Keyboard Serial Terminal
Manual

Quickly get to the real fun. Copy-paste software and send keystrokes from a modern computer to an Apple-1 or Apple][through a serial (or USB) interface.

Prerequisites

A serial cable is not included in the kit. It is needed to connect the **K**eyboard **S**erial **T**erminal (KST) to the modern computer using connector J9. Depending on what kind of connector is available on the “modern” computer a different cable is needed.

COM Port (male DE-9 connector)

Computers and laptops from the 80s until beginning 2000s have these COM ports. In this case an **RS232 Null Modem cable** is needed. Any female-to-female DB-9 serial cable can be used where the transmit and receive lines are cross linked. Among many others, a StarTech.com cable [1] is suitable for the job (*see* figure at the right).

An RS232 Null Modem cable

USB Port (USB-A connector)

In most cases an **USB to RS232 adapter cable** is preferred. A null modem cable (see above) is sometimes needed as well to change the gender and cross link the transmit and receive lines, depending on the serial ‘output’ gender of the USB cable.

A preferred single cable solution is, for example, the StarTech.com *1 Port FTDI USB to RS232 Serial Null Modem Adapter Cable* [2] (*see* the left figure below). This cable also has receive and transmit indicator leds, which are a nice plus.

The blueish and silver transparent cables (*see* the right figure) are very common, most of the time these cables need an extra null modem cable or converter. Always, cables with an *FTDI chipset* are preferred.

Single cable solution

Cable example that needs an extra null modem cable

Compatibility

The KST is compatible with the following vintage computers:

- an Apple-1 (being a clone or original),
- Apple][,
- Apple][Plus,
- Apple][Europlus
- or ITT 2020 (being an Apple][clone)

> The board is not compatible with later Apple][series, like the *II*c, *II*e or *II*gs.
>
> Furthermore, it is not tested to work with replica's of the compatible computers, for example Vince Briel's Replica 1 [3].

In theory the KST can work with any other vintage (single board) computer that takes a strobed parallel ASCII data signal as (keyboard) input. For example a KIM-1 [4] or Processor Technology SOL-20 [5]. However, this is uncharted territory and for the user to explore.

Assembly

Make sure you have all the parts, see the *Bill of Materials*. Always solder by vertical height of the components, from low to high:

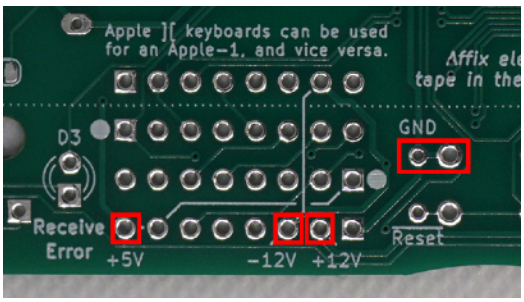
- Start with the three **resistors** (R1-R3). Take note that resistor R2 has to be put in at the back (solder) side and soldered at the front (component) side of the board. For best results use a plastic lead bend tool, or bend the leads by hand, before inserting the resistor into the board. Cut the leads after soldering, then reheat again to reflow the solder.
- Then the **jumper header** (JP1). It has to be put in on the back side and soldered on the front side of the board. Solder one pin first, then melt it again while aligning the header correctly. After that (re)solder all pins. Watch out to not touch the same pin that is being soldered, it gets hot!
- One shiny **crystal** (Y1) is next. Orientation does not matter.
- Solder the three **push buttons** (SW1-SW3). Cut the pins short on the backside, then resolder. When using external buttons (on the computer case for example) the *solder holes* can be used instead. See the markings of these holes at the back side of the board.
- Eight rows of 8-pin **female in-line machined pin sockets** (J1-J8) have to be soldered in.
  - Insert the four rows (J1-J4) on the back side and solder them at the front side. Do these in two pairs: J1 & J3 together and J2 & J4 together. Use the *16 pin male-male IC adapter* (not the IC socket as its pins are thicker) as a guide when soldering each pair. This way the in-line sockets will be soldered in vertically correct. Solder one pin of each row first. Then melt the solder of this pin again while moving the adapter on the other side in the perfect angle. Finish by soldering the rest of the pins. Finally remove the IC adapter in a vertical direction, preferably using an IC extractor tool, as the pins are bend easily.
  - Repeat the same trick for the other four rows (J5-J8). These are inserted at the front side and soldered at the back side. Use J5 & J7 and J6 & J8 as pairs with the 16 pin male-male IC adapter as a guide.
- Next, the four IC **sockets** (U1-U4). Make sure they are oriented correctly, check if the notch of the socket matches the outline on the board. Start with soldering two opposite pins, then melt the soldered pins again while pushing the socket all the way down on the top side. When the socket sits nicely, then (re)solder all pins. Watch out for the solder jumper JP2, keep it unsoldered.
- Put in the five **ceramic capacitors** (C7-C11). Insert them in the board (orientation does not matter) and solder them. Then cut the leads and reheat again to reflow for best results.

