The length of the ToneBurst is programmable using the command 'T', normally followed by one nibble. This nibble indicates the duration of the ToneBurst in data-bit periods (i.e. in nominal 1.5 ms increments) and so is usually set to 9. Note that because the actual duration of the ToneBurst varies with the bit-period, it may change when the carrier-frequency or keying ratio commands are used ('J n' and 'K n o p', see sections 5.6 and 5.7).

If the parameter character is a '1', then a subsequent nibble is read to form a Hexadecimal value up to '1F', i.e. 31 decimal. Thus ToneBursts with nominal periods from 2 ms (command 'T 2' = 2 bits) to 45.5 ms (command 'T 1 F' = 31 bits) can be generated.

If the parameter character is a '0' (zero), then the duration is determined by the subsequent nibble, but the continuous tone finishes at the end of the last bit-period, rather than at the point where the modulated ToneBurst ends (i.e. 1/3 through the last bit). Thus a command 'T 9' gives a nominal unmodulated ToneBurst of 12.5 ms whilst a command 'T 0 9' gives a burst of 13.5 ms.

5.6 Carrier Frequency Command ('J n')

Command	Nominal Offset	Actual Frequency	Actual Offset
J 8	-24%	16.6 kHz	-25%
J 9	-21%	17.6 kHz	-20%
J A	-18%	18.1 kHz	-18%
J B	-15%	18.6 kHz	-15%
J C	-12%	19.2 kHz	-13%
J D	-9%	19.8 kHz	-10%
J E	-6%	20.5 kHz	-7%
J F	-3%	21.2 kHz	-4%
J 0	0%	21.9 kHz	0%
J 1	+3%	22.8 kHz	+3%
J 2	+6%	23.6 kHz	+7%
J 3	+9%	24.1 kHz	+9%
J 4	+12%	24.6 kHz	+12%
J 5	+15%	25.6 kHz	+16%
J 6	+18%	26.1 kHz	+18%
J 7	+21%	26.7 kHz	+21%

Table 5 : Carrier Frequency Command Parameters

The command for changing the (nominal 22 kHz) carrier-tone frequency is 'J' followed by a single nibble. The nibble uses its most significant bit as a signing bit (-8) so that 'F' represents a step of "minus 1" and '1' represents a step of "+1", where each frequency step is approximately 3%, as shown in Table 5.

Each cycle of the carrier tone is actually generated by counting machine (program execution) cycles of the internal microprocessor, partly by hardware and partly in software. To change the tone frequency, the number of machine cycles is altered, and this can only occur in integer steps (and preferably in pairs of machine cycles for better waveform control). There are only typically 50 machine cycles in each 22 kHz cycle so it is not possible to achieve exactly 3% increments in the carrier frequency. Also, frequencies above the nominal (i.e. above about 23 kHz) have slight limitations of edge-jitter (< 1 µs) and square-wave asymmetry (<3 µs).

5.7 Data-bit Keying Command ('K n o p')

The duty cycle of the DiSEqC™ message data (i.e. the proportion of each bit-period that the 22 kHz tone is switched on) is set by the command 'K', followed by 3 nibbles because each nominal 1.5 ms bit-period is divided into three parts. Each part is nominally 0.5 ms, i.e. 11 (eleven) cycles of the 22 kHz tone. For a data '1', only the first part contains the tone, and for a data '0' the first two parts contain the tone. In the third part, no 22 kHz cycles are actually transmitted, except during an unmodulated ToneBurst.

The 3 nibbles in the 'K' command (e.g. 'K n o p') indicate the deviation in the number of effective cycles from the nominal 11, in the 3 respective sections, as shown in Figure 2. Thus the nominal keying ratios can be restored with the command 'K 0 0 0' (zeros). The nibbles use their most significant bit for signing such that 'F' represents "minus 1" (i.e. 10 cycles) and '1' represents +1 (i.e. 12 cycles), etc., as shown in Table 6. It is thus possible to independently change the "Keying time" (duty cycle) of the '0' and '1' bits and the overall bit-period. This command can produce keying ratios and periods which are considerably outside the DiSEqC™ Bus Specification so the parameters should be selected with some care. Note that the parameters always refer to the number of cycles of the carrier tone, which may not be at the nominal 22 kHz frequency (see section 5.6).

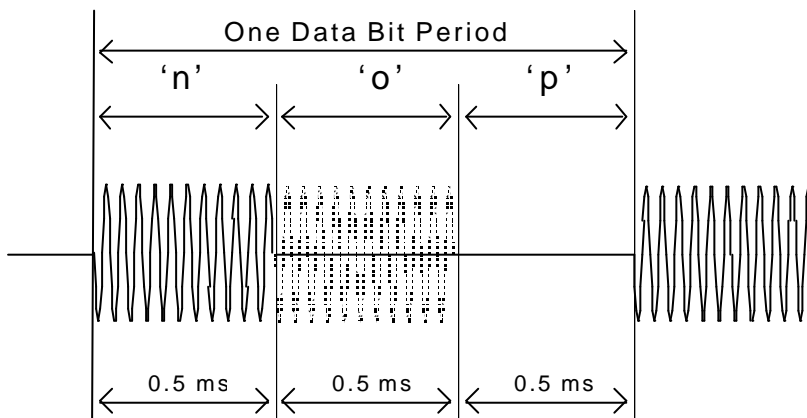


Figure 2 : Data-bit Keying

A change of one carrier cycle in any one of the "Keying" periods (number of cycles per DiSEqC™ data-bit) introduces a change in the bit-period (and frequency) of about 3% (i.e. 1 cycle in 33). Therefore, changes in the keying ratios can be arranged to complement any carrier frequency change and maintain a nominal bit-period. Thus if the frequency has been set approximately 9% high with the command 'J 3', then the bit period may be returned to 1.5 ms with the command 'K 1 1 1', i.e. with 36 cycles (+1 in each of the 3 intervals) at about 24 kHz. Similarly, to keep a substantially constant bit-period, the parameter in the 'J' command should be changed by the same number as the (signed) sum of the parameters in the 'K' command.

Parameter	Carrier Cycles
0	11
1	12
2	13
3	14
4	15
5	16
6	17
7	18

Parameter	Carrier Cycles
8	3
9	4
A	5
B	6
C	7
D	8
E	9
F	10

Table 6 : Data-bit Keying Command

5.8 Low-pass Filtering Command ('L n')

To give some immunity to “spikes” and “missing” carrier cycles on the Bus, the Test Tool Bus Monitor software “tone-decoding” (demodulation) uses an up-down counter between numerical thresholds, to detect the onset and termination of the carrier tone. Polling occurs at an **average** rate of about 17 kHz, regardless of the carrier frequency and whether it is transmitted locally or received over the Bus. The counter is incremented if a (positive) carrier transition has occurred (on the Bus) in the previous polling period (i.e. typically 60 µs), and is decremented otherwise. The default threshold is 3 counts (i.e. equivalent to command 'L 3'), and a typical functional range of the parameter is 1 - 5, although this depends on the frequency and keying ratios of the signals.

6. User Flag Commands ('U n', 'V n')

Flag Number	Control Function	'U n' / 'V n' commands	See Section
0	Report (typically once per second) if Bus is “Quiet”	OFF / on	4.
1	Report (typically once per second) if the Bus is “Blocked”	off / ON	4.
2	Report DiSEqC™ message durations and d.c. risetime	OFF / on	4.
3	Report DiSEqC™ message gaps as numeric values	off / ON	6.1
4	Send timing data as Hexadecimal or Decimal numbers	hex / Decimal	6.1
5	Report Data in DiSEqC™ (transmit) Buffer at transmission	OFF / on	6.2
6	Report Backwards-compatible status at transmission	OFF / on	6.2
7	Minimise delay from Reply to Backwards-compatible tones	off / ON	5.2
8	Echo RS 232 Commands (when interpreted)	off / ON	3.1
9	Receive RS 232 commands into “background” buffer	OFF / on	6.3
A	Test trigger output = Pulse at each message-start	off / ON	6.4
B	Test trigger output = Pulse at each ToneBurst	OFF / on	6.4
C	Test trigger output = Pulse at command initiation	OFF / on	6.4
D	Monitor d.c. signalling (> 15 v.) levels on the Bus	off / ON	6.5
E	Slave Emulation Mode (i.e. no power to Bus)	OFF / on	6.6
F	Test trigger output = Carrier Tone	OFF / on	6.4

Table 7 : User Flags Commands

The operating mode of the DiSEqC™ Test Tool is determined by up to 16 “User-Flags”. The presently-defined flag functions are listed in Table 7, with typical default settings shown in bold capitals. The functions of flags not explained elsewhere in this manual are described below.

