8-bit RISC Processor

Design Description and Instructions

Created by Md. Alvir Islam Nobel

Functional Modules

- Program Counter Unit (PCU)
- Instruction Memory
- Data Memory
- Control Unit
- Register Set
- Arithmetic & Logical Unit (ALU)
- I/O Module
- Interrupt Module
- Serial Module

Program Counter Unit (PCU)

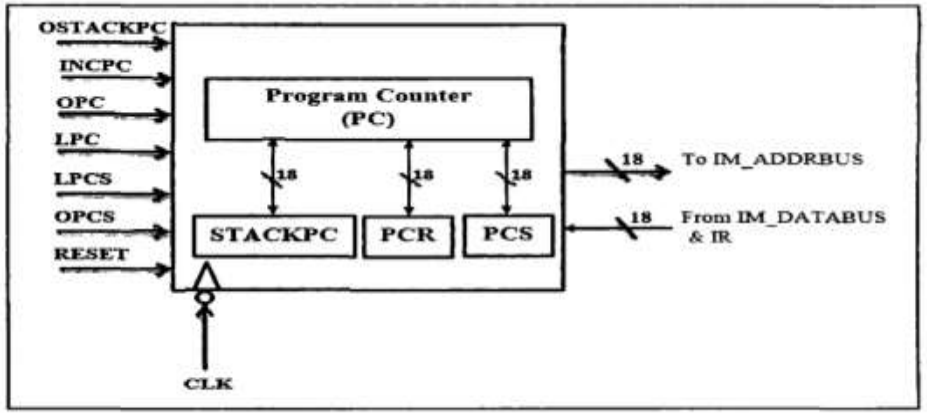


Figure 3.3 Block diagram of PCU

- Consists of Program Counter (PC), Program Counter Save (PCS), Program Counter Register (PCR) and STACKPC
- PC is an 18 bit wide register that holds the location of the current instruction being executed
- As an instruction is fetched the PC updates it stored value by adding 1.
- IM_ADDRBUS gets the 18 bit PC content with the activation of corresponding control signal.
- When LPCS signal is enabled, PCS saves the PC content incremented by 2.
- PCR is used to store next instruction to be executed while branching instruction is executed.
- STACKPC stores the PC content while Interrupt Module instruction is being executed.
- All Loading operations to the registers take place in the falling edge of the clock.

Instruction Memory

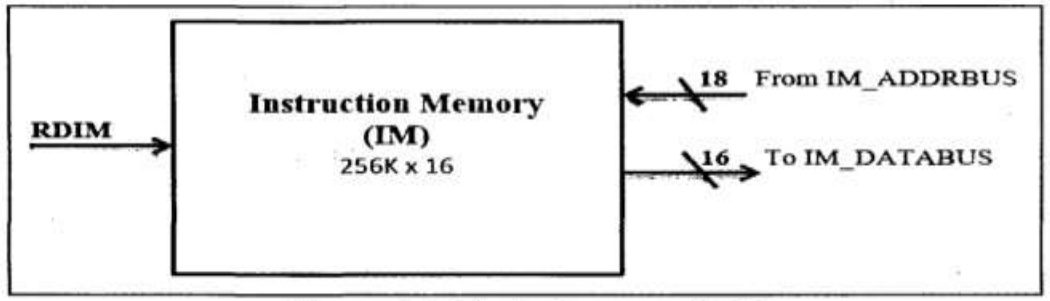


Figure 3.4 Block diagram of IM

- There are two cycles in a pipeline processor : Instruction Fetch, Instruction Execution.
- Instruction Memory is 16-bit wide and having 262,144 address locations, so that any practical real time programs can be fitted into it.
- In the Fetch cycle, IM content is loaded to IM_DATABUS for the corresponding IM_ADDRBUS

