International Journal of LNCT, Vol 2(5)
ISSN (Online): 2456-9895

Design of power and area optimized ALU for FPGA

Shikha Gupta[1], Prashant Chaturvedi[2] and Monika Kapoor[3]
M.tech Scholar, Department of Electronics and Communication, L.N.C.T. Bhopal[1]
Professor, Department of Electronics and Communication, L.N.C.T. Bhopal[2]
Professor & Head, Department of Electronics and Communication, L.N.C.T. Bhopal[3]

Abstract
In this work we have implemented an arithmetic and logic unit with clock gating and hardware sharing method. In our design we have used clock gating technique to implement arithmetic and logic unit. Using this method we have successfully reduced the dynamic power consumption by reducing switching activity inside the design. The dynamic power consumption is decreased by 21% for 8 operation design compared to other design available in literature and 12.2% for 15 operation design.. We have also used hardware sharing method for four instructions; this reduces the hardware usage of FPGA. The Resource usage is reduced by 29% for 8 operations design and by 2.2 & for 15 operation design.

Keywords
Clock gating, Dynamic power consumption, FPGA, Hardware sharing, Arithmetic and logic unit.

1.Introduction
This is an era of hand held devices and equipments, most of these devices runs on battery, this puts a constraint on standby time, to increase standby time more and more battery life is needed, one way of solving this issue is to reduce power consumption of device or equipment. These days almost every device is intelligent, this intelligence came from using processors, and in forthcoming years this trend is likely to be increase. But these processors consume lot of the power of device as lot of switching activity is going inside. ALU (Arithmetic and Logic Unit) is the heart of any processor; this also consumes most of the processor power. In this work we worked in order to reduce power consumption of ALU. We have designed a sixty four bit optimized ALU. A two level optimization is implemented, first we have reduced the FPGA resource consumption by reusing them for different operations, details are given in forthcoming sections, this will cut down FPGA resource consumption and also power consumption of design, several blocks are designed to implement specified operations, in second level of optimization we enable only one block at a time which is currently selected and all other blocks are disabled using clock gating technique, this reduces dynamic power consumption of device and makes our design more greener.

In this work four architectures of ALU are presented as listed below:

1) Conventional 8 operations design
2) Area and delay optimized 8 operations design
3) Conventional 15 operations design
4) Area and delay optimized 15 operation design

The architecture of different designs and their performance analysis is discussed in forthcoming sections.

2.Conventional eight operation design
The operation supported by the reference are listed in Table 1 and the block diagram of reference design is shown in Figure 1. As shown in table 1 that the reference design can perform 8 logical and arithmetic and logical operations only. Shift, Rotate and BCD operations are not supported by this design. Eight units are used to implement the eight operations. At any given time instant only one of the eight units will be in use to perform one of the eight operations but clock signal CLK is assigned to all the eight units at all times, this increases the dynamic power consumption of the design.

Table 1 Operations supported by 8 operations conventional design

Serial No.	Selection	Operation
1	0000	Logical ANDing
2	0001	Logical XNORing
3	0010	Logical XORing
4	0011	Logical ORing
5	0100	Binary Addition
6	0101	Binary Subtraction
7	0110	Binary Increment
8	0111	Binary Decrement

Figure 2 shows the internal structure of logical AND operation. The AND operation is implemented by an array of AND gates followed by a D flip flop.

Similarly internal structures of all the units are shown in Figure 2 to Figure 9.

