

Disciplina: ELE1717 - Sistemas Digitais
Aluno:

Período: 2017.2
Data: 29/08/2017

Implemente o circuito de um microprocessador (Figura 1) com arquitetura do tipo Von-Neumann (RAM 256x8-Bits), com 4 registradores de uso geral (A,B,C,D) de 8 Bits cada, com um ponteiro para pilha (SP) de 8 Bits, com 2 Bits de sinalização Z (True se o resultado da ULA for zero) e C (True se a ULA apresentar um carry de saída), com 24 posições de I/O representadas pelos endereços de 232 até 255 da memória RAM e com um conjunto de instruções apresentados na Tabela 1.

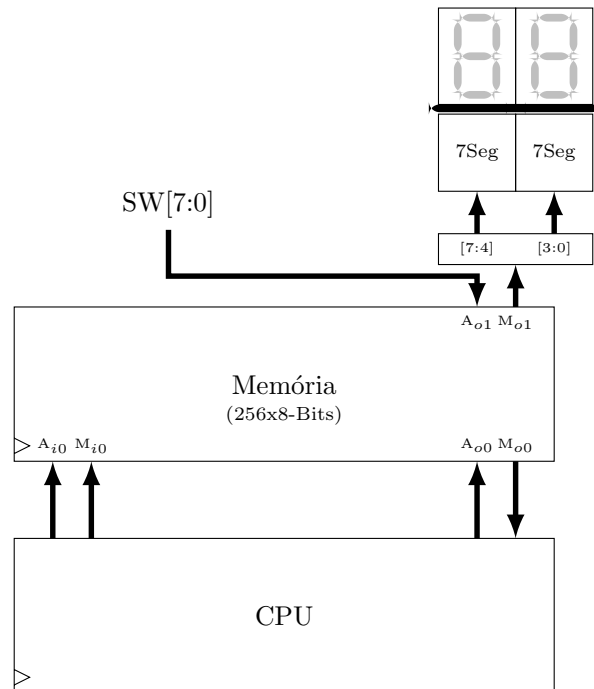


Figura 1: Diagrama de blocos do *datapath* do microprocessador

O processador será construído por um conjunto de blocos constituídos por: **Contador de programa (PC)**, o qual se trata de um registrador que retém o endereço da instrução a ser executada. **Ponteiro da pilha (SP)**, o qual se trata de um registrador que retém o endereço do topo da pilha. Quatro **Registradores de uso geral**, o qual se trata de um banco de registradores que retém os dados que estão sendo trabalhados. **Unidade lógica aritmética (ULA)**, o qual se trata de um bloco que realiza operações lógicas e/ou aritmética como soma, subtração entre outras. **Registrador de instrução (IR)**, o qual se trata de um registrador que retém a instrução corrente enquanto ela é executada. O processador possui instruções de três tipos, uma com dois operandos, uma outra com apenas um operando e a última sem a utilização de operandos.

7	6	5	4	3	2	1	0
opcode							
operando 1							
operando 2							

7	6	5	4	3	2	1	0
opcode							
operando 1							

7	6	5	4	3	2	1	0
opcode							

-	Descrição (2 operandos)
opcode	Código da função a ser executada
operando 1	Valor a ser utilizado na operação
operando 2	Valor a ser utilizado na operação

-	Descrição (1 operando)
opcode	Código da função a ser executada
operando 1	Valor a ser utilizado na operação

-	Descrição (sem operando)
opcode	Código da função a ser executada

No processador proposto os dados possuem 8 Bits e o endereçamento da memória 8 Bits. A memória deverá possuir pelo menos uma entrada de dados e duas saídas de dados. A primeira saída de dados (A_{o0}, M_{o0}) será destinada ao microprocessador e a segunda (A_{o1}, M_{o1}) será utilizada para a verificação do funcionamento do microprocessador. A saída destinada para a verificação terá os seus 8 Bits divididos em 2 grupos de 4 Bits, os quais deverão ser exibidos nos displays de 7-segmentos HEX1,0 através do conversor 7Seg. O microprocessador será implementado no kit DE2, o sinal de *clock* será oriundo do relógio de 27MHz, o sinal de *clock* do sistema deverá ser atribuído ao LEDG[0], a saída da ULA deverá ser direcionada para os LEDR[15:0] e o botão KEY[1] será utilizado para forçar o *reset* do sistema.

Id	Instrução	Descrição
1	MOV reg, regX	reg=regX
2	MOV reg, address	reg=mem[address]
3	MOV reg, constant	reg=constant
4	MOV address, reg	mem[address]=reg
5	MOV address, Constant	mem[address]=constant
1	ADD reg, regX	reg=reg+regX
1	ADD reg, address	reg=reg+mem[address]
1	ADD reg, constant	reg=reg+constant
1	SUB reg, regX	reg=reg-regX
1	SUB reg, address	reg=reg-mem[address]
1	SUB reg, constant	reg=reg-constant
1	AND reg, regX	reg=reg AND regX
1	AND reg, address	reg=reg AND mem[address]
1	AND reg, constant	reg=reg AND constant
1	OR reg, regX	reg=reg OR regX
1	OR reg, address	reg=reg OR mem[address]
1	OR reg, constant	reg=reg OR constant
1	XOR reg, regX	reg=reg XOR regX
1	XOR reg, address	reg=reg XOR mem[address]
1	XOR reg, constant	reg=reg XOR constant
1	SHL reg, regX	reg=reg<<regX
1	SHL reg, address	reg=reg<<mem[address]
1	SHL reg, constant	reg=reg<<constant
1	SHR reg, regX	reg=reg>>regX
1	SHR reg, address	reg=reg>>mem[address]
1	SHR reg, constant	reg=reg>>constant
1	CMP reg, regX	if reg==regX, Z=1
1	CMP reg, address	if reg==mem[address], Z=1
1	CMP reg, constant	if reg==constant, Z=1
1	INC reg	reg=reg+1
1	DEC reg	reg=reg-1
1	NOT reg	reg=not(reg)
1	CALL address	[SP]=PC; SP=SP-1; PC=mem[address]
1	PUSH reg	[SP]=reg; SP=SP-1
1	PUSH address	[SP]=mem[address]; SP=SP-1
1	PUSH constant	[SP]=constant; SP=SP-1
1	POP reg	reg=[SP]; SP=SP+1
1	JMP address	PC=mem[address]
1	JC	PC=mem[address], if carry=true
1	JNC	PC=mem[address], if carry=false
1	JZ	PC=mem[address], if zero=true
1	JNZ	PC=mem[address], if zero=false
1	JA	PC=mem[address], if carry=false e zero=false
1	JNBE	PC=mem[address], if carry=false e zero=false
1	JAE	PC=mem[address], if carry=false
1	JNB	PC=mem[address], if carry=false
1	JB	PC=mem[address], if carry=true
1	JNAE	PC=mem[address], if carry=true
1	JNBE	PC=mem[address], if carry=true ou zero=true
1	JNA	PC=mem[address], if carry=true ou zero=true
1	JE	PC=mem[address], if zero=true
1	JNE	PC=mem[address], if zero=false
1	RET	PC=[SP]; SP=SP+1
1	HLT	parar o processador

Tabela 1: Conjunto de instruções do processador.

1 (2,0) - Projete e implemente um processador no kit DE2 de acordo com as especificações apresentadas.