Control Unit

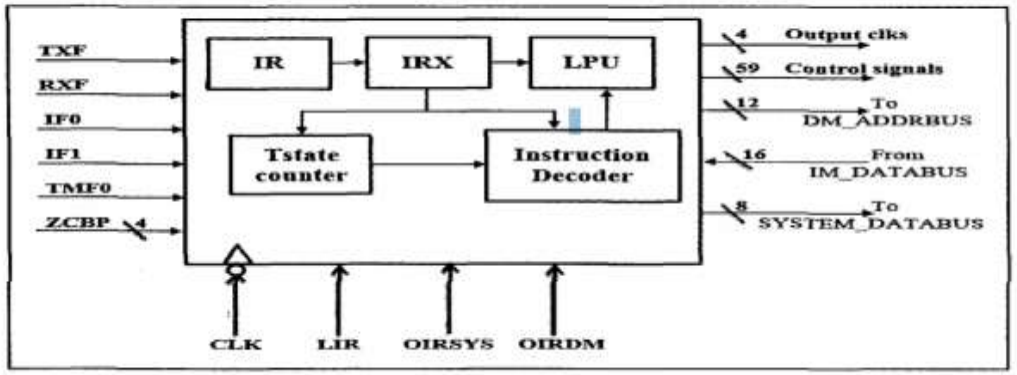


Figure 3.5 Block diagram of Control unit

- Consists of Instruction Registers (IR, IRX), Tstate Counter , Low Power Unit (LPU), Instruction Decoder.
- While executing one instruction , another instruction is fetched; that's why we need 2 Instruction Registers. IR content is moved to the IRX in rising edge of the Execution cycle.
- Tstate counter generates fetch and execution cycles required for the proper working of processor. The tstates TFl (fetch cycle), TXl (execution cycle), TX2 (execution cycle2, only for branching instruction) are generated at rising edge of clock. Interrupt priority and exceptions in instructions, like jump, are also taken in account in Tstate counter module.
- Instruction decoder will generate controls signals required for the modules whenever IRX is loaded with a valid instruction (every rising edge of the clock).

- Clock Gating is done at the Low Power (LP) unit for the Data Memory and General purpose Registers as they are very power consuming elements .
- The Control Unit takes input clock from source clock of FPGA and generates 59 control signals and 4 clock signals for the proper working of all modules. The four clock signals include gated clock signals for register set and Data Memory modules, baud rate clock for Serial Module and 25 MHz output clock signal for all other modules.
- The Control Unit receives inputs from Interrupt Module about the states of its three interrupts through IFO, IFl and TMFO flags, which are taken into account in Tstate counter module. Also receives inputs from Flag register regarding the states of its four flags, which are used for control signal generation related to branching instructions.
- The Control Unit receives inputs from Serial Module regarding transmission or reception of data through TXF and RXF flags.

Data Memory

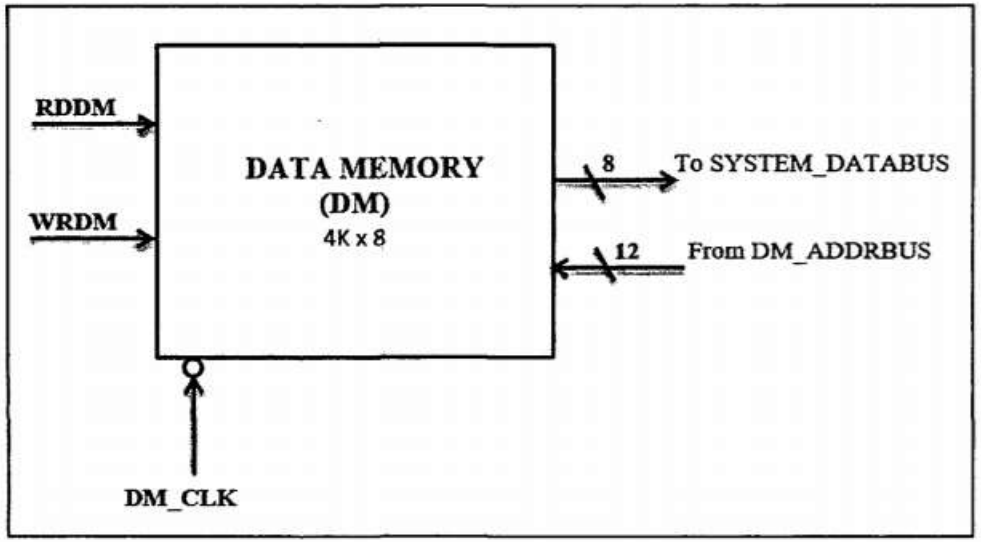


Figure 3.6 Block diagram of DM

- 8 bit wide and has 4096 address locations
- Gets required address from Control Unit by 16 bit DM_ADDRBUS
- Gives reading and writing access thorough the 8 bit SYSTEM_DATABUS databus
- Control Unit provides the required clock when it is in use
- All loading to Data Memory occurs in the falling edge of the clock

