



FPGAs 101: Everything You Need to Know to Get Started

Dr HAMADI CHAREF Brahim

February 2023
(First presented October 2015)

Motivation and main message ...

- **FPGA is not complicated**
- **FPGA is not expensive**
- Think “hardware”
- Learn from examples
- Study FPGA literature eBooks
- Join FPGA Community Singapore and our meeting/workshop

FPGA Community Singapore

The screenshot shows the Facebook group page for 'FPGA Community Singapore'. The cover photo is a close-up of an FPGA development board with various components labeled, including a central 'SPARTAN-6' chip. The group name 'FPGA Community Singapore' and 'Public Group' are displayed at the top left. A 'Joined' button with a dropdown arrow is in the top right. Below the header are navigation tabs for 'Discussion', 'Members', 'Events', 'Photos', and 'Files', followed by a search bar. The main content area displays an event titled 'FPGA.SG v0.1 Meeting / Workshop October 2015' scheduled for October 31.

This is a detailed view of the event page for the 'FPGA.SG v0.1 Meeting / Workshop October 2015'.
- **Event Details:** OCT 31, Public - Hosted by Brahim Hamadicharef and Michał J. Gajda. Options to 'Hosting', 'Invite', 'Edit', and more are available.
- **Location:** 31 October at 10:00–18:00, Next Week. Location is HackerspaceSG, 344B, KING GEORGE'S AVENUE, KING GEORGE'S BUILDING, SINGAPORE. A 'Show Map' link is provided.
- **Creator:** Created for FPGA Community Singapore.
- **Description:** This will our first FPGA.SG Meeting / Workshop. Plan is to introduce FPGA topic, programming tool and language, development board, and spend the afternoon going through some tutorials and demos, and lot of discussions.
- **Attendees:** 21 going, 15 maybe, 50 invited. A grid of 12 profile pictures shows who is attending, with 9 marked with a green checkmark.
- **Friends:** Michał, Shanmugam and 8 other friends are going.

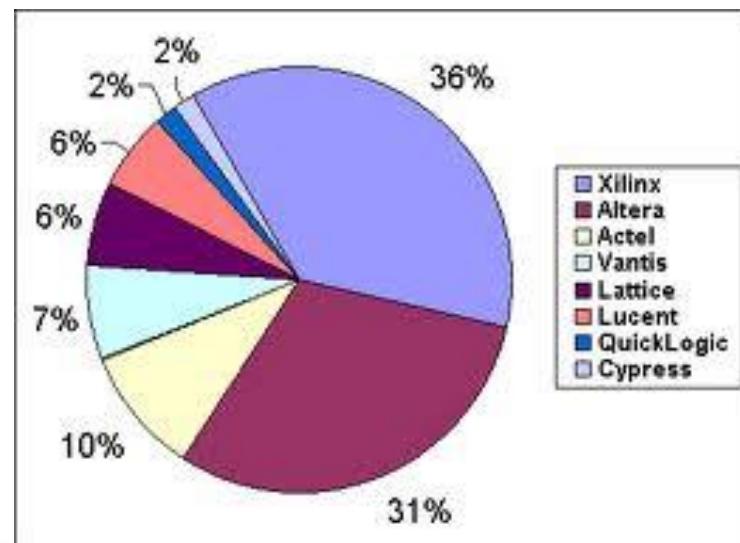
Agenda

- Manufacturers, distributors
- Few boards (S3E and ZC706)
- FPGA inside
- Tools Xilinx ISE / Vivado
- VHDL Hello world !
- Demonstrations examples
- FPGA Literature eBooks
- Research and challenges

FPGA - Manufacturers

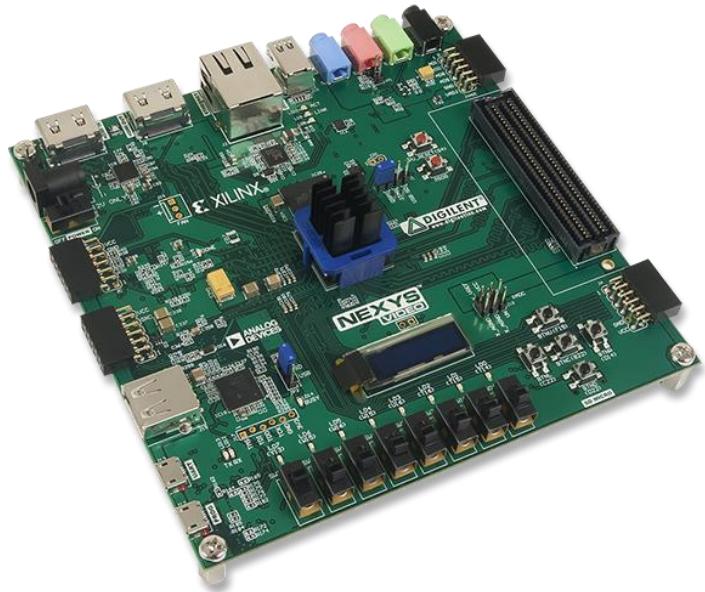
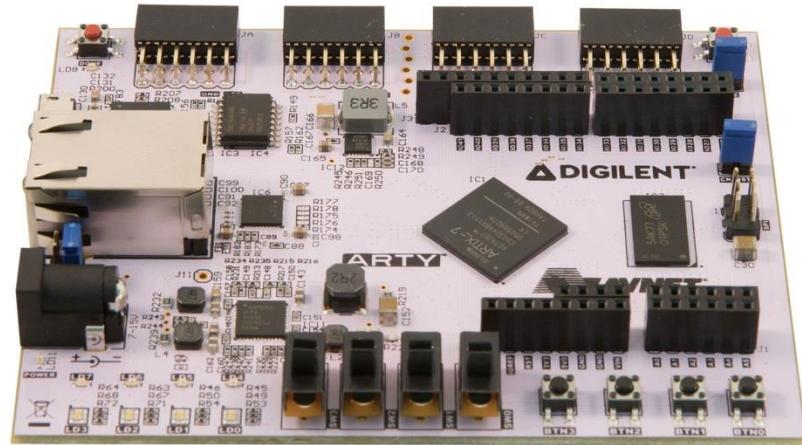


ALL PROGRAMMABLE™



FPGA - Distributors

- Lots of choice ... Arty, Zybo, ZedBoard, Anvyl, Genesys, Nexys , Basys, CoolRunner-II, Mimas, Elbert, Saturn, etc



NUMATO LAB

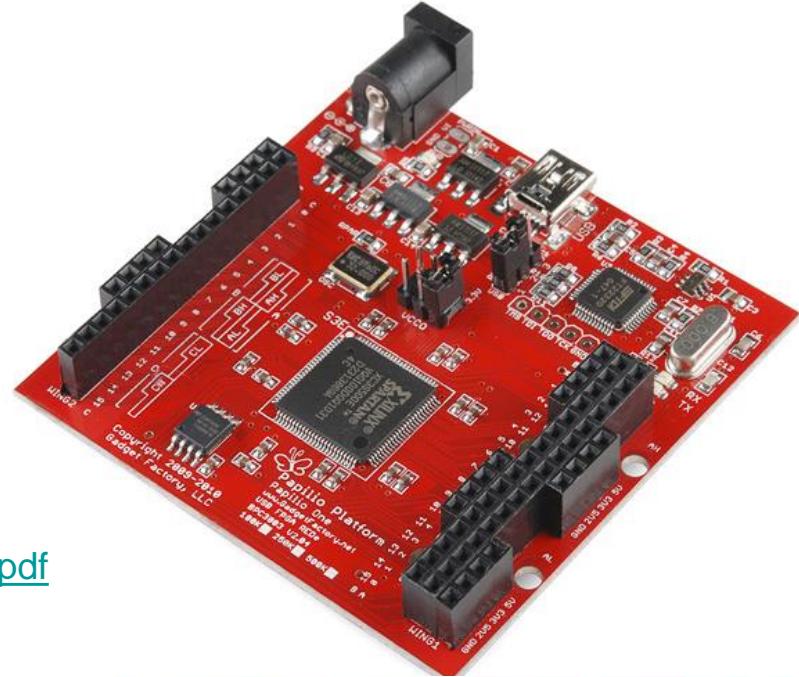
<http://numato.com/fpga-boards.html>



<http://www.digilentinc.com/Products/Catalog.cfm?NavPath=2,400&Cat=10&FPGA>

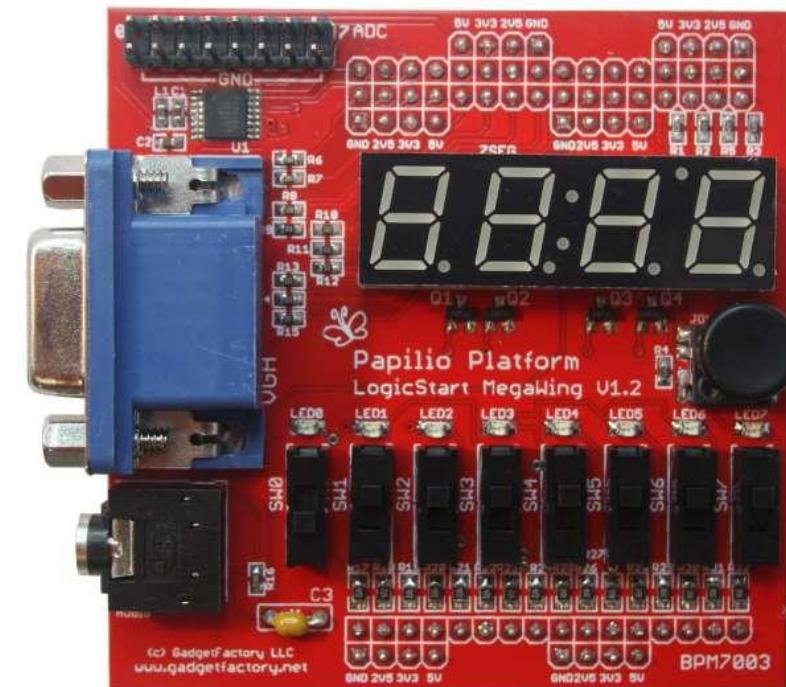
Papilio One 250k

- **Papilio One 250k (S\$70)**
 - Spartan 3E - xc3s250e-vq100-4
 - 5508K Logic Cells, 12 Multipliers
 - 216K RAM blocks
 - 32 MHz oscillator (DCM)



http://www.xilinx.com/support/documentation/data_sheets/ds312.pdf

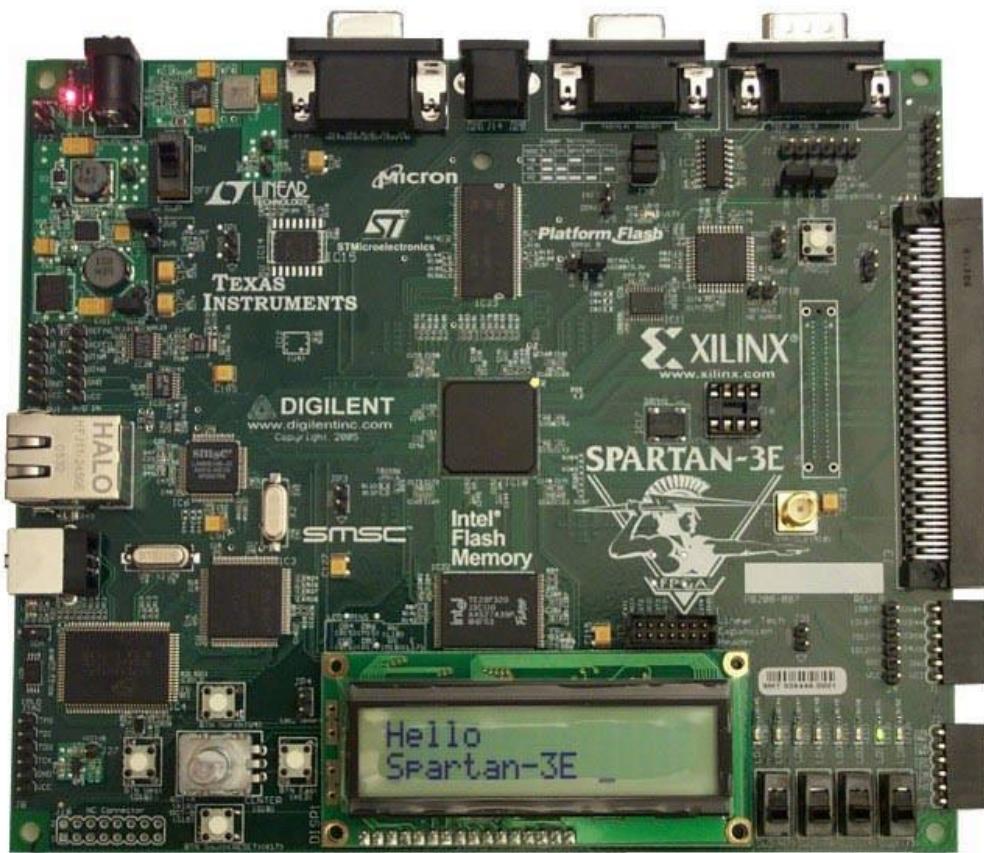
- **Logic Start Mega Wing (S\$50)**
 - 7 Segment Display – 4-digits
 - VGA Port (3R,3G,2B Output)
 - Mono Audio Jack - 1/8" Jack
 - RC Low Pass Filter
 - Micro joystick - 5 directions
 - SPI ADC - 12-bit, 1Msps, 8 Channel
 - 8 LED's - User Feedback
 - 8 Slide Switches - User Input



