

Arquitetura e Organização de Computadores

Conjunto de Instruções da
Arquitetura – CompSim





Agenda

- Operações Básicas de Entrada e Saída
 - Utilizando o Arduino com o CompSim
 - Detalhando a Interface CompSim-Arduino UNO
 - Experimentos



Operações Básicas de Entrada e Saída

- **Utilizando o Arduino com o CompSim**

- Antes de iniciarmos, vamos ao checklist:

- Arduino configurado com o Firmata?
- Arduino conectado à USB do computador?
- CompSim rodando?
- Plataforma Mandacaru criada (menu “Simulator” → “New”)?
- Dominando a interface de I/O do CompSim (Aula 9)?

- Se a resposta for “sim” para todas essas perguntas, vamos então aos experimentos!

Operações Básicas de Entrada e Saída

- Periféricos do CompSim (Recapitulando...)**

Periférico	Modo de Operação	Endereço (AC High)		Dado (AC Low)	Retorno (AC)
		Bin	Dec		
Vídeo	Output (Digital)	00000000	0	Byte	-
Teclado	Input (Digital)	00000001	1	-	Byte
Arduino – Porta B (Pinos 8 a 13)	Input/Output (Digital)	00000010	2	Byte*	Byte
Arduino – Porta D (Pinos 0 a 7)	Input/Output (Digital)	00000011	3	Byte**	Byte
Arduino – Porta C (Pinos A0 a A5)	Input (Analógico)	00000100	4	Byte***	Inteiro ⁺
Arduino – Porta PWM (Pino 3)	Output (Analógico)	00000101	5	Byte	-
Arduino – Porta PWM (Pino 5)	Output (Analógico)	00000110	6	Byte	-
Arduino – Porta PWM (Pino 6)	Output (Analógico)	00000111	7	Byte	-
Arduino – Porta PWM (Pino 7)	Output (Analógico)	00001000	8	Byte	-
Arduino – Porta PWM (Pino 10)	Output (Analógico)	00001001	9	Byte	-
Arduino – Porta PWM (Pino 11)	Output (Analógico)	00001010	10	Byte	-

* Os 2 bits mais significativos são descartados (Mask 00111111).

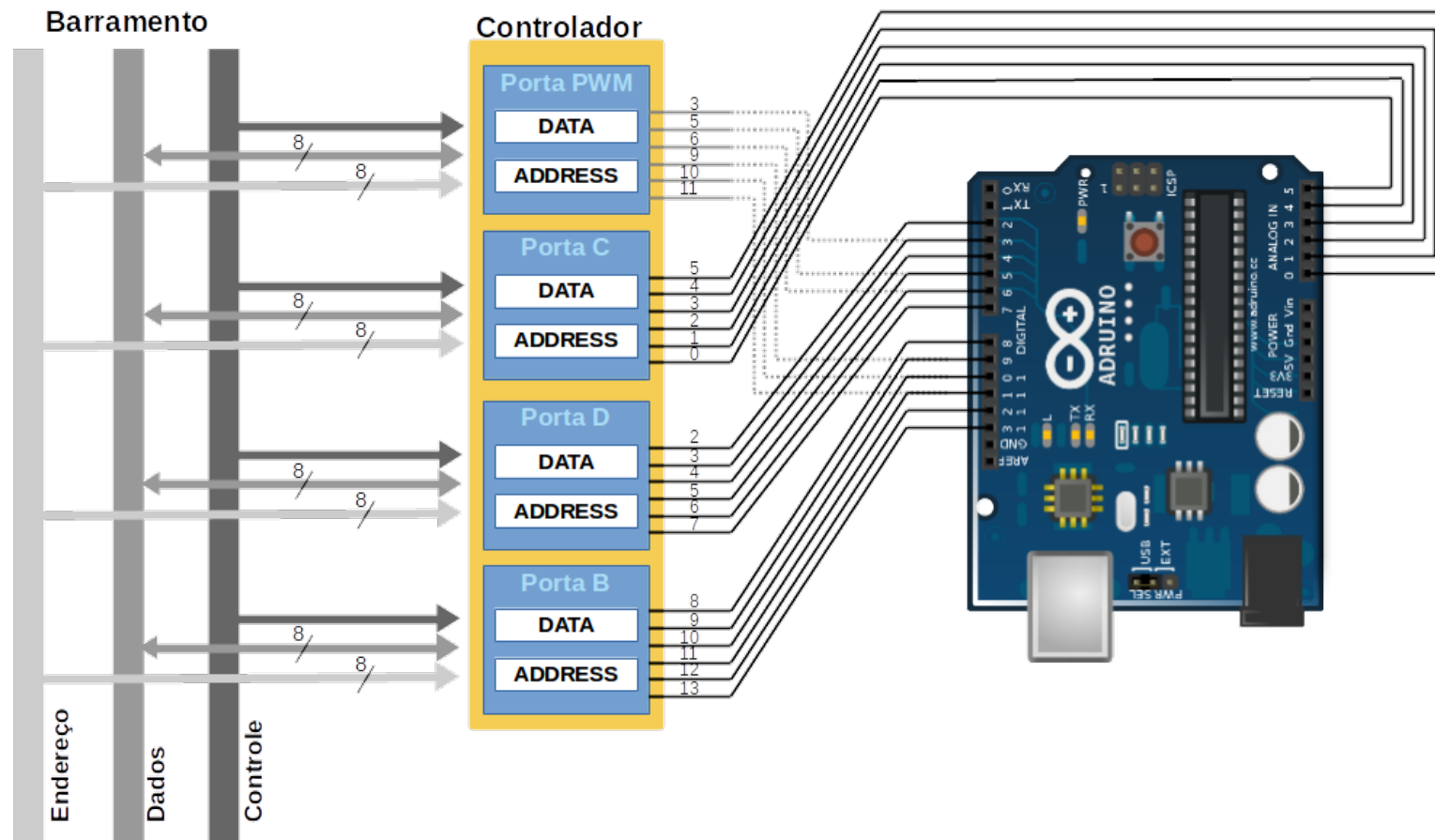
** Os 2 bits menos significativos são descartados (Mask 11111100).