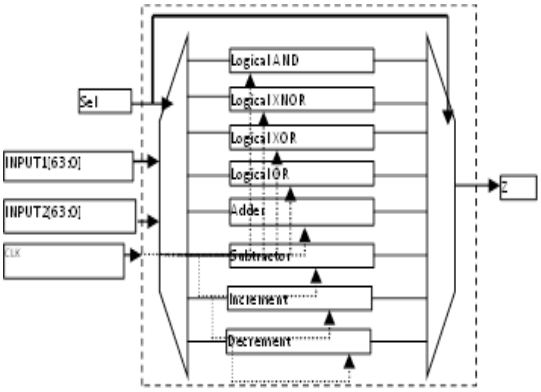

Figure 1 Conventional 8 operation design - internal architecture

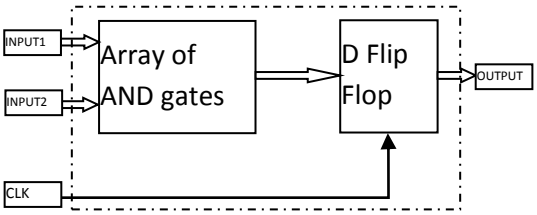

Figure 2 Internal structure - logical AND

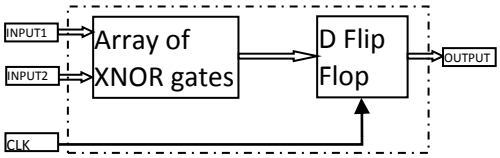

Figure 3 Internal structure - logical XNOR

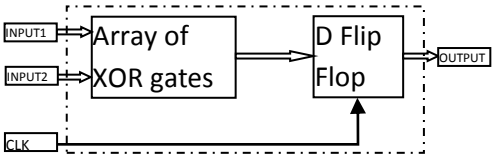

Figure 4 Internal structure - logical XOR

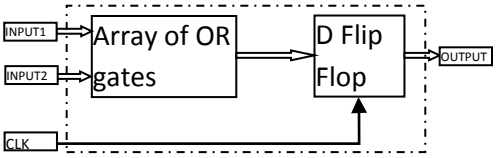

Figure 5 Internal structure - logical OR

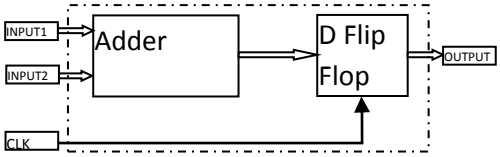

Figure 6 Internal structure – adder

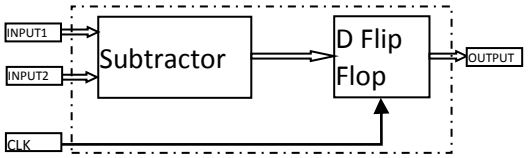

Figure 7 Internal structure - subtractor

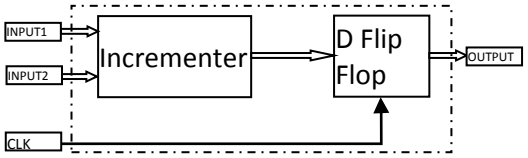

Figure 8 Internal structure - incrementer

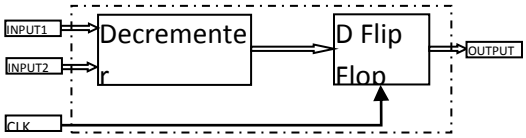

Figure 9 Internal structure-decrementer

3.Area and power optimized eight operation design

In the conventional design all the eight units are connected to the clock at all times which increases the dynamic power consumption, so reduce the dynamic power consumption the clock gating technique is adopted. The clock gating logic shown in figure 10 divides the input clock to 5 different clock signals and only one out of 5 clock signals is active at any given time instant. This reduces the dynamic power consumption of the design.

The second level of optimization is the reduction in the number of units from 8 to 5. In this design the addition, subtraction, increment and decrement operation are implemented using a single unit called the adder + 2's complement unit.

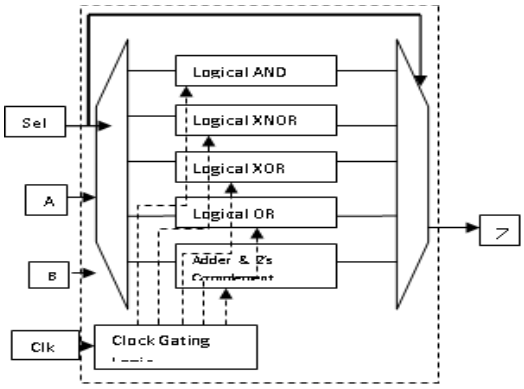

Figure 10 Area and power optimized 8 operation design - internal architecture

In the earlier designs to implement 8 operations, 8 separate blocks are implemented. But there are similar operations like addition, subtraction, increment and decrement. These 4 operations are similar to each other.

Addition is needed in all the four operations, subtraction can be implemented by first doing 2's complement to a number and then adding the other number to it. Similarly increment can be implemented. So to implement these 4 operations we have implemented a single block called addition and 2's complement block. By reusing the adder hardware we have reduced the hardware cost of the design.

