



PAK-VI Data Sheet

© 1999 by AWC

AWC
310 Ivy Glen Court
League City, TX 77573
(281) 334-4341
<http://www.al-williams.com/awce.htm>
V1.0



Overview

The Stamp PAK VI allows you to use an ordinary PS/2 or AT-style keyboard with any computing device capable of handling RS232 communications. The PAK VI is the perfect solution for microcontrollers that need real input. It provides a raw mode for ultimate control or a cooked mode that makes interfacing with the keyboard simple.

Here are some of the PAK V's major features:

- Serial interface uses TTL level RS232 at 9600
- Uses as few as 1 pin to connect to the host
- Buffers up to 16 keystrokes or scan codes
- Easy to use

The PAK VI is a standard 18-pin IC. In order to operate it must have a regulated supply of 5V and connection to a clock element. The Stamp PAK VI includes a 50MHz ceramic resonator that you can use to clock the chip.

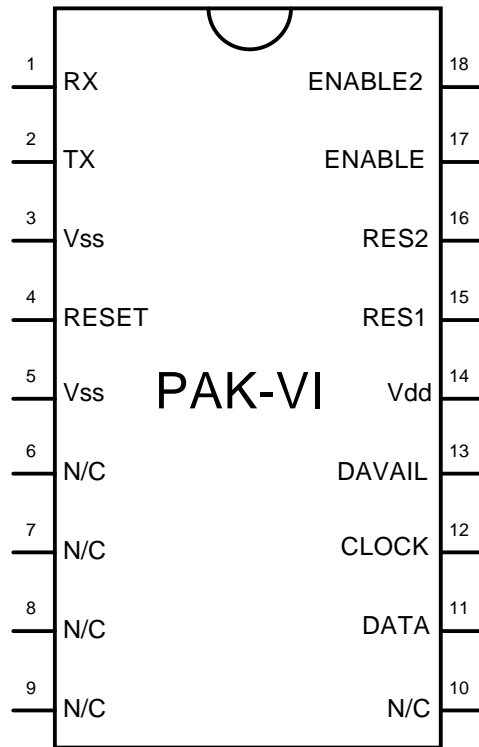
Other than the power and clock connections, the Pak VI requires at least one wire to connect between your microcontroller (the host) and itself. Depending on the host's capabilities and your application, you may want to use a few additional pins to control the PAK

If You Need Help

If you require assistance with your PAK VI, please feel free to contact us. The best way to get support is via e-mail (stamp@al-williams.com). However, you may also call between 9AM - 4PM Central Time at (281) 334-4341. You can also fax to (281) 754-4462. Be sure to check out our Web page for updates at www.al-williams.com/awce.htm.

Registering Your PAK-VI

Please take a moment to register your e-mail address with AWC. Simply send an e-mail message to pakreg@al-williams.com. AWC will not make your address available to other companies, but we may periodically send you updated technical notes. You'll also receive information about new microcontroller products and specials.



WARNING: The PAK VI is a static-sensitive, CMOS device. Observe static precautions when handling. Operating the device without both Vss pins grounded or with RES1 or RES2 disconnected may damage the chip.

Pin Connections

Pin	Name	Type	Description
1	RX	Input	TTL-level RS232 input
2	TX	Output	TTL-level RS232 output
17	Enable	Input	If this pin is not connected or high, the PAK transmits codes it receives from the keyboard; the PAK always responds to commands
18	Enable2	Input	If this pin is low, the PAK transmits codes it receives from the keyboard; the PAK always responds to commands
13	DAvail	Output	High when data is available
4	RESET	Input	Hardware resets the PAK when low. Must be high for normal operation
3,5	Vss	Power	Ground (please ground both pins)
14	Vdd	Power	+5V
15	RES1	Clock	Connects to resonator
16	RES2	Clock	Connects to resonator
11	DATA	I/O	Keyboard data line
12	CLOCK	I/O	Keyboard clock line
6-10	N/C	N/C	Not used

Software Commands

Code	Name	Description
\$01	COOK	Sets cooked mode (default)
\$02	RAW	Sets raw mode
\$03	SLOFF	Turns off scroll lock LED
\$04	SLON	Turns on scroll lock LED
\$05	NLOFF	Turns off num lock LED
\$06	NLON	Turns on num lock LED
\$07	CLOFF	Turns off cap lock LED
\$08	CLON	Turns on cap lock LED
\$09	AION	Turns on auto indicator mode (cooked mode only; default)
\$0A	AIOFF	Turns off auto indicator mode
\$0B	ESCAPE	Sends next byte directly to keyboard
\$FF	RESET	Warm reset PAK and reset keyboard

Operating Modes

By default the PAK VI starts in cooked mode. Keys you press generate their ASCII equivalent on the serial output. The PAK manages the indicator lights. Most non-ASCII characters generate a special code greater than or equal to \$80.

