

PSG32

Table of Contents

| | |
|--|----|
| Overview | 3 |
| Features | 3 |
| Changes From PSG16..... | 3 |
| Core Hierarchy..... | 4 |
| Clocks | 4 |
| Computing Frequency Resolution | 4 |
| Maximum Frequency Generated..... | 5 |
| Example Tone Frequency Calc..... | 5 |
| Registers:..... | 6 |
| Frequency Register | 8 |
| Pulse Width Register..... | 8 |
| Control Register | 8 |
| ADSR Register..... | 9 |
| Wave Table Base Address | 11 |
| Global Registers..... | 11 |
| I/O Ports | 12 |
| Operation: | 13 |
| Frequency Synthesis | 13 |
| Wave Table | 13 |
| WISHBONE Compatibility Datasheet | 16 |

Overview

PSG32 is an audio interface circuit (sound interface device) for use within a programmable system to interface the system to an audio output. It supports four ADSR audio channels with a wavetable option. The wave table option allows arbitrary waveforms to be played.

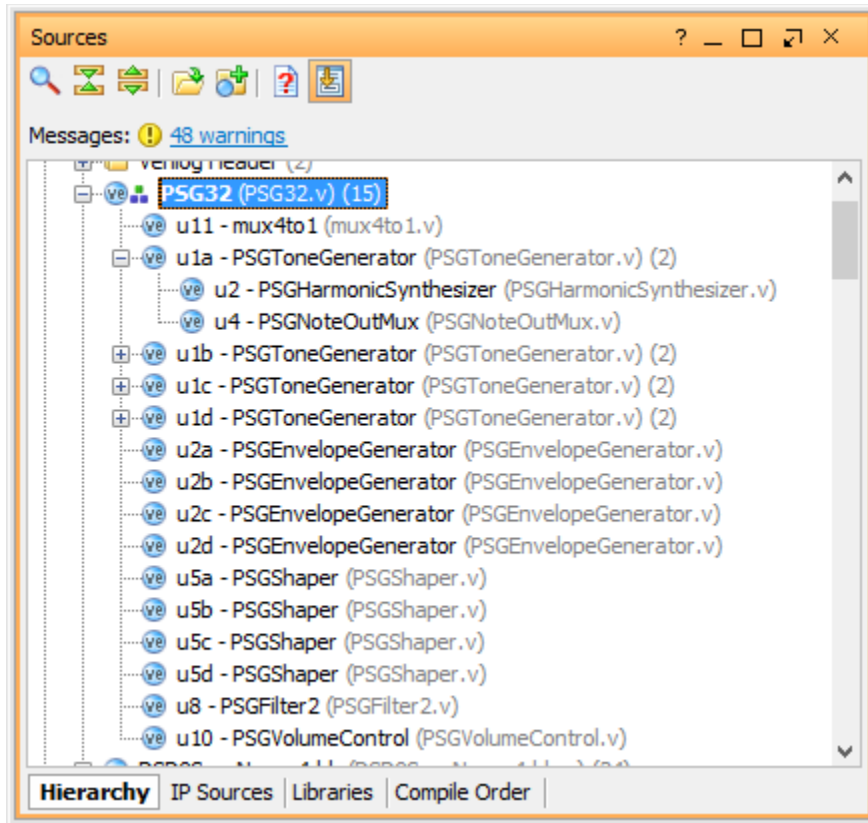
Features

- four ADSR / wave table channels (“voices”)
- programmable frequency and pulse width control
- 0.0116 Hz frequency resolution (with 50.0MHz reference clock)
- attack, decay, sustain and release
- test, ringmod, sync and gate controls
- six voice types: triangle, sawtooth, pulse, noise and wave
- frequency modulation
- digital exponential decay and release modelling (2^{**n})
- 31 tap digital FIR filter

Changes From PSG16

- the register set is 32 bit wide
- decoding of the register address range is externally supplied
- the core no longer uses TDM (time domain multiplexing) during signal generation
- the core no longer has a clock prescaler. Instead the frequency accumulators are wider.
- the core is capable of higher resolution frequencies
- 32 bit frequency synthesis accumulators are used rather than 24 bit.