Accumulator

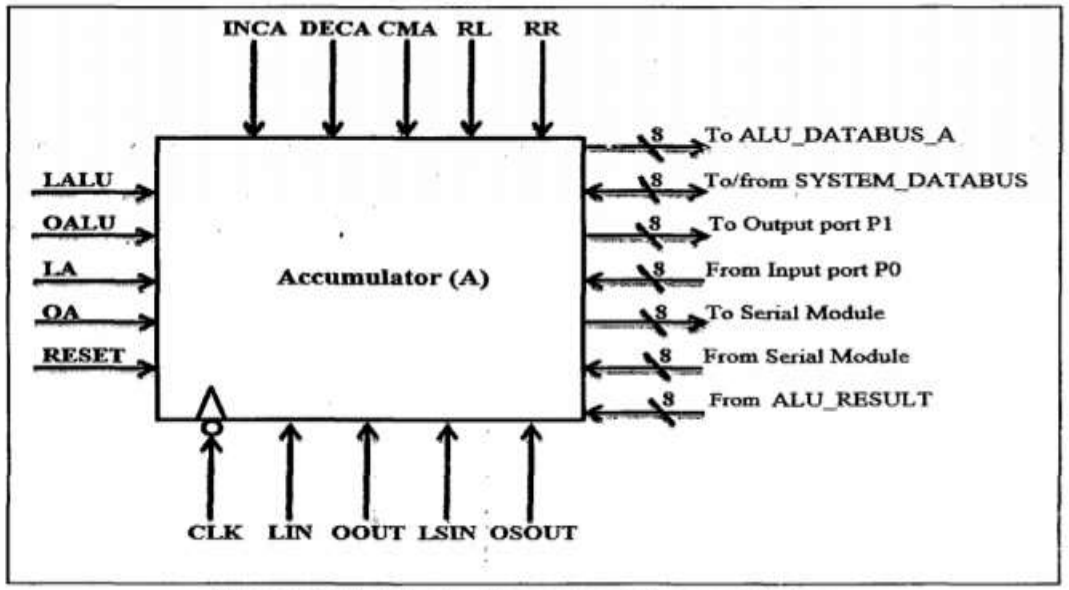


Figure 3.7 Block diagram of Accumulator

- 8 bit wide register
- Connected to SYSTEM_DATABUS through bidirectional data bus ; can load or transmit data. Used for data transfer instruction.
- Connected to ALU_DATABUS_A and ALU_RESULT data buses to transfer logical and arithmetic instructions
- Connected to two eight bit data buses for communicating with I/O module
- Connected to TBUFF register in Serial Module for sending data required for transmission through serial-out port ‘txout’. Also connected to RBUFF register for storing data received through serial in port ‘rxin’.

Register Set

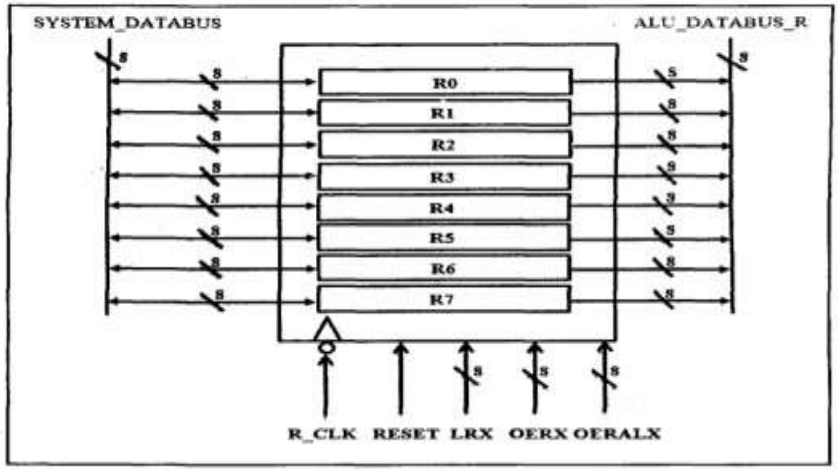


Figure 3.8 Block diagram of Register set

- Contains eight 8- bit registers : R0,R1,R2,R3,R4,R5,R6,R7
- Connected to ALU unit through ALU_DATABUS_R data bus to execute Arithmetic and Logic operation
- Connected to SYSTEM_DATABUS by a bidirectional data bus with loading and storing facilities
- Clock is provided by the control unit. This set is enabled when one of the registers needs to be used