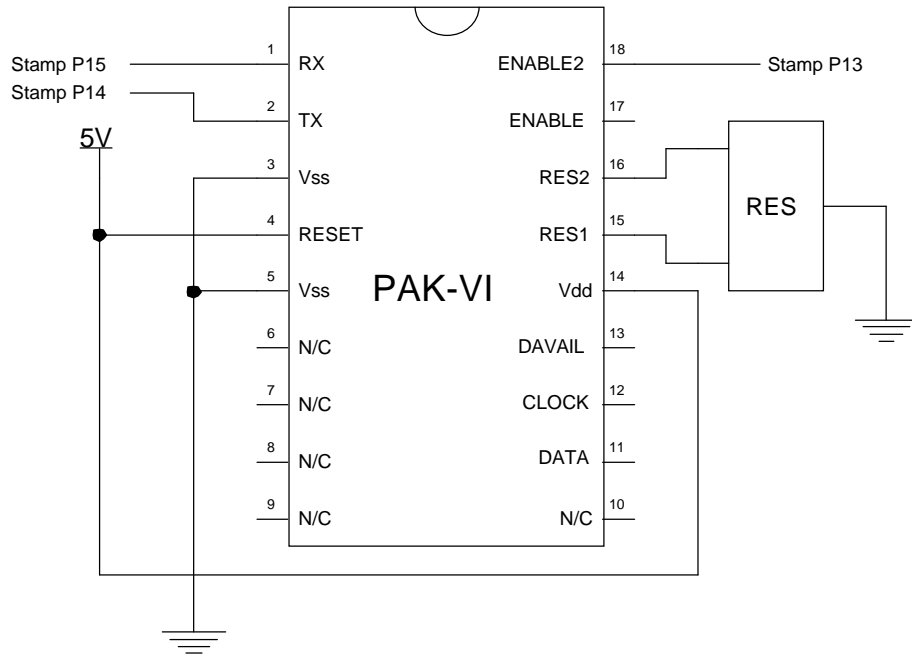
If you prefer, you can disable the autoindicator mode and take control of the keyboard LEDs in your software by issuing the AIOFF command.

You can also receive raw scan codes instead of ASCII codes by issuing the RAW command. In this mode you'll receive make and break codes for each key pressed.

Typical Circuits and Software

Connecting the Pak to its external components is simple. Supply 5V to Vdd and ground the Vss pins. Connect the two outer pins of the ceramic resonator to the RES1 and RES2 pins (the order does not matter). Ground the center pin of the resonator. Connect the RESET pin to 5V. For the simplest mode of operation leave ENABLE open and jumper ENABLE2 to ground. Of course, you'll also need to connect the RX and TX pin to the host microprocessor. If the host uses true RS232, you'll need a line driver (like the Maxim 232 chip). For connection to a TTL inverted device (like the Basic Stamp) you can directly connect the pins.

If you are not planning to send the PAK any commands, you can omit the RX line. If you'd like



an indication of when the PAK has data available, you can monitor the DAVAIL pin which is high when data is available.

If you are using the Basic Stamp you'll have no problems using the SEROUT command to send codes to the PAK. Many Basic Stamp-compatible compilers have these commands too.

Operating Modes

There are two major modes that you can use with the PAK-VI. The default is cooked mode. This provides the keyboard input as ASCII. Special characters (like function keys) appear as special characters above \$7F.

If you want more control, you can switch to raw mode. Raw mode sends you every byte the keyboard generates. There is no ASCII conversion, nor does the PAK-VI automatically track the shift, caps lock, or other state keys. This mode provides the ultimate flexibility, but requires more work to use.

In raw mode, you may get several bytes for each key pressed. Most keyboards start in mode 2 (although you can change the mode on many keyboards). In mode 2, each key emits at least one scan code for a make (key press) and at least two codes for a break. For example, the space bar emits \$29 for make, and \$F0 \$29 for break. Some keys, particularly extended keys, have more codes. For example, the home key that is *not* part of the numeric key pad is scan code \$E0 \$6C and the break code is \$E0 \$F0 \$6C.