<http://papilio.cc/index.php?n=Papilio.PapilioOne>

Diligent Spartan-3E Starter Kit Board

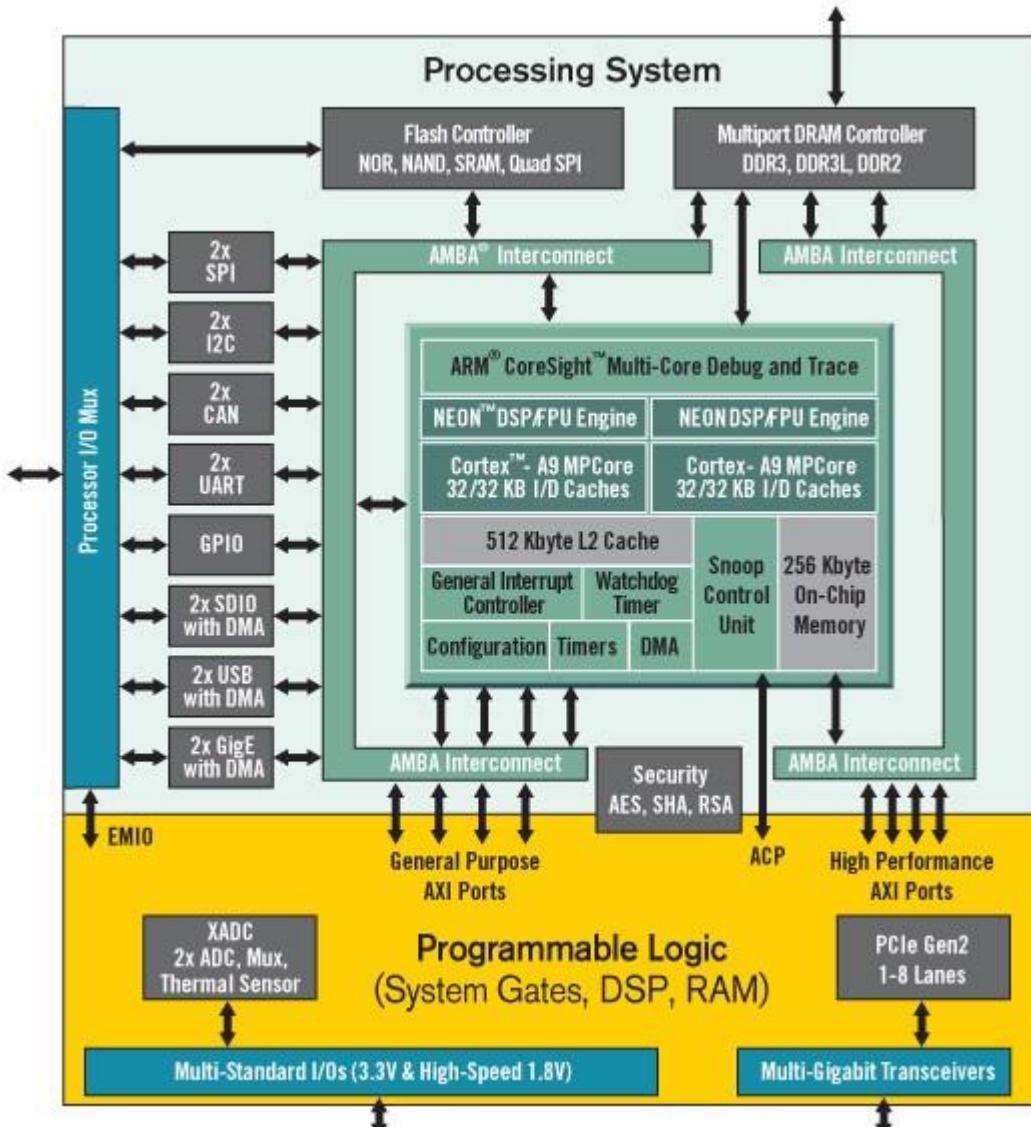
- **Spartan-3E Starter Kit Board (\$199)**
 - Spartan 3E - XC3S500E-4FG320C
 - 10476 Logic Cells, 20 dedicated multipliers
 - 360K Block RAM bits
 - 50 MHz oscillator (DCM)
 - 2x 16 LCD display
 - 4 buttons + 4 switches
 - Rotatory encoder
 - 10/100 Ethernet RJ45
 - VGA 3-Bit Display
 - PS/2 Mouse/Keyboard Port
 - Quad DAC LTC2624
 - Dual ADC LTC1407A-1
 - SPI Serial Flash M25P16
 - DDR SDRAM 512 Mb
 - Serial output
 - 2x Pmod™ connectors



Xilinx Zynq-7000 SoC ZC607

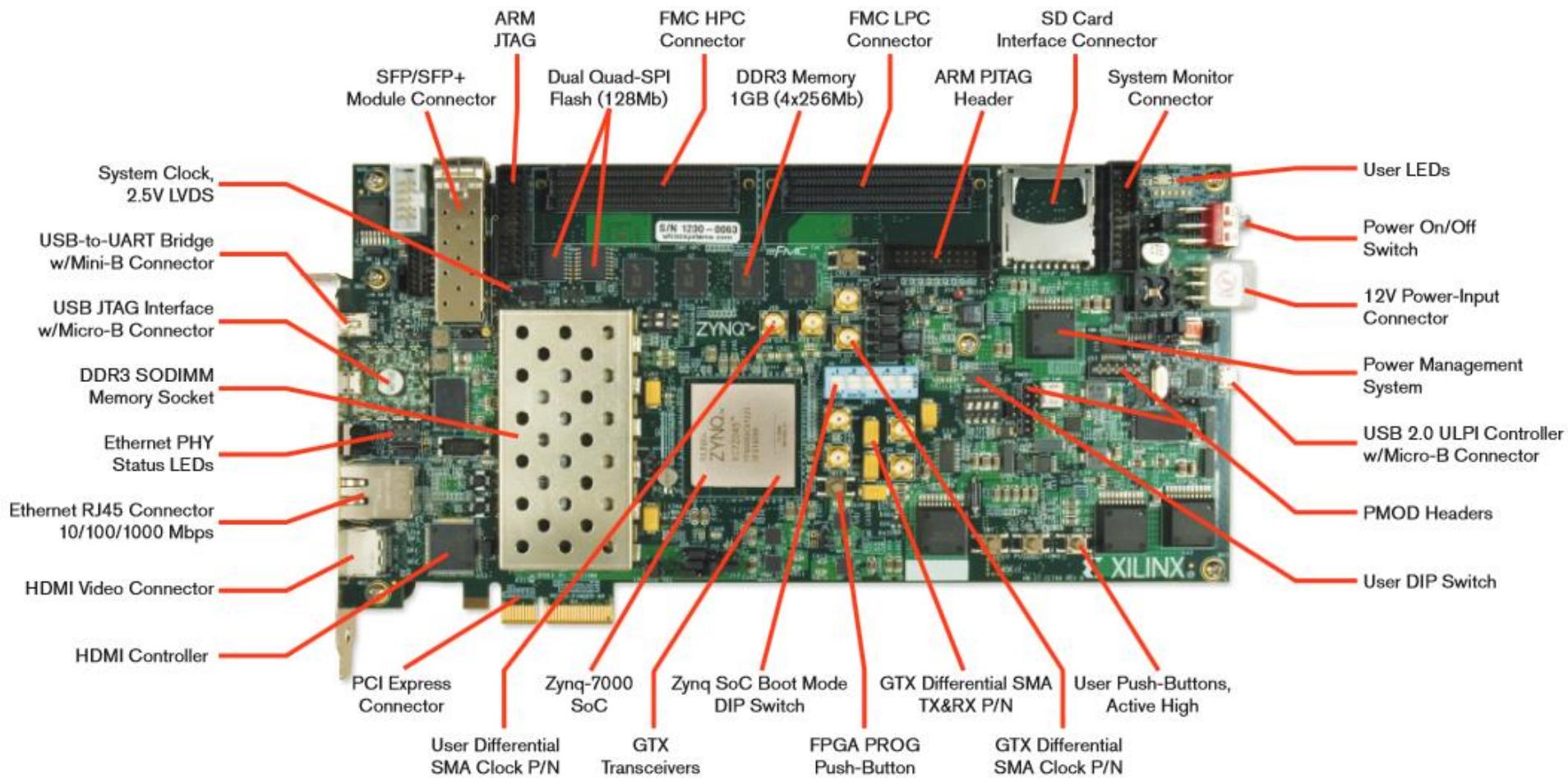
- **Xilinx Zynq-7000 All Programmable SoC ZC706 Evaluation Kit**
 - Kintex-7 FPGA XC7Z045 FFG900 -2
 - 350K Logic Cells
 - 900 DSP Slices
 - 437200 36Kb RAM blocks
 - 218600 LUTs
 - DDR3 SODIM Memory 1Gb
 - HDMI, PCIe
 - XADC, GigE, etc.

EK-Z7-ZC706-G
US\$2495

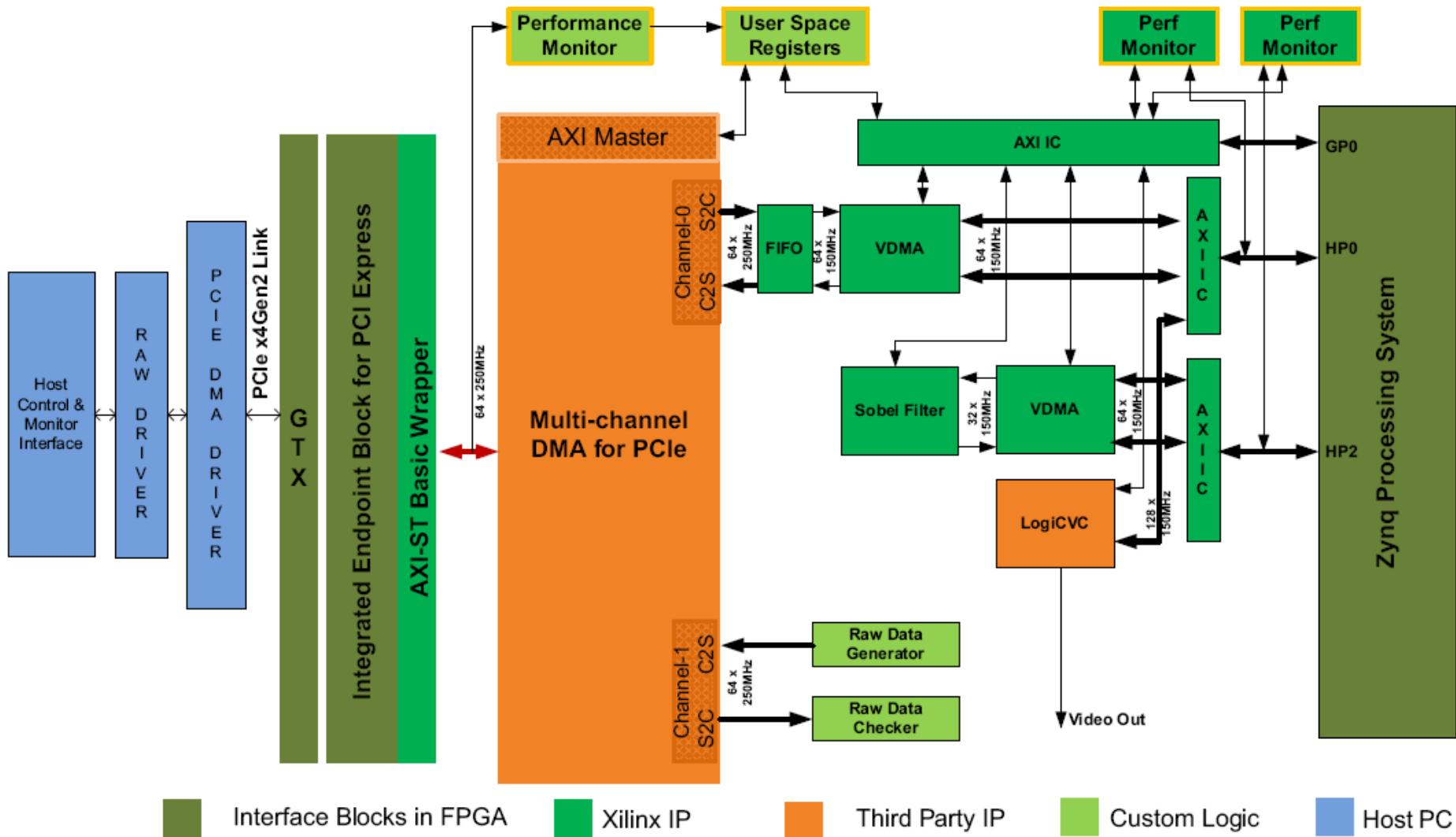


<http://www.xilinx.com/products/boards-and-kits/ek-z7-zc706-g.html>

Xilinx Zynq-7000 SoC ZC607



Xilinx ZC706 PCIe Targeted Reference Design



FPGArduino - FPGA-based Arduino



[Terasic DE0-Nano
\(Altera Cyclone-IV\)](#)

