AWC

PAK-VIIa/b Pulse Coprocessor Data Sheet

AWC
1279 FM 518 Rd #2
Kemah, TX 77564
(281) 334-4341
http://www.al-williams.com/awce.htm
V1.4 10 Aug 2004

Table of Contents

Overview

The PAK-VIIb is an 8 channel pulse input coprocessor. It can perform a variety of functions:

- Measure pulse widths in 5uS intervals.
- Count the number of pulses that occur on each channel.
- Measure the period of a pulse train.
- Measure time in 1 second or 200mS intervals.
- Provide the static state of each input pin on demand.
- Read quadrature inputs
- Robust, speed-independent serial interface.
- Works with Basic Stamp's SHIFTIN and SHIFTOUT.
- Synchronous - read results when you are ready for them.
- Easy to use.

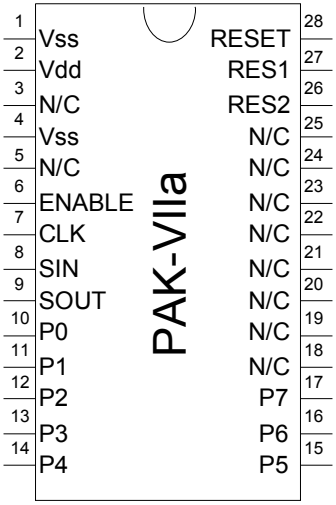
Like all PAKs, the PAK-VIIb is simple to connect to a Stamp or any microcontroller. If your microcontroller can switch pins between input and output status, you can connect a single PAK with as few as 2 I/O lines. However, the PAK will allow you to use separate I/O pins (3 lines) if necessary. You can also connect multiple PAKs together using the same two or three lines if you provide an additional enable line for each PAK.

The PAK-VIIb is a standard 28-pin IC. In order to operate, it must have a regulated supply of 5V and connection to a clock element. The PAK-VIIb includes a 50MHz ceramic resonator that you can

use to clock the chip. If you need more accuracy, a crystal or external oscillator (up to 75MHz) may be used.

If You Need Help

If you require assistance with your PAK VIIb, please feel free to contact us. For best support, 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.



WARNING: The PAK VIIb 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
8	SIN	Input	Data Input
9	SOUT	Open Collector Output	Data Output
6	Enable	Input	If this pin is not connected or high, the PAK is active. Otherwise, the PAK does not drive the SOUT line, nor does it respond to commands.
7	CLK	Input	Used to clock data to and from the PAK.
28	RESET	Input	Hardware resets the PAK when low. Must be high for normal operation
1,4	Vss	Power	Ground (please ground both pins)
2	Vdd	Power	+5V
27	RES1	Clock	Connects to resonator
26	RES2	Clock	Connects to resonator
10 to 17	P0-P7	Inputs	Pulse input channels (each input has an optional built-in weak pull up resistor).
18 to 25	Q0-Q7	Inputs	Enable inputs for each channel (these inputs have Schmitt triggers and weak pull up resistors)

Command Format

The PAK-VII uses a very simple command byte. Each channel (P0 to P7) has 6 associated registers. To read a register you form a single byte with the following bits:

7	6	5	4	3	2	1	0
Ca	Cr	R2	R1	R0	C2	C1	C0

Where:

- Ca is 1 if you want to clear all registers for this channel after reading the register. (Cr must be 0).
- Cr is 1 if you want to clear this register after reading it. (Ca must be 0).
- R2-R0 is the register number from the table below.
- C2-C0 is the channel number in binary (0=000, 1=001, 2=010, 3=011, 4=100, 5=101, 6=110, 7=111).

