4 BIT MULTIPLIER

Aditya Sundararajan, Sagar Bhanushali, Dishant Gala, Rahul Ghuge

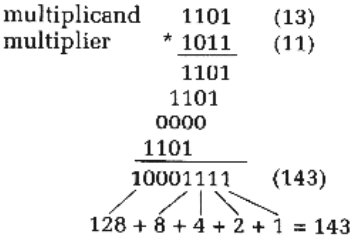
April 23, 2012

1 Abstract

A combinational multiplier is a good example of how simple logic functions (gates, half adders and full adders) can be combined to construct a much more complex function. In particular, it is possible to construct a 4x4 combinational multiplier from an array of AND gates, half-adders and full-adders, taking what you have learned recently and extending it to a more complex circuit. The purpose of this document is to introduce how a relatively complex arithmetic function, such as binary multiplication, can be realized using simple logic building blocks.

2 Introduction

It is useful to consider how binary multiplication can be performed. Consider the following binary multiplication of two positive 4-bit integer values.



In the course of multiplying two binary numbers, each bit in the multiplier is multiplied with the multiplicand. Each of the four products is aligned (shifted left) according to the position of the bit in the multiplier that is being multiplied with the multiplicand. The four resulting products are added to form the final product.

Since we are dealing with binary numbers, forming the products is particularly easy. If the multiplier bit is a 1, then the corresponding product is simply an appropriately shifted copy of the multiplicand. If the multiplier bit is a zero, then the product is zero. 1-bit binary multiplication is thus just an AND operation.

For an N-bit multiplier and multiplicand (an NxN bit product), the result is 2N bits wide. The result of our desired 4x4 bit multiplication is thus an 8-bit result.

3 Analysis

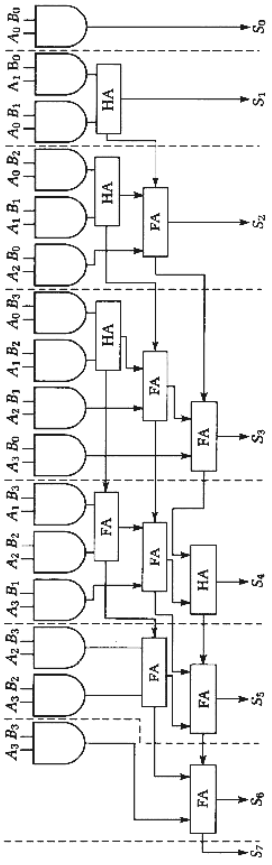
a] Method One:

Consider computing the product of two 4-bit integer numbers given by A(3)A(2)A(1)A(0) (multiplicand) and B(3)B(2)B(1)B(0) (multiplier). Each of the ANDed terms is referred to as a partial product. The final product (the result) is formed by accumulating (summing) down each column of partial products. Any carries must be propagated from the right to the left across the columns.

Since we are dealing with binary numbers, the partial products reduce to simple AND operations between the corresponding bits in the multiplier and multiplicand. The sums down each column can be

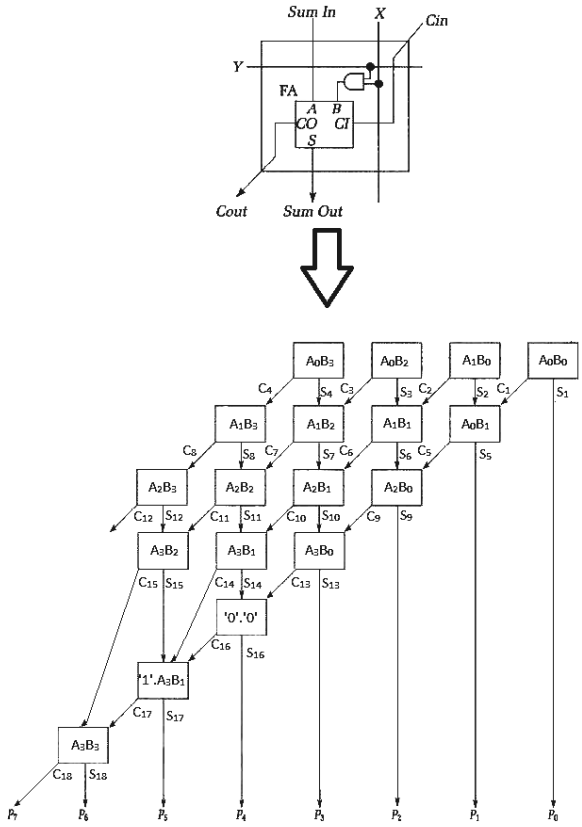
implemented using one or more 1-bit binary adders. Any adder that may need to accept a carry from the right must be a full adder. If there is no possibility of a carry propagating in from the right, then a half adder can be used instead, if desired (a full adder can always be used to implement a half adder if the carry-in is tied low). The diagram below illustrates a combinational circuit for performing the 4x4 binary multiplication.