[Xilinx Spartan 3E-
500 Starter Kit](#)

[Xilinx Spartan 3E-
1600 Dev. Board](#)

[Xilinx Spartan
3A/3AN Starter Kit](#)

[Digilent Basys-3
\(Xilinx Artix-7\)](#)



[Digilent Nexys-3
\(Xilinx Spartan-6\)](#)

[Digilent ZYBO
\(Xilinx Zynq\)](#)

[FER ULX2S
\(Lattice XP2\)](#)

[E2LP
\(Xilinx Spartan-6\)](#)

[Lattice Brevia
\(Lattice XP2\)](#)



[Lattice Brevia 2
\(Lattice XP2\)](#)

[No-name TB276
\(Altera Cyclone-IV\)](#)

[No-name TB299
\(Xilinx Spartan-6\)](#)

[Scarab
MiniSpartan6+
\(Xilinx Spartan-6\)](#)

[Numato Mimas V2
\(Xilinx Spartan-6\)](#)

ZPU - FPGA-based Arduino

ZPUino demo
for Spartan 3E
Evaluation
Board (Starter
Kit)

ZPU toolchain

Description	File	Size
ZPU GCC toolchain linux x86	zpu-toolchain-linux-1.0.tar.bz2	24M

ZPUino HDL core

Board name	FPGA	Core features	Download
Papilio One 250	S3E 250	Regular	download
		Many PWM	download
		VGA ZX	download
Papilio One 500	S3E 500	Regular	download
		SID	download
		SID + VGA 160x120x8	download
Papilio Plus LX4	S6LX4	Regular	download
Papilio Plus LX9	S6LX9	Regular	download
Spartan3E Starter Kit	S3E 500	Regular	download
Digilent Nexys2 Board	S3E 1200	Regular	download

Other boards

- Contact us at zpuino@alvie.com for other platforms and boards

Source downloads

ZPU toolchain

- GIT source: <git://repo.or.cz/zpugcc.git>
- Patch: [0001-libgloss-_mask-belongs-in-.rodata-section-not-in-.da.patch](#)
You will need to apply the above patches before building the ZPU GCC toolchain.

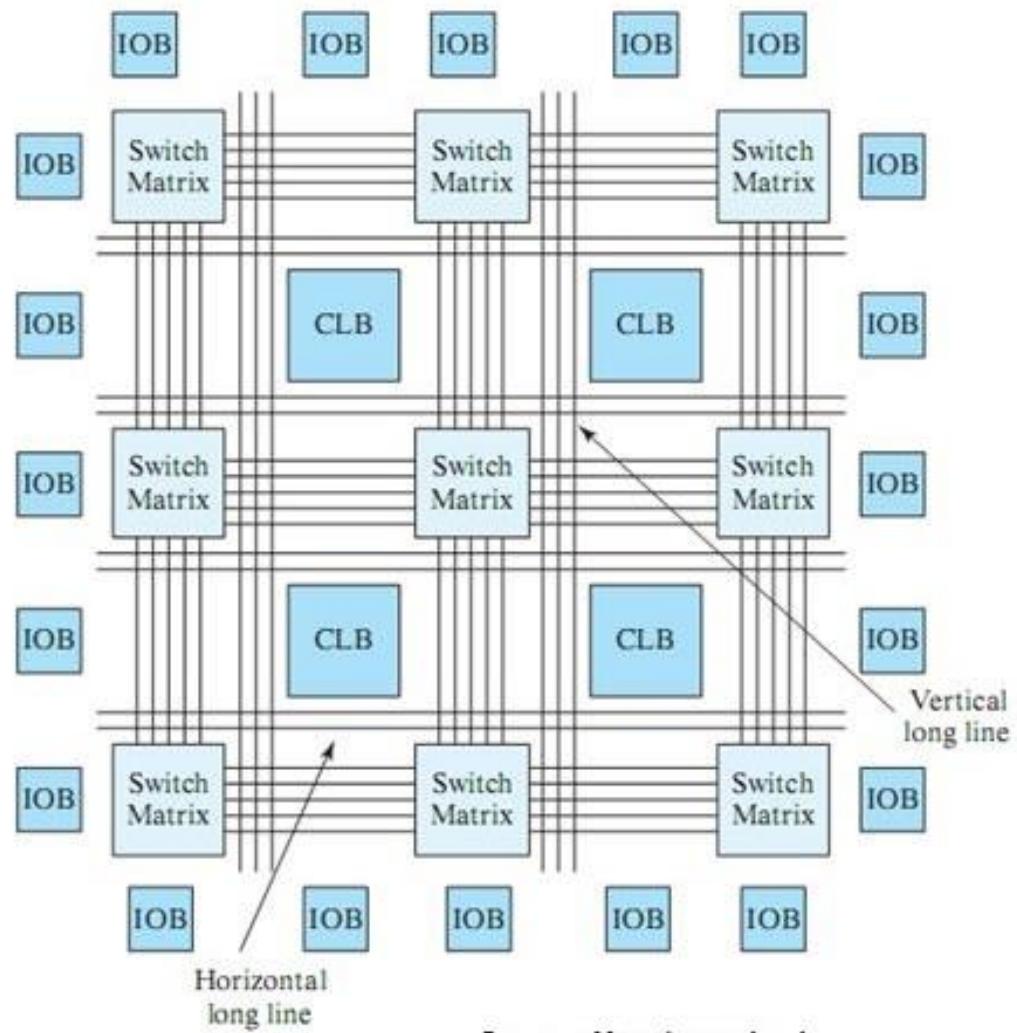
ZPUino IDE

- <http://github.com/alvieboy/ZPUino>

FPGA – Field Programmable Gate Array

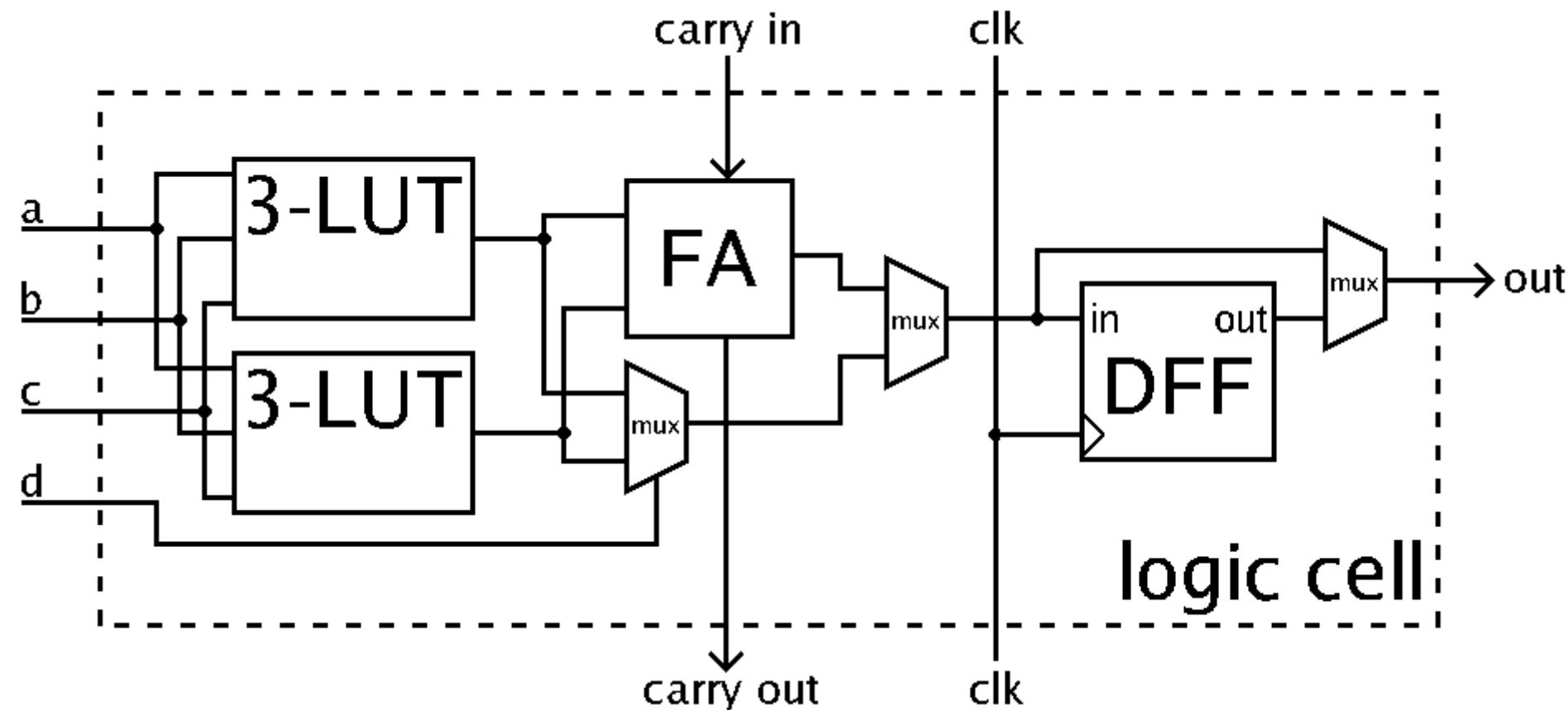
Xilinx FPGAs

- Configurable Logic Block (CLB)
 - Programmable logic and FFs
- Programmable Interconnects
 - Switch Matrices
 - Horizontal/vertical lines
- I/O Block (IOB)
 - Programmable I/O pins



Source: Mano's textbook

FPGA inside - Configurable Logic Block (CLB)