Core Hierarchy



Clocks

The PSG32 core uses two clocks. The first of which is the system bus clock. The second clock is a 50MHz timing reference clock. The 50MHz reference clock is used during tone and the envelope generation. In order to have consistent values for frequency settings and envelope settings between systems a 50MHz reference clock should be used.

Computing Frequency Resolution

The frequency resolution depends on the reference clock used. 32 bit harmonic synthesizers are used as frequency generators. The minimum frequency resolution is then the clock frequency divided by 2^{32} . For a 50MHz clock this would be $50\text{MHz}/(2^{32}) = 0.0116 \text{ Hz}$.

Maximum Frequency Generated

The maximum frequency that can be generated is 2^{20} * the minimum frequency resolution. For a 50MHz clock this would be 12.2kHz.

Example Tone Frequency Calc.

For a tone of 1kHz with a 50MHz clock, the value needed in the frequency control register is $1\text{kHz}/0.0116 = 85899$.

Registers:

Registers are selected with the s_cs_i input port. A 256 byte register range is required.
All register read / write operations are 32 bit.

| reg | Bits | R/W | Brief | |
|-----|-------------------------------------|-----|--|--|
| 00 | ----- nnnn nnnnnnnn nnnnnnnn | R/W | channel 0 frequency (24 bits) | |
| 04 | ----- nnnn nnnnnnnn | R/W | channel 0 pulse width (16 bits) | |
| 08 | -----R trsgFefo vvvvvv-- | R/W | channel 0 control | |
| 0C | ---aaaaa aaaaaaaa aaaaaaaa aaaaaaaa | R/W | attack (29 bits) | |
| 10 | ----- dddddddd dddddddd dddddddd | R/W | decay (24 bits) | |
| 14 | ----- ssssssss | R/W | sustain (8 bits) | |
| 18 | ----- rrrrrrrr rrrrrrrr rrrrrrrr | R/W | release (24 bits) | |
| 1C | ----- --aaaaaaaaaaaa-- | R/W | wave table base address (14 bits, lsb=0) | |
| 20 | ----- nnnn nnnnnnnn nnnnnnnn | R/W | channel 1 frequency | |
| 24 | ----- nnnn nnnnnnnn | R/W | channel 1 pulse width | |
| 28 | -----R trsgFefo vvvvvv-- | R/W | channel 1 control | |
| 2C | ---aaaaa aaaaaaaa aaaaaaaa aaaaaaaa | R/W | attack (29 bits) | |
| 30 | ----- dddddddd dddddddd dddddddd | R/W | decay (24 bits) | |
| 34 | ----- ssssssss | R/W | sustain (8 bits) | |
| 38 | ----- rrrrrrrr rrrrrrrr rrrrrrrr | R/W | release (24 bits) | |
| 3C | ----- --aaaaaaaaaaaa-- | R/W | wave table base address (14 bits, lsb=0) | |
| 40 | ----- nnnn nnnnnnnn nnnnnnnn | R/W | channel 2 frequency | |
| 44 | ----- nnnn nnnnnnnn | R/W | channel 2 pulse width | |
| 48 | -----R trsgFefo vvvvvv-- | R/W | channel 2 control | |
| 4C | ---aaaaa aaaaaaaa aaaaaaaa aaaaaaaa | R/W | attack (29 bits) | |
| 50 | ----- dddddddd dddddddd dddddddd | R/W | decay (24 bits) | |
| 54 | ----- ssssssss | R/W | sustain (8 bits) | |
| 58 | ----- rrrrrrrr rrrrrrrr rrrrrrrr | R/W | release (24 bits) | |
| 5C | ----- --aaaaaaaaaaaa-- | R/W | wave table base address (14 bits, lsb=0) | |
| 60 | ----- nnnn nnnnnnnn nnnnnnnn | R/W | channel 3 frequency | |
| 64 | ----- nnnn nnnnnnnn | R/W | channel 3 pulse width | |
| 68 | -----R trsgFefo vvvvvv-- | R/W | channel 3 control | |
| 6C | ---aaaaa aaaaaaaa aaaaaaaa aaaaaaaa | R/W | attack (29 bits) | |
| 70 | ----- dddddddd dddddddd dddddddd | R/W | decay (24 bits) | |
| 74 | ----- ssssssss | R/W | sustain (8 bits) | |
| 78 | ----- rrrrrrrr rrrrrrrr rrrrrrrr | R/W | release (24 bits) | |
| 7C | ----- --aaaaaaaaaaaa-- | R/W | wave table base address (14 bits, lsb=0) | |
| A0 | uuuuuuuu uuuuuuuu uuuuuuuu uuuuuuuu | R/W | scratchpad0 | |
| A4 | uuuuuuuu uuuuuuuu uuuuuuuu uuuuuuuu | R/W | scratchpad1 | |
| A8 | uuuuuuuu uuuuuuuu uuuuuuuu uuuuuuuu | R/W | scratchpad2 | |
| AC | uuuuuuuu uuuuuuuu uuuuuuuu uuuuuuuu | R/W | scratchpad3 | |
| B0 | ----- mmmmm | R/W | master volume | |
| B4 | nnnnnnnn nnnnnnnn nnnnnnnn nnnnnnnn | R | osc3 oscillator 3 output | |
| B8 | ----- nnnnnnnn | R | env3 envelope 3 output | |
| BC | ----- -sss-sss-sss-sss | R | envelope state | |

