L3 Systems

KeyAT-4
Version 4.0

On the device label:
↑RS-232↑
KeyAT-4
Part # SA0064
Pb
RoHS
FCC CE
Serial No. 2602249
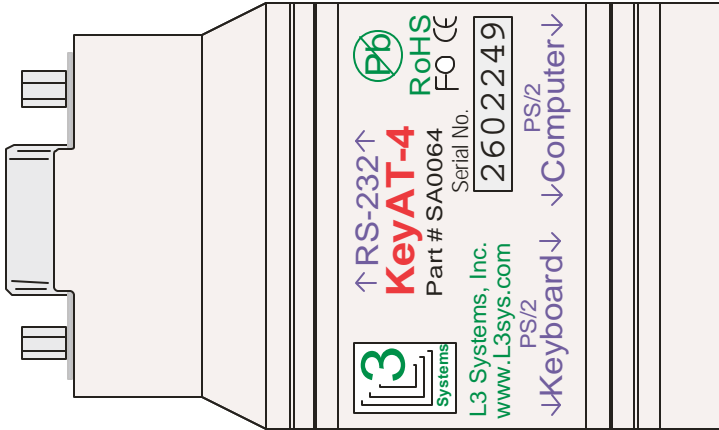PS/2 ↓Computer↓
L3 Systems, Inc.
www.L3sys.com
PS/2 ↓Keyboard↓
L3 Systems

**RS-232
Keyboard
Port Adapter**

Copyright 2006
L3 Systems, Inc.
Redmond, WA

Quick Reference

Reference	Description	Pg.
Serial Connector	Pin-outs: RS-232 Connector	2
PS/2 Connectors	Pin-outs: Keyboard/Computer Connectors	3

Command		Description	Pg.
Set to character mode	~C	Send ASCII chars	4
ASCII hex mode	~H	Send Hex ASCII chars	5
Keyboard code mode	~K	Send direct key codes	5
DOS scan code mode	~S	Send DOS scan codes	5
Turn off Commands	~@	Turn off commands (~anything)	4
Send scan code	~:nn	Sends scan code nn	5
Scan code key down	~+nn	Sends key down, scan code nn	5
Scan code key up	~-nn	Sends key up, scan code nn	5
Immediate turn around	~I	Data interpreted as received	6
Line turn around*	~L	Data buffered until <Enter>	6
Monitor Mode On/Off	~M / ~N	Monitor keyboard, key codes	6
Qwerty Monitor On/Off	~Q / ~N	Monitor keyboard, ASCII	6
Display ON/OFF LEDs	~Dnn	Sets keyboard LED status to nn	7
Transfer to keyboard	~Xnn	Sends hex byte nn to keyboard	7
Status request	~?	Response = AA- EELLDBTM	6
Sleep	~Znn	Sleeps nn seconds	7
Set turnaround delay	~Pnn	Sets turnaround to nn milli-sec	6
Send CR mode	~F	Sends <CR> through	6
Setup Mode	~*SETUP	Enter Setup Mode, Commands below:	8

	Command	Pg.		Command	Pg.
D	Set Config to default	9	LK	Load Scan Code Table	9
LC	Load Config string,	9	PK	Print Scan Code Table	10
PC	Print Config string	8	EK	Erase Key Scan Codes	10
LP	Load Power-On String	10	T	Test Keyboard Interface	11
PP	Print Power-on String	10	H	Help – Prints help info	8
LR	Load Reset String	11	V	Print Version information	7
PR	Print Reset String	11	Q	Quit Setup Mode	7

Reset	~~~~~~~~~~	Reset KeyAT & Enable commands (ten ~ characters)	10

Reference	Description	Pg.
Special Characters	^C send Ctrl "C", ~41 Sends ASCII Hex 41	12
Examples	Examples of KeyAT Commands	12
Scan Code Table	Table of Scan Code values	13
ASCII Scan Code Table	Table of ASCII Characters & Scan Codes	15

NOTE: LED Solid=Keyboard activity, Blinking=RS-232 Activity

Introduction

The KeyAT keyboard port adapter allows you to send keyboard keystrokes to a computer via an RS-232 interface. It can also be used to convert a PS/2 keyboard to send RS-232 data. Typical uses for the KeyAT are:

- **Testing software** - With the help of a KeyAT, you can send keys that simulate user input from another computer using RS-232 communications.

- **Remote Access -** The KeyAT allows remote access to a system where the software does not provide any other method.

- **Monitoring Keyboard sessions** - The KeyAT can be used to monitor keyboard sessions. This can be helpful to develop regression test data or to detect keystrokes that are causing problems with target software.

- **Serial Keyboard** - You can use the KeyAT to send and receive from a Keyboard attached to an RS-232 port.

- **Attaching input devices** – Allows you to attach RS-232 devices to the keyboard port such as bar code readers, scales, and credit card readers.

Connections

The following illustrates a typical connection of the KeyAT.

