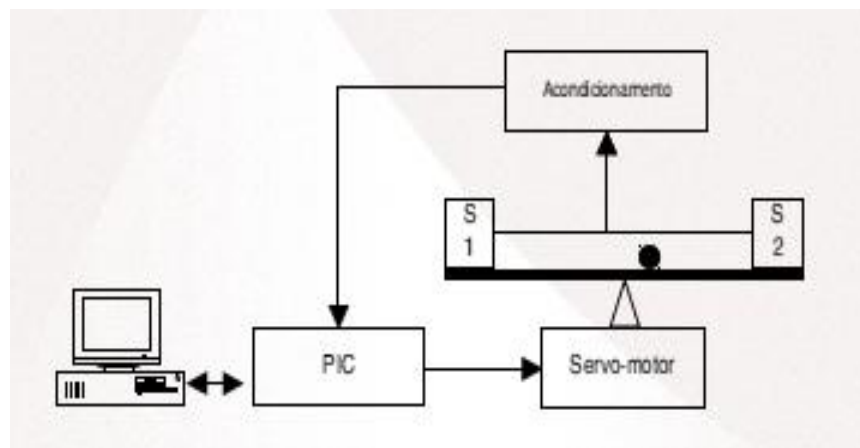


# RELATÓRIO DE TRABALHO PRÁTICO DE ELETRÓNICA IV



## *TRABALHO 8 - BOLA NA CALHA*

André Vicente      N° 74228

Ricardo Pina      N° 59796



# Índice

<b>1. Introdução</b>	<b>1</b>
<b>2. Projeção inicial do <i>hardware</i></b>	<b>2</b>
<b>3. Eletrônica de acondicionamento</b>	<b>3</b>
3.1. Dimensionamento do circuito de acondicionamento . . . . .	3
3.1.1. Ensaio prático com valores de resistências calculados . . . . .	4
3.2. Dimensionamento do regulador de tensão de referência . . . . .	5
<b>4. Estruturação e desenvolvimento de código</b>	<b>8</b>
4.1. Aquisição de pontos . . . . .	8
4.2. Temporizador do <i>PID</i> . . . . .	8
4.3. Comunicação Tensão e Posição . . . . .	8
<b>5. Controlador <i>PID</i></b>	<b>10</b>
<b>6. Conclusão</b>	<b>12</b>
<b>A. Diagrama de blocos do projeto</b>	<b>15</b>
<b>B. Circuito Elétrico</b>	<b>17</b>

# Lista de Figuras

2.1. Diagrama de blocos do <i>hardware</i> . . . . .	2
3.1. Capturas de osciloscópio com a bola no extremo longínquo(esq.) e próximo(dir.). . . . . .	3
3.2. Amplificador Diferencial. . . . .	4
3.3. Captura de osciloscópio de $V_{ref}$ . . . . .	5
3.4. Tensão de referência. . . . .	5
3.5. Capturas de osciloscópio com a bola no extremo longínquo (esq.) e próximo (dir.) . . . . .	7
5.1. Diagrama de blocos de <i>PID[3]</i> . . . . .	10

## Lista de Tabelas

3.1. Parâmetros de entrada para cálculo de $R_4$ . . . . .	4
3.2. Parâmetros de entrada para dimensionamento do regulador de tensão. . . . .	6
3.3. Valores de resistências calculadas para o regulador de tensão. . . . .	6

# 1. Introdução

No âmbito da componente prática da disciplina Eletrónica IV, pretende-se construir um sistema com a capacidade de controlar a posição de uma bola colocada numa calha, cuja inclinação é manipulada por um servo-motor.

A posição da bola é medida por 2 sensores de distância por infravermelhos. O sistema de medição deve ter a capacidade medir posição da bola com precisão de  $1\text{ mm}$ .

A posição da bola deve ser regulada por um controlador *PID* implementado dentro de um *MCU*<sup>1</sup>, com a capacidade de ajustar a posição com a precisão igualmente de  $1\text{ mm}$ . O *Setpoint* (posição pretendida) é regulado pelo utilizador que digita o seu valor num *PC* que está ligado ao *MCU* via *RS232*. O *MCU* usado é o *PIC32MX795F512H*.