Each User Flag is selected individually by a command character followed by a single ASCII-Hex parameter which corresponds to the flag number. Flags are cleared by the command ‘U n’ and set by the command ‘V n’, so that ‘U 4’ clears flag 4 and ‘V A’ sets flag ‘A’, etc. These flags may also be controlled in groups as described in subsection 6.7.

6.1 Numeric Timing Reports (User Flags 3 & 4)

Numerical timing reports are disabled with the command ‘U 3’ and enabled with the command ‘V 3’. The data may be presented either in decimal notation (selected by command ‘V 4’) or Hexadecimal (selected by command ‘U 4’). The decimal format always consists of three digits to discriminate it from the two-character Hex. data.

6.2 DiSEqC™ Transmission Reports (User Flags 5 & 6)

The Test Tool normally only reports (via RS 232) the messages actually carried on the DiSEqC™ Bus. However, if the reports are not as expected (i.e. if data is missing or corrupted) then the actual contents of the Test Tool’s transmit buffer is reported just prior to each transmission by issuing the command ‘V 5’. This should permit identification of whether errors are occurring on the DiSEqC™ Bus or RS 232 communications links. Then the buffer reports can be disabled with command ‘U 5’.

Similarly, reporting of the 3 “backwards-compatible” signals is enabled by the command ‘V 6’ and disabled with the command ‘U 6’. Each signal is reported as a single character, i.e. ‘H’ or ‘v’ for Polarisation (‘H’ represents >15 volts selected for the Bus), ‘T’ or ‘o’ for continuous Tone (High Band) or no tone (Low Band) and ‘B’, ‘a’ or ‘-’ for Modulated, Unmodulated or No ToneBursts respectively. In each case the capital letter corresponds to the equivalent DiSEqC™ logic ‘1’ level and the lower-case letter to the logic ‘0’ level.

6.3 RS232 Command Buffering (User Flag 9)

The microprocessor in the Test Tool performs the “Bus Monitor” function by testing for the presence of a carrier tone on the Bus about every 60 µs. Although it is possible to time-share this with most of the other functions of the Tool, it is not practicable to interpret the commands received from the RS232 communications whilst monitoring the Bus. The processor can simply receive RS232 command characters from the terminal and store them in an internal data buffer (of approximately 15 bytes length), whilst continuously monitoring the Bus. Then, when an “execute commands” character is received (<Enter> or <space>), the control software suspends the Bus Monitoring whilst interpreting the command(s) in the buffer. This mode is selected by the user-command ‘V 9’. The <space> command has the same effect as <Enter> in scanning the RS232 receive buffer data, but does not send the message in the DiSEqC™ transmit buffer over the Bus.

Sometimes it may be beneficial to process the incoming RS232 data immediately it is received, particularly to give some interaction with a user entering data directly from a keyboard. Therefore, the RS 232 buffer can be disabled so that the Bus monitoring activity is suspended as soon as a character is received from the terminal, until the next “execute” character is received. This mode is selected by the user-command ‘U 9’. In this mode some commands are actioned immediately (e.g. the tone amplitude command ‘H n’) whilst others await the next “execute” or “transmit” command.

6.4 Trigger output Pulses (User Flags A, B, C & F)

The 6-pin DIN connector on the front panel of the Test Tool carries various test signals (Table 8) including a programmable trigger pulse on the centre pin (pin 6). Each of the test pulses is synchronised to the internal (nominal) 22 kHz tone generation and is positive-going (between “TTL” levels) of approximately a half-cycle duration (20 µs). They can be enabled in any combination, although selection of the tone frequency output (command ‘V F’) will swamp the other pulses.

Command ‘V A’ selects a pulse which occurs just before the full DiSEqC™ message is transmitted and command ‘V B’ selects a similar pulse just before the ToneBurst. Command ‘V C’ gives a pulse as soon as execution of a complete DiSEqC™ sequence commences, i.e. at the time that a previous continuous tone is cancelled, the H/V voltage signalling is updated or a power-down sequence is initiated. The continuous tone output (command ‘V F’) is primarily intended for use with a frequency counter (although the frequencies generated are crystal-controlled and so actually very consistent).

All of the above trigger pulses are disabled with their complementary ‘U n’ commands.

6.5 Bus d.c. Detection (User Flag D)

The Test Tool Bus-monitor software normally uses any voltage-crossing of the (nominal) 15 volt d.c. threshold as a zero-reference for timing-measurements before any DiSEqC™ messages. However, if the actual Bus voltage is close to this level (perhaps because limit levels have been set for production-line testing) then false-reporting may occur. To prevent this, the detected d.c. level can be forced to “low” with the command ‘U D’, or restored to normal with command ‘V D’.

6.6 Bus Termination and Emulation (User Flag E)

When operated from local d.c. power, the Test Tool is sometimes required to act only as a Bus Monitor and at other times to emulate a DiSEqC™ Master. In the former case the Tool should not put any terminating impedance or d.c. voltage onto the Bus, whilst in the latter case it should. The Test Tool software always initialises (at power-on or hardware reset) in Bus-monitor mode (i.e. with no power or load on the Bus). When the first “transmit” command (i.e. <Enter>) is received (and also any subsequent messages requesting a “power-down” cycle) then the Bus d.c. voltage is tested and power is only applied if there is less than 5 volts d.c. already on the Bus. Thus the Test Tool normally “intelligently” chooses its mode. However, it is possible to force the Tool not to apply a voltage to the Bus, nor generate the Backwards-compatible tone signals, with the command ‘V E’. Normal operation is restored with command ‘U E’.

6.7 Byte control of Flags or External Ports

If a mask-programmed microprocessor is used in the Test Tool, there are two ports (16 pins) available for auxiliary control purposes (corresponding to the Data and Address buses to the normal external Program EPROM). An optional implementation of the DiSEqC™ Test Tool may use an OTP microprocessor with the unused external address lines brought out to a connector on the rear panel of the unit. Since normal 8051 hardware uses an address latch for the 8 low address lines (multiplexed with the data on Port 0), this port can be allocated as an output with driving characteristics defined by the type of latch installed (74LS or 74HC, etc.). The 8 high address lines (from Port 2) are normally wired directly to the external Program Memory (EPROM), and thus this port may be used for input or output using the standard 8051 “quasi-bidirectional” conventions.

Some command characters have been provisionally reserved for the control of these port-pins. The commands again consist of a single letter followed by an ASCII-Hex pair of characters which correspond to the 8 bits of the relevant port. Thus a combination of individual pins can be controlled by summing their appropriate weightings (i.e. 2^n where n is the bit number from 0 to 7). The provisional commands are:

- ‘N nn’ : NOT byte nn to Port 0 (i.e. Clear any bits which are set in the byte nn)
- ‘O nn’ : OR byte nn to Port 0 (i.e. Set any bits which are set in the byte nn)
- ‘Q’ : Read byte from Port 0 (Reads the output driver, not the output pins)
- ‘R’ : Read byte from Port 2 (Previously-written ‘low’ pins must be set ‘high’ first)
- ‘W nn’ : Write byte nn to Port 2
- ‘X nn’ : Exclusive OR byte nn to Port 0 (i.e. Change any bits which are set in the byte nn)
- ‘Y nn’ : Write byte nn to Port 0

In the normal Test Tool implementation, these commands are not needed to control the external Port-Pins, so instead they are mapped onto the “User Flags” (in two groups of 8). Thus up to 8 User Flags in each of two groups (0 - 7 and 8 - F) can be changed with a single command to “Port 0” and “Port 2” respectively.

7. Example Commands to the Test Tool

All the operating parameters of the Test Tool are initialised to nominal or typical values at power-up, and a sample DiSEqC™ Command is pre-loaded into the transmit buffer. However, when the Test Tool software initialises, it does not apply any voltage onto the Bus because it may be required to act as a passive Bus Monitor. It first sends the “sign-on” message to the RS 232 terminal and if there is no active Master on the Bus, it sends an “underscore” (`_`) character about once per second to indicate that there is no d.c. so the Bus cannot function.

