

**UFPR – Bacharelado em Ciência da Computação**  
**CI210 – Projetos Digitais e Microprocessadores**  
**Trabalho Semestral - Prof. Marco A. Zanata Alves**

O **Mico X1** é um processador extremamente simples que, a cada ciclo, busca e executa uma instrução de lógica/aritmética, ou uma instrução de controle de fluxo (salto/desvio). **Sua tarefa** é implementar o circuito de dados e o de controle da versão X1 do Mico.

Esse processador possui instruções que trabalham com registradores e também com um valor constante que está dentro da instrução. Cada instrução do Mico X1 tem 32 bits e o **formato de cada instrução** é dada pelos campos definidos na Tabela 1.

<b>Código da operação</b>	<b>Registrador a</b> (end. reg. origem)	<b>Registrador b</b> (end. reg. origem)	<b>Registrador c</b> (end. reg. destino)	<b>Constante</b> (valor fixo)
4 bits	4 bits	4 bits	4 bits	16 bits

**Tabela 1:** Formato da instrução do Mico X1.

O **conjunto de instruções** do Mico X1 está definido na Tabela 2.

<b>Op.Code</b>	<b>Instrução</b>	<b>Semântica</b>	<b>Comentário</b>
0	nop	--	No operation
1,2	Op c, a, b	$R(c) \leftarrow R(a) \text{ op } R(b)$	op = {add, sub}
3,4,5	Op c, a, b	$R(c) \leftarrow R(a) \text{ op } R(b)$	op = {and, or, xor}
6	Not c, a	$R(c) \leftarrow \text{not } R(a)$	Complemento
7	sll	$R(c) \leftarrow R(a) \ll R(b)$	Deslocamento lógico esq. *
8	srl	$R(c) \leftarrow R(a) \gg R(b)$	Deslocamento lógico dir. *
9	ori	$R(c) \leftarrow R(a) \text{ OR extZero}(K)$	OR com constante lógica
a	xori	$R(c) \leftarrow R(a) \text{ XOR extZero}(K)$	XOR com constante lógica
b	addi	$R(c) \leftarrow R(a) + \text{extZero}(K)$	SOMA com constante lógica
c	Show a	Display R(a)	Exibe valor na saída
d	Jump E	$IP \leftarrow E$	Salto incondicional
e	Branch a, b, E	$IP \leftarrow ((R(a) == R(b)) ? E : IP + 1)$	Desvio condicional **
f	halt	$IP \leftarrow IP + 0$	Termina a simulação

\*Completa com zeros. \*\*Você pode usar o subtrator da ULA e um fio de (ZERO?) para fazer a comparação  
 $E = IP + \text{SignExt}(\text{Const})$

**Tabela 2:** Conjunto de instruções da arquitetura Mico X1.

O **banco de registradores** contém  $2^4$  registradores de 32 bits, duas portas de leitura (A e B) e uma porta de escrita (C). Sendo que o registrador 0 contém a constante zero ( $R(0)=0$ ). Logo, escritas neste registrador zero não tem nenhum efeito. Em nossa descrição, a,b,c são nomes de registradores (endereços dos registradores) enquanto que A,B,C são os conteúdos dos respectivos registradores ( $R(a)=A$ ).

A **ULA** executa as operações de lógica e aritmética definidas na Tabela 3 que serão úteis para a implementação do processador. Essa ULA deverá ter duas entradas: A e B de 32 bits cada e um seletor. A saída da ULA será o resultado e um fio indicando se houve saída igual a zero (em todos os bits).



~~inicializada no próprio programa. O programa deverá funcionar normalmente mesmo se a constante F for alterada.~~

~~Seu programa deve exibir o operando F (com show), e exibir cada número da sequência com a instrução show. Ao final programa deverá encerrar com a instrução halt.~~

~~Para tanto você deve:~~

- ~~—— (i) escrever o programa iterativo em C;~~
- ~~—— (ii) traduzir este programa para o assembly do Mico;~~
- ~~—— (iii) traduzir o assembly para binário e editar a memória de instruções.~~

~~O IP deve ter 32 bits. Caso você utilize uma memória de instruções com apenas 16bits de endereço (por limitação do software) você deverá ligar apenas os bits menos significativos do PC.~~

## **Regras Gerais de Entrega e Apresentação**

A implementação será feita no simulador **Logisim Evolution**, onde poderão ser utilizados os componentes localizados nas categorias:

- Wiring (fios, splitters, etc.)
- Gates (and, or, not, etc.)
- Plexers (mux, demux, encoders, decoders, etc.)
- Memory (registers, memory, etc.)

Cada aluno fará a entrega individual do circuito e também de um relatório PDF pelo Moodle, contendo:

Nome

Discussão da escolha do somador.

O código teste de instruções,

~~Código fibonacci em C e Assembly.~~

Também deverá ser apresentado no relatório o print do bloco operativo do seu processador implementado no **logisim evolution**.

Esse relatório não deve passar de duas páginas A4 (frente e verso) com fonte tamanho mínimo 11 pontos.

As datas limite de entrega serão sempre às 6h (a.m.) do dia, impreterivelmente. (não confundir com 18h)

Casos não tratados no enunciado deverão ser discutidos com o professor.

Os trabalhos devem ser feitos individualmente. A cópia do trabalho (plágio) não será tolerada e acarretará em nota igual a Zero para todos os envolvidos. É do interesse de todos que alunos conversem sobre o projeto mas cada um deverá desenvolver seu próprio trabalho.

Os trabalhos deverão ser apresentados de forma oral pelo aluno. A nota irá considerar domínio do tema, robustez da solução e rigorosidade da metodologia.

## **Sugestões:**

1. Assegure-se de que entendeu a especificação antes de iniciar o projeto do(s) modelo(s).