| | | | | |
|---------|-------------------------|-----|----------------------------|--|
| C0 | ----- RRRRRRRR RRRRRR | R/W | filter sample rate divider | |
| 100-178 | ----- s---kkkk kkkkkkkk | W | filter coefficients | |
| | | | | |

Frequency Register

This register sets the tone frequency for the voice. In order to set the frequency specify a value that is a multiple of the base frequency step. For example for an 800 Hz tone with a 50MHz clock, $800/0.0116 = 68719$ would need to be specified.

Pulse Width Register

This register controls the pulse-width when the pulse output waveform is selected. Pulse frequency is controlled by the frequency register.

Control Register

‘R’ bit is a software reset bit for the envelope generator

‘o’ bit enables the output for the voice

‘F’ bit enables frequency modulation by the output of the previous voice.

‘e’ bit when set routes the tone generator through the envelope generator. When clear the raw tone generator is used without an envelope. This is primarily for debugging.

‘f’ bit tells the sound generator to route the voice’s output to the filter

‘vvvvvv’ sets the output voice type

101000 = reverse sawtooth

010000 = triangle wave

001000 = sawtooth wave

000100 = pulse (or possibly square)

000010 = noise

000001 = wave

‘g’ bit ‘gates’ the envelop generator which when set causes it to begin generating the envelope for the voice. When the gate is turned off, the envelope generator enters the release phase.

ADSR Register

‘a’ - Attack

The attack code controls the attack rate of the sound envelope. The attack slope is triggered when the gate signal is activated. The envelope travels from a zero level to it's peak during the attack phase.

The value required in the rate register can be calculated as:

reg value = $1 / ((1 / \text{clock frequency}) / \text{desired time})$

Example: reg value = $1 / ((1 / 50e6) / 2e-3)$
= 390.625

Some typical attack values:

| Rate Divider | Attack Time |
|--------------|-------------|
| 100000 | 2 ms |
| 400000 | 8 ms |
| 800000 | 16 ms |
| 1200000 | 24 ms |
| | 38 ms |
| | 56 ms |
| | 68 ms |
| | 80 ms |
| | 100 ms |
| | 239 ms |
| | 500 ms |
| 40000000 | 800 ms |
| | 1 s |
| | 3.2 s |
| | 5.3 s |
| 400000000 | 8 s |

‘d’ = Decay

The decay code controls the decay rate of the sound envelope just after the peak has been reached from the attack phase. The envelop decays from it’s peak value down to the value set by the sustain code.