The initial layer of AND gates forms the sixteen partial products that result from ANDing all combinations of the four multiplier bits with the four multiplicand bits. The column sums are formed using a combination of half and full adders.



The speed of an adder constructed in this manner is a function of how long it takes the carries from the less significant bits on the right to propagate through to the left. The scheme that is shown above is often referred to as ripple carry since each more significant column in the sum must wait for the carries to be computed in the less significant columns before its corresponding sum bit can be computed. Its possible to include carry lookahead circuitry to compute the carries in parallel, thus greatly speeding up the computation at an increase in gate count.

b] Method Two:



A slightly different implementation scheme is shown in the figure below. This multiplier is constructed from an array of building blocks, shown in the first figure below. Each building block consists of an AND gate for computing locally the corresponding partial product (X and Y), an input passed into the block from above (Sum In), and a

carry (Cin) passed from a block diagonally above. It generates a carry out bit (COUT) and a new sum out bit (Sum Out). The second figure illustrates the interconnection of these building blocks to construct a 4x4 combinational multiplier. The Ai values are distributed along block diagonals, and the Bi values are passed along the rows.

4 Applications

1. Generic multiplier for any number of bits.
2. Digital Signal Processing.
3. In Data transmitter and receiver circuits.

5 Logic used in VHDL program

a] Method One:

In this method, each bit of the multiplier is first ANDed with every bit of the multiplicand. The output of this step is stored in signals w, x, y, z, where,
w(i)=A(i) AND B(0);
x(i)=A(i) AND B(1);
y(i)=A(i) AND B(2);
z(i)=A(i) AND B(3);
Here, i = 0 to 3.
Now, the inputs and the signals are 'port mapped' in stages (stage1 to stage12) to full adder program. The outputs of the full adder program are stored in 'fas(j)' and 'fac(j)' signals, where, 'fas(j)' is the sum output and 'fac(j)' is the carry output of the jth stage. Here, j = 1 to 12. The 'fas' and 'fac' from each stage are appropriately substituted in each successive stage as shown in figure for Method One(Analysis - above) and thus the final output is generated for 4-bit multiplier using Method One.

b] Method Two:

In this method, unlike Method One, each bit is not ANDed in the main program itself. Instead, each bit of both multiplicand and multiplied is 'port mapped' in stages (stage1 to stage18) to the program which contains the 'building block' for the Method Two. This 'building block' contains code for ANDing the inputs and then for full adder. The output of this 'building block' are stored in 'S(k)' and 'C(k)' signals, where, 'S(k)' is the sum output and 'C(k)' is the carry output of the kth stage. Here, k = 1 to 18. The 'S' and 'C' from each stage are appropriately substituted in each successive stage as shown in figure for Method Two(Analysis - above) and thus the final output is generated for 4-bit multiplier using Method Two.

Now, to give a choice to the user to choose any of the two methods, a code for 2:1 Multiplexer is written.
If the user chooses the 'select input' as '0', then Method One is selected and the 4-bit Multiplier output is generated using the algorithm for Method One.
If the user chooses the 'select input' as '1', then Method Two is selected and the 4-bit Multiplier output is generated using the algorithm for Method Two.