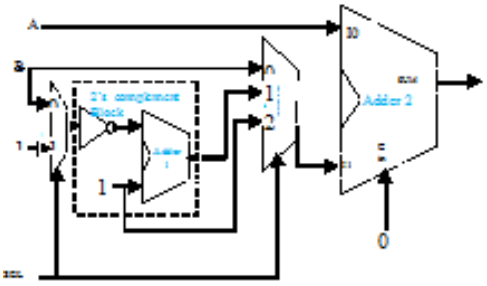

Figure 11 Adder + 2's complement Unit

Figure 11 shows the architecture of adder and 2's complement block. The architecture consists of two 8 bit adders adder 1 and adder 2. Adder 1 is employed in 2's complement unit along with a 8 bit inverter unit. Adder 2 is the main adder which is used to perform the six operations in conjunction with 2's complement block. Three customized multiplexer are also used to implement the adder and 2's complement block. Table 2 shows the input lines selected of two multiplexer for different values of SEL operation selection signal.

Table 2 Operation of Adder + 2's complement Unit

SEL	**Operation**	**MUX 1**	**MUX 2**
0111	DEC A	1	1
0110	INC A	2	0
0100	A + B	0	0
0101	A – B	1	0

The clock gating logic shown in figure 10 bifurcates the input CLK signal to five different clock signals. Table 3 shows the active clock signals with respect to the currently selected operation. The first four operation activates the respective logical units and the next four selection signals selects the adder+2's complement unit to perform the addition, subtraction, increment and decrement operation.

Table 3 Clock Gating

S.No	**SEL**	**Selected CLK IN**
1	000	CLK_Logical_AND
2	001	CLK_Logical_XNOR
3	010	CLK_Logical_XOR
4	011	CLK_Logical_OR
5	100	CLK_Adder_2's_Complement_Unit
6	101	
7	110	
8	111	

4.Conventional fifteen operation design

This section discusses the conventional ALU supporting fifteen operations, this ALU is based on the same design as the reference ALU supporting eight operations i.e. clock is available to all the units at all time and different units are used for performing addition, subtraction, increment and decrement. Here in this design we have incorporated 7 more instructions to perform shift, rotate and BCD operations. Table 4 shows the supported operations by this design.

Table 4 Reference ALU-15 instructions

Serial No.	**Selection**	**Operation**
1	0000	Logical ANDing
2	0001	Logical XNORing
3	0010	Logical XORing
4	0011	Logical ORing
5	0100	Binary Addition
6	0101	Binary Subtraction
7	0110	Binary Increment
8	0111	Binary Decrement
9	1000	Rotate Right
10	1001	Rotate Left
11	1010	Shift Right
12	1011	Shift Left
13	1100	BCD Addition
14	1101	BCD Subtraction
15	1110	BCD Multiplication
16	1111	NOP

Figure 12 shows the internal architecture of reference ALU-15 operations, no clock gating logic is used and separate blocks are used to perform addition, subtraction, increment and decrement.

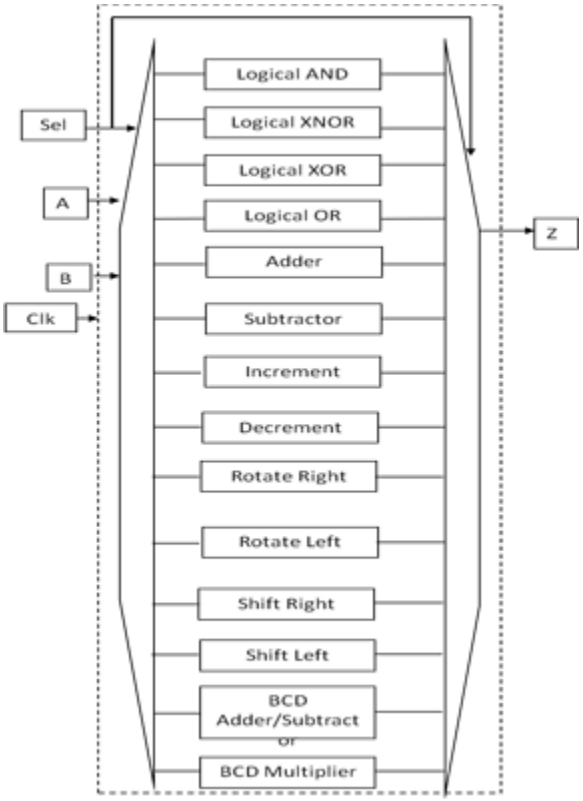

Figure 12 Conventional 15 operation ALU – internal architecture

The rotate right operation is shown in Figure 13, it consists of connections with input and out ports and 64 D flip flops (FD) as shown in figure. The first input signal I0 is connected to output signal O63, I1 to O0, I2 to O1 and so on.