Here are the available registers:

Register number	Name	Function	Units
000	DURLOW	Duration of last logic low pulse	5uS
001	DURHIGH	Duration of last logic high pulse	5uS
010	RAWLOW	Current count for low pulse	5uS
011	RAWHIGH	Current count for high pulse	5uS
100	RISE	Number of rising edges detected	
101	FALL	Number of falling edges detected	
110		Special registers	
	TICK200	Channel 0 - Timer tick	200mS
	TICK	Channel 1 - Timer tick	1S
	VER	Channel 6 - Version # (1 for PAK-Va, 2 for PAK-Vb)	
	INP	Channel 7 - Current input status	
111	SUM	Sum of registers 000 and 001	5uS

Each time you send a register read command to the PAK, it returns two bytes. The first byte is the least significant byte of a 16-bit word. The second byte is the most significant byte. For the current input status (register 6, channel 7), the first byte is the real time state of the input pins and the second byte is image of the input pins as they were when the last 5uS sample was taken.

In addition, you can use the following special commands which do not return any data:

Command	Description
11111111	Reset everything
1100PPPP	Set prescaler (affects all timing functions)
	PPPP = 0000 = 1:2 (PAK-VIIb default)
	PPPP = 0001 = 1:4
	PPPP = 0010 = 1:8
	PPPP = 0011 = 1:16
	PPPP = 0100 = 1:32
	PPPP = 0101 = 1:64
	PPPP = 0110 = 1:128
	PPPP = 0111 = 1:256
	PPPP = 1000 = 1:1 (PAK-VIIa default)
11010000	Reads next byte and uses it to set internal weak pull up resistors. Each bit corresponds to one channel (1=no pull up ; 0=pull up). By default all pull up resistors are off.
11010001	Reads next byte and uses it to set input thresholds. Each bit corresponds to one channel (1=TTL; 0=CMOS). By default the input thresholds are TTL.
11010010	Reads next byte and uses it to set Schmitt trigger input mode. Each bit corresponds to one channel (1=normal; 0=Schmitt trigger). By default, all inputs are normal.

Note: With 1:1 prescaler, connecting more than 3 inputs may cause slight timing inaccuracies. For maximum accuracy, use a higher prescale factor or limit changing signals to 3 channels and enable internal pull ups on unused channels (or tie unused channels to ground or the supply voltage).

Resetting the PAK

There are three ways you might reset the PAK-VII. When you start, your host microcontroller might take a short time to take control of the PAK's I/O lines. That's why it is important to initially reset the PAK's communications system.

Resetting the communications system is easy. Simply pull the SIN line low and raise the CLK input. Then, while CLK is high, raise SIN. When you release the clock, the PAK will reset its communications. This does not affect any running counters or timers. It also does not reset any options (for example, pull up resistors). You can perform this reset using the FReset subroutine included with the example Basic Stamp library.

If you want to force a hard reset, which will clear all of the aforementioned items, send the PAK a $FF or use the library's FTotalReset command.

Another way to force a hard reset is to pull the RESET pin low briefly and then restore it to high. This will physically reset the PAK-VII, destroying any settings, counts, or intervals in progress. Notice that most of the time this is not necessary so there is no need to connect the RESET pin to the host microcontroller. Simply connect it to +5V.

Registers

Each channel has 6 identical 16-bit registers. When the PAK observes a logic low on an input, it records the time using the RAWLOW register (010). Each count in this register represents 5uS, so the register will overflow at just over 327mS. Logic highs record in the RAWHIGH register (011). When the PAK senses an edge it does several things. First, it records the edge in the RISE or FALL registers. Second, it transfers the old RAW register into the appropriate DUR register. In other words, for a low to high transition, RISE increases by 1, and DURLOW receives the value

of RAWLOW. Finally, the PAK clears RAWLOW in preparation for the next count.

You'll rarely read the RAW registers, since they can change rapidly. The DUR registers have the last completed pulse width. Remember that using the normal clock, the DUR registers will overflow on pulses over 327mS. However, the RISE and FALL registers will still be correct.

Each channel also provides a SUM register (111). This is simply the sum of DURLOW and DURHIGH. Again, this result is 16 bits, so the total of both durations must be 327mS or less. If this is not the case, you may have to read each duration separately.