ALU

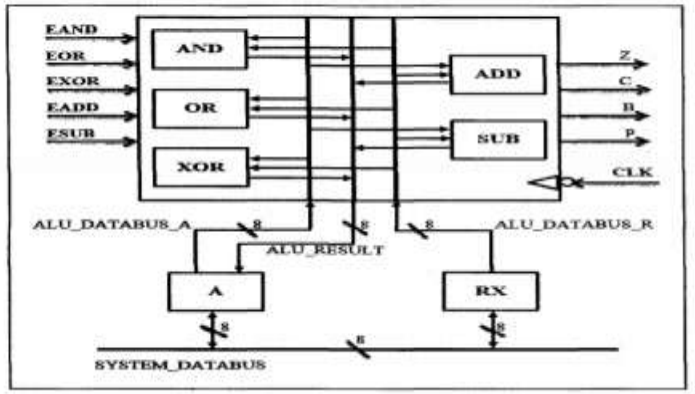


Figure 3.9 Block Diagram of ALU unit

- It is connected to Accumulator by ALU_DATABUS_A and ALU_DATABUS_RESULT . It is also connected to Register Set by ALU_DATABUS_R.
- During execution , inputs are loaded through ALU_DATABUS_R and ALU_DATABUS_A.
- Result is stored to the accumulator through ALU_DATABUS_RESULT.
- ALU can perform AND, OR, XOR, ADD, SUB operations for their corresponding control signals: EAND, EOR, EXOR, EADD, ESUB
- There are four flags Z (Zero), C (Carry), B(Borrow), P(Parity) in a set of 4-bit flag register. Parity gets high when there is even number of 1’s.

Flag Register

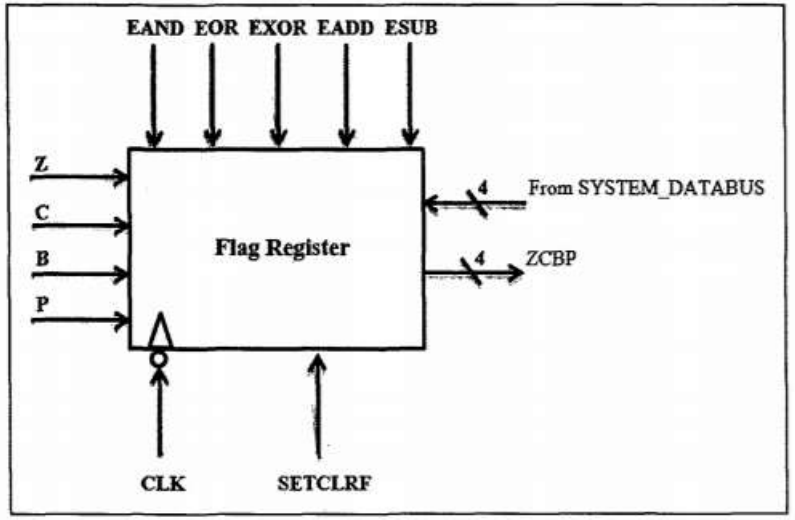


Figure 3.15 Block diagram of Flag register

- 4 bit Register consisting of Zero flag (Z), Carry flag (C), Borrow flag (B), Parity flag (P). Follows this manner [ZCBP] ; where Z is MSB and P is LSB.
- Gets updated for ALU operations only.
- For instance: During addition, only Z,C,P bits get updated ; B stays the same.