- The **LEDs** (D1-D3) follow. Note that D3 (red) has to be put in at the back side and soldered at the front side of the board. The long leg is the anode (+) and the short leg the cathode (-). The short leg should go through the square pad and the long leg through the round pad. When using external LEDs the three solder holes marked T,R,K can be used, *see* the Schematics.

- The six **capacitors** (C1-C6) go next, make sure the cathode (-, short leg) side of the capacitor is in the shaded, 'filled' half of the circle on the board. The anode (+, long leg) should be in the 'empty' half of the circle. Take note that one of the capacitors has a larger capacitance (C6, 47 µF), it should go in the top left and has a bigger footprint on the board.

- Mount the **RS232 connector** (J9). Push the connector down gently until the big feet 'snap' into the two bigger holes. Solder one pin and check if the connector is mounted correctly. If needed, adjust it by melting the solder and pressing the connector at the same time. Then solder all the pins, also the two bigger mounting pins (better to use a bigger soldering tip and thicker solder for those).

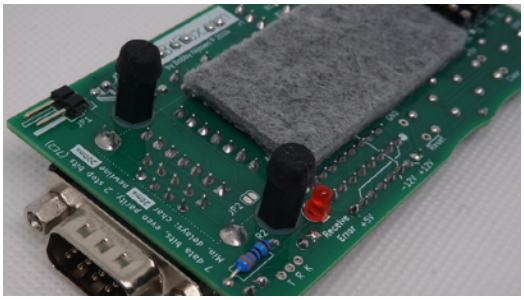
- Using a multimeter in continuity mode, **test for shorts** between all side to side pins J1-J8. Also check for shorts between all combinations of the GND, +5V, -12V and +12V rails. *See* the figure at the right with the points to check.



Points to check for shorts (back side PCB).

- Install the two **standoffs** with the **washer** (in between the stands and the PCB) on the solder side. Screw (turn clockwise) and and at the same time push the **rubber end caps** in the bottom side of the stands. *See* lower figure. A drip of glue (rubber suitable) makes it even more solid. This way the motherboard of the Apple computer is protected against scratches.

- Insert the four **IC**'s (U1-U4), make sure the notch of the IC (first-pin side) matches the notch-side of the socket (and the outline of the board). It is handy to use an IC straightener tool to bend the legs of the IC correctly before inserting them.



Mounting of the standoffs with caps and washers. Keep JP2 clear when affixing the felt.

- **Important last step:** affix the **adhesive felt** on top of the marked area at the backside of the PCB. *See* figure. Make sure to push it well but gently to minimize the height of the felt. This is to make sure the soldered pins do not touch the computer and will cause a short circuit. Keep solder jumper JP2 clear, this makes it possible to solder it when needed in the future (*see* page 10).

First Power Up

If at hand, it is advised to use the KST on top of an ASCII Keyboard Tester [6] first, before using it with an actual computer. The KST receives power from the computer (or tester), so no extra power supply is needed.

Follow these steps:

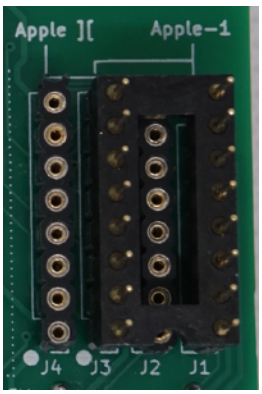
1. Make sure the correct serial cable is present, compatibility with the vintage computer is checked and all assembly steps have been followed closely. *See* all corresponding previous chapters. Before turning the power on, or connecting anything to the KST, make sure the board was tested for shorts using a multimeter (*read* the Assembly chapter).