You can clear a register by setting bit 6 when you request the register's contents (the Cr bit). However, you can't clear the SUM register since it is not a real register, but computed from two others.

In addition to the normal 6 registers, there is a special register (110) that provides extra functions depending on the channel you use. On channel 0, for example, this is the TICK200 channel. Here, you'll find a running count that increments every 200mS. You may clear this register when you read it by setting the Cr bit in the request. For channel 1, the special register is TICK. This is also a running timer, but runs at 1 second. You can clear TICK using Cr, also. Note that the 1 second and 200mS interval depend on the default prescaler setting. If you change the timing prescaler from the default (1:1 for PAK-VIIa; 1:2 for PAK-VIIb), the TICK and TICK200 registers will change proportionately.

Channel 7's special register is INP. This reads back two bytes. The first byte is the actual state of the input pins at the moment of command execution. The second byte contains the state of the input pins at the last 5uS sample. You can't clear this register.

You can access these registers by forming the appropriate request byte and sending it to the PAK. The example Basic Stamp library has a function, FCommand, that makes this easier. To use FCommand, you simply set ClearChan and ClearReg to 1 or 0. A 1 will cause the command to clear the entire channel, or the register respectively. You also set Register to the register number you want and Chan to the channel number. The result appears in the fpx variable.

The routines do not change the input variables, so if you set ClearReg to 0, for example, you don't have to set it again until you want to change its value no matter how many calls to FCommand you make.

Special Commands

In addition to reading registers, you may execute any of four special commands. Sending all 1's to the PAK, as mentioned before, causes a hard reset. This will terminate and reset all counting and measuring. It also resets all options including prescaling, pull ups, and thresholds.

If you need to measure pulses longer than the maximum allowed, you can reduce the speed of the PAK using a slower ceramic resonator or crystal. However, an easier solution is to engage the prescaler. Using the prescaler, you can change the sampling time from 5uS to 10uS to 1280uS in 8 steps. This changes the sample time, and also affects the TICK and TICK200 registers.

As an example, consider setting the prescaler to 1:2. This means that all pulse widths now are in units of 10uS. In addition, the TICK200 register's units will be 400uS and the TICK register's units will be 2 seconds instead of 1 second. This is the default for the PAK-VIIb. Note that when using the 1:1 prescaler

You can set the prescaler using the Basic Stamp library's FPrescale function (set fpb to the correct prescale code you want to use). The prescale command does not return any data to the host.

The other three special commands allow you to control the I/O characteristics of each channel. Each command allows you to send a byte immediately after you issue the command. Each bit in this byte represents a channel (bit 7 is channel 7; bit 0 is channel 0). Depending on the command you issue, you can set the input threshold level to CMOS or TTL (TTL is the default), enable weak pull up resistors on each input (the default is off), or set a Schmitt trigger input mode.

The CMOS threshold is 2.5V, TTL threshold is typically 1.4 to 1.5V. When using a Schmitt trigger, the input must rise to about 3.1V to read as a 1. Then it must drop to about 1.4V to read as a 0.