I/O Module

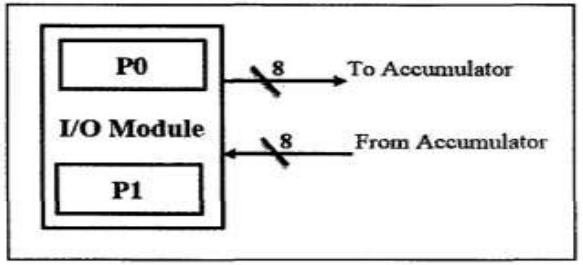


Figure 3.17 Block diagram of I/O Module

- It has one 8-bit input port P1 and one 8-bit P0. They are directly connected to Accumulator, which can control the data flow for the corresponding signals.

Interrupt Module

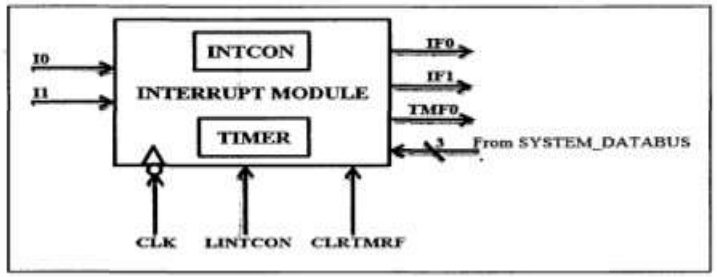


Figure 3.19 Block diagram of Interrupt Module

- Contains 2 External Hardware interrupts and a timer interrupt
- External Hardware interrupts are I0,I1; where I0 has the highest priority and I1, timer interrupts have the lowest priority.
- I1 interrupt is maskable but I0 is not.
- External Hardware interrupts are level triggered and have to be high for at least 2 cycles for the proper working of this pipeline.
- It has a 3 bit register named INTCON where the MSB indicates the activation of the timer interrupt, the second bit is high when external inputs are active , the LSB is used for masking I1.
- Flag IF0 is set when bit 1 of INTCON is high and I0 is high. IF1 flag will be set only if the bit 1 of INTCON is high, the bit 0 of INTCON is low and I1 is high .
- Contains a 10 bit register named as TIMER; it can count up to a predefined value. In this case, it is 1023. When it reaches max value flag TMF0 gets set. This flag TMF0 is cleared by the instruction CLRTMRF. When timer interrupt is turned on, flag will be raised at predefined intervals.

Serial Module

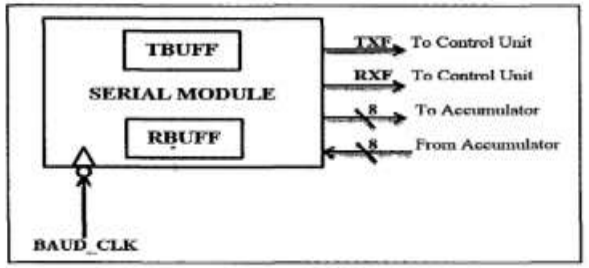


Figure 3.21 Block diagram of Serial Module

- Consists of ‘rxin’ as serial-in port and ‘txout’ as serial-out port.
- The serial communication is based on UART protocol shown in Figure 3.20. The baud rate used for this Serial Module is 115200 per second.
- The data transmission starts with a start bit of 0, followed by the individual data bits of the word with the Least Significant Bit (LSB) being sent first and then stop bit 1. (0,LSB_ _ _ _ _ _MSB,1)
- The Serial Module consists of two 8-bit registers TBUFF and RBUFF for storing data while transmission and reception. TBUFF is shifted out for transmission and similarly for reception RBUFF is shifted out.
- Serial Module will provide TXF and RXF flags to Control Unit regarding transmission or reception of data, which will be set whenever a transmission or reception respectively is completed.