O trabalho divide-se em 3 Fases:

- I. Realizar um diagrama de blocos de todo o projeto, esboçar o circuito elétrico inicial, verificação dos sinais medida nos intervalo de distâncias em questão e definição das variáveis de principais de controlo (por exemplo frequência de sinal a enviar para o servomotor).
- II. Escrever *device-drivers*, montar circuitos elétricos, construir controlador *PID*, e estabelecer a interface com o operador. Nesta mesma fase inicia-se o relatório. Poderão ser feitas correções no trabalho realizado na Fase I.
- III. Concluir toda a parte prática do trabalho, assim como se termina o relatório e escreve-se o manual de utilizador.

No início do projeto realizou-se um diagrama de blocos de todo o trabalho a desenvolver ao longo das aulas. Este diagrama encontra-se no anexo A.

---

<sup>1</sup>*Microcontroller Unit*

## 2. Projeção inicial do *hardware*

Como já referido e exigido, na Fase I é exigido a elaboração de um diagrama de blocos do *hardware* de todo o projeto. Isto implica que é necessário definir quais os processos (blocos) terão que ser realizados por cada elemento do *hardware*.

Na figura é ilustrado o diagrama de blocos de todo o *hardware* do projeto.

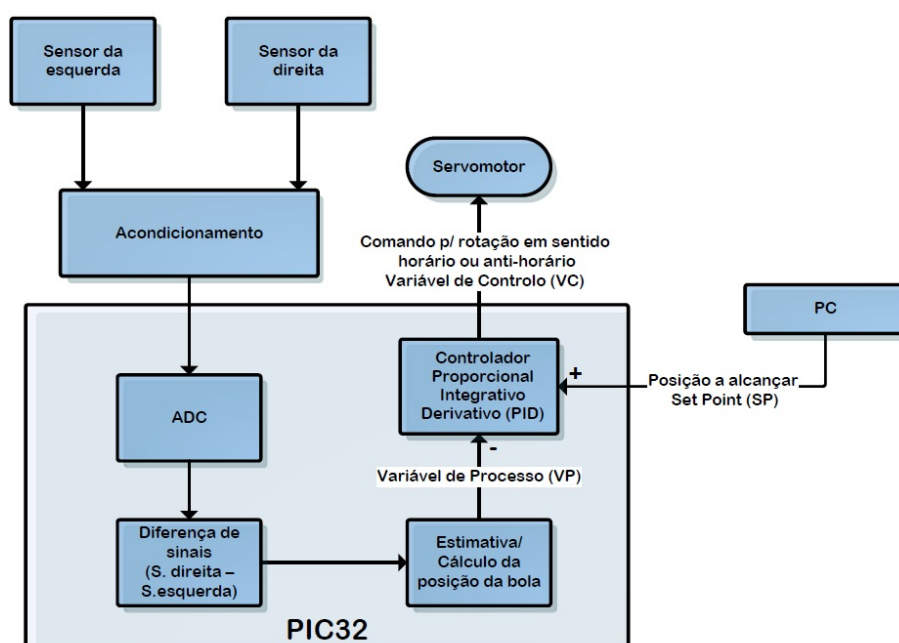


Figura 2.1.: Diagrama de blocos do *hardware*

Tal como demonstra o diagrama de blocos os sensores não serão ligados diretamente ao *PIC*, será realizado um acondicionamento de sinais, como justificado e explicado na capítulo 3.

No *PIC* será feita a conversão analógico para digital dos sinais de ambos sensores de distância, o cálculo da posição. A abordagem destes processos será explicada no capítulo 4. Outro importante processo realizado no *PIC* é controlador *PID*, processo este que é explicado no capítulo 5.

### 3. Eletrônica de acondicionamento

Os sensores de distância disponíveis no *Set-Up* do projeto atual, para as distâncias a medir apresentam uma tensão máxima de mínima de 3,1 V e 0,8 V respectivamente. Este últimos valores foram retirados de medições realizadas na prática conforme ilustra a figura 3.1.