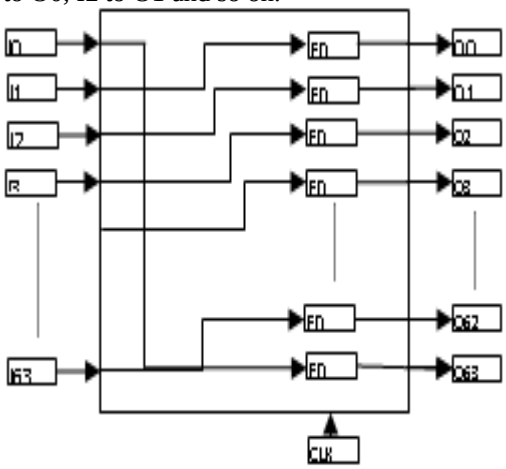

Figure 13 Rotate right – internal structure

Similarly architectures of rotate and shift operations are shown in Figures 13 – 16.

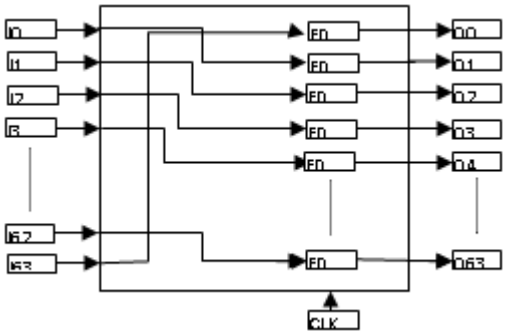

Figure 14 Rotate left – internal structure

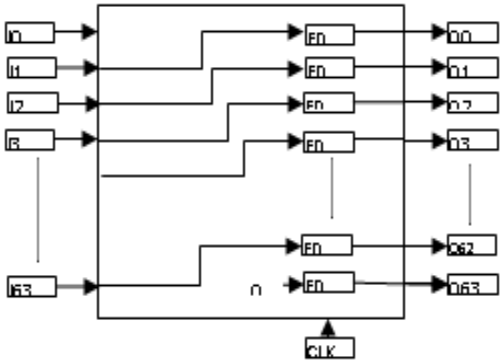

Figure 15: Shift right – internal structure

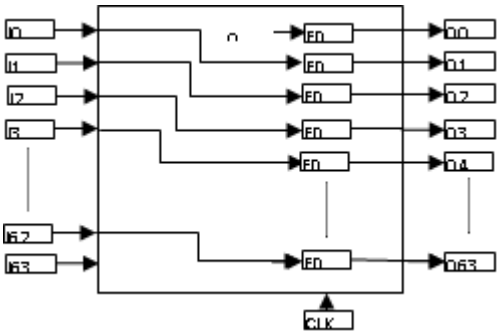

Figure 16 Shift Left – Internal Structure

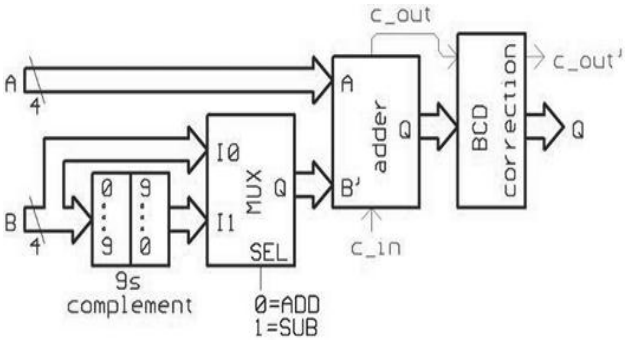

Figure 17 BCD Adder/Subtractor

The BCD adder/Subtractor unit shown in figure 17 consists of a binary adder with a BCD correction unit to perform the addition, to perform the subtraction also a 9'complement unit is also incorporated shown in figure 18. Here in figure 17 a multiplexer is also incorporated to select between the BCD addition and BCD subtraction. To perform BCD addition the data path is the simply the multiplexer, then the binary adder and then the BCD correction unit. To perform the subtraction first the 9's complement of the second number is calculated using the 9'complement unit and the multiplexer passes this complemented number to the binary adder to perform the subtraction, the final result of BCD subtraction is obtained after the BCD correction has been performed.

