

## Projeto 2 - Controlador Proporcional

### Objetivo:

Projetar um sistema de controle de temperatura capaz de aquecer e resfriar o ambiente baseado na temperatura atual e na temperatura programada. O controlador deverá ser do tipo Proporcional e as saídas para o drive do tipo PWM. Um led de vida do sistema (Heart Beat), deverá oscilar em 0,5Hz sempre que o sistema estiver energizado e o drive estiver habilitado. Caso driver seja desabilitado o Led deverá permanecer aceso.

### Diagrama de Blocos do Sistema:

O diagrama de blocos do sistema é apresentado na figura 1.

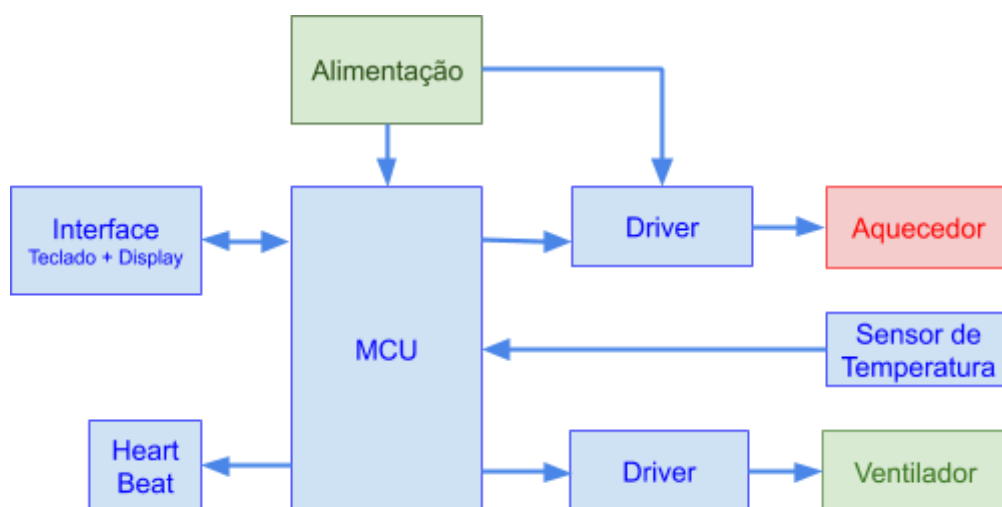


Figura 1 - Diagrama de Blocos do Sistema

### Interface:

O bloco de interface deve conter um display de LCD 16x2 e 4 botões. O display deverá apresentar a temperatura atual, a temperatura desejada (set-point) e o estado atual da saída (aquecendo, resfriando, desabilitado). Os botões são utilizados para alterar os parâmetros do controlador e valor de set point:

#### Teclado:

P: Programar/Alternar telas.

+: Incrementa um valor numérico, modifica uma saída para ON.

-: Decrementa um valor numérico, modifica uma saída para OFF.

S: Confirma o ajuste atual.



Os botões do teclado servem para ajustar:

- Um novo set-point.
- Valor do ganho proporcional do controlador.
- Desabilitar o controlador (desligar o driver independente do ajuste de temperatura)
- Ligar/Desligar manualmente o aquecedor.
- Ajustar o valor da saída do aquecedor no modo manual (0...100%).
- Ligar/Desligar manualmente o ventilador.
- Ajustar o valor da saída do ventilador no modo manual (0...100%).