http://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf

FPGA inside – SLICEL and SLICEM

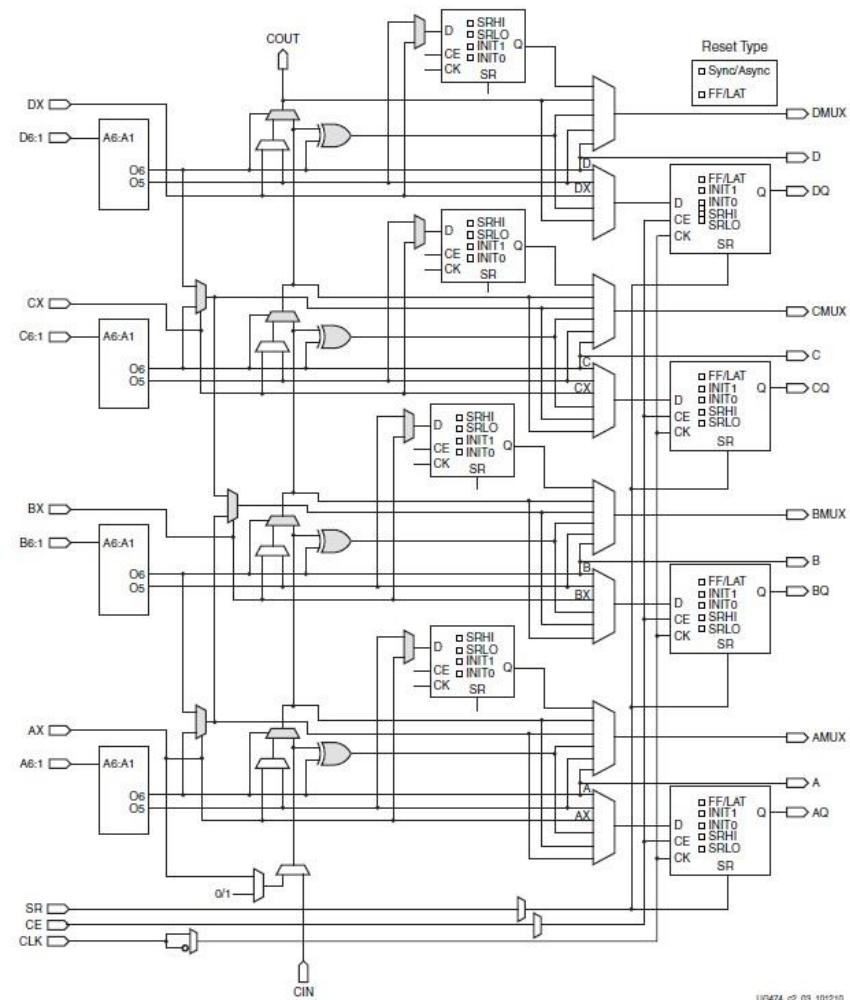


Figure 2-4: Diagram of SLICEL

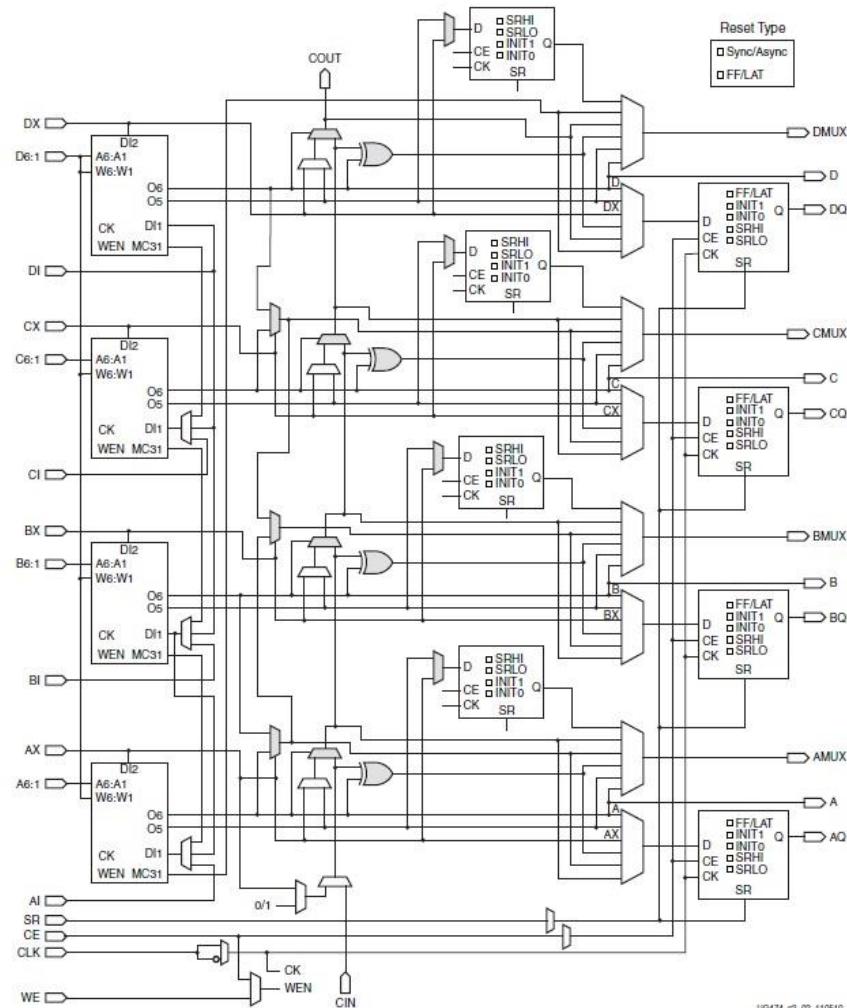
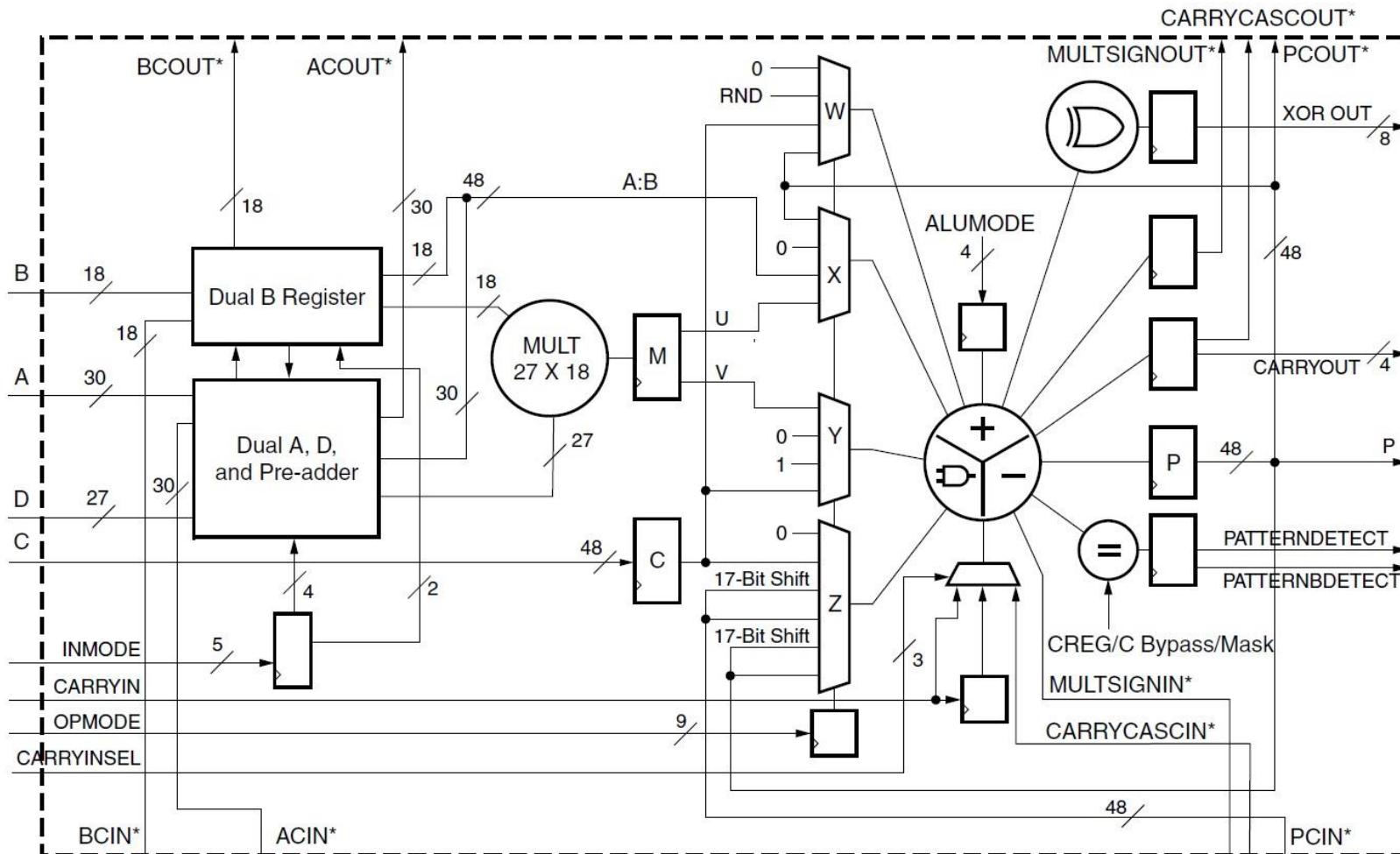


Figure 2-3: Diagram of SLICEM

http://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf

FPGA inside – DSP Slice (Xilinx DSP48E2)



*These signals are dedicated routing paths internal to the DSP48E2 column. They are not accessible via general-purpose routing resources.

UG579_c2_01_093013

Figure 2-1: Detailed DSP48E2 Functionality

http://www.xilinx.com/support/documentation/user_guides/ug579-ultrascale-dsp.pdf

Tools – Xilinx ISE Webpack (FREE)

XILINX ALL PROGRAMMABLE. APPLICATIONS PRODUCTS DEVELOPER ZONE SUPPORT ABOUT

Home - Products - Developer Zone - ISE Design Suite - ISE WebPACK Design Software

ISE WebPACK Design Software

ISE® WebPACK™ design software is the industry's only FREE, fully featured front-to-back FPGA design solution for Linux, Windows XP, and Windows 7. ISE WebPACK is the ideal downloadable solution for FPGA and CPLD design offering HDL synthesis and simulation, implementation, device fitting, and JTAG programming. ISE WebPACK delivers a complete, front-to-back design flow providing instant access to the ISE features and functionality at no cost. Xilinx has created a solution that allows convenient productivity by providing a design solution that is always up to date with error-free downloading and single file installation.

And new in ISE Design Suite 14 - WebPACK now supports embedded processing design for the Zynq®-7000 All Programmable SoC for the Z-7010, Z-7020, and Z-7030.

Download ISE WebPACK Now!

- Download ISE WebPACK software for Windows and Linux.

Key Features

- A free, downloadable PLD design environment for both Microsoft Windows and Linux!
- Embedded processing design support for the Zynq-7000 All Programmable SoC family the Z-7010, Z-7020, and Z-7030
- The industry's fastest timing closure with Xilinx SmartCompile technology
- Complete, front-to-back design environment, including the Xilinx CORE Generator™ system and the full PlanAhead design and analysis tool — with new RTL to Bitstream design flow for Logic Designers!
- Integrated HDL verification with the Lite version of the ISE Simulator (ISim)
- The easiest, lowest cost way to get started with the industry leader for productivity, performance, and power
- Easily upgrade able to any of the ISE Design Suite Editions from the Xilinx Online Store.

ISE Design Features Table

Features	ISE WebPACK
System Generator for DSP	
Platform Studio	(Device locked to three smallest Zynq devices)
Software Development Kit	✓
MicroBlaze Soft Processor	
MicroBlaze Microcontroller System	✓
Design Preservation	✓
Project Navigator	✓
CORE Generator	✓

Quick Links

- Free Evaluation
- Support and Documentation
- Supported Targeted Reference Designs
- IP Center
- Licensing Solutions

> More

Key Documentation

- ISE Design Suite Manuals
- ISE Design Suite Product Brief
- ISE Design Suite Product Table

Tools – Xilinx ISE

The screenshot shows the Xilinx ISE Project Navigator interface. The project is named "a2d_graph" and is set up for the "xc3s250e-4vq100" target device. The "Design Overview" pane on the left lists various reports and properties. The "Project Status" pane in the center displays the following information:

