ChipTune

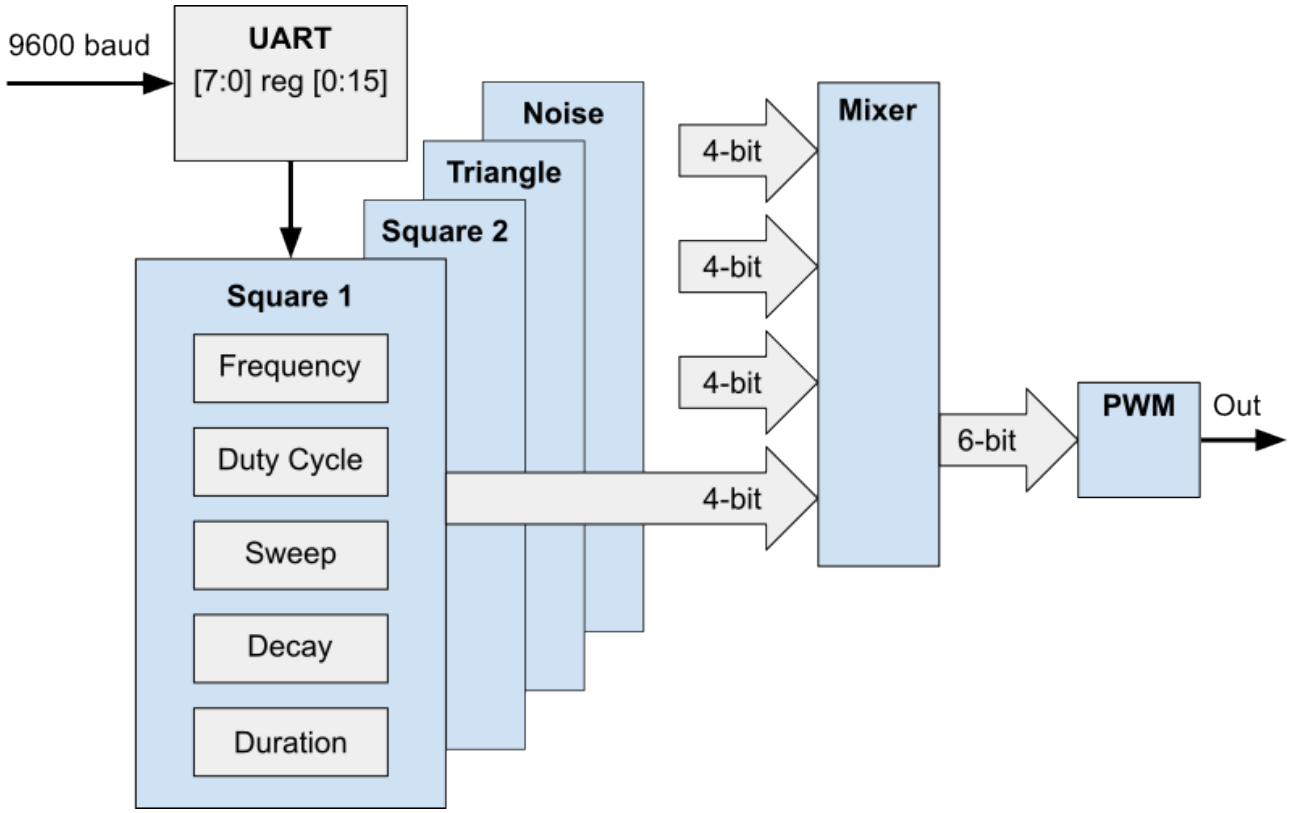

Figure 1: picture

- Author: Wallace Everest
- Description: Vintage 8-bit sound generator
- Language: Verilog

How it works

ChipTune implements an 8-bit Programmable Sound Generator (PSG). Input is from a serial UART interface. Output is PWM audio.

Overview

This project replicates the Audio Processing Unit (APU) of vintage video games.

Statistics

- Tiles: 2x2
- DFF: 388
- Total Cells: 2366
- Utilization: 30%

TinyTapeout 4 Configuration

TT04 devices from the eFabless Multi-Project Wafer (MPW) shuttle are delivered in QFN-64 packages, mounted on a daughterboard for breakout.

Based on data from:

- https://github.com/efabless/caravel_board/blob/main/hardware/breakout/caravel-M.2-card-QFN/caravel-M.2-card-QFN.pdf
- https://github.com/TinyTapeout/tt-multiplexer/blob/main/docs/INFO.md
- https://open-source-silicon.slack.com/archives/C016N88BX44/p1688915892223379

MPRJ_IO Pin Assignments

Signal	Name	Dir	QFN	PCB
mprj_io[0]	jtag	in	31	
mprj_io[1]	sdo	out	32	
mprj_io[2]	sdi	in	33	
mprj_io[3]	csb	in	34	
mprj_io[4]	sck	in	35	
mprj_io[5]	user_clk	out	36	
mprj_io[6]	clk	in	37	
mprj_io[7]	rst_n	in	41	
mprj_io[8]	ui_in[0]	in	42	
mprj_io[9]	ui_in[1]	in	43	
mprj_io[10]	ui_in[2]	in	44	
mprj_io[11]	ui_in[3]	in	45	
mprj_io[12]	ui_in[4]	in	46	
mprj_io[13]	ui_in[5]	in	48	
mprj_io[14]	ui_in[6]	in	50	
mprj_io[15]	ui_in[7]	in	51	
mprj_io[16]	uo_out[0]	out	53	