Sending just a single `<Enter>` character from the RS 232 terminal is sufficient to transmit the default message and, if there is no Slave on the Bus, a typical report would be:

```
=> <105> E2 10 00 - <200> ' 012 '
```

where => indicates that the message originated from the Test Tool and 105 is the typical delay in milliseconds from power-up to the start of the message. In this case the command is a ‘Reset’ (00) to all LNBs and Switchers (10 is the “wildcard” for all addresses 11 to 1F). The E2 framing byte requested a

reply and so there is a “long” pause (to wait for a reply) of 200 ms before a 12 ms unmodulated ToneBurst is sent for “simple” switchers.

The following examples all assume that there is a Slave on the Bus, preferably an Evaluation Board (such as APG 145B) so that the states of all the switch outputs can be seen. Note that for all early samples of APG 145B (serial numbers less than 100) it is recommended to connect a resistor of about 22 kΩ in parallel with, or in place of, the reset diode D2 (next to the SKEW connector) to give a short reset time consistent with the latest DiSEqC™ Bus recommendations. If a Slave is connected, the above Bus report would be typically :

=> <105> E2 10 00 - <060> E4 - <016> ' 012 '

where there is a 60 millisecond random delay (which can be any value from about 10 to 110 ms) before the Slave’s “OK” reply (E4). Then the Test Tool automatically waited for a period of only about 16 milliseconds, close to the recommended minimum, before sending the ToneBurst.

Because the actual command sent is a “Reset”, the Slave stays in its “Backwards-compatible” mode and thus the Band, Polarity and Satellite LEDs (1 - 3 on the Evaluation Board) should display the status of the continuous tone, d.c. voltage and ToneBurst signals. The default setting of the Test Tool is to cycle around the 8 combinations of these signals, so if the <Enter> key is hit a total of 8 times (spaced a few seconds apart), the “Reset” command continues to be sent and the LEDs on the Slave should cycle around the 8 states.

A new DiSEqC™ command can be put into the transmit buffer and sent with a key sequence such as: **E2 10 01 <Enter>** (type no spaces, but the Test Tool should insert them). This command instructs the addressed Slave to use its minimum Reply delay (about 10 ms) instead of a random delay. Since most of the command sequence is similar to the previous (Reset) command, the buffer can more easily be “edited” with the sequence: **= = i <Enter>** (i.e. first two bytes are the same, the 3rd is incremented by 1). In either case, a typical monitor report would be:

=> <020> E2 10 01 - <009> E4 - <016> ' 012 '

which has a 20 ms pause before the message and a 9 ms Reply delay by the Slave. The Slave LEDs should no longer follow the Backwards-compatible signals because it is now under the control of DiSEqC™ commands.

A typical satellite-signal switching command is **E2 10 24 <Enter>** (or just **= = 24 <Enter>** after the previous command), which turns the “Band” LED (1) on (if off), and gives a Bus report such as:

=> <020> E2 10 24 - <009> E4 - <016> FF - <018> ~~~~~~

where in this case the ToneBurst is modulated (FF report) and there is a pause of about 18 ms before a continuous (back-compatible) tone is put on the Bus by the Test Tool.

To illustrate how the “User-flags” can change the amount of data presented (in all subsequent reports), the RS 232 string: **/U 3 <Enter>** produces a power-down without timing information:

=> \ E2 10 24 = E4 = FF =

In the above case the d.c. voltage has risen above 15 volts (= markers) and the continuous tone is off. Alternatively, the command string: / **V 3 V 2** <Enter> reverses the above flag and adds risetime and message duration data to give a full report such as:

=> \ **040 / 5 /** <100> **E2 10 24 [039]** = <009> **E4 [013]** = <016> **FF [012]** =

To demonstrate a “one way” DiSEqC™ command (and restore normal reporting), the character string **E0 10 20 U 2** <Enter> should turn the “Band” LED off, but there will be no Reply and the Test Tool gives only a short delay (about 15 ms) before the ToneBurst, because a Reply is not expected:

=> <020> **E0 10 20** = <015> **FF** =

To send a “double” message (with version 0.8 software only), the ‘.’ (dot) separator can be used in a command sequence such as **E2 10 38 F5** <dot> **E2 10 14** <Enter> to give a report such as:

=> <020> **E2 10 38 F5** = <035> **E4** = <017> **E2 10 14** = <041> **E4 57** = <017> **FF** =

where the reply parameter (‘57’) reports the status of the control lines which have just been setup.

8. Description of the Test Tools Hardware

The DiSEqC™ Test Tool is constructed on a Eurocard-style (100 mm wide) Printed Circuit Board which is installed in a suitable metal enclosure. Note, however, that the length of about 177 mm is slightly longer than a standard Eurocard and, if used, the rear connector is of “half-height” format. The layout of the Front Panel is shown in Figure 3, with the pinning of the connectors in Tables 8 and 9.

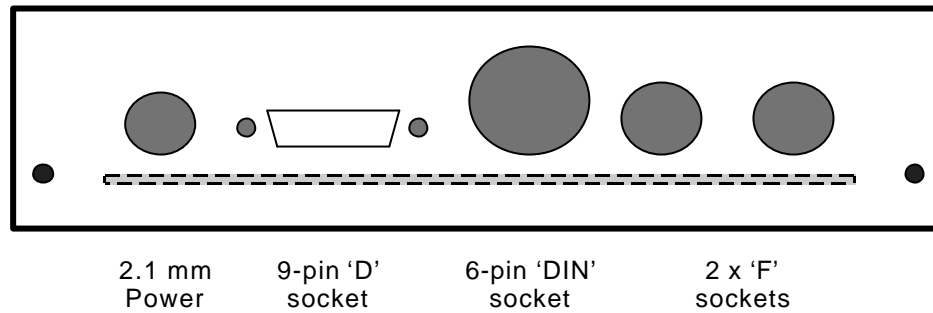


Figure 3 : Front Panel Layout

Pin number	DIN (audio)
Screen	GND
1	+24 v.
2	+5 v.
3	GND
4	22kHz detector
5	Bus
6	Trigger Pulse(s)

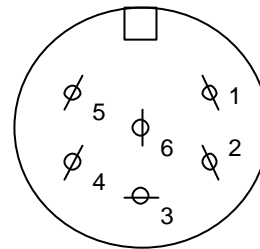


Table 8 : Test Connector Pinning

Pin Number	'D' connector
1	++ via 4k7Ω
2	TXD
3	RXD
4	
5	GND
6	++ via 4k7Ω
7	CTS
8	RTS
9	

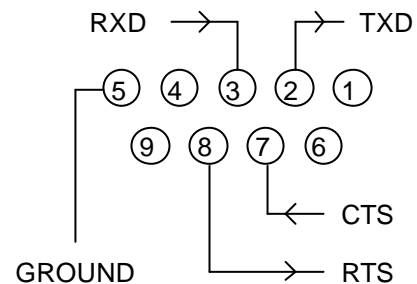


Table 9 : Serial Communications Connector Pinning

8.1 Access to the Printed Circuit Board

The Printed Circuit Board may be either slid out from the enclosure, or the top cover removed to give access with the PCB in place. To slide out the PCB, loosen the two screws in the corners of the front panel and partly pull out the panel. Next remove the screw attaching the earthing straps to Regulator 1 and then pull out the complete assembly. Alternatively, to remove the top cover, remove the four top-cover screws, one at each corner. Note, however that these screws are a self-tapping type into plastic mouldings so should be handled with some care.