6 Working

The synthesis (hardware implementation) of the project was done on CPLD using Xilinx IC XC9572/XC95180 Module. The CPLD provides very high clock frequencies (100KHz-5MHz) for high speed applications which can be reduced using clock reduction code to suit the circuit requirements. The inputs and enable pins are assigned switches through which they can be toggled. All the four 7-segment displays are common anode(thus has active low inputs)and the outputs are multiplexed(all four displays show same output). Each display has an active low enable input. Particular segments can be hindered by using the appropriate switches provided alongside the SSD's.

7 About Xilinx

Xilinx designs, develops and markets programmable logic products including integrated circuits (ICs), software design tools, predefined system functions delivered as intellectual property (IP) cores, design services, customer training, field engineering and technical support. Xilinx sells both FPGAs and CPLDs programmable logic devices for electronic equipment manufacturers in end markets such as communications, industrial, consumer, automotive and data processing. Xilinx FPGAs can run a regular embedded OS (such as Linux or vxWorks) and can implement processor peripherals in programmable logic. Xilinx delivers programmable platforms to help design engineers make their vision a reality. Xilinx programmable chips are the innovation platform of choice for today's leading companies for the design of tens of thousands of products that improve the quality of our everyday lives. Xilinx designs, develops and markets programmable logic products including integrated circuits (ICs), software design tools, predefined system functions delivered as intellectual property (IP) cores, design services, customer training, field engineering and technical support. Xilinx sells both FPGAs and CPLDs programmable logic devices for electronic equipment manufacturers in end markets such as communications, industrial, consumer, automotive and data processing. Xilinx's FPGAs have been used for the ALICE (A Large Ion Collider Experiment) at the CERN European laboratory on the French-Swiss border to map and disentangle the trajectories of thousands of subatomic particles. Xilinx has also engaged in a partnership with the United States Air Force Research Laboratories Space Vehicles Directorate to develop FPGAs to withstand the damaging effects of radiation in space for deployment in new satellites, which are 1,000 times less sensitive to space radiation than the commercial equivalent. The Virtex-II Pro, Virtex-4, Virtex-5, and Virtex-6 FPGA families are focused on system-on-chip (SoC) designers because they include up to two embedded IBM PowerPC cores. There are no PowerPC blocks in any Xilinx devices other than Virtex-II Pro. Xilinx FPGAs can run a regular embedded OS (such as Linux or vxWorks) and can implement processor peripherals in programmable logic. Xilinx's IP cores include IP for simple functions (BCD encoders, counters, etc.), for domain specific cores (digital signal processing, FFT and FIR cores) to complex systems (multi-gigabit networking cores, MicroBlaze soft microprocessor, and the compact Picoblaze microcontroller). Xilinx also creates custom cores for a fee. The ISE Design Suite is the central electronic design automation (EDA) product family sold by Xilinx. The ISE Design Suite features include design entry and synthesis supporting Verilog or VHDL, place-and-route (PAR), completed verification and debug using ChipScope Pro tools, and creation of the bit files that are used to configure the chip. Xilinx's Embedded Developer's Kit (EDK) supports the embedded PowerPC 405 and 440 cores (in Virtex-II Pro and some Virtex-4 and -5 chips) and the Microblaze core. Xilinx's System Generator for DSP implements DSP designs on Xilinx FPGAs. A freeware version of its EDA software called ISE Web-PACK is used with some of its non-high-performance chips. Xilinx is the only (as of 2007) FPGA vendor to distribute a native Linux freeware synthesis tool chain. Xilinx announced the architecture for an Extensible Processing Platform, which licenses the

ARM Cortex-A9 MPCore processor for embedded systems designers familiar with the ARM platform. The Extensible Processing Platform architecture abstracts much of the hardware burden away from the embedded software developers' point of view, giving them an unprecedented level of control in the development process. With this platform, software developers can leverage their existing system code based on ARM technology and utilize vast off-the-shelf open-source and commercially available software component libraries. Because the system boots an OS at reset, software development can get under way quickly within familiar development and debug environments using tools such as ARMs RealView development suite and related third-party tools, Eclipse-based IDEs, GNU, the Xilinx Software Development Kit and others. The platform targets embedded designers working on market applications that require multifunctionality and real-time responsiveness, such as automotive driver assistance, intelligent video surveillance, industrial automation, aerospace and defense, and next-generation wireless.Xilinx announced, in early 2011, a new Zynq product family specifically based on its extensible processing platform.