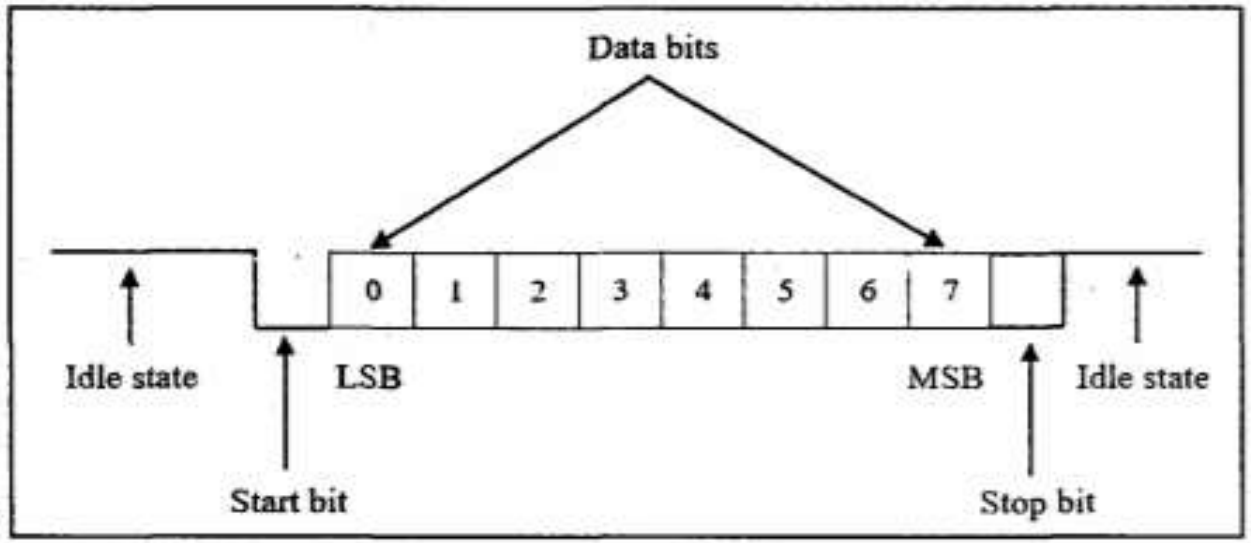


Figure 3.20 UART serial communication protocol

Instruction Set Architecture

- The Instruction set architecture contains four type of instructions: data transfer instruction, arithmetic and logical instruction, branching instruction and machine control and I/O instruction
- The instruction set architecture contains only 34 basic instructions and total number of opcodes is 83
- The opcode contains 16 bits from 0 to 15, of which 15th and 14th bits decides type of instruction being performed: if the 15th and 14th bit are 00, 01, 10 or 11, type of instruction will be data transfer instruction, arithmetic and logic instruction, branching instruction or machine control and I/O instruction respectively
- Table 3.1 shows instruction set of proposed RISC processor. In the Table 3.1 Opcode column, rrr is register select code from 000 to 111, x is don't care, a is address and d is data. In Mnemonic column, RX is register representing any one of R0 - R7 registers, A is accumulator, M is Data Memory address and add 18 is Instruction Memory address.

Data Transfer Instructions

Opcode	Mnemonic	Operation
00000rrrXXXXXXXX	MOV A, RX	RX => A
00010rrrXXXXXXXX	MOV RX, A	A => RX
0010aaaaaaaaaaa	MOV A, M	M =>A
0011aaaaaaaaaaa	MOV M, A	A =>M
00001xxxddddddddd	MOV A, 8 bit Data	8 bit Data => A

Arithmetic and Logic Operation

Opcode	Mnemonic	Operation
0100000rrrXXXXXX	AND A,RX	A <= A and RX update flags
0100001rrrXXXXXX	XOR A, RX	A <= A xor RX update flags
0100010rrrXXXXXX	OR A , RX	A <= A or RX update flags
0100011rrrXXXXXX	ADD A, RX	A <= A + RX update flags
0100100rrrXXXXXX	SUBA,RX	A <= A - RX update flags
0101000XXXXXXXXXX	CM A	A<= ~A
0101001XXXXXXXXXX	INC A	A<= A+1

0101010xxxxxxxx	DEC A	A <= A - 1
0101011xxxxxxxx	RR A	rotate accumulator right by 1 bit
0101100xxxxxxxx	RL A	rotate accumulator left by 1 bit
011000xxxxxxdddd	SETCLRF 4-bit data	clear/set flags
011001xxxxxxxxxx	CLRTMRF	clear timer flag
0111xxxxxxxxddd	EDINTER 3-bitdata	To control interrupt's operations; enable, disable and mask