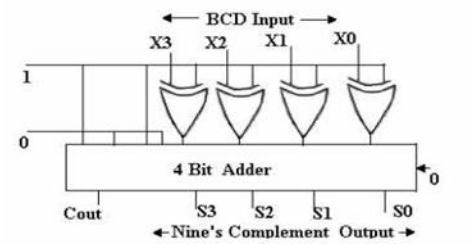

Figure 18 9's Complement Unit

The 9's complement is performed using 4 xor gates, which XORes the input number with 1 and a binary adder to perform the addition of 0110 with the xored number.

Figure 19 represents an area optimized BCD digit multiplier. This multiplier produces the result of multiplication in binary and we need a binary to BCD converter shown in figure 20. The B is the higher nibble of the multiplication and C is the lower nibble of multiplication.

Figure 21 shows the parallel multiplication process of a 4 x 4 BCD multiplier. Xi and yi are single digit BCD numbers. These numbers are multiplied using the single digit BCD multiplier shown in figure 19 and 20. pyixiH and pyixiL are higher and lower nibble of multiplication respectively.

Figure 22 depicts the 4 x 4 multiplier architecture to implement the algorithm shown figure 4.24. In the process of floating point multiplication this 4x4 multiplier is extended to implement 16 x 16 multiplier.

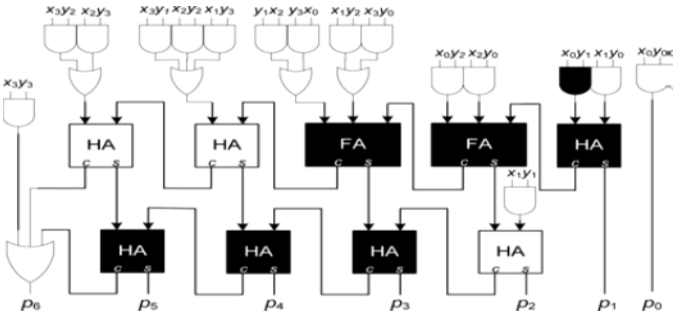

Figure 19 Area optimized – Single digit BCD Multiplier

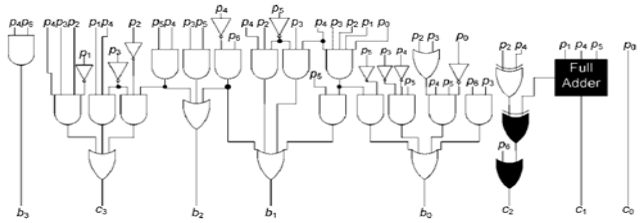

Figure 20Binary to BCD Converter

				x3 y3	x2 y2	x1 y1	x0 y0
				P0003L	P0002L	P0001L	p0000L
			P0003H	P0002H	P0001H	P0000H	
			P0103L	P0102L	P0101L	P0100L	
		P0103H	P0102H	P0101H	P0100H		
		P0203L	P0202L	P0201L	P0200L		
	P0203H	P0202H	P0201H	P0200H			
	P0303L	P0302L	P0301L	P0300L			
P0303H	P0302H	P0301H	P0300H				
P7	P6	P5	P4	P3	P2	P1	P0