1. The keyboard connects to the 6-pin mini-DIN "To Keyboard" connector.
2. A 6 PIN male-to-male Mini-DIN cable (PN CL0008) is supplied to connect the KeyAT to the computer.
3. The Serial port with RS232 cable, typically a DB9S to DB9P cable, [PN CL0005, not supplied] See **Serial Port Operation** section for more details.
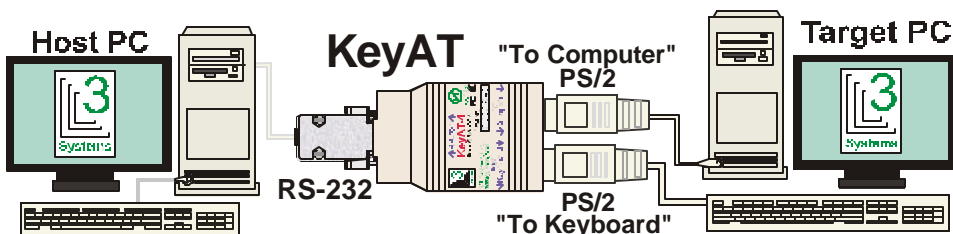

When a keystroke is received from the keyboard, the RS232 port is blocked. Likewise, when an RS232 command is received, the keyboard is blocked. Note: simultaneous use in any configuration is discouraged.

Status Indicator

A red status indicator located on the side of the KeyAT indicates port activity. It turns solid red when the attached keyboard port is being used. It blinks whenever the RS-232 port is active.
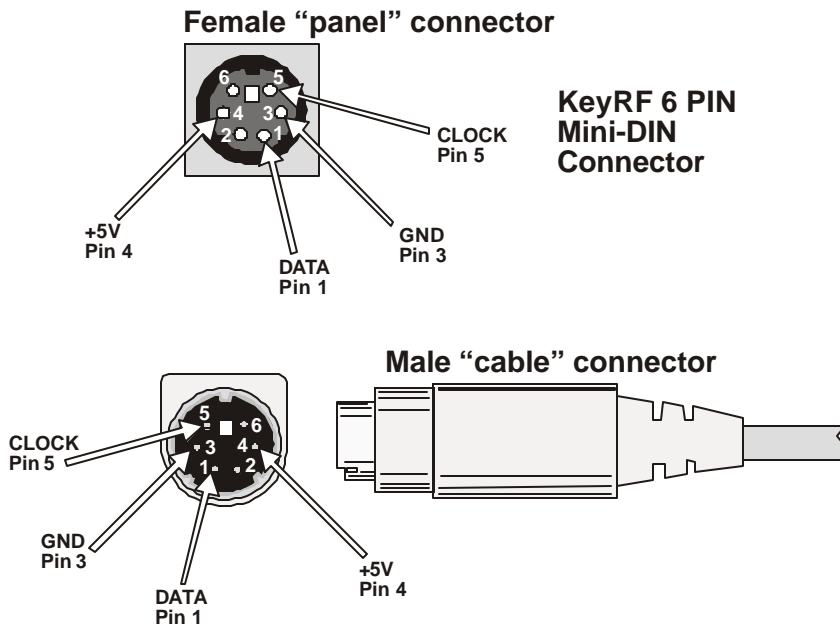
Serial Port Connector

The port is configured as a DCE device (it connects to a computer like a modem). On the 9 pin DB-9S connector, it transmits data on pin 2 and receives data on pin 3. No other RS-232 signal is required for it to operate.
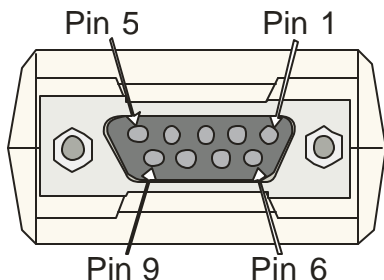
Pin	Signal	KB232 Function
1	DCD	RS-232 Output (High when Ready)
2	RX (Receive) from KB232	RS-232 Output
3	TX (Transmit) to KB232	RS-232 Input
4	DTR	RS-232 Input (Not Required)
5	Signal Ground	Ground
6	DSR	RS-232 Output (High when Ready)
7	RTS	Connected to RTS
8	CTS	Connected to CTS
9	RI / +5V Input	If K1 installed, connected to +5V Warning: do not connect this unless powering KeyAT from external source.

KeyAT Keyboard & PC Mini-DIN Connector

The following connector shows the pin definitions for the two 6 pin Min-DIN connectors. If the application is an RS-232 keyboard, the computer Mini-DIN connector can be used to provide 5 volts power.

Female "panel" connector

KeyRF 6 PIN Mini-DIN Connector

CLOCK
Pin 5

+5V
Pin 4

DATA
Pin 1

GND
Pin 3

Male "cable" connector

CLOCK
Pin 5

GND
Pin 3

DATA
Pin 1

+5V
Pin 4

K1 Jumper

K1 connects DB-9 Pin 9 to 5 volts. This can be used to power KeyAT from DB-9 connector. **Warning: This is not normally connected. If 5V power is connected to PC, then damage can occur if also connected to external supply!**

Sending Data

The main job of the KeyAT is to transfer data from the RS-232 port and send it through to the computer's keyboard port. Data can be sent using ASCII, ASCII control codes, DOS scan codes, keyboard scan codes or ASCII hex. A line can have a maximum length of 80 characters and should end with an <ENTER> (CR, hex 0D). When using immediate mode, data and commands are interpreted as received, and buffered up to 80 characters. If you get more than 80 characters ahead of the computer, the additional characters will be ignored.