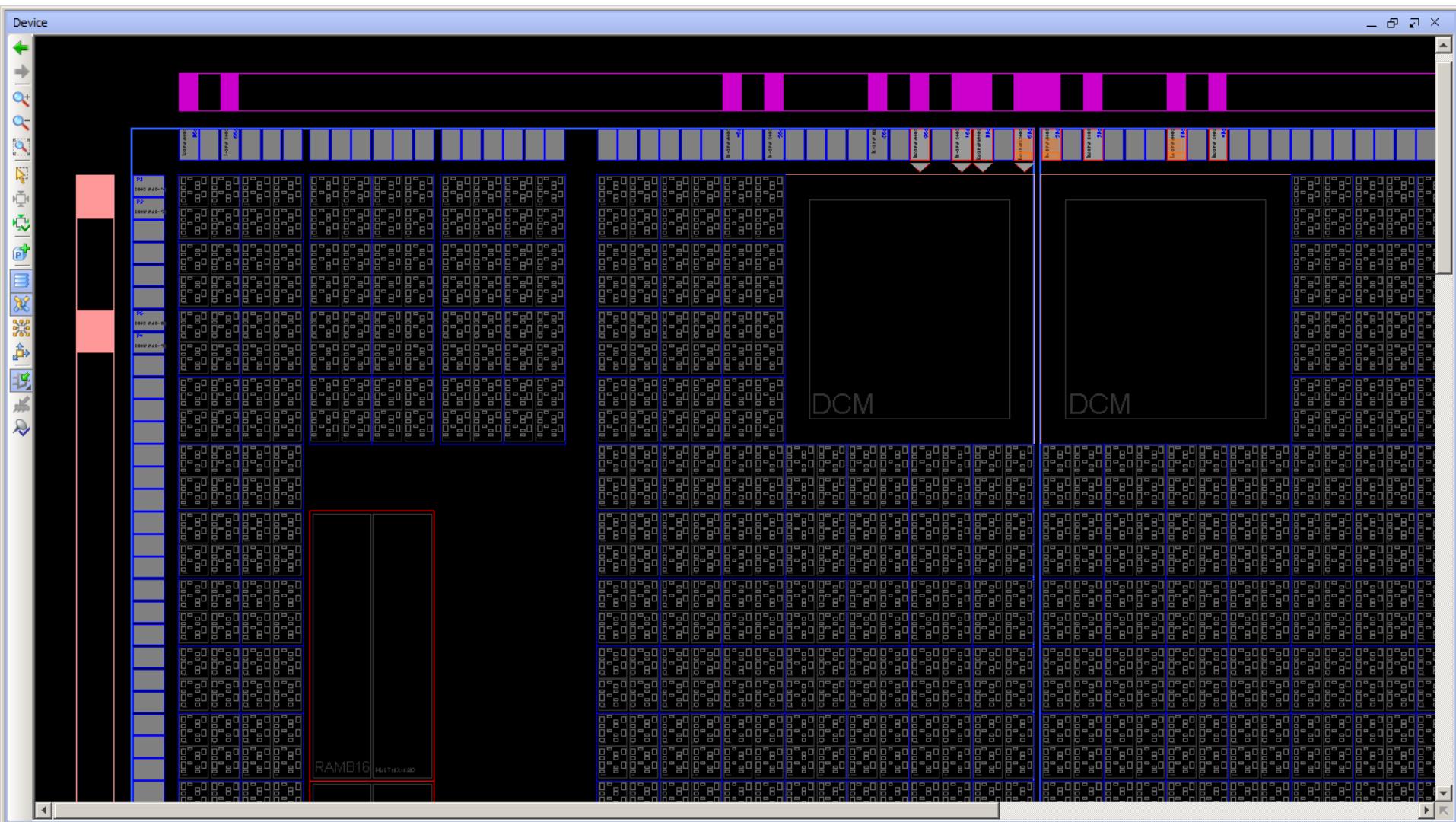
AtoD_graph Project Status (09/15/2015 - 18:04:56)			
Project File:	a2d_graph.xise	Parser Errors:	No Errors
Module Name:	AtoD_graph	Implementation State:	Programming File Generated
Target Device:	xc3s250e-4vq100	• Errors:	No Errors
Product Version:	ISE 14.7	• Warnings:	20 Warnings (20 new)
Design Goal:	Balanced	• Routing Results:	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	All Constraints Met
Environment:	System Settings	• Final Timing Score:	0 (Timing Report)

The "Device Utilization Summary" table below provides details on logic usage:

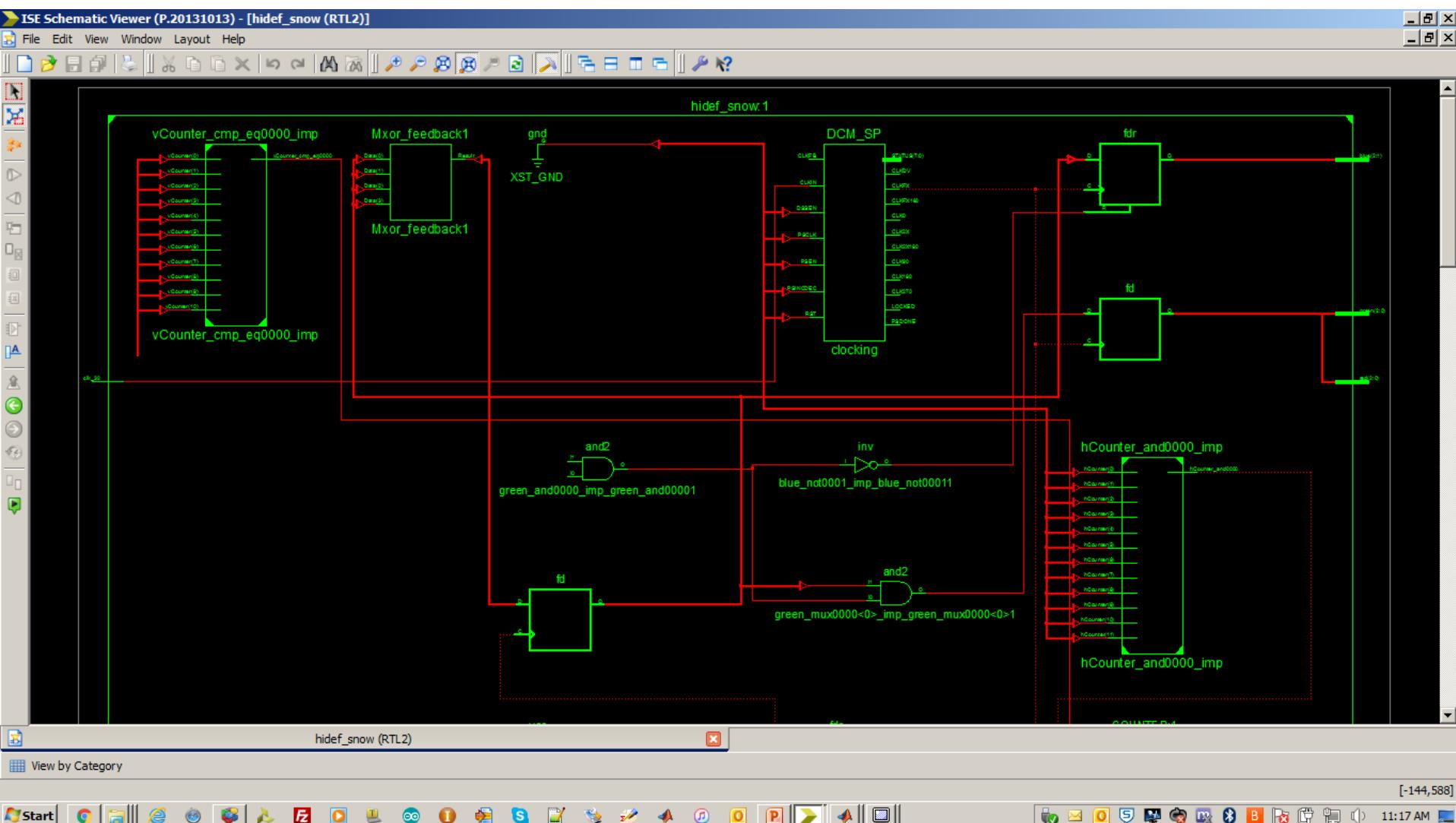
Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	105	4,896	2%	
Number of 4 input LUTs	59	4,896	1%	
Number of occupied Slices	93	2,448	3%	
Number of Slices containing only related logic	93	93	100%	
Number of Slices containing unrelated logic	0	93	0%	
Total Number of 4 input LUTs	109	4,896	2%	
Number used as logic	59			
Number used as a route-thru	50			
Number of bonded IOBs	15	66	22%	
Number of RAMB16s	1	12	8%	
Number of BUFGMUXs	2	24	8%	

The "Console" window at the bottom shows a message indicating a successful WebTalk report sent to Xilinx.

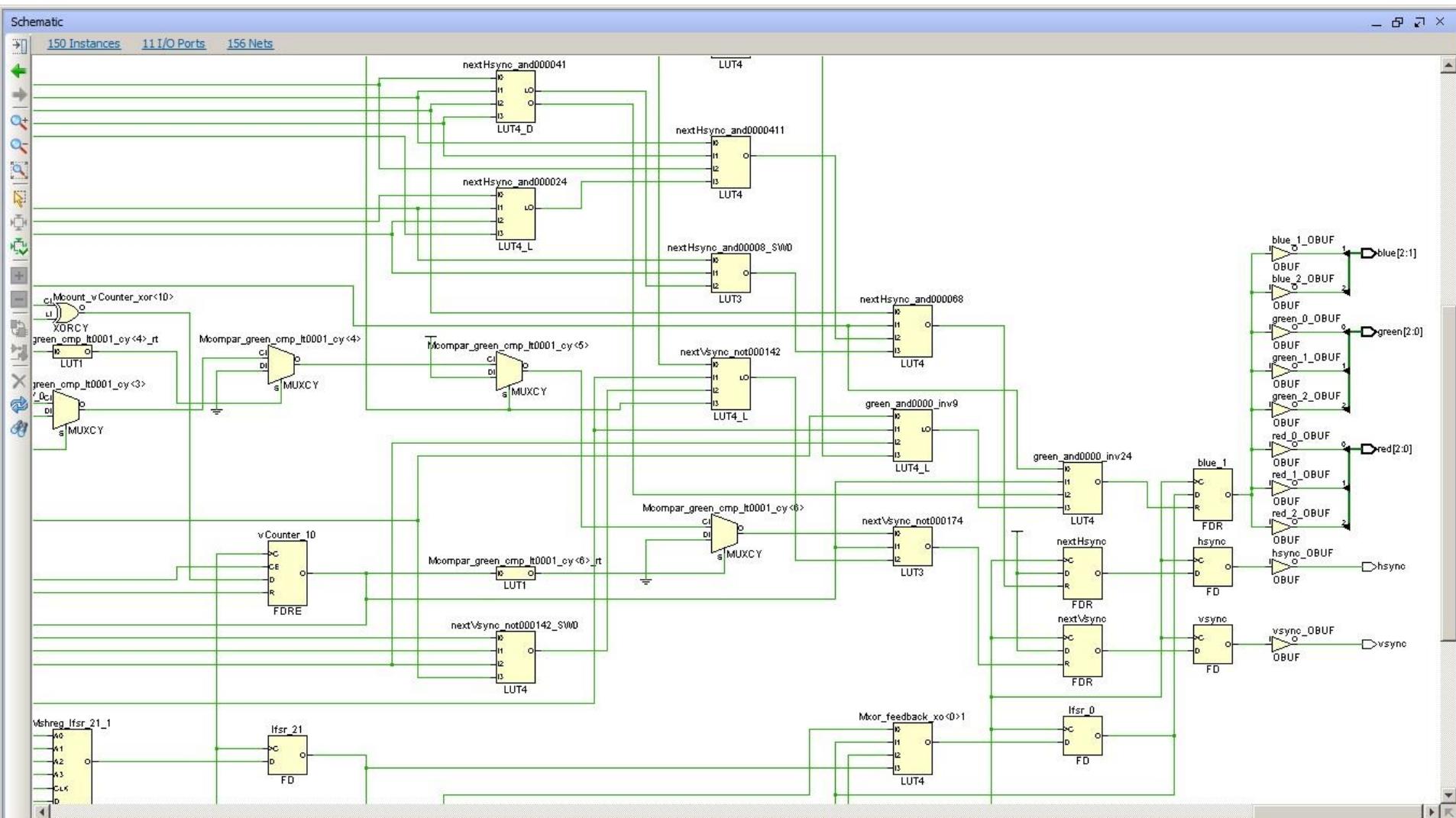
Tools – Xilinx ISE PlanAhead Device



Tools – Xilinx ISE (RTL Schematic)



Tools – Xilinx ISE (Technology Schematic)



Tools – Xilinx Vivado

The screenshot shows the Xilinx Vivado Design Suite homepage. At the top, there's a navigation bar with links for XILINX, APPLICATIONS, PRODUCTS, DEVELOPER ZONE, SUPPORT, ABOUT, and a search icon. Below the navigation bar, the page title "Vivado Design Suite" is displayed, followed by the tagline "Productivity. Multiplied." To the right, there are four download options: "Download Vivado Design Suite", "30-day Vivado Design Suite Evaluation", "Download WebPACK Edition", and "ISE Design Suite". Below these options are five menu items: "Overview" (selected), "What's New", "Documentation", "Download / Buy", and "Training & Support".

The Industry's First SoC-Strength Design Suite

The Vivado® Design Suite delivers a SoC-strength, IP-centric and system-centric, next generation development environment that has been built from the ground up to address the productivity bottlenecks in system-level integration and implementation. The Vivado Design suite is a Generation Ahead in overall productivity, ease-of-use, and system level integration capabilities.