2. Use the supplied 16-pin male-male DIP IC adapter to 'select' Apple-1 or Apple][usage. The thicker-and-shorter-pin side of the adapter go into J1-J4 of the board, the thinner-and-

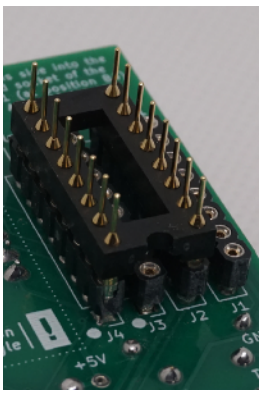
longer-pin side goes into the keyboard socket of the vintage computer. *See* the figures at the right.

3. Then connect the serial cable to J9 of the KST and secure its screws, leave the other end of the serial cable unconnected.

   Important: it is not recommended to disconnect (or connect) the serial cable to the KST when it sits on top of the Apple computer. Doing this can cause damage to the keyboard socket of the vintage computer because it is stressed by the wiggling of the KST.



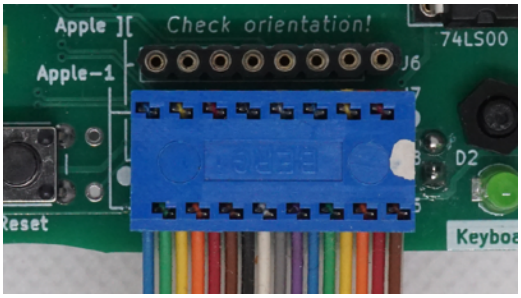
*Position of adapter for **Apple-1** usage.*



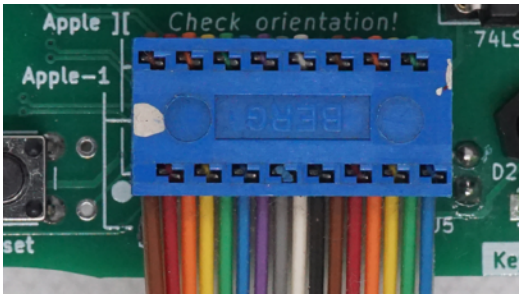
*Adapter positioned for the **Apple][**.*

4. Connect an Apple-1 or Apple][keyboard to the *Passthrough*-connector to be able to use the regular keyboard of the Apple computer. Apple][keyboards can be used with an Apple-1 computer, or vice versa. Please note that +12V is not available on the Apple][.

   Important: make sure to double check the orientation of the ribbon cable of the keyboard. Connect Apple-1 keyboards to J5 + J7 with the cable marking (pin 1, white dot on the board) to the right side. Apple][keyboards connect to J6 + J8 with the marking on the left side. This improves the cable handling for both keyboards. *See* the pictures below.



*An **Apple-1** keyboard connected with the pin-1 marking pointing to the right (J5 + J7).*



***Apple][** keyboards connect the other way around, pin-1 pointing to the left (J6 + J8).*

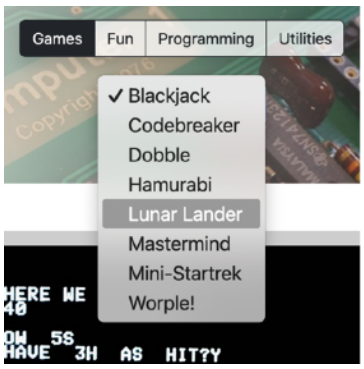
5. When all cables are connected, the KST can be mounted on top of the keyboard socket of the Apple computer. **Important:** make sure the pins of the male-male adapter are not misaligned with the keyboard socket on the mainboard, all pins should go in. Doing this wrong will damage the KST and / or the computer.

6. Turn on the Apple computer. At boot up of the KST, notice the **green** LEDs moving quickly from *Keyboard* to *Terminal*. In the meantime the **red** *Receive Error* LED (bottom side of the board) will flash for 1 second. This is normal behavior and tests if all the LEDs work. Also check if the lamp (or LED) of the regular Apple keyboard lights up and stays on, this means it is getting power correctly as well.

   - Press the *Toggle Input* button to switch between *Keyboard* and *Terminal* (serial) usage. The **green** LED indicates the active input.
   - Use the *Clear Screen* button to clear the screen of an Apple-1. Also handy when an Apple][keyboard is connected, as these do not have this button.
   - Use the *Reset* button to reset the Apple-1 or Apple][.

7. Finally connect the serial (or USB) cable to the modern computer and proceed with the rest of this chapter.

Quick Start: use Apple1Software.com



Picking a software package on Apple1Software.com.