Figure 21 Array BCD multiplication process

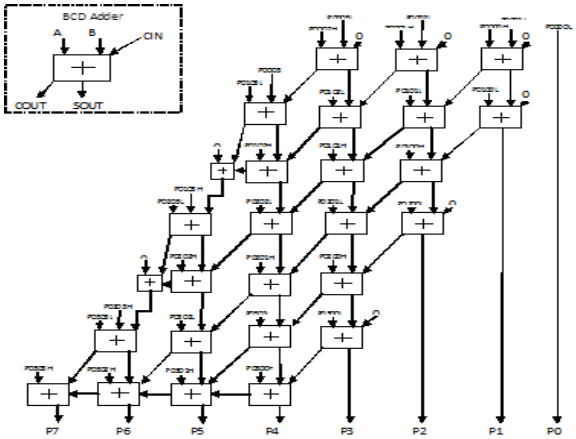

Figure 22 4 digit BCD Multiplier

5.Area and power fifteen operation design

In the reference design all the fifteen units are connected to the clock at all times which increases the dynamic power consumption, so reduce the dynamic power consumption the clock gating technique is adopted. The clock gating logic shown in figure 23 divides the input clock to 11 different clock signals and only one out of 11 clock signals is active at any given time instant. This reduces the dynamic power consumption of the design.

The second level of optimization is the reduction in the number of units from 14 to 11. In this design the addition, subtraction, increment and decrement operation are implemented using a single unit called the adder + 2's complement unit.

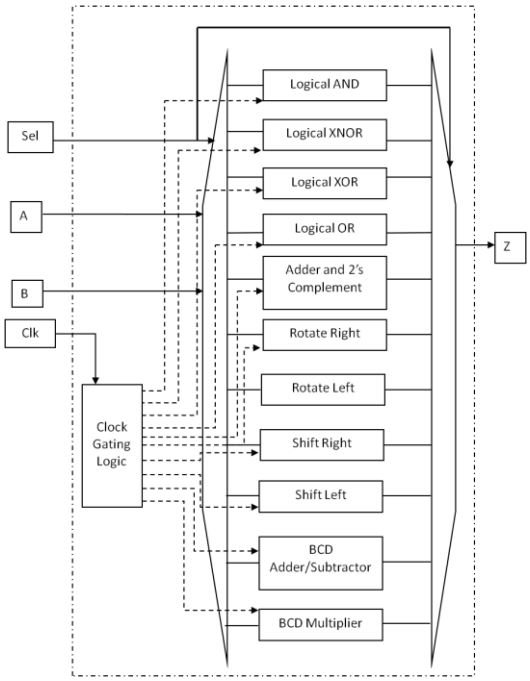

Figure 23 Area and power optimized 15 operation ALU – internal architecture

The clock gating technique is used to reduce the dynamic power consumption of the system. Table 5 depicts the active clock signals for currently selected operation.

6.Performance analysis

Table 6 shows the resource and power consumption of reference ALU and area and power efficient ALU. The base paper design and the conventional design are same, we have implemented the base paper design [1] and named it conventional design, and then we have optimized this conventional design in order to reduce the resource usage and dynamic power consumption and named it low power and low area design. As seen from table 6 that the logics used in optimized low power and low area design is least among the other designs available in literature. The percentage change is 29%. The dynamic power consumption of the base paper design is 43.53mW whereas the dynamic power consumption of proposed design is only 34mW, so the percentage change is 21.8%. So it can be concludes from the table the proposed low power and low area design has low resource usage and lower power consumption as compared to the other designs available in literature.

Table 7 shows the resource and power consumption of conventional ALU and area and power efficient ALU. The base paper design and the conventional design are same, we have implemented the base paper design and named it conventional design, and then we have optimized this conventional design in order to reduce the resource usage and dynamic power consumption and named it low power and low area design. As seen from table 7 that the logics used in optimized low power and low area design is least among the other designs available in literature. The percentage change is 2.2%. The dynamic power consumption of the conventional design is 228mW whereas the dynamic power consumption of proposed design is only 200mW, so the percentage change is 12.2%. So it can be concludes from the table the proposed low power and low area design has low resource usage and lower power consumption as compared to the other designs available in literature.