Important Note

Not all keyboards are the same. Raw mode will work with any standard keyboard, but your keyboard may not support all features that other keyboards support. By the same token, your

keyboard may support more features than other keyboards. Check the keyboard's documentation

Keyboard Commands

Most keyboards allow you to send a variety of commands:

- \$FF - Reset keyboard
- \$FE - Resend last byte
- \$F5 - Disable keyboard
- \$F4 - Enable keyboard
- \$F3 - Set typematic rate
- \$F2 - Return ID
- \$F0 00 - Return current code set
- \$F0 01 - Set code set 1 (XT compatible)
- \$F0 02 - Set code set 2 (AT compatible)
- \$F0 03 - Set code set 3 (make codes only)
- \$EE - Echos \$EE back
- \$ED - Sets indicator LEDs

Programming the keyboard is beyond the scope of this manual. However, most commands return an acknowledge byte (\$FA). You can find more about raw keyboard programming on the Web (see the end of this manual).

Note that you should not use \$ED to set the indicator LEDs in cooked mode because it will confuse the PAK. Instead, use the PAK commands to set the indicators.

Before sending commands directly to the keyboard, you should be in raw mode. In addition, you should prefix each byte with an escape command. If you want to send 3 bytes to the keyboard, you must prefix each byte with escape.

Hints

If you are using a Stamp or other microcontroller, you may want to switch character sets when using raw mode. Use the \$F0 command to change sets. Set #1 uses a single-byte scan code for each key. Bit 7 indicates if the code is for make (0) or break (1). This simplifies coding for the Stamp and other similar processors.

PS/2 keyboards require 4 wires: two for power and ground and two for clock and data. For prototyping, you may find it convenient to acquire a PS/2 connector designed to connect PS/2 keyboards to PC motherboards (you can find these at any place that sells PC components for a few dollars). These PS/2 connectors terminate in a .1" single inline pin socket. You can either cut the wires to connect to the PAK, or use some .1" single inline pins to make the connection.

Special Codes

In cooked mode, the PAK sends characters above \$7F for many special keys:

\$80 - F1	\$90 - Home
\$81 - F2	\$91 - Up
\$82 - F3	\$92 - PgUp
\$83 - F4	\$93 - Left
\$84 - F5	\$94 - Numeric 5
\$85 - F6	\$95 - Right
\$86 - F7	\$96 - End
\$87 - F8	\$97 - Down
\$88 - F9	\$98 - PgDn
\$89 - F10	\$99 - Ins
\$8A - F11	\$9A - Del
\$8B - F12	

Notes:

If shift key is down, add \$40 to above values

If num lock is on, \$90-\$9A appear as numbers

Frequently Asked Questions

Q: Can I run the Pak VI from a different clock source?

A: You may be able to use a lower clock speed, but it isn't recommended. The PWM and serial communications depend on specific timing.

If you do run at a different clock speed, you can expect the baud rates to scale accordingly. So using a 25MHz clock, for example, will cause the PAK to communicate at 4800 and 1200 baud. AWC does not support operation at any speed other than 50MHz.

Q: Is there a second source for the PAK VI?

A: No. However, if you have a high-volume application and you are concerned about availability, contact AWC about obtaining a license to produce your own PAKs or obtain them from third parties.

Q: What is the state of the PAK at startup?

A: Initially the PAK resets the keyboard, forces keyboard mode 2 (which should be the default) and is in cooked mode with autoindicator mode set.

Further Information

To learn more about programming PC keyboards in raw mode, you can refer to several resources on the World Wide Web:

[http://members.tripod.com/~ilkerf/c64tower/
F_Keyboard_FAQ.html](http://members.tripod.com/~ilkerf/c64tower/F_Keyboard_FAQ.html)

[http://www.geocities.com/SiliconValley/
Bay/8302/keybrd.htm](http://www.geocities.com/SiliconValley/Bay/8302/keybrd.htm)

Specifications

Ambient temperature under bias	-40°C to +85°C
Storage temperature	-65°C to +150°C
Voltage on VDD with respect to VSS	0 to +7.5V
Maximum current out of VSS pin	100 mA
Maximum current into VDD pin	100 mA
Maximum output current sunk by any I/O pin	45 mA
Maximum output current sourced by any I/O pin	45 mA

Parameter	Minimum	Typical	Maximum
Supply voltage	2.5V	5V	5.5V
Vdd rise time on power up	.05V/ms	-	-
Supply current @ 5V	-	60mA	