*** Utiliza apenas 3 bits para representar o intervalo de 0 a 5. Os 5 bits mais significativos são descartados (Mask 00000111).

⁺ Utiliza AC High e AC Low para retornar um inteiro entre 0 e 1024.

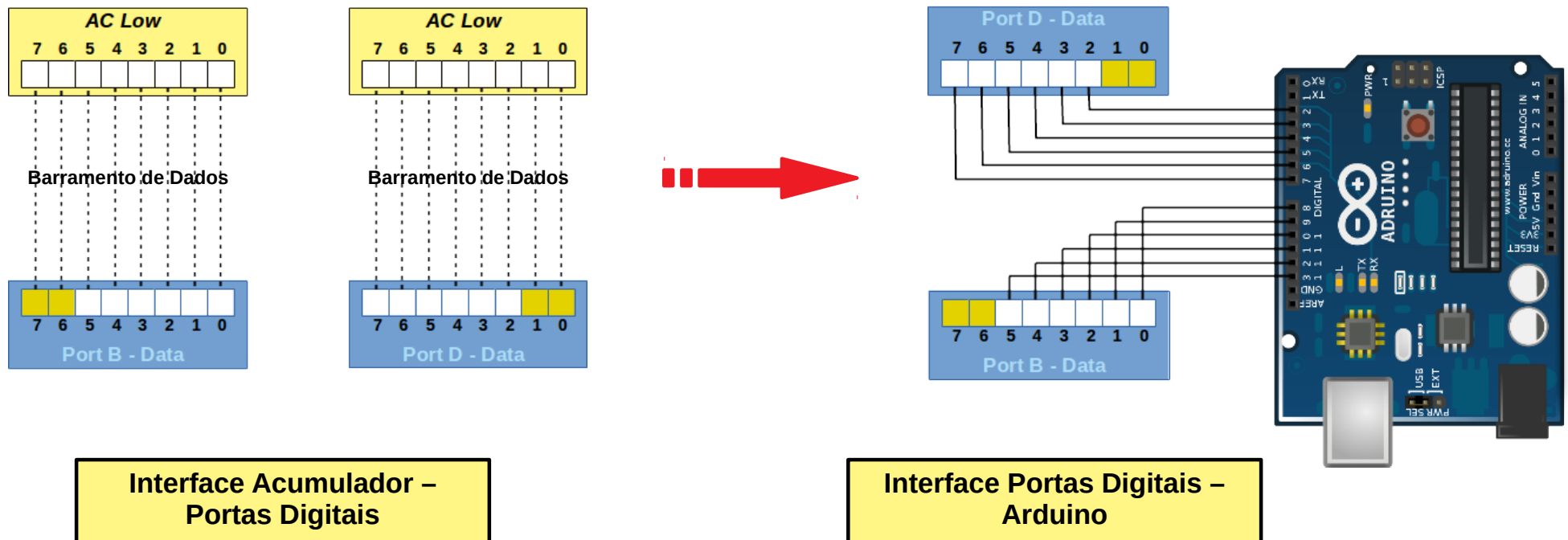
Operações Básicas de Entrada e Saída

- Detalhando a Interface CompSim-Arduino UNO**



Operações Básicas de Entrada e Saída

- **Detalhando a Interface CompSim-Arduino UNO**
 - Entrada e Saída Digital





Operações Básicas de Entrada e Saída

- **Utilizando o Arduino com o CompSim**

- **Experimento 0:** Blink do led *built-in*

- Na board Arduino UNO há um led conectado ao pino 13.
- Isso quer dizer que, quando o pino 13 estiver setado, o led estará acesso. Quando resetado, estará desligado.
- Assim, no CopmSim devemos realizar uma operação de escrita na port B, setando o bit correspondente, para acender o Led.