Branching Instructions

10000aaxxxxxxxxxxaaaaaaaaaaaaaa	JUMP add18	Jump to 18-bit address add 18
10001aaxxxxxxxxxxaaaaaaaaaaaaaa	JZ add18	Jump to 18-bit address add 18 if Z=1
10010aaxxxxxxxxxxaaaaaaaaaaaaaa	JC add18	Jump to 18-bit address add 18 if C=1
10001aaxxxxxxxxxxaaaaaaaaaaaaaa	JB add18	Jump to 18-bit address add 18 if B=1
10001aaxxxxxxxxxxaaaaaaaaaaaaaa	JP add18	Jump to 18-bit address add 18 if P=1

Machine Control and I/O Instructions

110000xxxxxxxxxx	HALT	To stop operations
110001xxxxxxxxxx	RESET	To reset Accumulator, Register set and PC
110010xxxxxxxxxx	SAV PC	Save PC+2 value in PCS register
110011xxxxxxxxxx	RES PC	Restore PC with PCS content
110100xxxxxxxxxx	RETI	Return fi-om interrupt service routine
110101xxxxxxxxxx	WAIT TXF	Wait until TXF flag gets set
110110xxxxxxxxxx	WAIT RXF	Wait until RXF flag gets set
11100xxxxxxxxxx	IN P0	Accumulator gets Port P0 content
11101xxxxxxxxxx	OUT P1	Accumulator sends it content to Port P1
11110xxxxxxxxxx	SIN RBUFF	Accumulator gets RBUFF register content
11111xxxxxxxxxx	SOUT TBUFF	Accumulator sends its content to TBUFF register

Control Signals

Program Counter Unit

Signal	Function
CLK	System Clock
OPC	To Output PC data on IM_ADDRBUS
PC	To Load PC with PCR content
INCPC	To Increment PC value by one
OPCS	To Load PC with PCS content
LPCS	To Load PCS with PC content
RESET	To Reset PC
OSTACKPC	To Load PC with STACKPC content

Instruction Memory

Signal	Function
RDIM	To output IM content to IMDATABUS corresponding to the address provided by the IM_ADDRBUS

Data Memory

Signal	Function
DMCLK	Data Memory Clock
RDDM	To output DM content to SYSTEM_DATABUS corresponding to the address provided by the DM_ADDRBUS
WRDM	To input SYSTEM_DATABUS content to DM corresponding to the address provided by the DM_ADDRBUS

Accumulator

Signal	Function
CLK	System Clock
RESET	To Reset Accumulator
LA	To Load Accumulator from SYSTEM__DATABUS

Accumulator

Signal	Function
RESET	To Reset Accumulator
LA	To Load Accumulator from SYSTEM_DATABUS
OA	To Output Accumulator to SYSTEM_DATABUS
INCA	To Increment Accumulator
DECA	To Decrement Accumulator
CMA	To Complement all the bits of Accumulator
LALU	To Load Accumulator from ALU_RESULT
OALU	To Output Accumulator to ALU_DATABUS_A
RL	To Rotate the bits of Accumulator in left direction
LIN	To Load Accumulator from 8-bit input port PO
OOUT	To Send Accumulator content to 8-bit output port
LSIN	To Load Accumulator from 8-bit register RBUFF in Serial Module
OSOUT	To Send Accumulator content to 8-bit register TBUFF in Serial Module

Register set

Signal	Function
R_CLK	Register set clock
LRX	To Load RX from SYSTEMDATABUS
OERX	To Output RX to SYSTEM_DATABUS
OERALX	To Output RX to ALUDATABUS_R
RESET	To Reset all eight registers

Flag Register

Signal	Function
SETCLRF	To Set/Clear flags bits by SYSTEMDATABUS
EAND	A and RX, Parity (P) and Zero (C) flags are updated
EOR	A or RX, Parity (P) and Zero (C) flags are updated
EXOR	A xor RX, Parity (P) and Zero (C) flags are updated
EADD	A + RX, Carry (C), Parity (P) and Zero (C) flags are updated
ESUB	A - RX, Borrow (B), Parity (P) and Zero (C) flags are updated