Signal	Name	Dir	QFN	PCB
mprj_io[17]	uo_out[1]	out	54	
mprj_io[18]	uo_out[2]	out	55	
mprj_io[19]	uo_out[3]	out	57	
mprj_io[20]	uo_out[4]	out	58	
mprj_io[21]	uo_out[5]	out	59	
mprj_io[22]	uo_out[6]	out	60	
mprj_io[23]	uo_out[7]	out	61	
mprj_io[24]	uio[0]	bid	62	
mprj_io[25]	uio[1]	bid	2	
mprj_io[26]	uio[2]	bid	3	
mprj_io[27]	uio[3]	bid	4	
mprj_io[28]	uio[4]	bid	5	
mprj_io[29]	uio[5]	bid	6	
mprj_io[30]	uio[6]	bid	7	
mprj_io[31]	uio[7]	bid	8	
mprj_io[32]	sel_ena	in	11	
mprj_io[33]	spare		12	
mprj_io[34]	sel_inc	in	13	
mprj_io[35]	spare		14	
mprj_io[36]	sel_rst_n	in	15	
mprj_io[37]	spare		16	

APU Operation

The audio portion of the project consists of two rectangular pulse generators, a triangle wave generator, and a noise generator. Each module is controlled by four 8-bit registers. Configurable parameters are the frequency, duty cycle, sweep, decay, and note duration.

An explanation of register functions can be found on the NESDEV website. Only the lower 4-bits of the address are decoded.

- https://www.nesdev.org/wiki/APU

UART Operation

- A register address and data are recovered from two consecutive serial bytes.
- A byte with the msb=0 is considered the first byte with 7-bits of data.

- A byte with msb=1 is considered the second byte with the remaining 1-bit of data and a 6-bit address.
- A ready flag is generated after receiving the second byte.

```
Byte 1                                  Byte 2
Start D0 D1 D2 D3 D4 D5 D6 0 Stop  Start D7 A0 A1 A2 A3 A4 A5 1 Stop
```

Pin Assignments

Signal	Name	Signal	Name
clk	12 MHz	ena	spare
rst_n	spare	uio_oe[7:0]	spare
ui_in[0]	spare	uo_out[0]	spare
ui_in[1]	spare	uo_out[1]	spare
ui_in[2]	rx	uo_out[2]	tx
ui_in[3]	spare	uo_out[3]	pwm
ui_in[4]	spare	uo_out[4]	spare
ui_in[5]	spare	uo_out[5]	spare
ui_in[6]	spare	uo_out[6]	blink
ui_in[7]	spare	uo_out[7]	link
uio_in[7:0]	spare	uio_out[7:0]	spare

- CLK is a 1.789733 MHz clock
- BLINK is an LED status indicator with a 1 Hz rate
- LINK is an LED activity indicator of the RX signal
- PWM is the pulse-width modulated audio output
- RX is a UART input (9600,8,N,1)
- TX is a loopback signal from RX

How to test

The ChipTune project can be interfaced to a computer COM port (9600,n,8,1). An analog PWM filter and audio driver are needed for the test rig.

The following serial strings will activate example functions:

```
# Square 1
08 82 30 80 00 84 00 86 #Clear
```

```
27 83 02 81 7E 85 08 86 #PlaySmallJump
27 83 02 81 7C 84 09 86 #PlayBigJump
13 83 1E 81 3A 84 0A 86 #PlayBump
19 83 1E 81 0A 84 08 86 #PlayFireballThrow
4B 83 1F 81 6F 85 08 86 #PlaySmackEnemy
1C 83 1E 81 7E 85 08 86 #PlaySwimStomp
08 82 3F 81 17 84 01 86 #400Hz
# Square 2
30 88 08 8A 00 8C 00 8E #Clear
18 89 7F 8A 71 8C 08 8E #PlayTimerTick
0D 89 7F 8A 71 8C 08 8E #PlayCoinGrab
1F 89 14 8B 79 8D 0A 8E #PlayBlast
7F 88 5D 8A 71 8C 08 8E #PlayPowerUpGrab
3F 89 08 8A 17 8C 01 8E #400Hz
# Triangle
00 91 00 92 00 94 00 96 #Clear
00 92 0B 95 00 96 40 91 #400Hz
# Noise
30 98 00 9A 00 9C 00 9E #Clear
00 9A 00 9E 05 9D 3F 98 #300Hz
```

IO

#	Input	Output	Bidirectional
0	None	None	None
1	None	None	None
2	RX	TX	None
3	None	PWM	None
4	None	None	None
5	None	None	None
6	None	BLINK	None
7	None	LINK	None