- No caso, deve ser setado o 6o. Bit do AC Low (veja a ilustração ao lado).



Operações Básicas de Entrada e Saída

• Utilizando o Arduino com o CompSim

– Experimento 0: Blink do led *built-in*

- **LDA** liga

AC High								AC Low							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								0	0	1	0	0	0	0	0

- **ADD** portB

AC High								AC Low							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0

- **INT** output

AC Low							
7	6	5	4	3	2	1	0
0	0	1	0	0	0	0	0

		1	0	0	0	0	0
13	12	11	10	9	8		
Port B - Data							

```
.code

blink:
    LDA liga
    ADD portB
    INT output
    ;encerra a aplicacao
    INT exit

.data
    ;seta 6o. bit
    liga: DD 00100000b

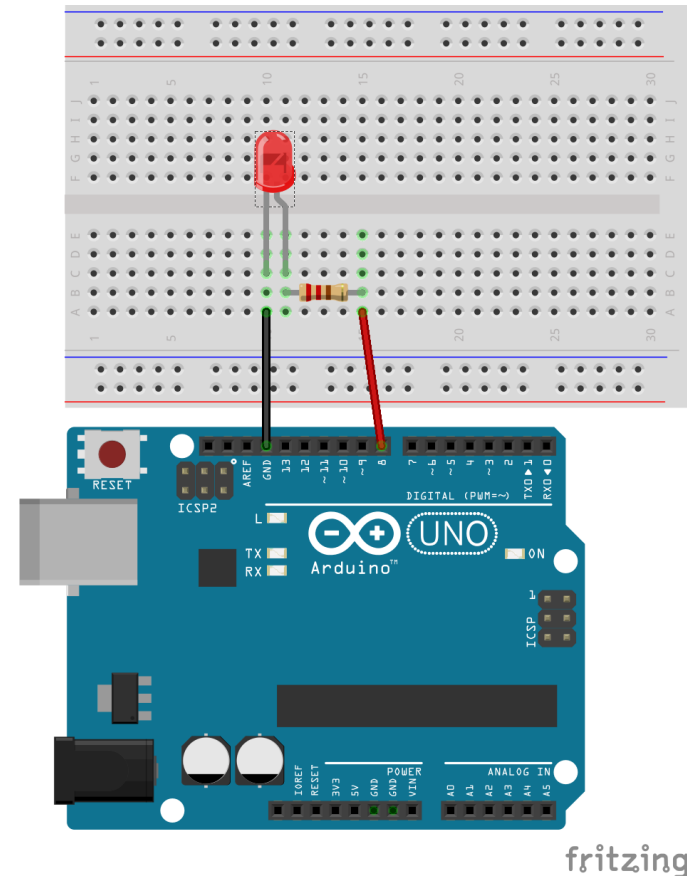
    ;Arduino port B:
    ; AC High  Ac Low
    ; 00000010 00000000
    portB: DD 000000100000000b

    ;syscall output
    output: DD 21
    ;syscall exit
    exit: DD 25

.stack 1
```