Display:

O projeto deve apresentar (no mínimo) as seguintes telas:

Tela 1: Estado Atual do Sistema

SP: Set Point (Valor ajustado de temperatura)

PV: Process Value (Valor atual de temperatura)

Heat: Aquecendo (Ligado = ON, Desligado = OFF)

Driver (Habilitado = ON, Desabilitado = OFF)



SP: 28.0C      Heat  
PV: 14.2C      ON

Tela 2: Ajuste do Set Point

SP: Set Point (Valor ajustado de temperatura)



New Set Point  
SP: 28.0

Tela 3: Ajuste do Ganho Proporcional

Kp: Proportional Gain (Valor do ganho proporcional)



New Prop. Gain  
Kp: 1.0

Tela 4: Modo Manual Aquecedor

Ex.1:

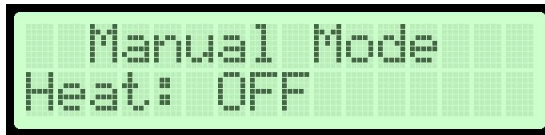
Estado: ON = Aquecedor Ligado, saída a 50%.



Manual Mode  
Heat: ON      50%

Ex.2:

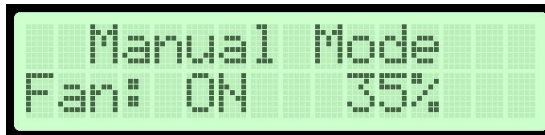
Estado: OFF = Aquecedor desligado, valor da saída não deve aparecer.



Tela 5: Modo Manual Ventilador

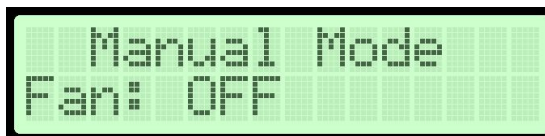
Ex.1:

Estado: ON = Ventilador Ligado, saída a 35%.



Ex.2:

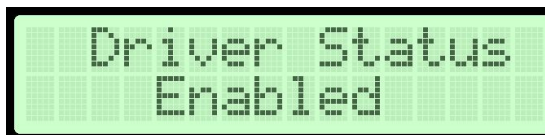
Estado: OFF = Ventilador desligado, valor da saída não deve aparecer.



Tela 6: Desabilitar Driver de saída

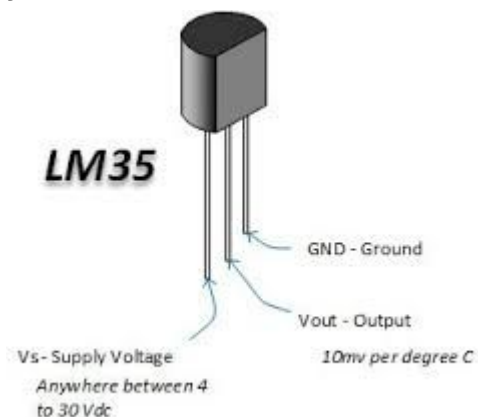
Ex. 1:

Driver habilitado, tanto para o aquecedor quanto para o ventilador



Sensor de Temperatura:

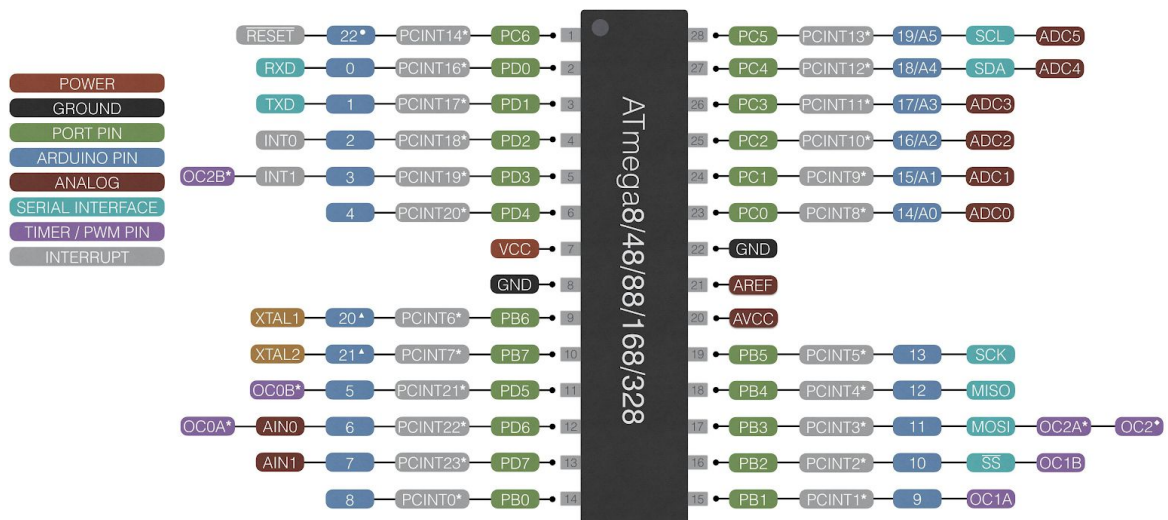
O Sensor de temperatura que deverá ser utilizado é o LM35, este sensor é do tipo analógico com a saída variando 10mV/°C, procure pelo datasheet para maiores informações.