8 Features of CPLD

1. High performance -5 ns pin to pin logic delay - Counter frequency 125 MHz
2. Large density range -36-288 macrocells with upto 6000 usable gates.
3. Enhanced pin locking architecture.
4. Programmable power reduction mode in each macrocell.
5. Slew rate control.
6. High drive 24 mA output.
7. Advanced CMOS 5v fast Flash technology.
8. Flexible 36V 18 function block-90 product terms drive any or all of 18 macro cells within function block-global and product term clocks, output enables, set and reset signals.
9. Extensive IEEE std 1149.1 boundary scan(JTAG) support.
10. Programmable power reduction mode in each macrocell.
11. Skew rate controlled on individual outputs.
12. User programmable ground pin capability.

9 CPLD Kit

The design of a digital system using PLD often requires the connection of several devices to produce the complete specification. For these type of applications, Complex Programmable Logic Devices(CPLD) are more suitable. A CPLD is a collection of individual PLDs on a single integrated circuit. A programmable interconnection structure allows the PLDs to be connected to each other in the same way that can be done with the individual PLDs.

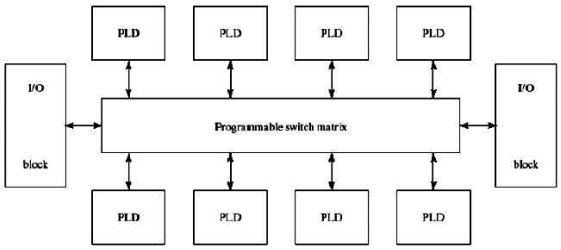
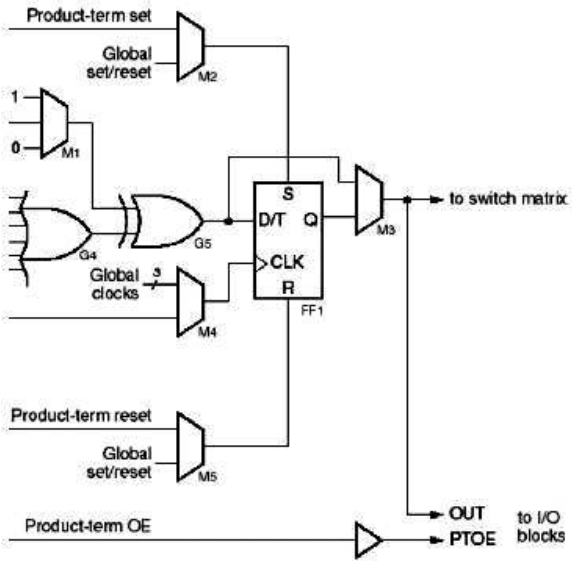


Fig 7-20 General CPLD Configuration

Now, the switch matrix receives inputs from the I/O block and directs it to the individual macrocells. Similarly, selected outputs from macrocells are sent to the outputs as needed. Each PLD typically contains from 8 to 16 macrocells. The macrocells within each PLD are usually fully connected. If a macrocell has unused product terms they can be used by other nearby macrocells. Different manufacturers use different approaches to make individual PLDs, the type of macrocells, I/O blocks and the programmable interconnection structure. The basic component used in PLD is the gate array. A gate array consists of pattern of gates fabricated in an area of silicon that is repeated thousands of times until the entire chip is covered with the gates.