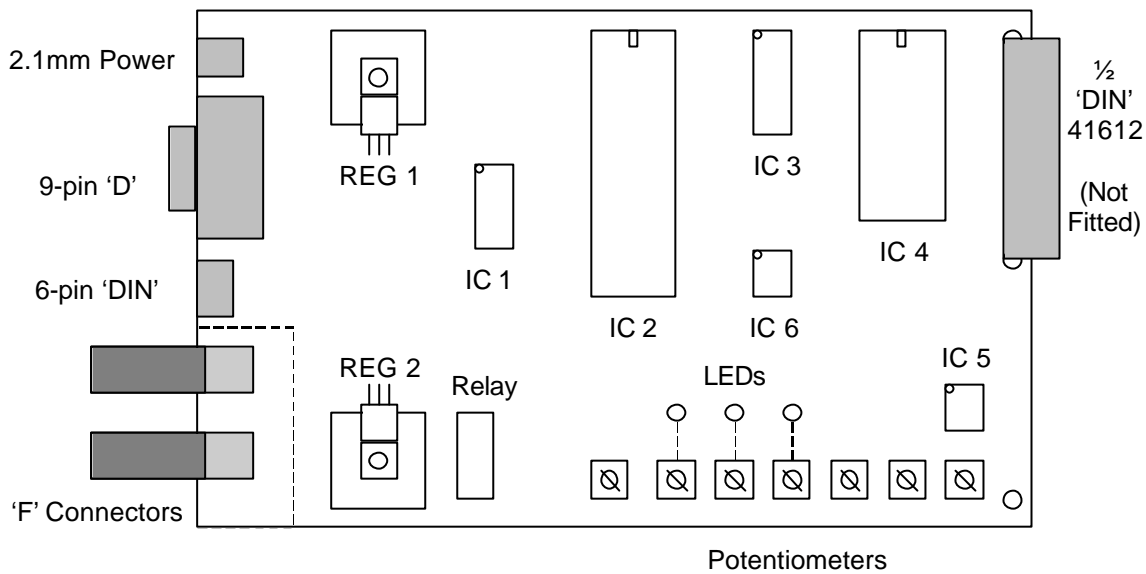


Figure 4 : Basic component layout of the Printed Circuit Board

8.2 Functions of the Adjustment Potentiometers

Seven potentiometers are carried along the lower edge of the Printed Circuit Board. However, the unit should operate satisfactorily if all these potentiometers are set to their central positions. Adjustment is only necessary if the user wishes to set up “High Precision” parameters (such as carrier tone amplitude) or specific “Pass” or “Fail” levels for production line testing. Table 10 lists the typical operating conditions which should be obtained over the adjustment range of the potentiometers. In each case the “Minimum” column corresponds to the fully anti-clockwise position of the wiper, and “Maximum” to the fully clockwise position. Note that clockwise rotation of the SENS potentiometer increases the signal amplitude which is needed for a tone on the Bus to be detected. Thus the direction of rotation is opposite to that normally associated with a gain or sensitivity control. Also, the sensitivity control operates by changing the d.c. bias level of the a.c.-coupled amplified tone relative to a “TTL” threshold. This d.c. level can be adjusted right up to the nominal threshold, thus giving effectively an “infinite” gain, i.e. a very high susceptibility to “noise” on the Bus.

Name	Function	Minimum	Nominal	Maximum
SENS	DiSEqC™ Bus Receiver Detection Level (Sensitivity)	-100 mV	400 mV	750 mV
AMP 1	DiSEqC™ Bus Transmit Amplitude 1 (low)	20 mV	125 mV	200 mV
AMP 2	DiSEqC™ Bus Transmit Amplitude 2 (medium)	175 mV	275 mV	375 mV
AMP 3	DiSEqC™ Bus Transmit Amplitude 3 (high)	400 mV	500 mV	600 mV
VOLTS	Low d.c. volts applied to Bus ("Vertical Pol.")	10.5 V	13 V	15.5 V
DIFF	Difference between "low" and "high" Bus volts	0 V	2.5 V	5 V
SLEW	Risetime of Bus d.c. voltage after a Power-down	1 ms	6 ms	12 ms

Table 10 : Potentiometer Adjustment Ranges

The method of setting the potentiometers will depend on the exact user requirements, but the following procedures may be of guidance:

Connect an oscilloscope to one of the 'F' connectors or between pins 5 and 3 (ground) of the front panel 'DIN' test connector. Set the oscilloscope to a.c. coupling and a typical sensitivity of 100 mV/cm. Set the Test Tool to transmit a tone continuously, for example with command 'M 2' <Enter>. Activate the "low amplitude" transmit stage for example with command 'H 1' and adjust potentiometer 'AMP 1' for the required signal, say 100 or 130 mV peak-peak. Repeat for the "medium" and "high" amplitude stages with commands 'H 2' and 'H 4' and adjust 'AMP 2' and 'AMP 3' respectively for signals of say 260 and 520 mV. Three LEDs are mounted next to their associated potentiometers to confirm which stages are actually active.

To set the voltage potentiometers, first select a "low" bus voltage with a command such as 'M 0'. Adjust potentiometer 'VOLTS' for the required d.c. voltage on the Bus, say 14 volts, using either the oscilloscope with d.c. coupling, or preferably a suitable voltmeter. Then select a "high" Bus voltage with a command such as 'M 3' and adjust the 'DIFF' potentiometer for the required Bus voltage, say 16 volts.

If there is a need to adjust the risetime (from 0 volts) of the d.c. voltage from the Test Tool then a command sequence such as '/ <Enter>' briefly drops the supply nearly to zero and then reinstates it. Use the oscilloscope and adjust potentiometer 'SLEW' as required. A risetime of about 5 ms should be suitable for most purposes. Note that the voltage rise is actually "slew rate limited" (with a range of about 0.1 to 1.0 volts per millisecond) so the risetime will be slightly longer when aiming for a "high" bus voltage than a "low" bus voltage. The fall time remains almost constant at about 2 ms but, if required, could be made similar to the risetime by installing a 1.2 volt reference diode (or two forward diodes in series, or a LED) in the empty location on the PCB allocated to 'D1'.

There are various ways to set the "receive sensitivity" potentiometer, 'SENS'. A suitable tone amplitude can be set up on the Bus (either from the Tool or externally) and the potentiometer adjusted until the "one second" tone markers are just being reported. Alternatively, an oscilloscope can monitor the tone detector output on the test connector pin 4 and set the point where the required tone on the Bus just causes the detected voltage to cross the nominal TTL detection threshold of about 1.4 volts. A convenient check is to use the alignment command 'H 9' and adjust 'SENS' so that the Bus Monitor correctly reports messages originated by the Tool, i.e. with a sensitivity of between about 260 mV ('level 2') and 390 mV ('level 3').

8.3 Brief Details of the Hardware Design

The Schematic Circuit Diagram of the Test Tool is shown in Figure 5. The microprocessor section is very similar to the original “DiSEqC™ Master Development Board” (APG 146B) but the processor is slightly upgraded to an 80C52 to give extra facilities such as a 3rd counter-timer and additional on-chip RAM. The clock frequency is increased to give more processing power, but is constrained to the highest (below 16 MHz) “standard” 8051 series division ratio for RS232 communications (14.7456 MHz). This retains compatibility with other 8051 derivatives. The allocation of the Port-pins is shown in Table 11 and the rear PCB connector (unused in the Test Tool) in Table 12.

The program memory is a (28-pin) EPROM, but a 32-pin socket is fitted which is suitable for single-voltage FLASH memories such as 29EE010. This may improve the software upgrade facilities in the future. An IIC bus (serial) EEPROM is also included on the PCB to allow data to be stored over a power-down, but again this is not implemented in the first software release.