This website makes it easy to transfer software to an Apple-1. Just choose a category, pick a software package and hit the *Serial-* button below the screenshot. It is meant to be self-explanatory and further options are explained on the website itself.

Apple1Software.com uses the Web Serial API [7,8] to connect the browser directly to the KST using a serial port. This operates locally (client side), i.e. the browser does not allow the server side to connect to the serial port. Currently this API is supported by Google Chrome and Microsoft Edge browsers only.

Another, more low-level alternative to experiment with, is the website WebSerial.io by William Kapke [9], this is a browser based terminal application. This website does not support to set character and/or newline delays, so direct copy pasting of software is not supported.

Use of a Terminal

As another option use a terminal application that supports serial connectivity. Choose a terminal application that supports setting a *character delay* and a *newline delay*, these are important to be able to copy paste software. For example **Minicom** [10, 11] or **Tera Term** [12] are terminal applications that support the needed settings.

For Apple][users, please note that the KST is not meant to be an alternative for **ADTPro** [13]. The KST uses the keyboard socket for communication, while ADTPro uses the cassette interface, or a serial interface inserted into an expansion slot. The KST does not support the loading of disk images, use ADTPro for this. However, the KST can be used to transfer / auto-type bytecode (using the System Monitor) or BASIC code. The ADTPro program can be started and controlled remotely on an Apple][by using the KST.

Configuration

In the terminal application, make the following settings:

- The baudrate depends on the setting of jumper JP1 (backside of the board). Leaving this jumper OFF sets the baudrate to **57.6 kbps** (default). Put the jumper ON for a slower baudrate of **9200 bps**.

  Important: change this jumper only while the vintage computer and the KST are turned off.

- Use **7 data bits**, **even parity** and **2 stop bits** (7E2) as the serial frame configuration.
- A minimum character delay of **20 ms** and a newline delay of **200 ms** has to be used when fragments of code are copy-pasted into the terminal. This is to give the vintage computer some extra processing time.

As an example, follow these steps to use Minicom on a Mac / Linux environment:

- Install *Brew* and install *Minicom*, watch this video [14] or search the web for a tutorial.
- After inserting the USB to serial cable (*see* Prerequisites) use the command *ls /dev/tty.usb** to list the name of the USB device. Take note of this, e.g. */dev/tty.usbserial-A4013JA5*.

  For some cables it is needed to install driver software first. This can be the cause of the device not appearing in the device list here. See the documentation or software that comes with the cable on how to install it properly.

- Enter the main setup of Minicom with *minicom -s*. *See* the figures below.

```
+-----[configuration]------+
| Filenames and paths      |
| File transfer protocols  |
| Serial port setup        |
| Modem and dialing        |
| Screen                   |
| Keyboard and Misc        |
| Save setup as dfl        |
| Save setup as..          |
| Exit                     |
| Exit from Minicom        |
+--------------------------+
```

Minicom main setup menu.

```
+-----------------------------------------------------------------------+
| A -    Serial Device      : /dev/tty.usbserial-A4013JA5               |
| B - Lockfile Location     : /usr/local/Cellar/minicom/2.9/var         |
| C -   Callin Program      :                                           |
| D -  Callout Program      :                                           |
| E -    Bps/Par/Bits       : 57600 7E2                                 |
| F - Hardware Flow Control : No                                        |
| G - Software Flow Control : No                                        |
| H -     RS485 Enable      : No                                        |
| I -   RS485 Rts On Send   : No                                        |
| J -  RS485 Rts After Send : No                                        |
| K -  RS485 Rx During Tx   : No                                        |
| L -  RS485 Terminate Bus  : No                                        |
| M - RS485 Delay Rts Before: 0                                         |
| N - RS485 Delay Rts After : 0                                         |
|                                                                       |
|    Change which setting?                                              |
+-----------------------------------------------------------------------+
```

*Minicom **Serial port setup**: option A and E should be configured.*

- Choose the *Serial port setup* from the menu. Press *A* and enter the serial device noted earlier. Press *E* to change the baudrate settings (*see* previous page). Note that if jumper JP1 is ON, a lower baudrate has to be used. Press *return* to go back to the main menu.

- Pick *Screen* from the main menu. Make sure *Add linefeed* and *Local echo* are both set to No. *See* figure at the right. Press *return* to go back to the main menu.

```
P - Add linefeed          : No
Q - Local echo            : No
```

*Minicom **Screen** settings, set options P and Q.*

- Choose *Save setup as dfl* to save the settings ("dfl" means default).