Vivado supports the following devices families: Ultrascale, Virtex-7, Kintex-7, Artix-7, and Zynq -7000

Accelerating Design and Integration

- Software-defined IP Generation with Vivado High-Level Synthesis
- Model-based DSP Design Integration with System Generator for DSP
- Block-based IP Integration with Vivado IP Integrator

Accelerating Verification

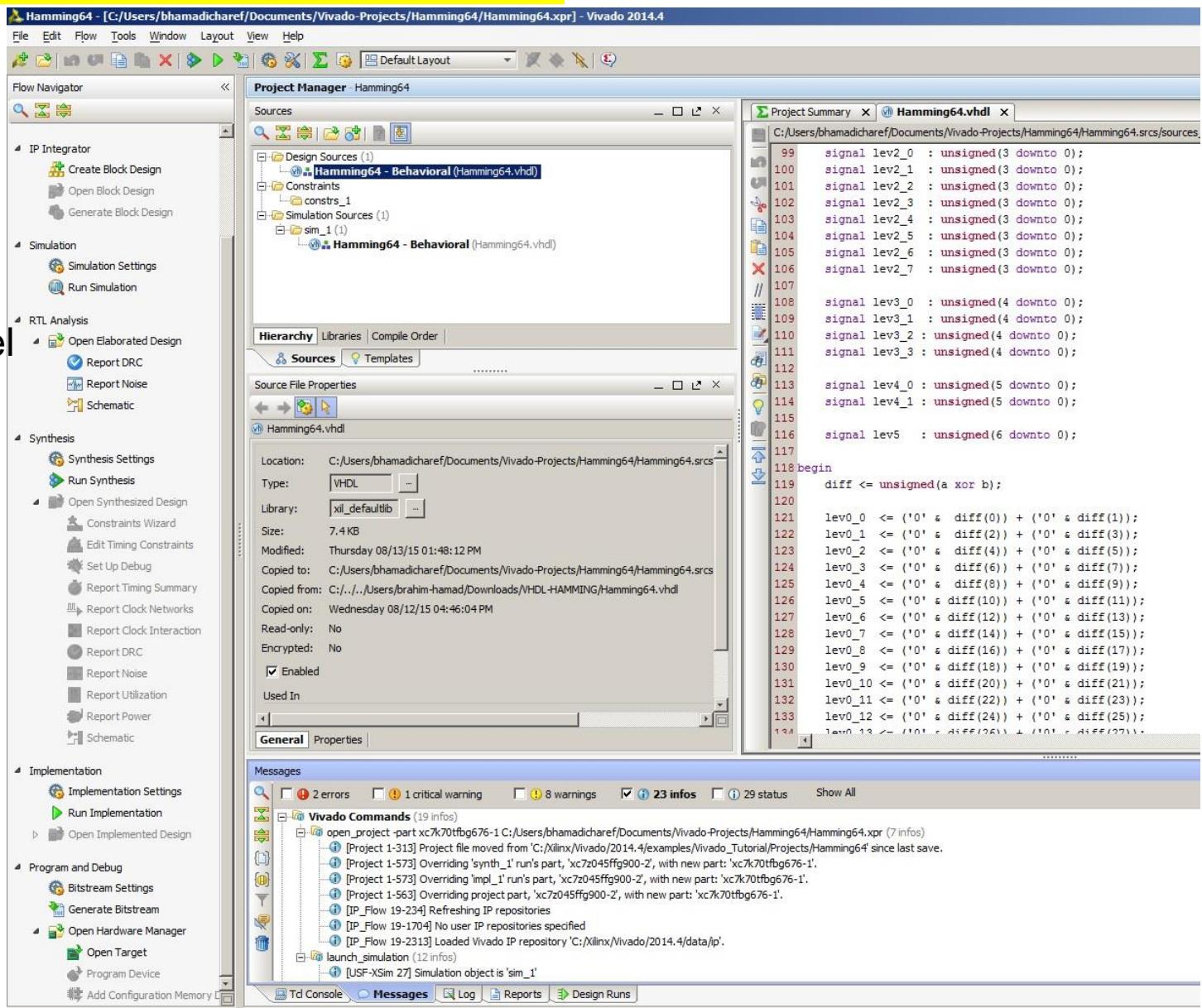
- Vivado Logic Simulation
- Integrated Mixed Language Simulator
- Integrated & Standalone Programming and Debug Environments
- Accelerate Verification by >100X with C, C++ or SystemC with Vivado HLS

Accelerating Implementation

- 4X Faster Implementation
- 20% Better Design Density
- Up to 3-Speedgrade Performance Advantage for the low-end & mid-range and 35% Power Advantage in the high-end

Tools – Xilinx Vivado

RTL is
Register-transfer level

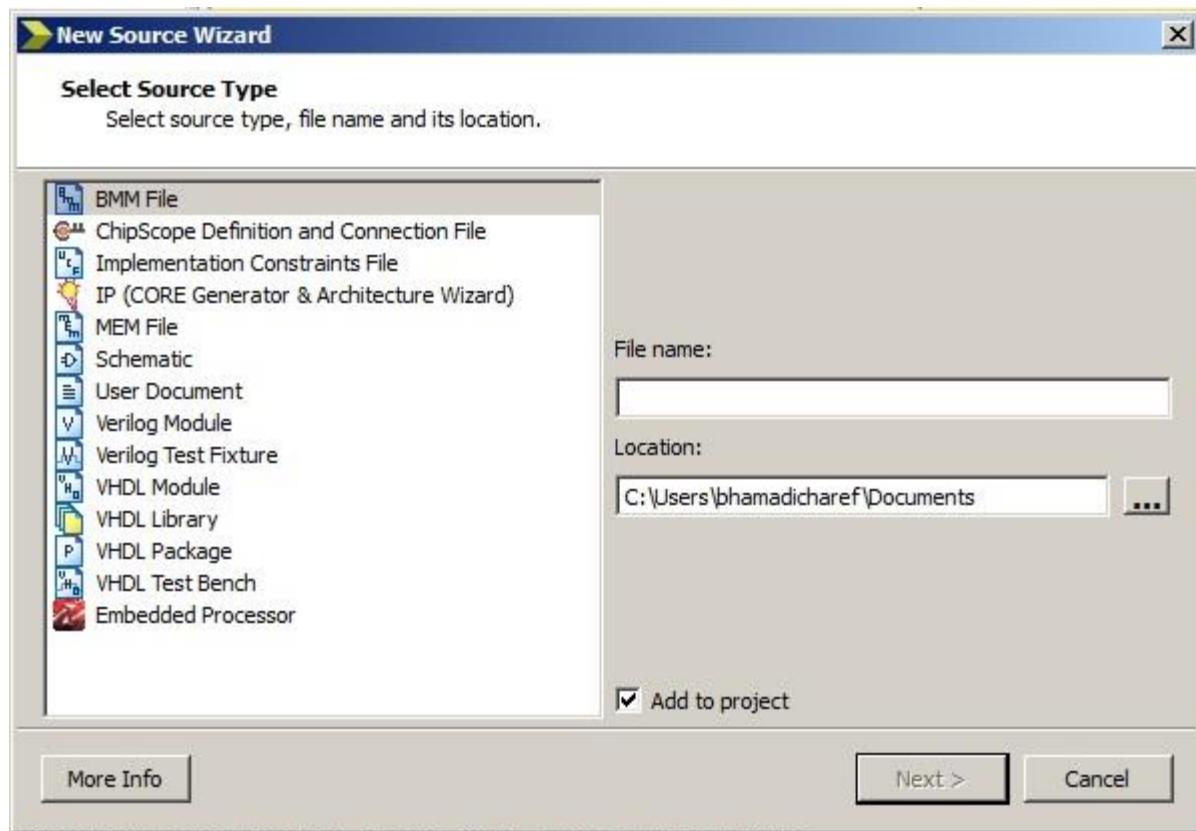


VHDL – Source files

Languages

- Verilog
- VHDL
- mix of both

- IP cores
- Source
- Test bench
- Library
- Package



Source VHDL (.vhd), source Verilog (.v)
Constraints file (.ucf) i.e. pin definitions
Memory content (.coe)

FPGA – Demo Hello World

```
## User Constraint File (UCF)
## Use the following constraints on a Papilio One
NET "clk" LOC = "P89" | IOSTANDARD = LVCMOS25 | PERIOD = 31.25ns;
NET "mystery" LOC = "P90" | IOSTANDARD = LVCMOS25 | DRIVE = 4 | SLEW = SLOW;

-----
library ieee;

use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity mystery is
Port (      clk : in STD_LOGIC;
            mystery : out STD_LOGIC);
end mystery;

architecture Behavioral of mystery is
signal counter : unsigned ( 11 downto 0 ) := (others => '0');
signal shift : std_logic_vector(135 downto 0) := x"FC255346D1B5ED025D57B49D1B44D584A1";

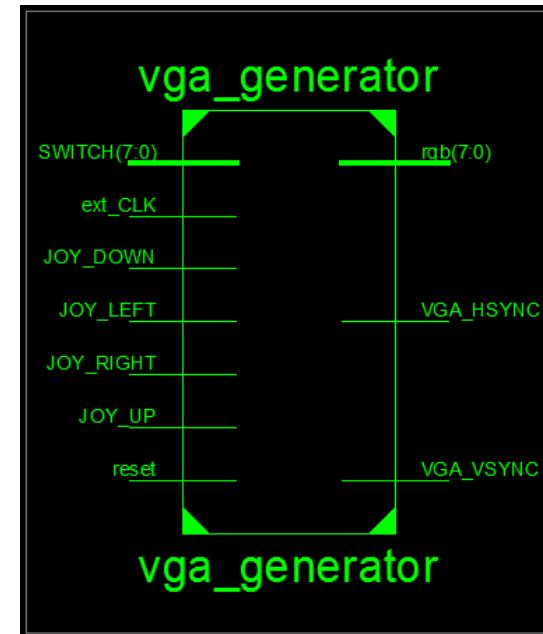
begin
mystery <= shift(135);
process(clk)
begin
  if rising_edge(clk) then
    if counter = 3332 then
      counter <= (others => '0');
      shift <= shift(134 downto 0) & shift(135);
    else
      counter <= counter + 1;
    end if;
  end if;
end process;
end Behavioral;
```

GadgetFactory.net
VHDL Puzzle Contest

[http://forum.gadgetfactory.net/index.php
?topic/1386-vhdl-puzzle-contest-over-
150-in-prizes/page-3](http://forum.gadgetfactory.net/index.php?topic/1386-vhdl-puzzle-contest-over-150-in-prizes/page-3)

FPGA – Demonstrations

- **VHDL-HELLO-WORLD**
Serial port output Hello World
- **VHDL-NIGHTRIDER**
NightRider LEDs horizontal motion
- **VHDL-LEDS**
LEDs on/off from switches
- **VHDL-DIGITALCLOCK**
4-digits 7-segment LEDs minutes + seconds
- **VHDL-STEPPER**
4-pins output driver stepper motor
- **VHDL-VGA-COLORS**
VGA ouput, switch select color RGB332
- **VHDL-VGA-SQUARE**
VGA ouput, square color RGB332
move position using joystick
change color on borders



FPGA – eBooks



PONG P. CHU

HDL Chip Design

A practical guide for designing, synthesizing and simulating ASICs and FPGAs using VHDL or Verilog

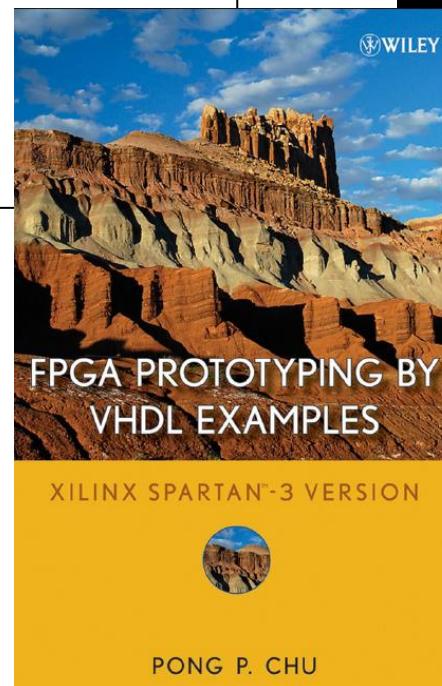
Douglas J Smith

Foreword by Alex Zamfirescu

Doone
Publications



28



PONG P. CHU

FREE RANGE VHDL



The no-frills
guide to writing
powerful code
for your digital
implementations

BRYAN MEALY
FABRIZIO TAPERO

erangefactory.org

FPGA – Online Sources of Knowledge

- **FPGA Projects** (Mike Field) with HDMI, ESP8266, Stepper, Fractal, SD card, Filtering, SPDIF, VGA, Synth, DisplayPort, etc.
http://hamsterworks.co.nz/mediawiki/index.php/FPGA_Projects
- **OpenCores.org** (Arithmetic core, Prototype board, Communication controller Coprocessor, Crypto core, DSP core, ECC core, Library, Memory core, Processor, System on Chip, System on Module, System controller, Testing / Verification, Video controller)
<http://opencores.org>
- **Digital Design and Embedded Programming**
PLD, SPLD, GAL, CPLD, FPGA Design
<http://www.edaboard.com/forum30.html>
- **StackOverflow** (vhdl, fpga, verilog, xilinx-ise, hdl, vivado, spartan, synthesis, zynq, etc.)
<http://stackoverflow.com/questions/tagged/xilinx>
- **FPGAfund.com**
<http://www.fpga4fun.com>

FPGA – Literature Xilinx

- **Xilinx FPGA literature** (DS - data sheet, UG - User Guide, XAPP - Application Note, WP - White papers)
<http://www.xilinx.com/support.html#documentation>
- Xilinx User Community Forums
<https://forums.xilinx.com>
- Xcell Daily Blog
<https://forums.xilinx.com/t5/Xcell-Daily-Blog/bg-p/Xcell>
- Xcell Journal (Quaterly)
www.xilinx.com/about/xcell-publications/xcell-journal.html
- Digilent Inc Support (schematics)
<https://www.digilentinc.com/Support/Support.cfm>

FPGA – Research Topics

- FPGA topics
 - Systolic architecture
 - Pipelined design, timing closure
 - Bit-Serial Digital Signal Processing
 - Sigma-Delta and One-bit DAC
 - Block and Distributed RAM
 - Bipartite tables (LUT)
 - Tree adder
 - Montgomery modular multiplication
 - Vedic Multiplier
 - CORDIC
 - FFT, DCT, FIR, IIR, DWT
 - High-Level Synthesis
 - Clock Domain Crossing (CDC)
 - Hardware Trojan Insertion

IEEE Transactions on Circuits and Systems I: Regular Papers

IEEE Transactions on Circuits and Systems II: Express Briefs.