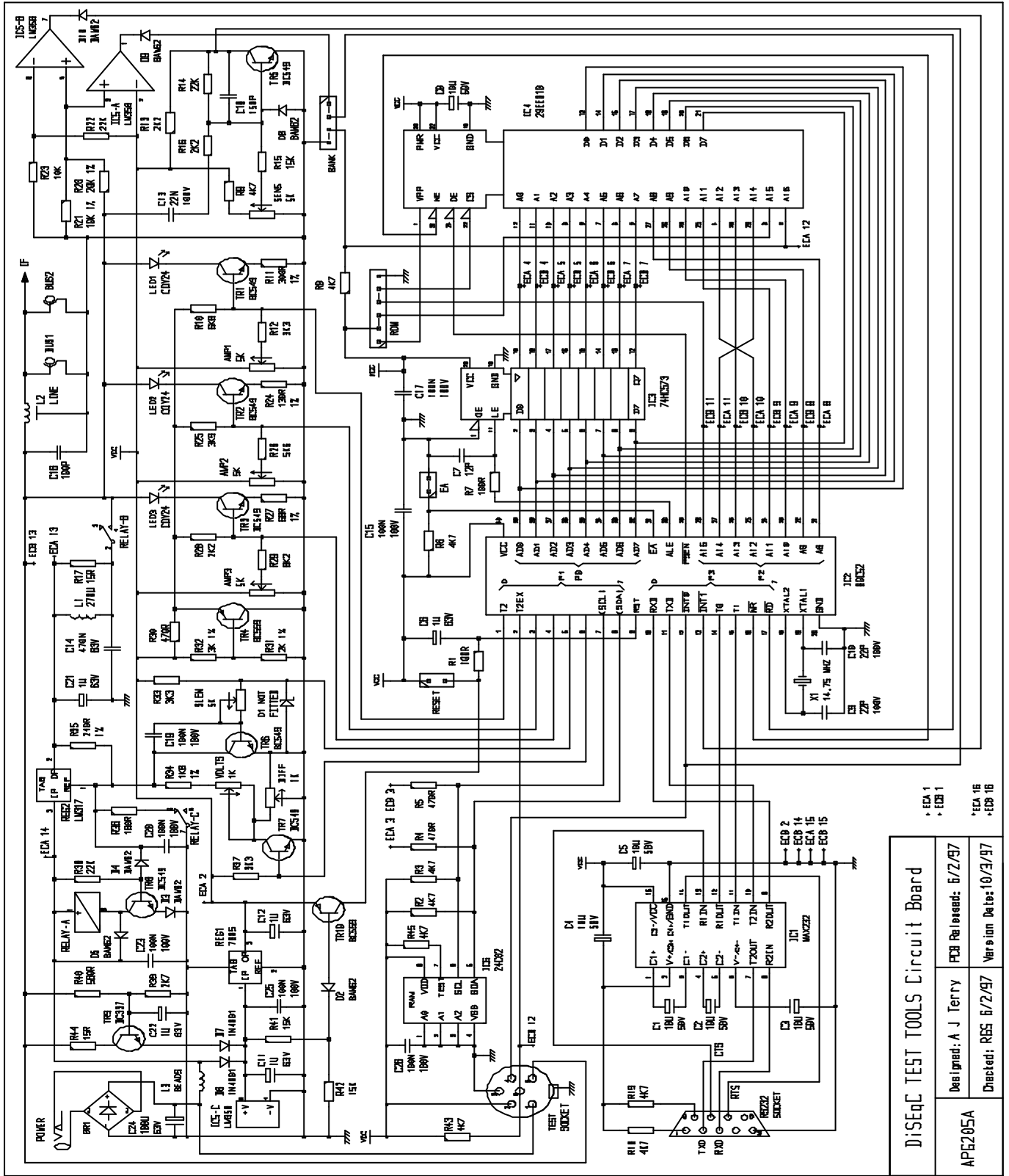


Figure 5 : DiSEqC™ Test Tool Circuit Diagram

DiSEqC TEST TOOLS Circuit Board	
Desig'nad: A J Terry	PCB Re'lased: 6/2/97
APE205A	Directad: RES 6/2/97 Ver's ion: De'be' 10/3/97

Port 1	Pin number	“Outputs”	Port 3	Pin number	I/O
P1.0 (T2)	1 (8x52)	DTX0	P3.0 (RXD)	10	RXD (in)
P1.1 (T2EX)	2 (8x52)	Test Trigger	P3.1 (TXD)	11	TXD (out)
P1.2	3	DTX1	P3.2 (INT 0)	12	DRX (in)
P1.3	4	DTX2	P3.3 (INT 1)	13	Bus > 5v. (in)
P1.4	5	d.c. On/Off	P3.4 (T 0)	14	CTS (in)
P1.5	6	d.c. High/Low	P3.5 (T 1)	15	RTS (out)
P1.6 (SCL)	7 (8x65x)	SCL	P3.6 (WR)	16	WR (out)
P1.7 (SDA)	8 (8x65x)	SDA	P3.7 (RD)	17	Bus > 15v. (in)

Table 11 : Microcontroller Ports Pins Allocations

The external power supply is connected via a bridge (BR 1) to accommodate either polarity of d.c. input voltage, because the connector polarity is not standardised. The 5 volt regulator (REG 1) can take current either from the external power supply (via D 6) or from the DiSEqC™ Bus (via D7). The stabilising capacitor on the input to REG 1 is isolated from the Bus by TR9.

The conventional microcontroller reset circuit has an added transistor (TR 10) which forces a hardware reset whenever the input voltage to the regulator (REG 1) is less than about 8 volts. Initialisation (of the software) thus should be reliable even if the voltage from the Bus (or from the external power supply) rises slowly.

To allow the Test Tool to be able to emulate a “Slave” application, a Bus-voltage-detecting operational-amplifier is included, similar to that on the Slave PCB (APG 145B). Alternatively, the pin-link (BANK) allows the port-pin to be used as a bank-switch (A 16) for the flash memory. A second op.-amp. pulls the interrupt pin (I1) Low if the Bus voltage falls below 5 volts, to allow emulation of a reset (the processor may be multi-tasking so it cannot be hardware-reset). Each of these op.-amps. is connected via diodes (D 9 and D 10) to prevent its output pulling the port pin above the 5 volt supply rail.

The tone-reception stage (TR 5) is basically the same as the previous designs, but includes a diode to protect against reverse (base-emitter) transients from the Bus. A “Miller” capacitor (C 18) has been introduced to roll off the gain at frequencies well above 22 kHz, to improve the immunity to spikes, etc. The added potentiometer (SENS) trims the bias level, allowing adjustment of the minimum signal level which just crosses the “TTL” detection threshold (nominally +1.4 volts).

The single “Bus-driver” transistor of the previous evaluation PCBs is replaced by 3 stages of modified design, all driven by a single port-write software instruction. The three stages are set to generate different amplitudes of 22 kHz tone, which may give up to 7 levels (plus off), or they may be set individually to “pass” and “fail” levels. The low-amplitude stage may alternatively inject a small-amplitude continuous tone (typically 100 mV peak-peak) onto the Bus, to evaluate the noise-rejection characteristics of devices under test. The LEDs are now used purely for display and are placed in the main current path so that they are much brighter than the previous design. To avoid being of an inconveniently low value, the potentiometers are in the base circuits, which are each biased by a common reference voltage of +2.0 volts (plus a Vbe) from TR4.

The voltage applied to the Bus (in “DiSEqC™ Master” applications) is adjustable via REG 2 to between about 10 and 20 volts. The regulator sets up a current source of typically 5 mA. in the reference resistor

chain so that when TR 7 is conducting, the potentiometer “VOLTS” sets the rail over a range of about 10 to 15 volts. When TR 7 is turned off, the rail rises by between 0 and 5 volts, dependent on the setting of the potentiometer “DIFF”. The rise and fall times of the voltage steps are partially defined by C20.

The voltage feed to the Bus is disabled via TR 6 which pulls the “reference” pin close to ground. The rise and fall times are further controlled by the integrator capacitor (C19) and the base-driving impedance (potentiometer “SLEW”). The optional zener diode (D 1) allows some flexibility in pre-setting the ratio of fall time to rise time.

When its “reference” pin is grounded, REG 2 still provides an output of about 1.25 volts. However, when the reference has fallen to about ½ volt, TR 8 comes out of conduction and releases the relay, thus isolating the Bus. The relay contacts (RELAY-B) also ensure that the Master terminating (source) resistance (R 17) and capacitance (C 14) are removed from the Bus in Slave-emulation and Bus-monitor modes.

The second pair of relay contacts (RELAY-C) ensure that the reference voltage cannot rise above about 1 volt whilst the relay is released. Thus, when the microcontroller disables TR 6, its collector voltage only rises sufficiently to turn on TR 8 and thus the relay. When these “normally closed” relay contacts open, then the reference and Bus voltages rise at a rate determined by C 19 and the SLEW potentiometer.

Pin Number	DIN (41612) connector		
	(Row c)	Row b	Row a
1			
2		GND	+5v.
3		SCL	SDA
4		A1	A0
5		A3	A2
6		A5	A4
7		A7	A6
8		A9	A8
9		A11	A10
10		A13	A12
11		A15	A14
12		Test Trigger	A16
13		Bus volts	Bus (not IF)
14		GND	+24v
15		GND	GND
16			

Table 12 : Rear Connector Pin Allocations

9. Slave Microcontroller Evaluation PCB

This section describes the setting up and operation of the printed circuit module which is available for the evaluation of the DiSEqC™ “Slave” Microcontroller for OEM use. The circuit board carries the DiSEqC™ Slave integrated circuit with versatile peripheral circuitry to permit most of the typical uses to be set up and changed. The other circuit board, the “Master” Development PCB (APG 146B), has been replaced by the DiSEqC™ Test Tool, which can perform all its functions.