- Then choose *Exit from Minicom*. The configuration of Minicom is now done.

Minicom can be started with the *minicom* command and it connects with the serial port right away. Test the connection by entering something in the terminal, it should appear on the vintage computer screen. Make sure the input is toggled to *Terminal* on the KST board. *See* figure at the side.

Input set to Terminal.

The delays have to be set for each terminal session and are not part of the default configuration:

- In the main terminal window of Minicom *ESC + Z* brings up another settings window.

- Press T to change the *Terminal settings*. Use option D and F to change the minimum delays. Check if option A is set to *VT102* (for proper arrow key handling). *See* the figure below.

```
Welcome to minicom 2.9

OPTIONS: I18n
Compiled on Apr  3 2024, 13:16:24.
Port /dev/tty.usbserial-A4013JA5, 10:52:21
                +---------------[Terminal settings]----------------+
Press Meta-Z f|                                                    |
              | A -      Terminal emulation : VT102                |
              | B -     Backspace key sends : BS                   |
              | C -           Status line is : enabled              |
              | D -   Newline tx delay (ms) : 200                  |
              | E -           ENQ answerback : Minicom2.9           |
              | F - Character tx delay (ms) : 20                   |
              |    Change which setting?                           |
              |                                                    |
              +----------------------------------------------------+
```

Setting the terminal emulation, character and newline delays in Minicom.

- To exit Minicom press *ESC + X*.

Note some **caveats** using Minicom (and possibly other terminal applications):

- Make sure the window of the terminal application has focus when typing something. To give focus to the terminal window, just click in the (black) empty area of the window.
- By default, characters that are typed in the terminal window will appear on the screen of the vintage computer only and do not appear in the terminal itself. This is because the echo'ing of characters interferes with high speed TurboType™ transfers and it is easier to see receive error notifications, which are always send to the terminal when they occur. More on this in later chapters.
- First exit Minicom (with *ESC* + *X*) before powering off the vintage computer (and KST) and before disconnecting the serial (or USB) cable from the modern computer. Otherwise Minicom will sometimes become unresponsive and is unable to reconnect.
- Insert the USB cable preferably in the same USB port at the time when Minicom was configured, otherwise the operating system will use a difference device path and the *Serial Device* (option *A*) of Minicom settings will have to reflect this.
- When waiting for a serial transfer after copy-pasting software, stay in Minicom and do not change focus to another screen / program. This sometimes causes errors.

Basic Interaction

Toggle input to *Terminal* on the KST. The display of characters in the terminal window is affected by the *Serial Echo* setting (*Ctrl* + *E*, *see* next chapter) and the *Local echo* setting of the terminal application itself.

Hex machine code **programs** or **data** can be typed in or copy-pasted as commands to a System Monitor. After a full reset (*Ctrl* + *Q*), the Apple-1 loads this monitor (or Wozmon) automatically. When in BASIC on the Apple][, use *Ctrl* + *S* to enter the System Monitor.

To transfer **BASIC** source code, make sure BASIC is loaded first. Thereafter, use *Ctrl* + *B, 1* to start it on the Apple-1. Take into account longer processing times by the Apple, so character and line delays can be increased (or doubled) when characters are skipped.

Try out the example programs on this page, on Apple1Software.com or search the web. After a program is transferred, press the *Toggle Input* on the KST to enjoy the program, or play the game, with the regular (vintage) keyboard.

```
0
:A9 00 AA 20 EF FF E8 8A
:4C 02 00
0R
```

Wozmon listing of the Apple-1 test program.

```
05 REM Fibonacci numbers
10 LET M = 5000
20 LET X = 1 : LET Y = 1
30 IF (X>M) THEN GOTO 100
40 PRINT X
50 X = X + Y
60 IF (Y>M) THEN GOTO 100
70 PRINT Y
80 Y = X + Y
90 GOTO 30
100 END
```

Example BASIC source code.

```
10 X = PEEK(49200):PRINT "HELICOPTER"
20 GOTO 10
```

Example BASIC program to make some sounds on an Apple][.

Keep **CAPS LOCK on** when using the KST with an Apple-1. Lowercase characters seem to work as they will appear as uppercase characters. However, they are not recognized correctly by the Apple-1 monitor.

The use of the **BACKSPACE** key will be put through as either an underscore (_), or as a regular backspace control character, depending on the *Arrow Keys* setting of the KST (*see* next chapter). This makes proper backspace handling possible on both the Apple-1 and Apple][.