In line turn-around mode, the KeyAT sends a colon prompt after the data has been processed, signaling that it is ready for another command. The example shows the ASCII characters 'DIR<ENTER>' being sent through the KeyAT.

```
Example:    KeyAT, V4.0x1, [C]L3 Systems, Inc. 2005
            :DIR^M<ENTER>
            :
```

Commands

Commands are instructions to the KeyAT to do something other than just pass data through to the computer. All commands begin with a ~ (tilde, hex 7E) character followed by a command character and then in some cases some data.

~@ **Disable Commands** – All commands are disabled until 10 consecutive tildes (~~~~~~~~~~) or reset/power cycle.

~~~~~~~~~~ **Reset/Enable Commands** – If commands are disabled, sending 10 consecutive tilde (~) characters will enable commands. If commands are enabled, this will reset the KeyAT (& run power-up string if loaded).

Data Mode Commands

The four mode commands set the way data is interpreted by the KeyAT. A mode is set until another mode is given or until a power failure occurs.

~C **Character Mode** - is the default mode of operation. When this mode is set, any ASCII character except ~ (tilde, hex 7E), ^ (caret, hex 5E), and CR (return, hex 0D) is converted to the scan codes for that character. Each ASCII character will be sent to the computer with the corresponding "make" and "release" codes. Allow enough time for the codes to be sent through to the computer.

Control codes may be sent by a combination of a ^ (caret, hex 5E), and the corresponding letter. For example, a control C (Hex 03) is ^C.

~H **Hex ASCII Mode** - Data is interpreted as hexadecimal encoded ASCII. For example, to send the computer the character 'A', you would need to send its hexadecimal ASCII value (Hex 41) as two ASCII digits, 4 and 1.

~S   **Scan Code Mode** - In this mode, data is translated to the corresponding keyboard scan codes. For an ASCII transfer, each pair of ASCII encoded characters is interpreted as a DOS scan code. DOS assigns each key a DOS scan code. Note that all DOS scan codes are sent complete with make and break sequences.

~K   **Key Code Mode** - In this mode, codes are sent directly to the computer without interpretation. For an ASCII transfer, each pair of ASCII encoded characters are interpreted as one keyboard code. Care should be taken in using keyboard scan codes, as the computer can be left in a very confused state if the scan code sequence is not properly completed.

Send a scan code

If you want to send a scan code without going into scan code mode, then use the ~:nn command. This is useful for the occasional function key, arrow key or any other keys that doesn't have an ASCII equivalent. See the scan code table later on for a list of valid scan codes.

~:nn   **Send Scan Code** – This scan code nn.

       Example: `:~:73<ENTER>` *(sends PgUp)*

Up/Down Scan Codes

Use these two commands to send a key down or a key up to the computer. A common use is for Shift, Alt and Cntl keys, which vary the meaning of successive scan codes. See the scan code table later on for a list of valid scan codes.

~+nn   **Key down Scan Code** – This sends just the key down (make) sequence for scan code nn.

~-nn   **Key up Scan Code** – This sends just the key up (break) sequence for scan code nn.

     Example:

       `:~+56~:62~-56<ENTER>` *(sends <Alt>F4)*

       where: **~+56** *Sends Alt down sequence*
                    ~:62 *Sends F4 scan code*
                    ~-56 *Sends Alt up sequence*

Turn around setting

Data can be sent and interpreted one line at a time, or be processed immediately. The default setting is for line turn around mode (~L). This is better for automated control and provides more reliable interaction. Immediate turn around mode (~I) can better simulate keyboard operation. The setting persists until the next ~I or ~L command, or until power is cycled.

~L **Line Turn Around** - Set to a line oriented basis, each line must end with an <ENTER> or <CR>, hex OD.

~F **Send <CR> mode** - This allows the <CR> to be sent on to the computer as an <ENTER> key.

~I **Immediate Turn-around** - Data is interpreted 'on-the-fly'. Note that in this mode <ENTER> or <CR> (Hex 0D) is not a special character, and passed on to the computer. XON/XOFF flow control is provided. However, caution should exercised, as it is easy to overrun the PC's keyboard receive buffer.

~Pnn **Set Turn Around Delay (nn milliseconds)** - This sets the delay that the KeyAT waits before responding to a command. The default setting is 0. (Note: you must send two digits. For 7 milliseconds, send ~P07.)

Keyboard Monitor

The KeyAT can send either binary key codes (~M) or ASCII translations (~Q) of keys struck at the local keyboard through to the RS-232 port. The QWERTY monitor can be used for constructing an RS-232 keyboard, and it will send only keys with ASCII translations.

~M **Monitor ON –** Monitor Key Scan codes.
~Q **QWERTY Monitor ON –** Monitor ASCII translation
~N **Monitor OFF -** Turn keyboard monitor off (default).

Status Request

~? **Request Status** - The format of the response is:

> **Stat=LDTM**   where:
> **L** LED status : "0"=All Off, "1"=Scroll-Lock ON, "2"=Num-Lock ON
>   "3"=Scroll-Lock & Num-Lock ON , "4"=Caps-Lock ON
>   "5"=Scroll-Lock & Caps-Lock ON , "6"=Scroll-Lock & Caps-Lock ON
>   "7"=Scroll-Lock, Num-Lock & Caps-Lock ON
> **D** Data mode (C = char, H = Hex, K = Key code, S = Scan code)
> **T** Turn Around (I = Immediate, L = Line oriented)
> **M** Monitor mode (M = Monitor on, Q = Qwerty monitor, N = None)

Sleep

~Znn **Sleep** – Use sleep to provide delays. The value nn is in seconds (approx). This can be helpful when doing scripting, allowing the KeyAT to help you pace keystrokes, not out-running the application.

> `*~Z15<ENTER>` *(delays 15 seconds)*

Keyboard Commands

The following two commands send data to the attached keyboard. *Note: Do not send keyboard commands at the same time you are sending data to the computer, as this can cause the keyboard to miss the command.*

Controlling Keyboard LED's

~Dnn **Set/Clear Keyboard LED's** - Use this command to turn on or off local keyboard LED indicators. The nn field is an ASCII hex encoded binary field, where:

b0=Scroll lock   b1=Num Lock   b2=Caps Lock

Examples: ~D01 Scroll Lock on, Caps & Num Lock off,
        ~D04   Caps Lock on, Scroll & Num Lock off,
        ~D00   Turn off all LEDs

Sending Data to the Keyboard

~Xnn **Send Data to Keyboard** - Use this command to send the ASCII hex encoded byte nn to the keyboard. This command requires technical knowledge of keyboards.

<u>Setup Mode</u>

The KeyAT has commands for configuration, status and testing which are accessed by entering setup mode with following command:

> `*~*SETUP<ENTER>`
> `Setup>`

Leaving Setup Mode

You can exit setup mode with the "Q" commad:

> `Setup>Q<ENTER>`
> `:`

Displaying Version

The firmware version can be displayed with the "V" commad:

> `Setup>V<ENTER>`
> `KeyAT, V4.0x1, [C]L3 Systems, Inc. 2006`
> `:`

Help Command

You can display a short setup help list with the "Q" commad:

```
Setup>H<ENTER>
~:nn Scancode nn
~-nn/~+nn Make/Break Scancode nn
~Knn Send Keycode nn
~Dnn Set Keyboard LEDs
~Xnn Send Keyboard byte nn
~Znn Delay nn
~nn ASCII byte
^X Control-X
--- Setup Commands ---
Pn-Print, Ln<String>-Load
  n=C - Config Buffer
  n=K - Key Scan Codes
  n=P - Power-On Buffer
  n=R - Reset Buffer
DC Set Default Config
EK Erase Key Codes
H Help, V Version, I LED Status
:
```

Configuration string

The KeyAT stores operating parameters in a configuration string located in non-volatile flash memory. These include parameters such as the settings of the baud rate, enabling command mode and determining the turn-around character.

Displaying Configuration

To display the configuration string, do the following command. (Note that the results shown below are the default settings.)

```
Setup>PC<ENTER>  (Displays Configuration Data)
9600,NoPar,YesCmd,NoCR,LM,NM,0D,7E
Setup>
```

The configuration settings are defined as follows. The underlined characters represent the minimum required entry in each field:

9600 – *Baud Rate*: 1200, 2400, 4800, 9600, 19200, 38400, and 57600 are valid parameters. 9600 is the default setting.

NoPar – *RS-232 Parity*: Valid settings are NoPar for no parity, Even for even parity, and Odd for odd parity. The default is NoPar.

YesCmd – *Command mode*: YesCmd enables KeyAT commands and NoCmd disables these commands. The default is YesCmd.

NoCR – *CR Pass through mode*: NoCR does not send through the <CR> or <ENTER> at end of command line. YesCR will send it through.

LM – *Entry Mode*: LM (default) enables "line Mode", IM enables Immediate mode

NM – *Monitor Mode:* NM (default) No Monitor Mode, MM enables Monitor Mode QM enables QWERTY mode

OD – *End-of-line character*: Default is 0D, (<CR> or <ENTER> character)

7E – *Command prefix character*: Default is 7E, tilde (~) character.

Changing Configuration

You can change operating parameters of the KeyAT adapter with the LC command. The following example shows the baud rate changed to 4800 from default, and enabling the "CR pass-through mode":

```
Setup>LC4800,NoPar,YesCmd,YesCR,LM,NM,0D,7E<ENTER>
```

The following shows the minimal entry for the same setting:

```
Setup>LC48,N,Y,Y,LM,NM,0D,7E<ENTER>
```

Powering on to Default Settings

Should you make a mistake in configuring the settings of the KeyAT configuration string, you can force it to use the default settings if you send an RS-232 "Break" condition to the when applying power, This will cause the KeyAT to return to the following settings:

```
9600,NoPar,YesCmd,NoCR,LM,NM,0D,7E
```

Specifically, this will force it operate at 9600 baud with no parity. If you use the application "Tera Term", send ten Alt-B's and then quickly connect the KeyAT while the break is in progress.

Restoring Configuration Default Settings

Restore the KeyAT default configuration with the D command:

```
Setup>D<ENTER>     (Restores default settings)
Setup>
```

Key Scan Code Table

The KeyAT allows you to define custom keyboard scan codes for any ASCII character in the Code Table.

Loading Key Scan Codes

The LK command loads a key scan code table entry:

LKaa=bb,cc…   **Loading Key Scan Codes**

   Example: `Setup>LK2A=79,F0,79<ENTER>` *(* maps to a "+" key)*

In the above example, an entry is made to define what happens when the ASCII asterisk character "*" is sent via the RS-232 port to the computer. Normal mappings are shown in the ASCII Scan Code table at the end of this manual. The custom scan code table can be used to map keys to any ASCII character.

The key code "79" defines the pressing of the "+" key, and the key codes "F0" and "79" define the removal of the "+" key. Refer to the "Scan codes" table at the end of this manual for key code values.

The table space allows for 50 entries

Displaying Code Table

The code table can be displayed to review entries:

```
        :PK<ENTER>       (Displays Code Table)
AS Codes
-- ------
2A 79 F0 79      ← Shows data of example above
:
```

Erasing the Key Code Table

The "EK" command deletes all entries in the Key Scan Code table. If you want to change existing entries in the code table, you must clear the table with the "EK" command and re-enter the codes with the "LK" command.

```
        :EK<ENTER>       (EraseKey Scan Code Table)
        :
```

Power On String

The KeyAT can execute a command string when power is applied. This string can include keys to send to the computer and/or KeyAT commands. It's strongly recommended when using this feature to start the string with a delay command (~Znn) to allow the KeyAT to wait for the computer to finish its power on sequence.

Loading the Power ON String

The LP command loads a code table entry:

LP=<*String*>   **Loading Power On String**

   Example: `Setup>LP=~Z80DIR^M<ENTER>` *(Delay 80 seconds, & send "DIR<ENTER>")*
            `Setup>`

In the above example, when power is applied to the KeyAT, it first delays 80 seconds, and then sends a "DIR<Enter>" command.

Displaying Power on string

The **PP** command displays the Power On string:

```
        Setup>PP<ENTER>
        ~Z80DIR^M              (Shows Power On string loaded in example above)
        Setup>
```

Reset String

The KeyAT can execute a command string whenever the computer sends the keyboard a reset. This string can include keys to send to the computer and/or KeyAT commands. It's strongly recommended when using this feature to start the string with a delay command (~Znn) to allow the KeyAT to wait for the computer to finish its reset sequence.

Loading the Reset String

The **LR** command loads a code table entry:

LR=<*String*>   **Loading Reset String**

> **Example**: `Setup>`<u>`LR=~Z80DIR^M<ENTER>`</u> *(Delay 80 seconds, & send "DIR<ENTER>")*
> `Setup>`

In the above example, when the computer sends the keyboard a reset, it first delays 80 seconds, and then sends a "DIR<Enter>" command.

Displaying Reset string

The **PR** command displays the Power On string:

> `Setup>`<u>`PR<ENTER>`</u>
> `~Z80DIR^M`   *(Shows Reset string loaded in example above)*
> `Setup>`

Test KeyAT

T   **Test KeyAT** - This command to the KeyAT causes it to undergo tests to validate the keyboard and computer interface. You will see the keyboard LED's cycle, testing the keyboard interface.

> `Setup>`<u>`Y<ENTER>`</u>
> `Test: <Enter> to stop` *(Keyboard LEDs cycle, pressing <ENTER> halts test)*
> `Setup>`

The test runs until<ENTER> is received.

Special Characters

ASCII Hex (~nn) - ASCII hex characters can be sent by sending the hex code preceded by a ~(tilde, hex 7E). For example, to send an M character in hex, send a ~4D.

Tilde and Caret Characters (~~ ^^) - To send a ~ (tilde, hex 7E) or ^ (caret, hex 5E) character, send the character twice. For example a ~~ will send the key codes for the ASCII character ~.

Tera Term

For use with Microsoft Windows, we suggest using the application called Tera Term. It's free to download and use. Tera Term is a terminal emulator that also has a macro (or scripting) capability. The terminal emulator provides a simple way to test and experiment the operation of the KeyAT. You can also create scripts to automate the keystrokes.

Examples:

Sending Keys that don't have ASCII eqvivalents:

```
:~:59<ENTER>            Sends F1 key
:~+29~:59~-29<ENTER>    Sends Ctrl-F1 key
:~+56~:59~-56<ENTER>    Sends Alt-F1 key
:~:72<ENTER>            Sends Up-Arrow key
```

Use of different data modes:

```
:~C<ENTER>              Sets to character mode
:DIR^M<ENTER>           Sends DOS a directory command

:~H<ENTER>              Sets to ASCII HEX mode
:4449520D<ENTER>        Sends DOS a directory command

:~S<ENTER>              Sets to scan code mode
:32231928<ENTER>        Sends DOS a directory command

:~K<ENTER>              Sets to key code mode
:23F02343F0432DF02D5AF05A<ENTER>
                        Sends DOS a directory command
```

Autoexec Command for Windows (uses boot sense):

```
:~LR=~Z99~:28~Z30~+56~:63~-56<ENTER>
```
On boot, waits 99 seconds for boot to finish, sends a <ENTER> (Win 95 Network logon), waits another 30 seconds, and sends an Alt-F5. The Alt-F5 would be assigned as a key shortcut for a desktop application that you wish to run.

Scan Codes

Below is the table of Scan definitions.

AT Scan Code	KeyAT SCAN Code	KEY	Keyboard Scan Codes	AT Scan Code	KeyAT SCAN Code	KEY	Keyboard Scan Codes
01	01	ESC	76 F0 76	37	37	K	42 F0 42
02	02	1 / !	16 F0 16	38	38	L	4B F0 4B
03	03	2 / @	1E F0 1E	39	39	; / :	4C F0 4C
04	04	3 / #	26 F0 26	40	40	' / "	52 F0 52
05	05	4 / $	25 F0 25	41	41	` / ~	0E F0 0E
06	06	5 / %	2E F0 2E	42	42	Left Shift	12 F0 12
07	07	6 / ^	36 F0 36	43	43	\ / |	5D F0 5D
08	08	7 / &	3D F0 3D	44	44	Z	1A F0 1A
09	09	8 / *	3E F0 3E	45	45	X	22 F0 22
10	10	9 / (	46 F0 46	46	46	C	21 F0 21
11	11	0 /)	45 F0 45	47	47	V	2A F0 2A
12	12	**- / _**	4E F0 4E	48	48	B	32 F0 32
13	13	**= / +**	55 F0 55	49	49	N	31 F0 31
14	14	Backspace	66 F0 66	50	50	M	3A F0 3A
15	15	Tab	0D F0 0D	51	51	**, / <**	41 F0 41
16	16	Q	15 F0 15	52	52	**. / >**	49 F0 49
17	17	W	1D F0 1D	53	53	**/ / ?**	4A F0 4A
18	18	E	24 F0 24	53	93	**/**	E0 4A E0 F0 4A
19	19	R	2D F0 2D	54	54	Right Shift	59 F0 59
20	20	T	2C F0 2C	55	55	*****	7C F0 7C
21	21	Y	35 F0 35	55	A4	Prt Scrn	E0 12 E0 7C E0 F0 7C E0 F0 12
22	22	U	3C F0 3C	56	56	Left Alt	11 F0 11
23	23	I	43 F0 43	56	A5	Right Alt	E0 11 E0 F0 11
24	24	O	44 F0 44	57	57	Space	29 F0 29
25	25	P	4D F0 4D	58	58	Caps Lock	58 F0 58
26	26	**[/ {**	54 F0 54	59	59	F1	05 F0 05
27	27	**] / }**	5B F0 5B	60	60	F2	06 F0 06
28	28	Enter	5A F0 5A	61	61	F3	04 F0 04
28	A0	Kpad Enter	E0 5A E0 F0 5A	62	62	F4	0C F0 0C
29	29	Left Ctrl	14 F0 14	63	63	F5	03 F0 03
29	A1	Right Ctrl	E0 14 E0 F0 14	64	64	F6	0B F0 0B
29+69	A2	PAUSE	E1 14 77 E1 F0 14 F0 77	65	65	F7	83 F0 83
30	30	A	1C F0 1C	66	66	F8	0A F0 0A
31	31	S	1B F0 1B	67	67	F9	01 F0 01
32	32	D	23 F0 23	68	68	F10	09 F0 09
33	33	F	2B F0 2B	69	69	Num Lock	77 F0 77
34	34	G	34 F0 34	70	70	Scr Lock	7E F0 7E
35	35	H	33 F0 33	71	71	Home	E0 6C E0 F0 6C
36	36	J	3B F0 3B	71	A6	Keypad Home / 7	6C F0 6C

Scan Codes (cont.)

AT Scan Code	KeyAT SCAN Code	KEY	Keyboard Scan Codes	AT Scan Code	KeyAT SCAN Code	KEY	Keyboard Scan Codes