9.1 General Hardware Description

The module is a double-sided printed circuit board employing standard through-hole (i.e. non surface-mounted) components. The circuit board (APG 145B) is about 9 cm by 10 cm and includes two 'F' connectors and a row of 8 red LEDs along one side. The schematic diagram is shown in Figure 7.

The two 'F' connectors are directly linked together so that the Slave may be used in a simple "loop-through" mode such as in an "Installer Aid" (for example giving a signal strength reading at the "dish" end of the download). Note that this does not represent the more sophisticated loop-through applications where the Intermediate Frequency (I.F.) signal and/or the DiSEqC™ bus commands are actually switched (or diverted) by a Slave under the control of the Bus.

The power supply from the cable to the centre pin of the Bus 'F' connectors passes along a printed circuit track (which is configured to isolate the satellite signal I.F.) to the remainder of the circuitry. A low power 5-volt regulator feeds all the basic components including the DiSEqC™ Slave integrated circuit. It is also used as a reference for the 13/18 volt level detection (backwards-compatible polarisation selection) and for a 6-volt switchable (power-down) rail for typical LNB applications. When the 6 volt rail is active (i.e. not in "standby") then the green LED is illuminated and the voltage is available on the 3-pin polarised connector (SW6V).

For the evaluation of devices with a local power supply (e.g. positioners) a 2.1 mm co-axial connector is provided, to receive a positive dc voltage on its outer sleeve and ground connection to its centre pin. This connector includes a switch which disconnects the Bus supply voltage from the power control/regulator transistor but a diode allows the DiSEqC™ microcontroller circuitry to remain powered from the Bus if the local power is not present. Conversely, if the Bus power is removed, a second diode applies the local power to the microcontroller to retain its status and any internally stored data. Note that with this configuration, the local and Bus voltages must not be the same level because the current switching may produce invalid impedances on the Bus and corrupt the 22 kHz data.

Two alternative sockets are provided for the DiSEqC™ Slave microcontroller, a 24 pin "skinny DIP" (7.5 mm spaced Dual In line Package) and a 28 pin PLCC (Plastic Leadless Chip Carrier). The voltage levels on the eight primary switching control outputs from the microcontroller are indicated on the row of eight red LEDs, which are buffered by a standard HCMOS inverter. Each LED is illuminated if the corresponding control pin is high. If it is desired to actually use the microcontroller outputs for control functions, then the buffer i.c. (IC4, 74HC540) may be removed from its socket and a DIL header plugged in to receive the signals (pins 2 - 9). If a PLCC i.c. is fitted, then the DIL socket can be used for test points or with a header plug.

The PCB supports a quasi-analogue (filtered Pulse Width Modulated) signal on the 2-pin polarised socket (SKEW). However, this is not supported by the present versions of the Slave software (versions 0.2 and 1.0 are totally identical), so this output just provides an addition binary (On/Off) control function.

The DiSEqC™ Slave microcontroller is intended to support a wide range of possible applications and configurations. This is achieved primarily by optionally linking certain pins to ground and connecting a few

signal diodes in programming arrays between “Sense” and “Strobe” pins. The evaluation board therefore needs to be able to be configured conveniently in a large number of different ways. The method selected to achieve this is the use of a number of vertical pins which may be joined in pairs by small bridging links. Some of the links connect signals to ground and others connect up pre-installed diodes for the arrays. Note that this is purely for evaluation purposes and in final applications the connections will be largely hard-wired, with perhaps a few diodes optionally installed for configuration options (such as LNB Local Oscillator frequency). The functions of the various pin-groups are described in section 9.3, and the overall methods of configuration are defined in the “DiSEqC™ Slave Microcontroller version 1.0 Specification”.

9.2 Setting up the Slave board

The Slave board should operate satisfactorily with no pin-link jumpers installed, except preferably ‘VOLTS - East’. However, generally it will be supplied with some jumpers fitted for testing and to give a more typical configuration. Many combinations of pin-linking are not meaningful, and may produce erroneous operation of the software, so it is important to carefully check the required characteristics against the Slave Microcontroller Specification and the information in Section 9.3 of this user manual. A typical configuration is given in the final column of Table 13. Note that the pin-linked configuration is only read when the Slave microcontroller is reset, so after changing any of the jumpers it is necessary to issue a Reset Command or remove and re-apply power to the unit.

The Slave board normally derives its power from the Bus so it is not necessary to make any power connection to the board. In fact, unless an application using a local power supply is to be evaluated, no connection should be made to the 2.1 mm connector because it includes an internal power-switching (disconnection) function. If external power is to be applied, the voltage polarity should be positive to the outer sleeve and negative (ground) to the inner pin.

Thus to operate the Slave board it is only necessary to connect a suitable coaxial cable terminated with ‘F’ connectors from a “Master” such as the Test Tool, to either Bus connector on the Slave board. When this is done, and power is applied to the Bus, the Slave should start up into the default state which has been set by the pin links. The amber “Transmit” LED (LED 10) should be completely extinguished and the green “Power” LED (LED 9) should be on. Depending on the configuration, most of the red LEDs should be off unless an active-low “decoder” function is selected, in which case all but one red LED should be illuminated. Until any DiSEqC™ commands are received the two “backwards compatible” functions (13 / 18 volt and 22 kHz tone switching) and the ToneBurst should control the appropriate LEDs.

9.3 Pin-Link Functions

To economise on the number of pin-links and matrix diodes used in the evaluation board, most links have been implemented in pairs with a common (central) signal pin. Since two separate jumpers cannot be installed on one (triple) pin-group, it may not be possible to evaluate every array configuration directly. However, if it is desired to check one of these conflicting configurations it may be possible to test the individual functions successively or, for final evaluation of an application, by soldering additional diodes in the necessary locations.

Figure 6 shows the positions of all the pin-links and other main interface components of the Slave PCB and Table 13 indicates the connections made when a jumper is bridged across each pair of pins. For the 3-pin groups, the pair of pins linked are indicated as N (north), S (south), E (east) or W (west).

The E-W group, VOLTS, allows either the 13 / 18 volt status or the external power supply voltage status to be applied to input IPX. Normally the East position is used, but if the West pair of pins are bridged (external supply) then the STR_D pin should be connected to Ground (a pin link is not available for this) to signal this configuration to the internal software (versions 0.2 and 1.0).

Link Name	Link From	To DiSEqC™ Slave pin name, number (DIL)	Function	Test Links
STBY	Ground	SBY (6)	Disables Standby function	
PWM	Ground	OPX (20)	Disables OPX output	
VOLTS (W)	6v rail	via D3 to IPX (19)	Local power availability sensing	
VOLTS (E)	13/18v sense	via D3 to IPX (19)	13/18v bus voltage sensing	✓
E1 (N)	Ground	OP 1 (5)	L.O. is fixed Low	
E1 (S)	STR_E via D4	"	L.O is fixed High	
E2 (N)	Ground	OP 2 (4)	Polarisation is fixed Vertical	
E2 (S)	STR_E via D5	"	Polarisation is fixed Horizontal	
E3 (N)	Ground	OP 3 (3)	No satellite selection	
E3 (S)	STR_E via D6	"	Satellite 'B'	
E4 (N)	Ground	OP 4 (2)	'Option' Switch is not available	✓
E4 (S)	STR_E via D7	"	Switch Option 'B'	
E5 (N)	Ground	OP 5 (1)	Switch S1 is not available	
E5 (S)	STR_E via D8	"	Not currently defined	
E6 (N)	Ground	OP 6 (23)	Switch S2 is not available	✓
E6 (S)	STR_E via D9	"	Decoding Mode 'Active High'	
E7 (N)	Ground	OP 7 (22)	Switch S3 is not available	
E7 (S)	STR_E via D10	"	Uncommitted switches Decoding	
E8 (N)	Ground	OP 8 (21)	Switch S4 is not available	
E8 (S)	STR_E via D11	"	Committed Switches Decoding	
G1 (N)	Ground	S0 (13)	Standard Configuration bit 0	
G1 (S)	STR_D via D12	"	L.O table number bit 0	
G2 (N)	Ground	S1 (14)	Standard Configuration bit 1	
G2 (S)	STR_D via D13	"	L.O table number bit 1	✓
G3 (N)	Ground	S2 (15)	Standard Configuration bit 2	
G3 (S)	STR_D via D14	"	L.O table number bit 2	✓
G4 (N)	Ground	S3 (16)	Standard Configuration bit 3	
G4 (S)	STR_D via D15	"	L.O table number bit 3	
H1 (N)	STR_E	via D16 to S0 (13)	Family (see Slave specification)	
H1 (S)	OP 1 (Low L.O.)	"	2nd L.O. table number bit 0	
H2 (N)	STR_E	via D17 to S1 (14)	Family (see Slave specification)	
H2 (S)	OP 1 (Low L.O.)	"	2nd L.O. table number bit 1	
H3 (N)	STR_E	via D18 to S2 (15)	Slave has loop-through facility	
H3 (S)	OP 1 (Low L.O.)	"	2nd L.O. table number bit 2	✓
H4 (N)	STR_E	via D19 to S3 (16)	Swap Backwards Compatible tone	
H4 (S)	OP 1 (Low L.O.)	"	2nd L.O. table number bit 3	✓

Table 13 : Pin-Link Functions.