Control Characters

The table below lists the actions that can be performed within a terminal program on the modern computer. The input of the board has to be set to *Terminal* (default at power on).

These control characters can also be used to orchestrate automatic demonstrations of an Apple-1 or Apple][. Use the macro [11] or script [15] functionality of the terminal program. Handy for museums or public events.

Key	Action	Behavior	Apple 1	Apple][
Ctrl + **E**	Serial **E**cho toggle on / off [1]	Toggled on: any non-control character is transmitted / echoed back on the serial line to confirm a receive. Make sure local echo of the terminal program is turned off.	✓	✓
Ctrl + **R**	**R**eset [2]	Resets the vintage computer (low going, *1.25ms* signal).	✓	✓
Ctrl + **C**	**C**lear Screen	Clears the screen of the vintage computer (high, *40ms*).	✓	
Ctrl + **Q**	Reset Se**q**uence [2]	Full reset sequence of the vintage computer. Holds both reset and clear (*40ms*), releases clear first and then releases reset (after *5ms*).	✓	
Ctrl + **K**	Switch to **K**eyboard [3]	Switches the input to the regular vintage computer keyboard, attached to the passthrough connector.	✓	✓
Ctrl + **P**	Auto **P**ower-On Reset toggle on / off [1]	Toggled on: the vintage computer is automatically reset (by a full sequence) at power-on of the KST.	✓	✓
Ctrl + **X**	E**x**tend Strobe signal toggle on / off [1]	Toggled on: the strobe signal is hold longer (*1ms* instead of *5µs*) when data is set on the output. Also the character and line delays used by the KST auto type features are doubled. Can be used to fix compatibility issues with other vintage computers that take parallel ASCII data as input.	✓	✓
Ctrl + **A**	**A**rrow Keys toggle on / off [1]	Toggled on: pressing the arrow keys of the modern computer in the terminal will generate the appropriate cursor moving escape codes for the Apple][. A regular backspace control character is used for the backspace key. Toggled off: the use of the backspace key will be put through as an underscore (_) for Apple-1 usage.		✓
Ctrl + **B**, **1** [4]	Start **B**ASIC	Auto types “E000R” followed by *return* to start BASIC.	✓	
Ctrl + **B**, **2** [4]	Return to **B**ASIC	Auto types “E2B3R” followed by *return* to return to BASIC.	✓	
Ctrl + **B**, **3** [4]	Run (within **B**ASIC)	Auto types “RUN” followed by *return* to run a program from within BASIC.	✓	✓
Ctrl + **L**	Auto type Turbo **L**oader for TurboType™	Auto types and runs the *Stack Sanitizer* and *Turbo Loader* programs for a TurboType™ transfer. *See* the next chapter.	✓	

TABLE CONTINUES ON THE NEXT PAGE

Table footnotes:

1) Toggles are visually confirmed by the **red** led: a **single flash** means the function is toggled on, **two flashes** mean it is turned off. This setting is remembered across power cycles.
2) Any previous error condition is reset after this action, i.e. the **red** led is always turned off.
3) The **green** led indicator switches from *Terminal* to *Keyboard*.
4) After the *Ctrl + B* character, the KST waits for input of a second character (1, 2 or 3).

Key	Action	Behavior	Apple 1	Apple][
Ctrl + **T**	Auto type **T**est Program	Auto types and runs the Test Program from the Apple-1 Operation Manual [16].	✓	
Ctrl + **S**	Enter **S**ystem Monitor	Auto types “CALL -151” followed by *return* to enter the System Monitor.		✓
ESC or Ctrl + **[**	**Escape** KST control character handling	Send a control character to the vintage computer without changing settings of the KST. To use, press and release the *ESC* key. Thereafter send the control character for the Apple][. For example *Ctrl + G* to ring the ‘bell’ or *Ctrl + C* to break execution of a BASIC program.		✓
Ctrl + **F**	Show **F**irmware Version of the KST	The version of the active firmware is transmitted to the terminal screen on the modern computer, like “v1.0”.	✓	✓

Go Fast with TurboType™

With TurboType™ higher transfer speeds can be obtained on an Apple-1. The method skips the display of the transferred characters on the screen of the Apple-1 and so it saves cpu cycles. It is part of the **Apple-1 Toolkit**, soon to be released by Uncle Bernie.

Visit Apple1Software.com and see how it is built up. TurboType™ Wozmon listings can be transferred within the browser or can be downloaded for offline use.