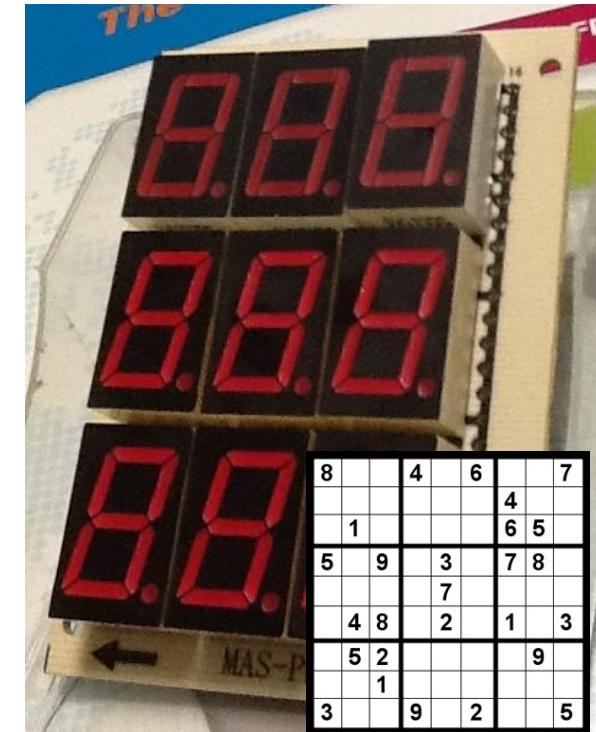
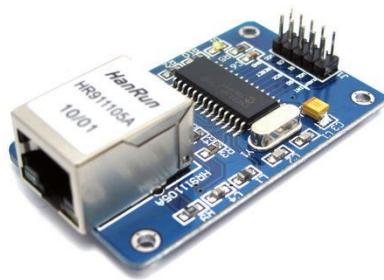
IEEE Transactions on Computers

Publish or Perish

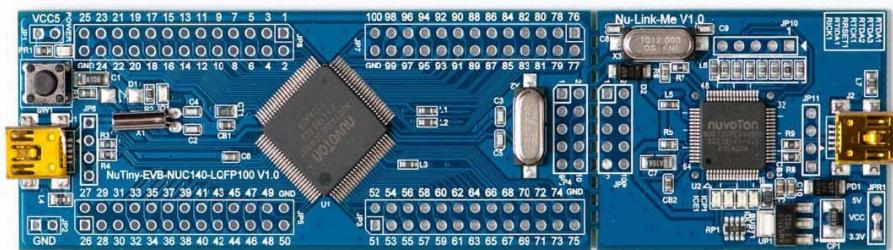
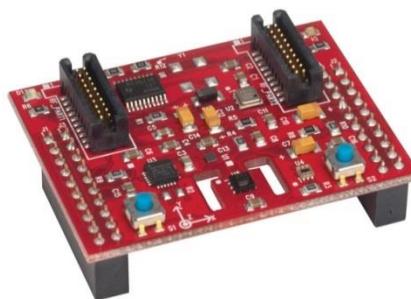
<http://www.harzing.com>

FPGA - Let's get started ...

SPI



pulse



I2S



FPGA – Waveguide Modeling

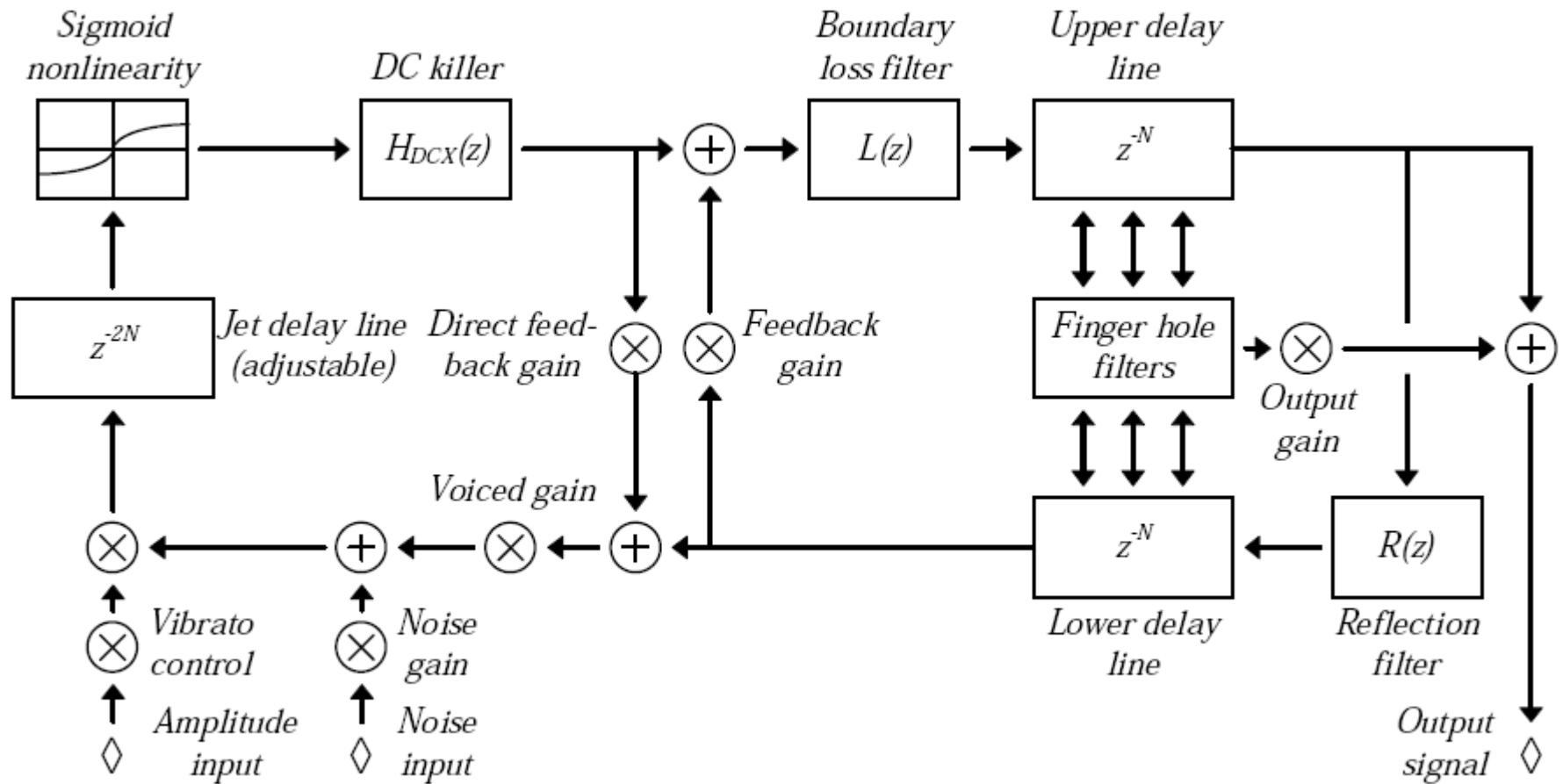
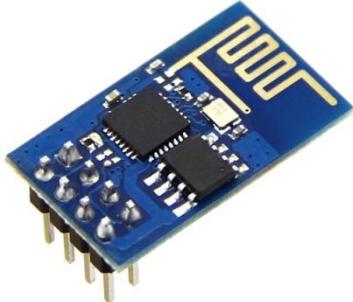


Figure 1: Block diagram of the new waveguide flute model

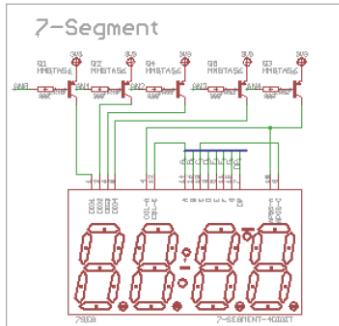
FPGA - WIFI 4D Lottery

- Web query from Singapore lottery results of last draw
 - 1st, 2nd and 3rd prices
 - Starter and consolidation prices (10x 4-digits)



A black right-pointing arrow icon.

http://hamsterworks.co.nz/mediawiki/index.php/FPGA_ESP8266



Sat, 24 Oct 2015 (Draw at 6.30pm)		Wed, 21 Oct 2015		Sun, 18 Oct 2015		Draw No. 3862		Draw No. 3861		Sat, 17 Oct 2015		Draw No. 3860	
1st Prize	9512	1st Prize	5787	1st Prize	7512	1st Prize	5787	1st Prize	7512	1st Prize	5787	1st Prize	7512
2nd Prize	1278	2nd Prize	2124	2nd Prize	4949	2nd Prize	2124	2nd Prize	4949	2nd Prize	2124	2nd Prize	4949
3rd Prize	3342	3rd Prize	1052	3rd Prize	5959	3rd Prize	1052	3rd Prize	5959	3rd Prize	1052	3rd Prize	5959
Starter Prizes		Starter Prizes		Starter Prizes		Starter Prizes		Starter Prizes		Starter Prizes		Starter Prizes	
0179	2445	0622	0767	0178	0567	0622	0767	0178	0567	0622	0767	0178	0567
2557	3947	1699	3108	5021	7638	1699	3108	5021	7638	1699	3108	5021	7638
5163	5814	6167	6293	7665	8318	6167	6293	7665	8318	6167	6293	7665	8318
6876	7109	6636	7243	8751	8883	6636	7243	8751	8883	6636	7243	8751	8883
7303	7428	8306	9217	9353	9690	8306	9217	9353	9690	8306	9217	9353	9690
Consolation Prizes		Consolation Prizes		Consolation Prizes		Consolation Prizes		Consolation Prizes		Consolation Prizes		Consolation Prizes	
1351	1704	0423	0746	0650	1335	0423	0746	0650	1335	0423	0746	0650	1335
2724	3296	1050	3140	2155	3696	1050	3140	2155	3696	1050	3140	2155	3696
3667	3787	3942	4310	3736	4110	3942	4310	3736	4110	3942	4310	3736	4110
4008	5107	4879	8031	6760	6886	4879	8031	6760	6886	4879	8031	6760	6886
8291	9281	9267	9341	9403	9839	9267	9341	9403	9839	9267	9341	9403	9839

FPGA - Large multiplication

- **Shakuntala Devi**

- Mental calculation

- June 18 , 1980, she demonstrated
Multiplication of two 13-digit numbers

$7,686,369,774,870 \times 2,465,099,745,779$

(picked at random by the Computer
Department of Imperial College, London)

She correctly answered

18,947,668,177,995,426,462,773,730

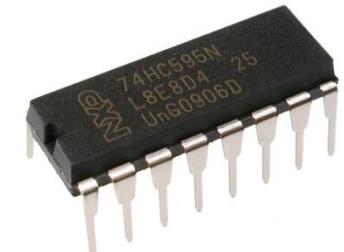
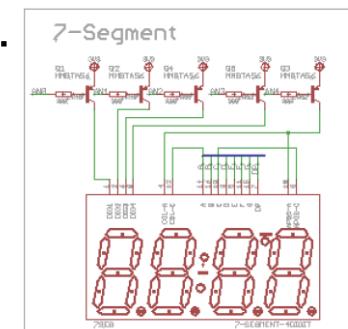
In 28 seconds !!



- How one do this with an FPGA (vs Arduino for e.g.)

- large data, data representation, large multiplication, Binary-to-BCD

- 48 I/O lines to drive XX-digits display ..



FPGA - Binary-to-BCD Conversion

688

IEEE TRANSACTIONS ON COMPUTERS, JULY 1977

Correspondence

Developing Large Binary to BCD Conversion Structures

MICHAEL BENEDEK

Abstract—Fast binary to BCD conversions consist of static implementation of a sequential algorithm and are limited to relatively small number of bits as the hardware increases rapidly with each added bit.