72	72	Up Arrow	E0 12 E0 75 E0 F0 75 E0 F0 12	81	B3	Keypad PgDn / 3	7A F0 7A
72	A7	Kpad Up Arrow / 8	75 F0 75	82	82	Insert	E0 70 E0 F0 70
73	73	Page Up	E0 7D E0 F0 7D	82	B4	Ins/0	70 F0 70
75	75	Lt Arrow	E0 12 E0 6B E0 F0 6B E0 F0 12	83	83	Delete	E0 12 E0 71 E0 F0 71 E0 F0 12
75	A9	Kpad Left Arrow / 4	6B F0 6B	83	B5	Kpad Del / .	71 F0 71
76	76	5	73 F0 73	84		Undefined	
77	77	Rt Arrow	E0 12 E0 74 E0 F0 74 E0 F0 12	85		Undefined	
77	B0	Kpad Rt Arrow / 6	74 F0 74	86		Undefined	
78	78	+	79 F0 79	87	87	F11	78 F0 78
79	79	End	E0 12 E0 69 E0 F0 69 E0 F0 12	88	88	F12	07 F0 07
79	B1	End/1	69 F0 69	89		Undefined	
80	80	Down Arrow	E0 12 E0 72 E0 F0 72 E0 F0 12	90	90	Left Window	E0 1F E0 F0 1F
80	B2	Kpad Dn Arrow / 2	72 F0 72	91	91	Right Window	E0 27 E0 F0 27
81	81	Page Down	E0 12 E0 7A E0 F0 7A E0 F0 12	92	92	Menu	E0 2F E0 F0 2F

ASCII Scan Codes

Below is the table of ASCII scan code definitions.

ASCII CHAR	ASCII HEX	KEYBOARD SCAN CODES	ASCII CHAR	ASCII HEX	KEYBOARD SCAN CODES
NUL ^@	00	14 1E F0 1E F0 14	Space	20	29 F0 29
SOH ^A	01	14 1C F0 1C F0 14	!	21	12 16 F0 16 F0 12
STX ^B	02	14 32 F0 32 F0 14	"	22	12 52 F0 52 F0 12
ETX ^C	03	14 21 F0 21 F0 14	#	23	12 26 F0 26 F0 12
EOT ^D	04	14 23 F0 23 F0 14	$	24	12 25 F0 25 F0 12
ENQ ^E	05	14 24 F0 24 F0 14	%	25	12 2E F0 2E F0 12
ACK ^F	06	14 2B F0 2B F0 14	&	26	12 3D F0 3D F0 12
BEL ^G	07	14 34 F0 34 F0 14	'	27	52 F0 52
BS ^H	08	66 F0 66	(	28	12 46 F0 46 F0 12
TAB ^I	09	0D F0 0D	)	29	12 45 F0 45 F0 12
LF ^J	0A	14 3B F0 3B F0 14	*	2A	7C F0 7C
VT ^K	0B	14 42 F0 42 F0 14	+	2B	79 F0 79
FF ^L	0C	14 4B F0 4B F0 14	,	2C	41 F0 41
CR ^M	0D	3A F0 3A	-	2D	7B F0 7B
SO ^N	0E	14 31 F0 31 F0 14	.	2E	49 F0 49
SI ^O	0F	14 44 F0 44 F0 14	/	2F	4A F0 4A
DLE ^P	10	14 4D F0 4D F0 14	0	30	45 F0 45
DC1 ^Q	11	14 15 F0 15 F0 14	1	31	16 F0 16
DC2 ^R	12	14 2D F0 2D F0 14	2	32	1E F0 1E
DC3 ^S	13	14 1B F0 1B F0 14	3	33	26 F0 26
DC4 ^T	14	14 2C F0 2C F0 14	4	34	25 F0 25
NAK ^U	15	14 3C F0 3C F0 14	5	35	2E F0 2E
SYN ^V	16	14 2A F0 2A F0 14	6	36	36 F0 36
ETB ^W	17	14 1D F0 1D F0 14	7	37	3D F0 3D
EM ^X	18	14 22 F0 22 F0 14	8	38	3E F0 3E
SUB ^Y	19	14 35 F0 35 F0 14	9	39	46 F0 46
SUB ^Z	1A	14 1A F0 1A F0 14	:	3A	12 4C F0 4C F0 12
ESC ^[	1B	76 F0 76	;	3B	4C F0 4C
FS ^\	1C	14 5D F0 5D F0 14	<	3C	12 41 F0 41 F0 12
GS ^]	1D	14 5B F0 5B F0 14	=	3D	12 55 F0 55 F0 12
RS ^^	1E	14 12 36 F0 36 F0 12 F0 14	>	3E	12 49 F0 49 F0 12
US ^_	1F	14 12 4E F0 4E F0 12 F0 14	?	3F	12 4A F0 4A F0 12

ASCII Scan Codes (cont.)

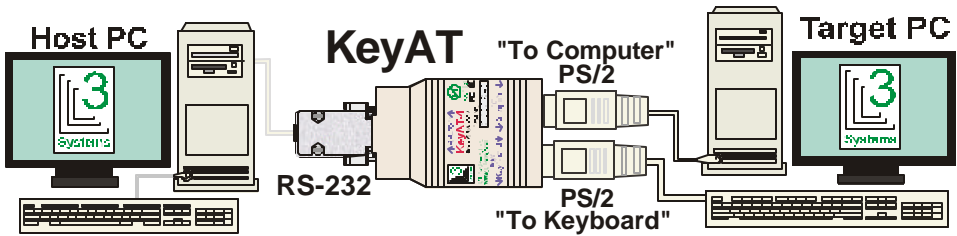| ASCII CHAR	ASCII HEX	KEYBOARD SCAN CODES	ASCII CHAR	ASCII HEX	KEYBOARD SCAN CODES
@	40	12 1E F0 1E F0 12	`	60	0E F0 0E
A	41	12 1C F0 1C F0 12	a	61	1C F0 1C
B	42	12 32 F0 32 F0 12	b	62	32 F0 32
C	43	12 21 F0 21 F0 12	c	63	21 F0 21
D	44	12 23 F0 23 F0 12	d	64	23 F0 23
E	45	12 24 F0 24 F0 12	e	65	24 F0 24
F	46	12 2B F0 2B F0 12	f	66	2B F0 2B
G	47	12 34 F0 34 F0 12	g	67	34 F0 34
H	48	12 33 F0 33 F0 12	h	68	33 F0 33
I	49	12 43 F0 43 F0 12	i	69	43 F0 43
J	4A	12 3B F0 3B F0 12	j	6A	3B F0 3B