Operações Básicas de Entrada e Saída

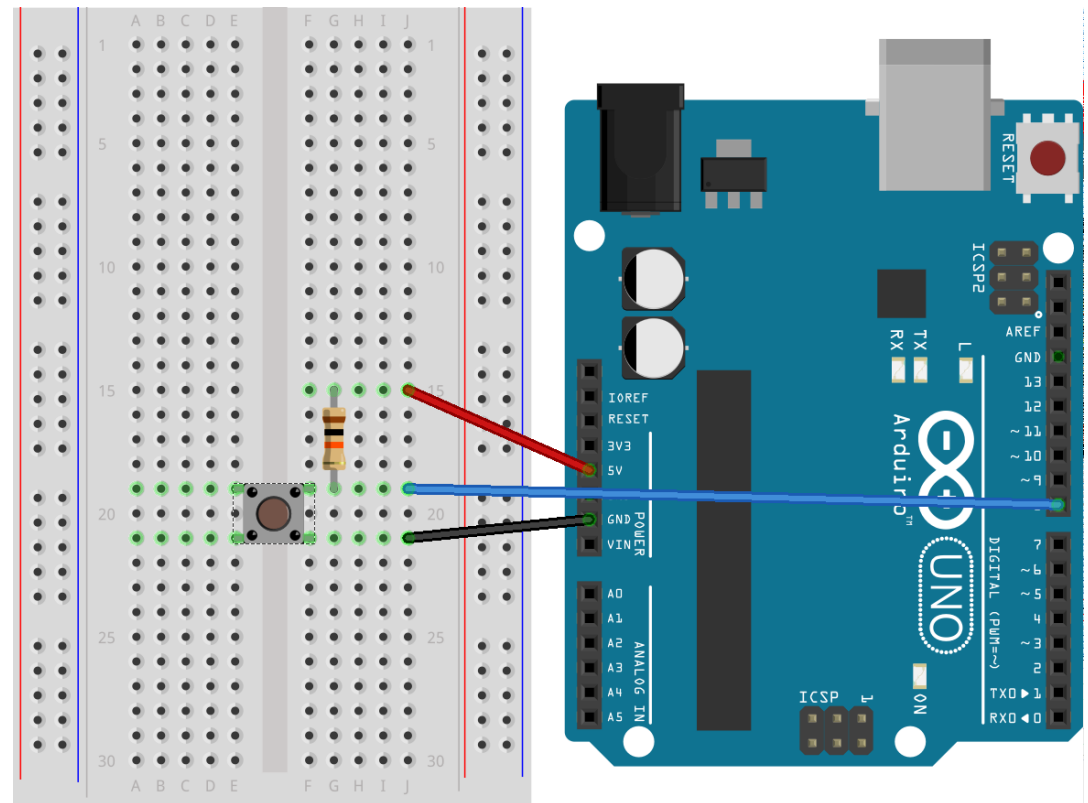
- **Utilizando o Arduino com o CompSim**
 - **Experimento 02:** Blink de led externo
 - Materiais:
 - Protoboard
 - Led
 - Resistor 220 Ohm
 - Código:
 - `blink_external_led.asm`



Operações Básicas de Entrada e Saída

- **Utilizando o Arduino com o CompSim**
 - **Experimento 03:** Lendo status de botão:

- Materiais:
 - Protoboard
 - Pushbutton
 - Resistor 10k Ohm
- Código:
 - pushbutton.asm



Atividades Práticas

- **Criar os seguintes experimentos:**

- 1) Acender um LED enquanto um botão estiver pressionado.
- 2) Acender um LED ao pressionar um botão e, ao pressioná-lo novamente, desligar o LED.
- 3) Fazer um LED ficar em modo “pisca-pisca” ao pressionar um botão e, ao pressioná-lo novamente, desligar o LED.