Table 5 Clock Gating - 15 operation design

Serial No.	Selection	CLK signal selected
1	0000	CLK_Logical_AND
2	0001	CLK_Logical_XNOR
3	0010	CLK_Logical_XOR
4	0011	CLK_Logical_OR

Serial No.	Selection	CLK signal selected
5	0100	CLK_Adder_2's_Complement_Unit
6	0101	
7	0110	
8	0111	
9	1000	CLK_Rotate_Right
10	1001	CLK_Rotate_Left
11	1010	CLK_Shift_Right
12	1011	CLK_Shift_Left
13	1100	CLK_BCD_Adder_Subtractor
14	1101	
15	1110	CLK_BCD_Multiplier

Table 6 Performance analysis – 8 Operation designs

On Chip	Base Paper Design		Conventional		Low Power & Low Area	
	Power (mW)	Resource	Power (mW)	Resource	Power (mW)	Resource
Clock	0.16	-	02	-	05	-
Logic	0.76	496	02	511	01	355
Signal	6.63	752	04	764	04	802
IOS	35.98	197	37	196	29	196
Static Power	45.32	-	45	-	45	-
Dynamic Power	43.53	-	44	-	34	-
Total	88.85	-	90	-	79	-

Table 7 Performance analysis-15 Operation designs

On Chip	Conventional		Low Power & Low Area	
	Power (mW)	Resource	Power (mW)	Resource
Clock	06	-	01	-
Logic	75	15253	74	14909
Signal	102	17618	95	17502
IOS	44	197	29	197
Static Power	46	-	46	-
Dynamic Power	228	-	200	-
Total	273	-	246	-

7.Conclusion

In this work we have successfully implemented an arithmetic and logic unit with clock gating and hardware sharing method. In this work four designs are implemented and compared, first design is the conventional ALU supporting 8 operations with no clock gating and hardware sharing method being used. In order to reduce dynamic power consumption of the design we have used clock gating method, in this method the clock is inactive for the units which are not in use for the currently selected operation and clock is active only for the unit which is currently being used, this reduces the switching activity inside the ALU and hence in-turn reduces the dynamic power consumption. The percentage decrease in

dynamic power consumption is around 21% for 8 instruction design.
Another level of optimization we made in this design is the inclusion of hardware sharing method, it is known that addition, subtraction, increment and subtraction can be implemented using a adder and 2's complement unit and hence in this work we have used this one unit called adder + 2'complmement unit to implement all the four operations namely: addition, subtraction, increment and decrement. This reduces the hardware resource usage of the optimized design and makes the design more cost effective.

The third design is conventional design supporting 15 instructions, in this design we have increased the functionality of the base paper design by employing 7 more instructions making the design to support a total of 15 operations. The new 7 operations employs rotate right, rotate left, shift right, shift left, BCD addition, BCD subtraction and BCD multiplication. This design does not use clock gating and hardware sharing method. The fourth design is proposed low power and low area design which employs the clock gating technique and hardware sharing method which in-turn reduces the dynamic power consumption and resource usage. The percentage decrease is 2.2% for resource usage and the percentage decrease is 12.2% for dynamic power consumption.

References

[1] Murgai S, Gupta A, Muthukrishnan G. Energy efficient and high performance 64-bit Arithmetic Logic Unit using 28nm Technology. In advances in computing, communications and informatics (ICACCI), International Conference on 2015 (pp. 453-6). IEEE.

[2] Shinde J, Salankar SS. Clock gating-A power optimizing technique for VLSI circuits. In india conference (INDICON), Annual IEEE 2011 (pp. 1-4). IEEE.

[3] Castro J, Parra P, Acosta AJ. Optimization of clock-gating structures for low-leakage high-performance applications. In circuits and systems (ISCAS), proceedings of 2010 IEEE International Symposium on 2010 May 30 (pp. 3220-3). IEEE.

[4] Prashanth BU, Kumar PA, Sreenivasulu G. Design & implementation of floating point ALU on a FPGA processor. In computing, electronics and electrical technologies (ICCEET), 2012 International Conference on 2012 (pp. 772-6). IEEE.

[5] Ortega-Cisneros S, Raygoza-Panduro JJ, Muro JS, Boemo E. Rapid prototyping of a self-timed ALU with FPGAs. In reconfigurable computing and FPGAs, 2005. ReConFig 2005. International Conference on 2005 (pp. 8-pp). IEEE.

[6] Ryu BS, Oh HS, Shim KH, Lee KY, Cho TW. Multi-level approaches to low power 16-bit ALU design. In ASICs, 1999. AP-ASIC'99. The First IEEE asia pacific conference on 1999 (pp. 158-61). IEEE.

[7] Lam TK, Yang X, Tang WC, Wu YL. On applying erroneous clock gating conditions to further cut down power. In proceedings of the 16th asia and south pacific design automation conference 2011 (pp. 509-14). IEEE Press.