First, the stack is sanitized with a small program running at memory location *0003*. Second, it runs the Turbo Loader bootstrap program at location *0100*, the Apple-1 then waits for the actual data. The data itself is marked with *T* and *X* characters, *see* the example on the side.

This *T-to-X* fragment consists of the ‘compressed’ main program and a CRC checker running at memory location *015E*. In case of a CRC failure it displays *EE*, otherwise it runs the main program that was transferred.

```
T
E000
:4CB0E2AD11D010FBAD10D0608A2920F0
:23A9A085E44CC9E3A920C524B00CA98D
…
:4D4CADE5D578D001184C02E120B7E54C
:36E820B7E54C5BE8E080D001884C0CE0
015E
:A0FF84268427C88424A000A9DF8525E6
:25A525C9F09020C000901CA98D20EFFF
…
:0CA52649108526A52749088527262626
:27680ACAD0E5C8F0B6C000D0DAF0B2
X
```

T-to-X fragment of a TurboType™ Wozmon listing.

To use **TurboType™ within a terminal**:

1. Make sure *Serial Echo* is turned off using **Ctrl + E** and all delays in the terminal application are set to zero (0), so full speeds can be obtained.
2. Use **Ctrl + L** to let the KST auto type and run the *Stack Sanitizer* and *Turbo Loader*. Notice that the KST is still using proper character and line delays when typing. When it finishes, the Apple-1 seems to hang, but in fact it waits for the T-to-X fragment.
3. The complete fragment including the beginning T and ending X can now be copy pasted into the terminal window. Note that no characters will appear on the screen of the Apple.
4. After the X is received by the Apple-1, the KST auto types “015ER” to run the CRC check of the transfer and after that run the program itself. In case of a CRC error, the characters “EE” are displayed and the program is not started.

Receive Error LED

This error LED is mounted on the backside / underside of the KST board. This is done to keep the board compact. Its **red** glow reflects on the mainboard of the Apple computer when it emits light.

It is being used in the following cases:

- At power-on of the KST the LED turns on for 1 second, to check if it works.
- When toggling a setting (by use of control characters) a confirmation is shown by this led: a **single flash** means the function is toggled on, **two flashes** mean it is turned off.
- The LED turns on and stays on when one of the following **receive errors** [17] are encountered during serial communication (transferring programs with Apple1Software.com or using a terminal application):

  ***F**rame Error*; means the first stop bit was incorrect / unexpected at the time the next readable serial frame is being processed. This is an out-of-sync condition. A character 'F' is transmitted over serial to the modern computer, independent of the current *Serial Echo* setting of the KST.

  ***D**ata OverRun*; indicates loss of data due to the receive buffer of the KST being full. One or more serial frames were lost. A character 'D' is transmitted to the modern computer.

  ***P**arity Error*; the next serial frame to be processed has a parity error. This means that the calculated parity bit of the data bits is not equal to the parity bit that was transmitted in the serial frame. A character 'P' is transmitted.

A *reset* of the vintage computer (by use of a control character) or a toggle of input on the KST clears the receive error condition and the LED will go off.

Problem Solving Guide

Problem	Cause	Remedy
Unable to select the connected USB to serial cable or it is not listed in the *dev*-directory on Linux / macOS.	The cable needs drivers or software to be installed.	See the documentation of the cable manufacturer to install it properly.
The serial interface does not seem to work at all.	The input on the KST is set to *Keyboard*. Some cheesy RS232C interfaces try to 'steal' the negative voltage line from the device they connect to in order to meet the serial communication standard.	Toggle the input selection to *Terminal*. Try a better cable [2] , or try to close solder jumper **JP2** (back side of the KST) with a solder blob.
When programs are copy-pasted in the terminal window, it seems that not all characters are shown by the Apple.	It could be the case the transfer is too quick for the vintage computer to process.	Make sure the proper **delays** are set (*see* page 6) in the terminal configuration. Or try the lower **baudrate** speed (*see* page 5). For TurboType™ listings it is normal that the characters in the *T-to-X* fragment are not displayed by the Apple.
When the *Serial Echo* is on, wrong characters appear in the terminal window when typing.	Incorrect serial communication settings are used.	Check the settings of the terminal application (*see* pages 5-6), also verify the setting of jumper JP1.