K	4B	12 42 F0 42 F0 12	k	6B	42 F0 42
L	4C	12 4B F0 4B F0 12	l	6C	4B F0 4B
M	4D	12 3A F0 3A F0 12	m	6D	3A F0 3A
N	4E	12 31 F0 31 F0 12	n	6E	31 F0 31
O	4F	12 44 F0 44 F0 12	p	70	4D F0 4D
P	50	12 4D F0 4D F0 12	q	71	15 F0 15
Q	51	12 15 F0 15 F0 12	r	72	2D F0 2D
R	52	12 2D F0 2D F0 12	s	73	1B F0 1B
S	53	12 1B F0 1B F0 12	t	74	2C F0 2C
T	54	12 2C F0 2C F0 12	u	75	3C F0 3C
U	55	12 3C F0 3C F0 12	v	76	2A F0 2A
V	56	12 2A F0 2A F0 12	w	77	1D F0 1D
W	57	12 1D F0 1D F0 12	x	78	22 F0 22
X	58	12 22 F0 22 F0 12	y	79	35 F0 35
Y	59	12 35 F0 35 F0 12	z	7A	1A F0 1A
Z	5A	12 1A F0 1A F0 12	{	7B	12 54 F0 54 F0 12
[	5B	54 F0 54	|	7C	12 5D F0 5D F0 12
\	5C	5D F0 5D	}	7D	12 5B F0 5B F0 12
]	5D	5B F0 5B	~	7E	12 0E F0 0E F0 12
^	5E	12 36 F0 36 F0 12	Del	7F	71 F0 71

<u>Warranty</u>: L3 Systems guarantees this product to be free of defects in material and workmanship for 180 days from date of shipment to the end user. L3 Systems will repair or replace (at our option) products within the warranty period at no charge for parts and labor. All returns must obtain a Return of Merchandise Authorization number (RMA) available on request from L3 Systems. Shipping costs (plus customs and duty, if any) to and from L3 Systems must be paid by the user. Damage or defect caused by accident, misuse or neglect is not covered. Damage or defect caused by shipping is excluded. L3 Systems shall not be liable for any consequential damage or losses from the use of, or inability to use its products. Any unauthorized repair or modification of the product voids the warranty. L3 Systems makes no other warranty, express or implied, nor have we authorized anyone to make representations to the contrary.

<u>FCC Class B Approval Information</u>

NOTE: This equipment has been tested and found to comply with the limits for a class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

KeyAT Keyboard Port Adapter

The KeyAT Keyboard Port Adapter allows you to make an RS-232 connection to a computer's keyboard port, and still use the keyboard. Typical uses are:

♦ **Testing Software** - Test software by sending it simulated keystrokes from another computer.

♦ **Remote Access** - Use the KeyAT to remotely access systems where software does not provide other options.

♦ **Monitor Keyboard Sessions** - You can monitor keyboard session to develop keyboard test scripts.

♦ **Connect a Keyboard to an RS-232 Port** - You can use a KeyAT to connect a keyboard to a computer's RS-232 port, and receive key codes from the keyboard.

♦ **Attaching input devices** – Allows you to attach RS-232 devices to the keyboard port such as bar code readers, scales, and credit card readers.

Some advanced features of the KeyAT Adapter are:

♦ **Accepts a Variety of Keyboard Data** - Accepts printable ASCII and encoded ASCII Hex, DOS scan codes, and direct keyboard key codes.

♦ **LED Use Indicator** - Status of remote connect.

♦ **Computer Keyboard Port Cable Provided** - Just plug it in and go.