[8] Pandey B, Pattanaik M. Clock gating aware low power ALU design and implementation on FPGA. International Journal of Future Computer and Communication. 2013; 2(5):461.

[9] Arbel E, Eisner C, Rokhlenko O. Resurrecting infeasible clock-gating functions. In design automation conference, 2009. DAC'09. 46th ACM/IEEE 2009 (pp. 160-5). IEEE.

[10] Semeraro G, Magklis G, Balasubramonian R, Albonesi DH, Dwarkadas S, Scott ML. Energy-efficient processor design using multiple clock domains with dynamic voltage and frequency scaling. In high-performance computer architecture, 2002. Proceedings. eighth international symposium on 2002 (pp. 29-40). IEEE.

[11] Burd TD, Brodersen RW. Design issues for dynamic voltage scaling. In low power electronics and design, 2000. ISLPED'00. Proceedings of the 2000 International Symposium on 2000 (pp. 9-14). IEEE.

[12] Pouwelse J, Langendoen K, Sips H. Energy priority scheduling for variable voltage processors. In Proceedings of the 2001 international symposium on Low power electronics and design 2001 (pp. 28-33). ACM.

[13] Lee C, Lee JK, Hwang T. Compiler optimization on instruction scheduling for low power. In System Synthesis, 2000. Proceedings. The 13th International Symposium on 2000 (pp. 55-60). IEEE.

[14] Parikh A, Kandemir M, Vijaykrishnan N, Irwin MJ. Instruction scheduling based on energy and performance constraints. In VLSI, 2000. Proceedings. IEEE Computer Society Workshop on 2000 (pp. 37-42). IEEE.

[15] Choi KW, Chatterjee A. Efficient instruction-level optimization methodology for low-power embedded systems. In Proceedings of the 14th international symposium on Systems synthesis 2001 (pp. 147-52). ACM.

[16] Toburen MC, Conte TM, Reilly M. Instruction scheduling for low power dissipation in high performance microprocessors. In Proceedings of the 1998 Power Driven Microarchitecture Workshop 1998 (pp. 14-19).

[17] Monteiro J, Rinderknecht J, Devadas S, Ghosh A. Optimization of combinational and sequential logic circuits for low power using precomputation. In Advanced Research in VLSI, 1995. Proceedings., Sixteenth Conference on 1995 (pp. 430-44). IEEE.

[18] Emnett F, Biegel M. Power reduction through RTL clock gating. SNUG, San Jose. 2000:1-1.

[19] Yeap GK. Practical low power digital VLSI design. Springer Science & Business Media; 2012.

[20] Perkins CE, Alpert SR, Woolf B. Mobile IP; Design Principles and Practices. Addison-Wesley Longman Publishing Co., Inc.; 1997.
[21] Kaeslin H. Digital integrated circuit design: from VLSI architectures to CMOS fabrication. Cambridge University Press; 2008.
[22] Schoenmakers PJ, Theeuwen JF. Clock gating on RT-level VHDL. In Proceeding of the int. Workshop on logic synthesis, Tahoe City, CA 1998 (pp. 387-91).
[23] Benini L, De Micheli G, Macii E, Poncino M, Scarsi R. Symbolic synthesis of clock-gating logic for power optimization of synchronous controllers. ACM Transactions on Design Automation of Electronic Systems (TODAES). 1999; 4(4):351-75.
[24] Huda S, Mallick M, Anderson JH. Clock gating architectures for FPGA power reduction. In field programmable logic and applications, 2009. FPL 2009. International Conference on 2009 (pp. 112-8). IEEE.
[25] Oklobdzija VG, Stojanovic VM, Markovic DM, Nedovic NM. Digital system clocking: high-performance and low-power aspects. John Wiley & Sons; 2005.
[26] Tirumalashetty V, Mahmoodi H. Clock gating and negative edge triggering for energy recovery clock. In Circuits and Systems. ISCAS 2007. IEEE International Symposium on 2007 (pp. 1141-4). IEEE.
[27] Pandey B, Yadav J, Pattanaik M, Rajoria N. Clock gating based energy efficient ALU design and implementation on FPGA. In energy efficient technologies for sustainability (ICEETS), 2013 International Conference on 2013 (pp. 93-97). IEEE.