Acknowledgements

This work is inspired by **Mike Willegal**'s *ASCII to PS/2 Keyboard Adapter*. Many thanks for his approval to use his design, the clear documentation on his website [18] and providing a detailed assembly and operations guide [19]. Make sure to check out his website: https://www.willegal.net/.

Thanks to **Uncle Bernie** for his coaching during development of this board and his provision of his TurboType™ algorithm.

DISCLAIMER

The use of our products and schematics is at your own risk. **We expect our audience and buyers to have experience with electronics and electricity.** The product images shown on the website are for illustration purposes only and may not be an exact representation of the actual product, which is subject to enhancement. Colors (of parts) can differ because of changes in supply.

In no event shall 8BitFlux.com be liable in warranty, contract, negligence, strict liability, tort or otherwise, for any damages, whether incidental or consequential, which are alleged to be caused by one or more of our products, beyond the cost to the purchaser of the subject product or products in question.

REFERENCE LIST

1. Startech. *2m Black DB9 RS232 Serial Null Modem Cable F/F*. Retrieved June 11, 2023, from https://www.startech.com/en-us/cables/scnm9ff2mbk
2. Startech. *1-Port USB to Serial RS232 Adapter*. Retrieved November 25, 2024, from https://www.startech.com/en-us/cards-adapters/icusb232ftn
3. Wikipedia. *Replica 1*. Retrieved January 9, 2025, from https://en.wikipedia.org/wiki/Replica_1
4. Otten, H. (2021). *ASCII PARALLEL KEYBOARD TO KIM-1*. Retrieved January 9, 2025, from http://retro.hansotten.nl/6502-sbc/kim-1-manuals-and-software/kim-1-articles/ascii-parallel-keyboard-to-kim-1/
5. Holden, H. (2020). *THE SOL-20 KEYBOARD'S CAPACITIVE KEY DETECT CIRCUITRY*. Retrieved January 9, 2025, from https://deramp.com/downloads/processor_technology/sol-20/hardware/Keyboard%20Repair%20Notes%20(H%20Holden).pdf
6. 8BitFlux.com. *ASCII Keyboard Tester*. Retrieved January 9, 2025, from https://8bitflux.com/#ascii-keyboard-tester.
7. MDN Web Docs. *Web Serial API*. Retrieved January 13, 2025, from https://developer.mozilla.org/en-US/docs/Web/API/Web_Serial_API
8. W3C Community Group. *Web Serial API*. Retrieved January 13, 2025, from https://wicg.github.io/serial/
9. Kapke, D. (2022). *A browser based Serial Port application*. Retrieved January 13, 2025, from https://webserial.io/
10. Wikipedia. *Minicom*. Retrieved January 13, 2025, from https://en.wikipedia.org/wiki/Minicom
11. Linux General Commands Manual. *minicom - friendly serial communication program*. Retrieved January 19, 2025, from https://man7.org/linux/man-pages/man1/minicom.1.html
12. Wikipedia. *Tera Term*. Retrieved January 13, 2025, from https://en.wikipedia.org/wiki/Tera_Term
13. ADTPro. *Apple Disk Transfer ProDOS*. Retrieved January 18, 2025, from https://adtpro.com/
14. YouTube. *How to install minicom on mac and connect to mbed*. Retrieved January 13, 2025, from https://www.youtube.com/watch?v=Y3DKVpXEUnc
15. Linux General Commands Manual. *runscript - script interpreter for minicom*. Retrieved January 19, 2025, from https://man7.org/linux/man-pages/man1/runscript.1.html
16. Apple Computer Company (1976). *Apple-1 Operation Manual,* p. 2. Retrieved January 15, 2025, from https://apple1software.com/downloads/Operation_Manual.pdf
17. Atmel. *ATtiny2313/V Datasheet*, p. 123. Retrieved January 20, 2025, from https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2543-AVR-ATtiny2313_Datasheet.pdf
18. Willegal, M. (2008). *ASCII to PS/2 Keyboard Adapter*. Retrieved June 11, 2024, from http://www.willegal.net/appleii/appleii-kb-int.htm
19. Willegal, M. (2008). *PS-2/RS232 to ASCII Keyboard Adapter Assembly and Operations Guide*, version 3.0. Retrieved June 11, 2024, from https://mirrors.apple2.org.za/ftp.apple.asimov.net/documentation/hardware/io/PS-2%2CRS-232%20to%20ASCII%20Keyboard%20Adapter%20Assembly%20and%20Operations%20Guide%20v3.0.pdf