CPLD FAMILIES

The basic construction of a CPLD remains a constant. The differentiator in various CPLD families is the different number of PLD blocks and the different number of I/O pins. Many CPLDs have fewer I/O pins than macrocells. The IC package size dictates the number of I/O pins but not the total number of macrocells. Typical CPLD families have devices with differing resources in the same IC package. The various families are described below:

CoolRunner-II 1.8 Volt CPLD:

Low power
Up to 512 macrocells Up
to 4 I/O banks Advanced
security features
1.5V interfacing

XC9500XV 2.5 Volt CPLD

Up to 4 I/O banks
2.5V, 3.3V I/O tolerance
Up to 288 macrocells
More product terms per macrocell

XC9500XL 3.3 Volt CPLD

5V tolerance
Up to 288 macrocells
More product terms per macrocell
Industry's highest I/O per package

CoolRunner XPLA3 3.3 Volt CPLD

Low power
5V tolerance
Up to 512 macrocells

XC9500 5 Volt CPLD

5V I/O tolerance
Up to 288 macrocells
24mA I/O drive

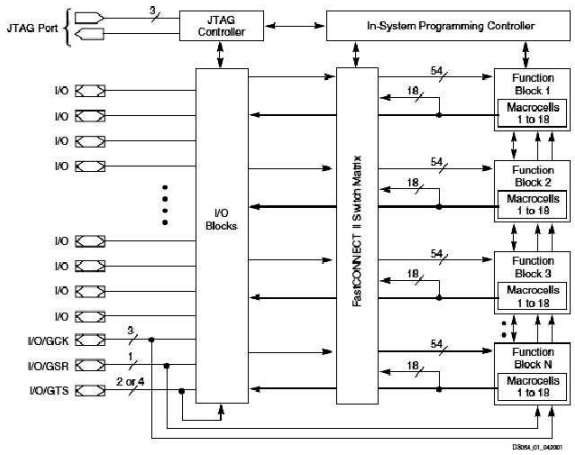
Now in the XC9500 family, various CPLDs can be distinguished as follows:

Xilinx XC9500 Product Family

	9536	9572	95108	95144	95216	95288
Macrocells	36	72	108	144	216	288
Usable Gates	800	1600	2400	3200	4800	6400
t_{PD} (ns)	5	7.5	7.5	7.5	10	10
Registers	36	72	108	144	216	288
Max I/O	34	72	108	133	166	192
Packages	VQ44 PC44	PC44 PC84 TQ100 PQ100	PC84 TQ100 PQ100 PQ160	PQ100 PQ160	PQ160 HQ208 BG352	HQ208 BG352

We have used the XC9572-15pc84 CPLD to implement our program. Since it belongs to the XC9500, the many features of the family are elaborated.

The architechture of the CPLD is as follows:



The above figure is a block diagram of the internal architecture of a typical XC9500 family CPLD. Each internal I/O pin can be used as a input, an output or a bidirectional pin according to the devices programming. The pins at the bottom of the figure can also be used for special purposes. Any of the three pins can be used as "Global Clocks" (GCK). One pin can be used as a "Global Set/Reset"(GSR).

Finally two or four pins can be used as "Global Three State Controls"(GTS). Only four FBs are shown in the figure, but the XC9500 architecture scales to accommodate 16 FBs in the XC95288. Regardless of the specific family member, each FB programmably receives 36 signals from the switch matrix. The inputs to the switch matrix are the 18 macrocell outputs from each of the FBs and the external inputs from the I/O pins. Each FB also has 18 outputs that run under the switch matrix.

10 Conclusion

A 4-Bit Multiplier device can be designed using a maximum of 18 Full Adders and few AND gates. Moreover, the device provides a good precision and accuracy. Also the option to choose either of the two methods gives a flexibility to the user.

11 References

1. Bookname: Circuit Design with VHDL
   Author: Volnei A. Pedroni
   Publication: The MIT Press

2. Bookname: Modern Digital Design.
   Author: R.P. Jain.
   Publication: Tata McGraw Hill.

3. Website: http://ecen3233.okstate.edu