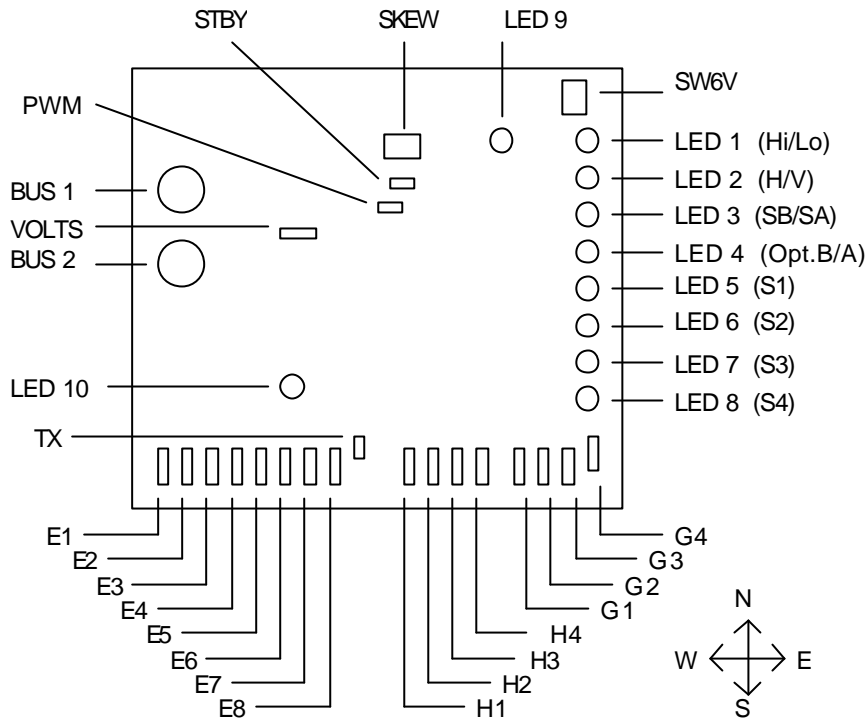


Figure 6 : Slave Circuit Board Pin-Links

9.4 Circuit Description

The DiSEqC system concept offers several new control facilities compared with existing LNBs and Switchers, for example the reverse-signalling (acknowledge) path, power-regulator capacitance-isolation and switching for power-down (standby) operation. In addition, the opportunity has been taken to demonstrate some alternative implementations for existing functions. Satellite equipment manufacturers can choose which elements and configurations best suit their particular purposes. The operation of the basic circuit elements is outlined below.

The schematic circuit diagram of the Slave PCB (APG 145B) is shown in Figure 7. Although the diagram appears to be in two isolated parts, this is done for clarity, and the parts are actually connected by their common signal names. The left-hand part of the diagram shows a typical use of the Slave chip, whilst the right-hand section provides many of the application options for configuration diode selection. Also included are the alternative PLCC package socket, and output buffers with status-display LEDs.

The board is fitted with two Bus F-connectors which are directly linked to provide a loop-through facility, as might be used in an Installer-Aid application, for example. Power from the 'F' connectors passes along a circuit track in the form of an elementary $\lambda/4$ capacitively-terminated (shorted) transmission line to feed the Slave circuitry whilst isolating the satellite signal Intermediate Frequency. Power for the peripheral hardware is normally also derived from the Bus, but a separate local supply can be applied via the power connector for Positioner or SMATV evaluation. Connection of the external supply disconnects the peripheral power supply from the bus, but the DiSEqC™ chip and its support components can derive power from either the local power supply (via D21) or the Bus (whichever is the higher voltage). Note that if the Bus and local power supply voltages are similar, this alternative supply switching may cause the Bus communications to

malfunction. A 22 volt zener diode, D1, is included to provide some protection against the accidental application of reverse voltage, or excess voltage.

The power regulator and power supply load capacitances are isolated from the DiSEqC™ Bus by the transistors TR1, 2 and 3, without the need for any inductors. The primary function of LED9 is actually as a reference voltage (~ 1.7 volts), with illumination (when active) only a secondary feature. The peripheral power supply (6v) will be off until the slave chip, IC1, sinks current (active low) via R4 and TR2. Then TR3 is brought into conduction and the output supply voltage rises until LED9 starts to “steal” TR2’s emitter current. At this point the regulator stabilises with about 4.3 volts at the junction of TR2 emitter and LED9 cathode. Although the LED is not a true reference diode, it does have a temperature coefficient which is similar to that of TR2’s base-emitter diode, so thermal stability should be quite good. To regulate to higher output voltages, LED9 could be replaced by a conventional regulator diode, but a series diode also will generally be needed to permit the stage to power-down without drawing reverse power via the reference diode.

The Bus-data transmitter, TR4, uses LED10 as a reference diode, also, to switch a reasonably stable voltage of about 1 volt across R14, giving a constant-current drive (sink) onto the bus of about 40 mA. This gives the nominal 600 mV swing across the 15Ω load at the far end (DiSEqC™ Master / IRD) of the bus cable.

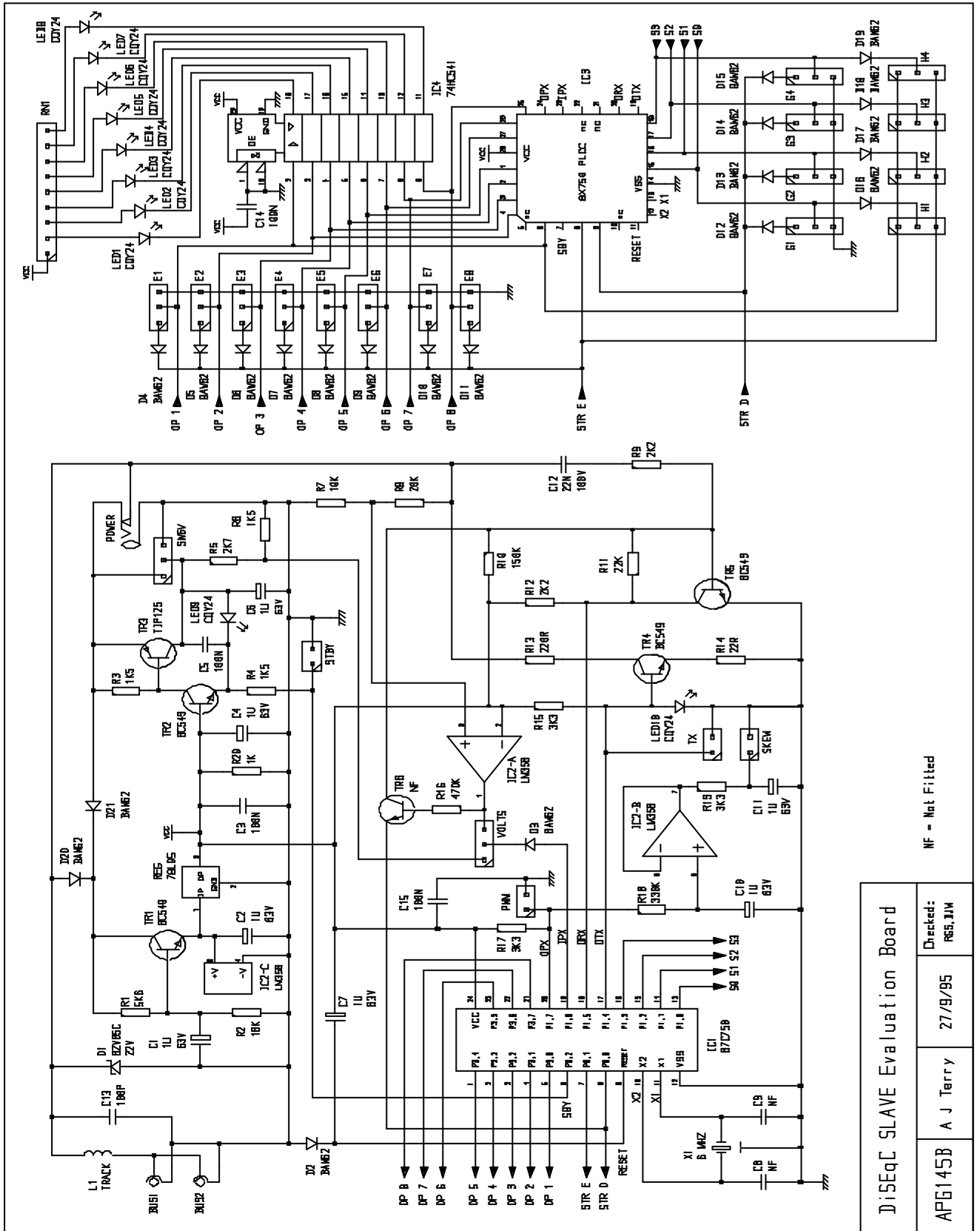
The bus-signal detector is TR5 which is biased close to saturation and with a shunt-feedback voltage gain of about 10 (R11 / R9). With a nominal minimum input tone of 150 mV peak (300 mV peak-peak), TR5 collector rises by about 1.5 volts to just reach IC1’s input threshold voltage, and just begin detection. The substantially virtual earth input impedance of TR5 gives the input circuit C12 / R9 a time constant of about 50 μs giving a high pass filtering characteristic to rapidly remove any low-frequency voltage disturbances on the Bus.