The value required in the rate register can be calculated as:

reg value = $1 / ((1 / \text{clock frequency}) / \text{desired time}) / 256$

Example: reg value = $1 / ((1 / 50e6) / 2e-3) / 256$
= 390.625

| Rate Divider | Decay/Release Time |
|--------------|--------------------|
| 1172 | 6 ms |
| 4688 | 24 ms |
| | 48 ms |
| | 72 ms |
| | 114 ms |
| | 168 ms |
| | 204 ms |
| | 240 ms |
| | 300 ms |
| | 750 ms |
| | 1.5 s |
| | 2.4 s |
| | 3.0 s |
| | 9.0 s |
| | 15.0 s |
| 4687500 | 24.0 s |

‘s’ = Sustain

Sustain sets the signal level at which the signal is ‘sustained’ relative to it’s peak value. There are 255 sustain levels from 0x0 to 0xFF with 0x0 being the lowest and 0xFF the maximum.

‘r’ = Release

The release code controls the rate at which the signal is ‘released’ after the gate is turned off. When the gate signal is made inactive, the release phase of the ADSR envelope begins. This is an exponential of 2 release.

| Rate Divider | Decay/Release Time |
|--------------|--------------------|
| | 6 ms |
| | 24 ms |
| | 48 ms |
| | 72 ms |
| | 114 ms |
| | 168 ms |
| | 204 ms |
| | 240 ms |
| | 300 ms |
| | 750 ms |
| | 1.5 s |
| | 2.4 s |
| | 3.0 s |
| | 9.0 s |
| | 15.0 s |
| | 24.0 s |

Wave Table Base Address

This register sets the beginning address for the wave table scan. Data values are read offset from this address by the output of the tone generator bits 17 to 27. Up to 2047 samples may be scanned. A repeating linear scan of the wave table can be accomplished by setting the tone generator to generate a sawtooth waveform.

Global Registers

A0,A4,A8,AC – these are 32 bit scratchpad registers which may be used to store data.

B0h – VOL – master volume.

BCh - ES – reflects the envelope state for each of the four envelope generators.

| SSS | Envelope State |
|-----|----------------|
| 0 | IDLE |
| 1 | ATTACK |
| 2 | DECAY |
| 3 | SUSTAIN |
| 4 | RELEASE |
| 5-7 | reserved |

I/O Ports

I/O is via a standard WISHBONE slave port with the addition of a circuit select line. An additional non-WISHBONE port is used to access the wave table memory. All accesses are 32 bit word wide accesses.

Reading the PSG has a three cycle latency before the core responds with an ack. Writing the PSG is single cycle.

| Name | Wid | I/O | Description | |
|---------|-----|-----|---|--|
| rst_i | 1 | I | This is the active high reset signal | |
| clk_i | 1 | I | system bus clock | |
| clk50_i | 1 | I | 50MHz timing reference clock | |
| s_cs_i | 1 | I | circuit select | |
| s_cyc_i | 1 | I | cycle active | |
| s_stb_i | 1 | I | data strobe | |
| s_ack_o | 1 | O | data transfer acknowledge | |
| s_we_i | 1 | I | write cycle | |
| s_adr_i | 8 | I | decode / register address, the two LSB's are not used in the core but must still be supplied. | |
| s_dat_i | 32 | I | data input | |
| s_dat_o | 32 | O | data output | |
| | | | | |
| m_adr_o | 14 | O | master address for wave table memory | |
| m_dat_i | 12 | I | master data input from wave table memory | |
| | | | | |
| o | 18 | O | 18 bit audio output | |
| | | | | |
| | | | | |

Operation:

Frequency Synthesis

The PSG uses a harmonic frequency synthesizer with a 32 bit accumulator. This gives the generator a base frequency step of 0.0116Hz. ($50e6 / 2^{32}$). The upper bits of the accumulator are used as a source for audio waves.