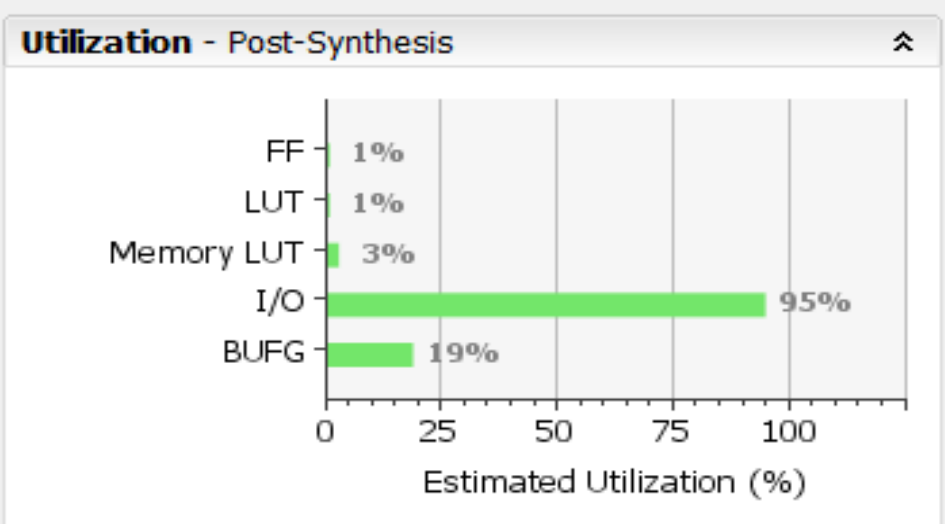
Interrupt Module

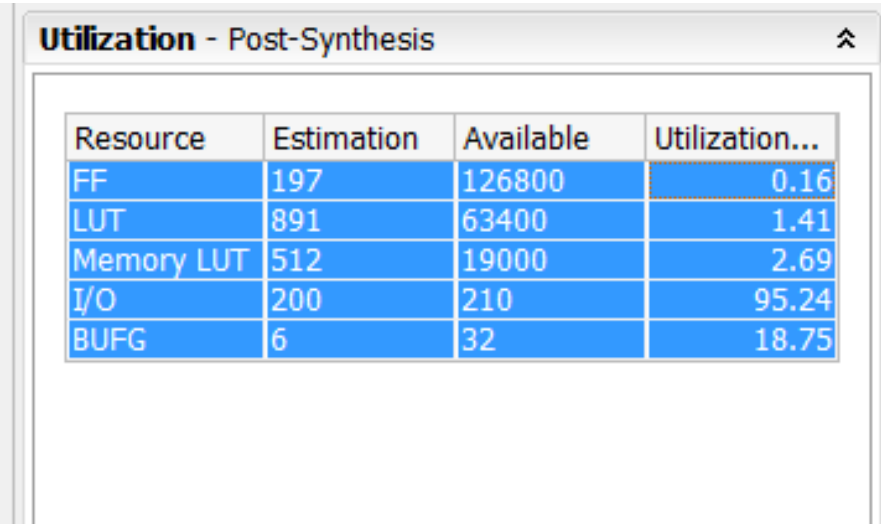
Signal	Function
CLK	System Clock
LINTCON	To enable/disable interrupts and mask I1 interrupt by SYSTEMDATABUS content.
CLRTMRF	Clear timer flag TMFO

Sample Instructions for Simulation

Opcode	Mnemonic	Operation
0000101101101001	MOVI A, 01101001	01101001=>A
1111100000000000	SOUT TBUFF	A=>TBUFF
1101010000000000	WAIT TXF	Wait until TXF flag gets set
1101100000000000	WAIT RXF	Wait until RXF flag gets set
1111000000000000	SIN RBUFF	RBUFF=>A
0011000000001111	MOV M,A	A=>DM[000000001111]
1100000000000000	HALT	Stop operations

DEVICE UTILIZATION OF THE XILINX Artix-7 BOARD FPGA FOR SAMPLE PROGRAM





Resource	Estimation	Available	Utilization...
FF	197	126800	0.16
LUT	891	63400	1.41
Memory LUT	512	19000	2.69
I/O	200	210	95.24
BUFG	6	32	18.75

Reference

[1] JIKKU JEEMON, “8-Bit RISC Processor Design using Verilog HDL on FPGA “ Thesis Dissertation, 2016