For backwards compatibility with 13 / 18 volt switching systems, a voltage detector with a nominal threshold of 15 volts and a tolerance of no more than ±1 volt is required, i.e. ± 6.7%. With a nominal 5 volt regulator tolerance of 4% - 5%, and voltage divider resistors of ± 1%, there is no scope for further errors. Therefore, a cheap Operational Amplifier, IC2-A, has been employed to give negligible offset voltage and negligible current loading on the divider resistors R7 and R8. The Op-Amp is operated from a higher voltage than the 5 volts VCC of the microprocessor so D3 is included to permit the Op-Amp to pull down the sensing pin (IPX) close to ground (VSS), but not pull it up above VCC. The second half of IC2 is used as a buffer to avoid loading the PWM low-pass filter R18 / C10. Note that the Operational Amplifier is a type which can operate with input and output signals very close to the negative (ground) supply potential.

During normal operation the configuration pins, S0 - S3 are (weakly) pulled high by the current sources internal to the DiSEqC™ chip. When the software needs to determine the external configuration, at power-up or reset, then the appropriate strobe pins are pulsed low in turn. When one or more diodes are present between appropriate pins, then the corresponding inputs are pulled low and detected by the internal software.

An exception to this scheme is that some simple applications may have some of the pins S0 - S3 permanently hard-wired low. The software detects this before attempting to strobe the pins, and sets up the appropriate default state. Another exception is that the Local Oscillator frequency-selection output is also

used as a strobe line (to signal the relevant LO frequency table value) so this remains low when the Low L.O. frequency is selected.



DiSEqC SLAVE Evaluation Board	
APG145B	A J Terry
Checked:	27/9/95
	RES, JIM

NF = Not Fitted

Figure 7 : DiSEqC™ Slave PCB Circuit Diagram

APPENDIX

Installation of User Software on a PC-Compatible Computer.

Usually, the DiSEqC™ Test Tool will be operated with a Personal Computer as the user's Input/Output device. This requires the PC to run a Terminal Emulator Utility, but a suitable program is included in all normal 'Windows™' installations, and so PC software is not included in the Test Tools Kit.

Windows™ 3.1

'Windows™ 3.1', and similar versions, normally include the utility 'TERMINAL.exe'. This is usually present in the "Accessories" program group, but, if not, it should be possible to install it from the Windows™ Installation Disks.

If the 'TERMINAL.exe' file is available, but there is no icon in the "Accessories" window, then use "Program Manager" and click **"File" : "Run"** (from the pulldown menu). Click **"Browse"** and move through the directory structure to click on 'TERMINAL.exe'. When 'TERMINAL.exe' is in the "File Name" box then click **"OK"**. The command line should now be in the "Run" window, so click on **"OK"** ("Run Minimised" should NOT be selected).

When "Terminal - [Untitled]" appears, click on **"File" : "Open"**. If a suitable configuration file such as 'TESTOOL.trm' or 'DISEQC.trm' is available, then select it, and click **"OK"**.

If a configuration file is not available (or if the Test Tool is not using the COM1 Port) then select **"Settings" : "Communications"** : and set up the parameters as shown in the box below (from the DiSEqC™ Test Tools User Manual, section 3.1, page 5). Then save the file the with a suitable name such as 'TESTOOL.trm'.

For Windows Terminal select :

"Settings" :	"Communications" :
B <u>a</u> ud Rate:	9600
D <u>a</u> ta Bits:	8
S <u>t</u> op Bits:	1
P <u>a</u> ridy:	None
F <u>l</u> ow Control:	Hardware
P <u>a</u> ridy C <u>h</u> eck:	Off
C <u>a</u> rrier Detect:	Off
C <u>o</u> nnector to suit the Port in use.	

For **"Settings" : "Terminal Preferences" :**

It is recommended to select the check-boxes :

"Line Wrap" and "Sound"

The setup may then be stored using **File : Save As...**

Finally, Power-up the Test Tool and proceed as described after the data-boxes on page 5 of the Test Tools User Manual.

Windows™ 95/98/NT4

Normally ‘Windows™ 95/98’ and ‘Windows™ NT4’ include the utility ‘HyperTerminal’. This should be found by clicking on:

“Start” : “Programs” : “Accessories” : “Hyperterminal” :

If a suitable configuration file is already available, click on it and proceed to Power-up the Test Tool, otherwise click on one of the available configurations (preferably “HyperTerminal”) and continue as follows:

When the “New Connection” panel is presented (if necessary click **“File” : “New Connection” :**) choose an icon and enter a suitable name such as ‘TESTOOL’. Select the appropriate COM port and click **“OK”**. (It may be necessary to enter a phone number before the window will close, but this will not be used for the Test Tool application.)

Set up the communication settings (if necessary click **“File” : “Properties” : “Configure”**) for the Test Tool (i.e. **9600** Bits per second, **8** Data bits, **No** Parity, **1** Stop bit, **Hardware** flow control) and click **“OK”**.

Click **“File” : “Properties” :** and the **“Settings”** tab, then choose an “Emulation” option such as “VT100”. Setting “Backscroll buffer lines” to zero is suggested. Click **“Terminal Setup”** and verify that the settings are suitable (the ASCII character set is recommended, and 132 columns is NOT recommended). Click **“ASCII Setup”** , tick the “Wrap lines ...” box and click the two **“OK”** boxes.

Save the configuration file with **“File” : “Save”** .

Power-up the Test Tool and its “sign-on” message should appear. Note that it may be necessary to move the “scroll bar” to the top of the window to see the cursor position. If the message is not displayed, check that the communication cable is plugged into the correct COM port socket, and see if any of the setup options have been missed.

If HyperTerminal is not present on the PC (note that the executable files are actually named ‘HYPERTRM.exe’ and ‘HYPERTRM.dll’), then it may be installed from the Windows Installation Disk by clicking: **“Start” : “Settings” : “Control Panel” : “Add/Remove Programs” : “Windows Setup” : “Communications” :** and ticking the “Hyperterminal” box.

If the above procedures do not find and setup a suitable Terminal Emulation Program in the PC’s ‘Windows™’ installation, then please send an e-mail report with the ‘Windows™’ version number (3.1, 95, 98, NT4, etc.) to: technical@diseqc.org