Wave Table

It is anticipated that the PSG core will be used in a system where dual port block memories are available and so the PSG core has a dedicated bus for the wave table memory. It's assumed that the wave table memory is capable of an access every clock cycle. The PSG uses the tone generator accumulator to generate 11 bit address offsets from which to read. The address used is the sum of the wave table base address register and 11 bits from the tone generator. Up to 16kiB of wave table memory is supported, allowing several different waveforms to be stored simultaneously. Access to the wave table is pipelined. Each channel of the PSG is given access to the wave table on successive clock cycles. Three clock cycles later data for the channel is latched in. There must be a memory latency of three clock cycles for wave table memory in order for the PSG's wave input to work correctly. Note that data is latched on every clock cycle for successive channels. The wave table is always being addressed by the core, however data latched in is not used unless selected in the control register for the channel. The wave table can be scanned at different rates depending on the frequency the channel is setup for. The same data value will be loaded from the wave table if the address does not change. The address may not change every clock cycle, however data will still be latched in.

Filter

The filter is a time domain multiplexed (TDM) filter in order to conserve resources. A digital FIR (finite impulse response) filter is used.

Filter Sample Frequency

The filter's sampling frequency is one of the characteristics controlling filter output. The sampling frequency factors into the calculations for the filter coefficients.

The sample frequency of the filter may be set using a sixteen bit control register which contains a clock divider value. This register is provided to make it easier to use the same filter coefficients in systems with different clock rates. The filter sample rate should be set to a rate substantially higher than the highest frequency

to be filtered. For example 100kHz. To get a 100kHz sample rate from a 50MHz reference clock the clock needs to be divided by 500. So the clock rate divider (CRD) register should be set to 500.

Taps

The filter contains a number of taps, which are points at which filter coefficients are applied to the input signal. The filter has a fixed number of 31 taps. Filter coefficients must be supplied for each tap.

Filter Coefficients

The filter coefficients control the resulting type of filter (low pass, band pass, high pass, band stop) and the frequency response of the filter. The filter coefficients are 12 fractional bits plus a sign bit. Filter coefficients range in value from -.9999 to +.9999

Software Sample

```
-----  
; Beep: a 800Hz tone for 1 sec.  
; Using a 37.5MHz bus clock.  
-----  
beep:  
    tgt  
    mark1  
    ldi        r5,$FFD50000  
    lea        r1,$B0[r5]  
    ldi        r1,$FF  
    stt        r1,$B0[r5]          ; set volume to 100%  
    ldi        r1,#91626          ; 800Hz  
    stt        r1,$00[r5]          ; frequency 0  
    ldi        r1,#200             ; attack  
    stt        r1,$0C[r5]  
    ldi        r1,#100             ; decay  
    stt        r1,$10[r5]  
    ldi        r1,$80             ; 128 sustain level  
    stt        r1,$14[r5]          ;  
    ldi        r1,#500             ; release  
    stt        r1,$18[r5]  
    ldi        r1,$1504           ; gate, output, triangle wave  
    stt        r1,$008[r5]  
    csrrw     r2,#2,r0             ; get tick  
    add       r1,r2,#37500000  
.  
.ipsg1:  
    csrrw     r2,#2,r0  
    bltu      r2,r1,.ipsg1  
    ldi        r1,$0000  
    stt        r1,$008[r5]          ; turn off gate  
    ret
```

WISHBONE Compatibility Datasheet

The PSG core may be directly interfaced to a WISHBONE compatible bus.

| | | |
|---|--|--|
| WISHBONE Datasheet WISHBONE SoC Architecture Specification, Revision B.3 | | |
| | | |
| Description: | Specifications: | |
| General Description: | PSG32 – programmable ADSR sound generator | |
| Supported Cycles: | SLAVE, READ / WRITE SLAVE, BLOCK READ / WRITE SLAVE, RMW | |
| Data port, size: | 32 bit 32 bit 32 bit Little Endian any (undefined) | |
| Data port, granularity: | | |
| Data port, maximum operand size: | | |
| Data transfer ordering: | | |
| Data transfer sequencing | | |
| Clock frequency constraints: | 16 MHz minimum to 300 MHz maximum | |
| Supported signal list and cross reference to equivalent WISHBONE signals | Signal Name: ack_o adr_i(7:0) clk_i dat_i(31:0) dat_o(31:0) cyc_i stb_i we_i | WISHBONE Equiv. ACK_O ADR_I() CLK_I DAT_I() DAT_O() CYC_I STB_I WE_I |
| Special Requirements: | | |