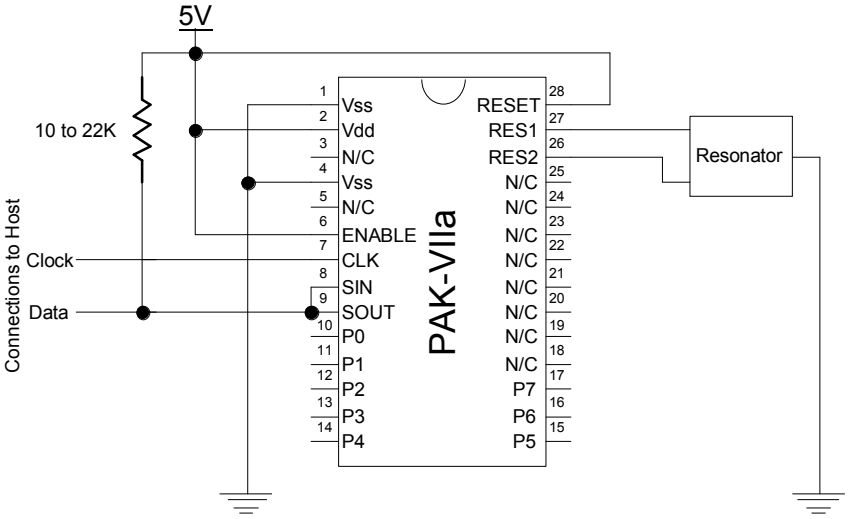
Using the Enable Inputs

The PAK-VIIb has 8 enable inputs on pin 18 to 25. Pin 18 enables input pin 10 (P0), pin 19 enables input pin 11 (P1), and so on. By default, these pins have weak pull up resistors internally, so by default, each channel is enabled. However, in noisy environments, you should actually connect the enable pin to +5V directly or through a 10K resistor.

The enable inputs do not affect the timing functions of the chip. However, they do control the counting in the RISE and FALL registers. When an input pin's corresponding enable pin is high (or not connected) the chip counts rising and falling edges as usual. If the enable pin is low, counting does not occur.

This can be used to read quadrature encoded inputs. Simply connect one phase to the input pin and the other phase to the corresponding enable pin. Then the RISE register will increase when the encoder moves in one direction and the FALL register will increase when the encoder moves in the other direction. Notice that the RISE and FALL registers do not decrease when using this scheme, so the host computer's program will need to periodically read and reset the registers.

Typical Circuit



Typical Circuit

Software

If you are using the Basic Stamp II or IISX, you can use the simple library included on the companion disk to work with the PAK-VII. You'll need to change the DATAP, DATAPIN, and CLK variables to match your circuit.

Here are the available subroutine calls:

Call	Arguments	Description
FCommand	clearreg = 1 to clear register clearchan = 1 to clear channel chan = channel number register = register number	Read register (do not set clearreg and clearchan simultaneously)
FReset	none	Reset PAK I/O
FTotalReset	none	Completely reset PAK
FPrescale	fpb=PPPP	Set prescale ratio
FPullUp	fpx=pull up byte	Set pull up byte (0=pullup on)
FThresh	fpx=threshold byte	Set threshold (0=CMOS)
FSchmitt	fpx=Schmitt byte	Set Schmitt trigger (0=on)

The library also defines constants for the register names if you want to use them. Remember that register 110 may be TICK, TICK200, or INP, depending on the channel you select.

Frequently Asked Questions

Q: Can I use a different clock?

A: Yes, you can use a different ceramic resonator or crystal to change the speed of the PAK-VII. Reducing the speed will reduce the power required, and also stretch the pulse measurement and timing functions. A 25MHz resonator, for example, will measure pulses in 10uS intervals, and double the time values for TICK and TICK200. The maximum input clock is 75MHz.

Q: I'm writing my own ShiftIn and ShiftOut instructions to communicate with the PAK. How fast can I go?

A: The Stamp II's ShiftIn and ShiftOut instructions operate at about 16kHz. However, with a 50MHz resonator, the PAK should operate fine when the CLK line operates at 100kHz.

Q: How accurate are the timing functions?

A: The timing functions are as accurate as the timing element connected. For the ceramic resonator supplied, that is reasonably accurate. However, you can replace the resonator with a 50MHz crystal (and appropriate capacitors) to improve the accuracy if you wish. Remember too that the time it takes you to communicate with the PAK is a factor as well. Using a Stamp II, reading 16 bits from the PAK takes about 1mS.

Q: How do I connect the resonator?

A: The center pin is ground. The outer two pins are interchangeable.

Notes

Notes

Specifications

Absolute Maximum Ratings

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 +7.0V
Maximum current out of VSS pin	130 mA
Maximum current into VDD pin	130 mA

DC Characteristics

Parameter	Minimum	Typical	Maximum
Supply voltage	3V	5V	5.5V
Vdd rise time on power up	.05V/ms	-	-
Supply current @ 5V/50MHz	-	77mA	