## MCU

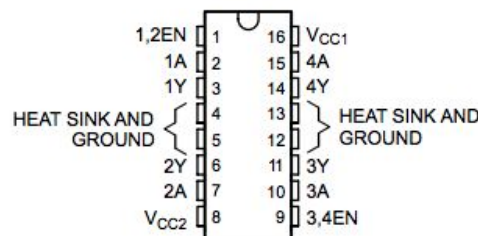
O MCU para a simulação pode ser escolhido pelo aluno, algumas opções disponíveis no Proteus são, PIC, ATMEGA (Arduino), PICCOLO (TMS320F28027), MCS51 e ARM Cortex-M3. O microcontrolador deverá ser responsável pelas tarefas de controle e interface do sistema. Na montagem prática todos devem utilizar o ATMEGA328P (Arduino).

## ATmega8/48/88/168/328 DIP pinout



## Driver

O driver utilizado é circuito integrado L293D - Quadruple Half-H Drivers, pinagem mostrada abaixo:

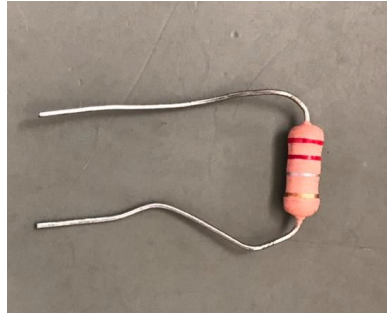


Pin Functions

PIN		TYPE	DESCRIPTION
NAME	NO.		
1,2EN	1	I	Enable driver channels 1 and 2 (active high input)
<1:4>A	2, 7, 10, 15	I	Driver inputs, noninverting
<1:4>Y	3, 6, 11, 14	O	Driver outputs
3,4EN	9	I	Enable driver channels 3 and 4 (active high input)
GROUND	4, 5, 12, 13	—	Device ground and heat sink pin. Connect to printed-circuit-board ground plane with multiple solid vias
V <sub>CC1</sub>	16	—	5-V supply for internal logic translation
V <sub>CC2</sub>	8	—	Power VCC for drivers 4.5 V to 36 V

### Aquecedor:

O aquecimento é realizado com um resistor de  $22\Omega$  1/2W.



Calcule a tensão máxima que pode ser aplicada nesse resistor e limite esse valor no Firmware.

$$\text{Equação: } P = \frac{V^2}{R}$$

### Ventilador:

O ventilador é composto por um motor e uma hélice, o modelo utilizado tem as seguintes características:

Alimentação: 12 VDC

Corrente: 130mA

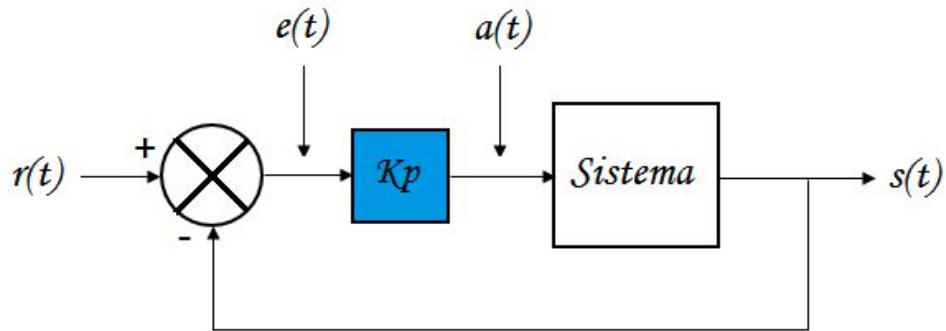


### Heart Beat:

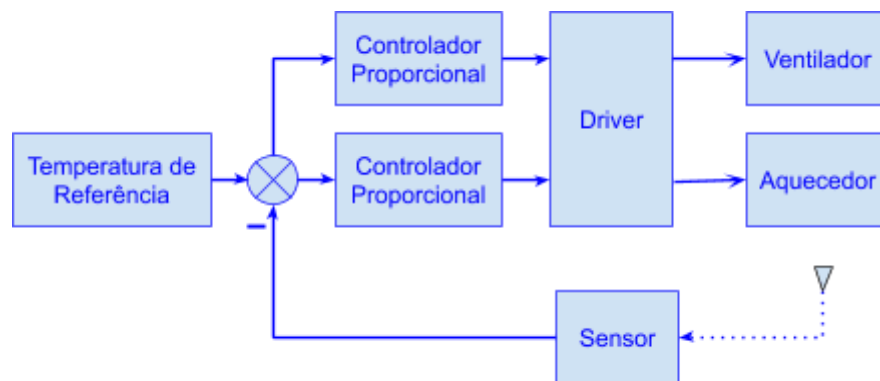
Um LED deverá ser utilizado para informar que o sistema está em execução. O LED deverá permanecer aceso caso o driver de saída esteja desabilitado.

## Controlador:

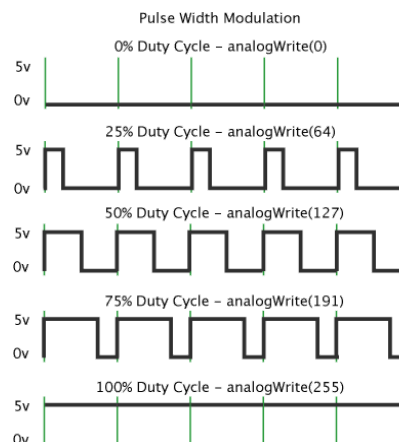
O controlador utilizado será do tipo Proporcional, neste controlador o sinal de erro, diferença entre o sinal de referência e o sinal de realimentação, é multiplicado pelo ganho proporcional ( $K_p$ ) e aplicado a saída.



O diagrama de blocos do controlador é mostrado abaixo:



A implementação do controlador deverá levar em conta os limites físicos das saídas (Saturação), lembrando que por padrão as saídas PWM do ATMEGA328P variam de 0 (0%) à 255 (100%) (\*para a função analogWrite). Ver imagem abaixo:



## Resultados:

Cada aluno deverá entregar:

- Desenho esquemático em formato PDF (0,1).
- Projeto do proteus com a simulação (0,1).
- Firmware com código modularizado utilizando a IDE Sloeber (0,4).
- Apresentar a montagem física do sistema (0,2).
- Fazer upload do firmware no github, colocar no arquivo README: (0,2)
  - Descrição do projeto
  - Diagrama esquemático
  - Fotos da montagem prática

Recomendações para o Firmware:

- Divida o código em 'pedaços' que possam ser testados separadamente.
- Fique atento às questões de hardware (pinos PWM, Analógicos, Digitais...).
- Modularizar o código (dividir em arquivos .h e .c ou .cpp)
- Utilize comentários em seções relevantes.
- Procure utilizar a abstração de Máquina de Estados.
- Utilize constantes na criação das telas para o display.
- Processe uma entrada do teclado somente quando necessário ou escolha uma taxa de atualização coerente.

Este projeto vale como a primeira nota da disciplina de Automação 1.