The present technique shows a geometrical expansion scheme of the Couleur algorithm [1] using static decoders and can be extended by similarity to very large number of bits. Groups of decoders with identical I/O pattern can be merged to reduce hardware complexity. The method will be demonstrated to 30 binary bits including programming of the large decoders. Furthermore, a hybrid (static/sequential) technique will be shown for applications where less than maximum conversion speed can be tolerated with substantial savings in hardware. Finally a reverse (BCD-binary) conversion is done by successive approximation.

Index Terms—Binary-BCD conversion, hybrid, static.

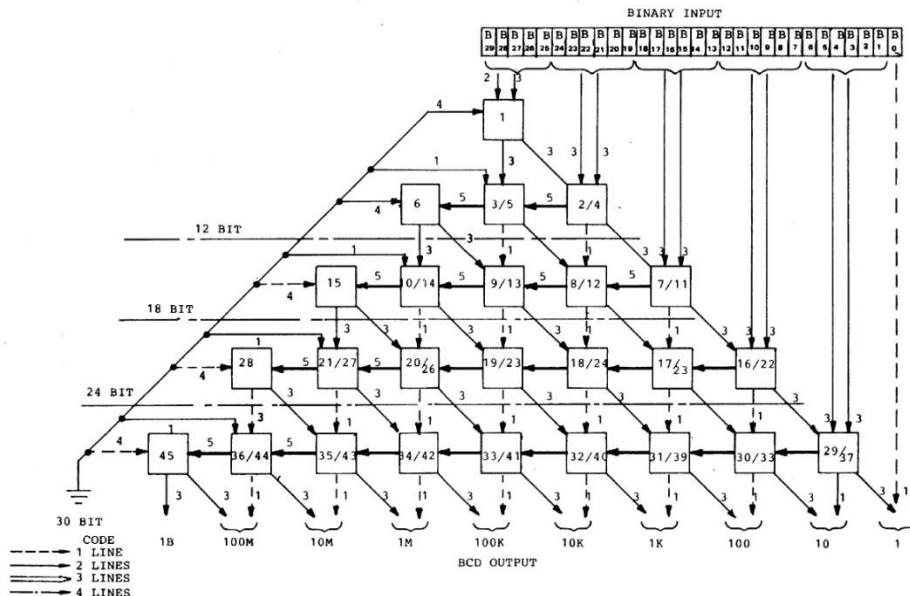


Fig. 5. BIN/BCD conversion map with merged (E9) decoding blocks. Decoder numbers referenced to Fig. 2.

FPGA - Moving LEDs

```
hide code 2 | compiled successfully | fork | parent | diff |  
  
1 #ifdef GL_ES  
2 precision highp float;  
3 #endif  
4  
5 uniform float time;  
6 uniform vec2 resolution;  
7  
8 float rand(vec2 co){  
9     return fract(sin(dot(co.xy, vec2(12.9898, 78.233))) * 43758.5453);  
10 }  
11  
12 void main (void) {  
13     // Divide the coordinates into a grid of squares  
14     vec2 v = gl_FragCoord.xy / 10.0;   
15     // Calculate a pseudo-random brightness value for each square  
16     float brightness = fract(rand(floor(vec2(v.x+floor(sin(time*2.)*5.),  
17                                     v.y+floor(cos(time*2.)*5.)))) + time);  
18     // Reduce brightness in pixels away from the square center  
19     brightness *= 0.5 - length(fract(v) - vec2(0.5, 0.5));   
20     gl_FragColor = vec4(brightness * v.x / resolution.x * 100.,  
21                         brightness * v.y / resolution.y * 100. * (cos(time)+1.),  
22                         brightness * v.y / resolution.y * 100.,  
23                         1.0);  
24 }
```

FPGA – Shades of Colors

hide code

2

compiled successfully

parent

diff

fullscreen

gallery

```
1 #ifdef GL_ES
2 precision mediump float;
3 #endif
4
5 uniform float time;
6 uniform vec2 mouse;
7 uniform vec2 resolution;
8
9 void main( void ) {
10
11     gl_FragColor = vec4(
12         gl_FragCoord.x / resolution.x,
13         gl_FragCoord.y / resolution.y,
14         0.5 * (1.0 - gl_FragCoord.x / resolution.x),
15         1.0) ;
16
17 }
```

<http://glslsandbox.com/e#11168.0>

FPGA – Yellow Swirl

hide code 2 compiled successfully parent diff

```
1 #ifdef GL_ES
2 precision mediump float;
3 #endif
4 // #I need Smoke and mirros !! anyone can help with the design? -Harley.
5 uniform float time;
6 uniform vec2 resolution;
7
8 #define N 6
9 //nice coffee ... kind of
10 void main( void ) {
11     vec2 v=(gl_FragCoord.xy-(resolution*0.5))/min(resolution.y,resolution.x)*10.0;
12     float t=time * 0.4,r=2.0;
13     for (int i=1;i<N;i++){
14         float d=(3.14159265 / float(N))*(float(i)+14.0);
15         r+=length(vec2(v.y,v.x))+1.21;
16         v = vec2(v.x+cos(r+sin(r)-d)+cos(t),v.y+cos(r+sin(r)+d)+sin(t));
17     }
18     r = (sin(r*0.05)*0.5)+0.5;
19     r = pow(r, 30.0);
20     gl_FragColor = vec4(r,pow(max(r-0.75,0.0)*4.0,2.0),pow(max(r-1.875,0.1)*5.0,4.0), 1.0 );
21 }
22 }
```

FPGA – FM Synthesis



United States Patent [19]
Peers, Jr.

[11] Patent Number: 5,581,045
[45] Date of Patent: Dec. 3, 1996

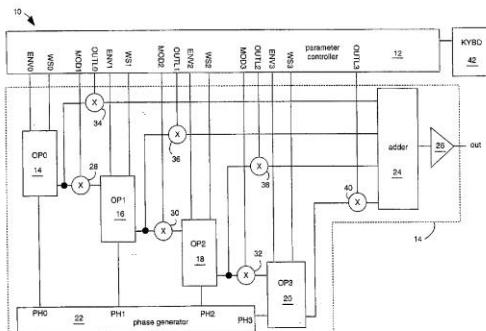
- | | | | | |
|------|---|---|--|--|
| [54] | METHOD AND INTEGRATED CIRCUIT FOR THE FLEXIBLE COMBINATION OF FOUR OPERATORS IN SOUND SYNTHESIS | 4,942,799
7/1990 Suzuki | 84/603
S05,202 6/1990 Oya | 84/603
S05,202 6/1990 Oya |
| [75] | Inventor: Roi N. Peers, Jr. , Livermore, Calif. | 5,033,252
7/1991 Kellogg | 84/658
S07,613 12/1991 Toda | 84/658
S07,613 12/1991 Toda |
| [73] | Assignee: ESS Technology, Inc. , Fremont, Calif. | 5,094,136
3/1992 Kado | 84/603
S13,892 3/1992 Ohya | 84/603
S13,892 3/1992 Ohya |
| [21] | Appl. No.: 305,099 | 5,182,415
1/1993 Kanimoto | 84/604
S19,161 5/1993 Oya | 84/604
S19,161 5/1993 Oya |
| [22] | Filed: Sep. 13, 1994 | 5,198,604
3/1993 Higashi et al. | 84/266
S218,156 6/1993 Izukawa | 84/266
S218,156 6/1993 Izukawa |
| [51] | Int. Cl. ⁵ G10H 5/00 | 5,357,575
10/1994 Kitamura | 84/625 X
S35,757 10/1994 Kitamura | 84/625 X
S35,757 10/1994 Kitamura |
| [52] | U.S. Cl. 84/647; 84/660; 84/663 | | | |
| [58] | Field of Search 84/625, 647, 648,
84/658-660, 663, 675-677, 697, 698 | | | |

References Cited

U.S. PATENT DOCUMENTS

- | | | | | |
|-----------|---------|-------------|-------|----------|
| 4,018,121 | 4/1977 | Chowning | | 84/659 X |
| 4,132,241 | 11/1979 | Nimis | | |
| 4,173,164 | 11/1979 | Adachi | | |
| 4,249,447 | 2/1981 | Tomisawa | | |
| 4,253,367 | 1/1981 | Hiyoshi | | |
| 4,297,933 | 11/1981 | Nishimoto | | |
| 4,301,704 | 11/1981 | Nagai | | |
| 4,406,204 | 9/1983 | Katoh | | |
| 4,422,321 | 12/1983 | Chibana | | |
| 4,459,513 | 6/1984 | Yamada | | |
| 4,553,101 | 11/1984 | Nishimoto | | |
| 4,616,546 | 12/1984 | Uchiyama | | |
| 4,636,036 | 2/1987 | Ozono | | |
| 4,644,839 | 2/1987 | Nishimoto | | |
| 4,655,115 | 4/1987 | Nishimoto | | |
| 4,773,332 | 5/1988 | Uchiyama | | |
| 4,788,888 | 6/1988 | Oyaa | | |
| 4,766,795 | 8/1988 | Takemoto | | |
| 4,788,896 | 12/1988 | Uchiyama | | |
| 4,813,326 | 3/1989 | Hirano | | |
| 4,875,100 | 10/1989 | Kuda et al. | | 84/626 |

7 Claims, 2 Drawing Sheets



United States Patent [19]

[11] Patent Number: 5,578,779
[45] Date of Patent: Nov. 26, 1996

- | | | | | | |
|---|--|-----------|----------|-----------------|--------|
| [54] | METHOD AND INTEGRATED CIRCUIT FOR ELECTRONIC WAVEFORM GENERATION OF VOICED AUDIO TONES | 4,813,326 | 3/19/90 | Hirano et al. | 846243 |
| [75] | Inventor: Roi N. Peers, Jr., Livermore, Calif. | 4,875,400 | 10/27/90 | Oka et al. | 846244 |
| [73] | Assignee: ESS Technology, Inc., Fremont, Calif. | 4,942,709 | 7/19/90 | Kondo et al. | 846245 |
| [21] | Appl. No.: 305,087 | 5,033,382 | 8/19/91 | Kellie et al. | 846246 |
| [22] | Filed: Sep. 13, 1994 | 5,076,133 | 11/19/91 | Toda et al. | 846247 |
| [51] | Int. Cl. 6 G10H 5/00 | 5,094,336 | 3/19/92 | Kudo et al. | 846248 |
| [52] | U.S. Cl. 844/67 | 5,138,924 | 8/19/92 | Ohyama et al. | 846249 |
| [58] | Field of Search 844/67-649, 658, 660-662, 675, 675-677, 697, 698 | 5,218,156 | 6/1993 | Ikusaka et al. | 846242 |
| | | 5,235,553 | 6/1993 | Kumimoto et al. | 846243 |
| Primary Examiner—William M. Shoop, Jr. | | | | | |
| Assistant Examiner—Jeffrey W. Donels | | | | | |
| Attorney, Agent, or Firm—Law Offices of Thomas E. Schatz, P.C., San Francisco, CA | | | | | |
| U.S. Patent and Trademark Office | | | | | |

References Cited

U.S. PATENT DOCUMENTS

- 13-232,141 1/1979 Niimi
4,173,164 11/1979 Adachi
249,447 2/1981 Tomisawa
4,253,367 3/1981 Hiroyoshi
2,297,933 11/1981 Nishimoto
3,401,704 11/1982 Nagai
4,406,204 9/1983 Kach
4,422,604 10/1983 Chiba
4,422,605 10/1983 Iwamura
4,554,857 11/1985 Nishimoto
6,164,546 10/1988 Uchiyama
4,633,066 2/1987 Oya
4,644,839 4/1987 Nishimoto
4,655,115 4/1987 Nishimoto
4,747,333 6/1988 Uchiyama
4,747,334 6/1988 Oya
4,766,785 8/1988 Takeuchi
4,788,896 10/1988 Uchiyama
designed to produce various tones
calculated for a next waveform. A series approximation
of a desired complex sound waveform is achieved by
calculating the contributions of twenty-nine time steps back
in time. Twenty-nine different address phases are respec-
tively applied to twenty-nine stacked arithmetic units. Each
arithmetic unit comprises a first adder that inputs the output
of a previous arithmetic unit and the input of the previous
arithmetic unit. A second adder inputs the result from the
first adder and one of the twenty-nine address phases. The
second adder then reads a waveform generator connected to
a multiplier that is controlled by a common multiplication
factor "B". The output of the twenty-nine unit produces the
desired tone without any of the stacked units feeding back
any signals.

14 Claims, 3 Drawing Sheets