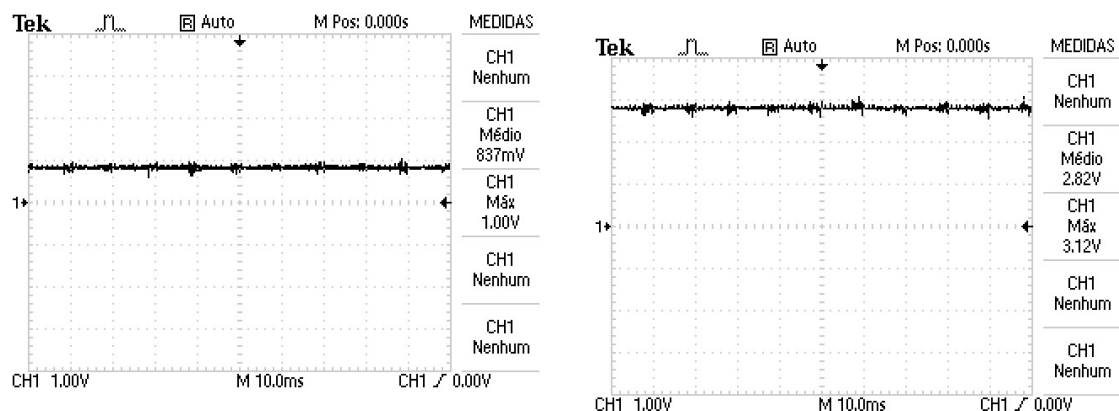


Figura 3.1.: Capturas de osciloscópio com a bola no extremo longínquo(esq.) e próximo(dir.).

Para conseguir aproveitar de toda excursão de sinal admissível nos pinos do *PIC32MX795F512H*, que está entre os 0 V e 3,3 V, realizou-se a subtração e amplificação dos sinais provenientes dos sensores de distância com amplificadores diferenciais (uma para cada sensor), como se verifica no anexoB. A subtração de sinal, que tem o objetivo de remover a componente *DC* do sinal de entrada, exige uma tensão de referência.

Outro facto signficante no acondicionamento que se pode observar, é a existência de um *buffer* para cada sensor distância. Estes *buffers* são necessários pois o efeito de carga nos circuitos de acondicionamento têm um efeito significativo nos níveis de tensão do sensores, como verificado em ensaios práticos.

#### 3.1. Dimensionamento do circuito de acondicionamento

Na figura 3.2, ilustra-se um exemplo de amplificador diferencial usado no presente projeto.



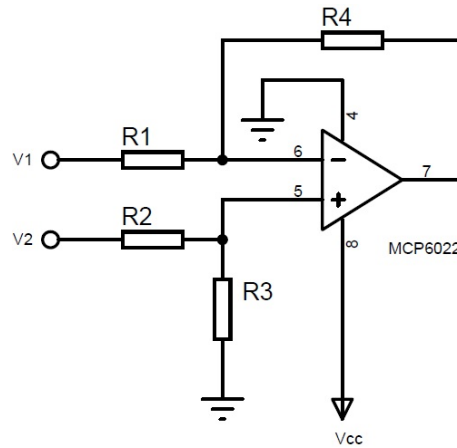


Figura 3.2.: Amplificador Diferencial.

Notar que o *OPAMP MCP6022* trata-se de um *OPAMP Rail-to-Rail* [2], o que significa que as suas tensões de saturação são as mesmas que as tensões de alimentação. A necessidade de usar *OPAMP's* deste tipo deve-se ao facto de estes deverem saturar negativamente a  $0\text{ V}$ , porque só assim se justifica usar um circuito de acondicionamento pois os sensores têm um tensão mínima baixa ( $0,8\text{ V}$ ).

A equação do sinal de saída é dada pela expressão explícita na equação 3.1.1.

$$V_o = \frac{R_3}{R_1} \cdot (V_2 - V_1) \quad \left| \begin{array}{l} R_1 = R_2 \\ R_3 = R_4 \end{array} \right. \quad (3.1.1)$$

Pela equação sabe-se que para conseguir uma tensão mínima de saída de  $0\text{ V}$ , é necessário que a tensão de referência  $V_1$  tenha o valor da componente *DC* de  $V_2$  ( $0,8\text{ V}$ ).

Na tabela 3.1, ilustram-se os parâmetros assumidos e considerados no dimensionamento do amplificador diferencial.

Tabela 3.1.: Parâmetros de entrada para cálculo de  $R_4$ .

$R_3$ [ $\Omega$ ]	$37\text{ k}$
$V_1$ [V]	$0,8$
$V_2$ [V]	$3,1$
$V_o$ [V]	$3,3$

Pela equação 3.1.1, pelos valores da tabela 3.1 calculou-se que  $R_1 = 25,8\text{ k}\Omega$ .

### 3.1.1. Ensaio práticos com valores de resistências calculados

Com vários ensaios práticos com associações de resistências o mais próximo possível ao calculado, verificou-se que a tensão de saída mínima (quando a bola está no extremo mais afastado do sensor) era significativamente superior a  $0\text{ V}$  e que a tensão máxima estava

próximo do pretendido. Para solucionar este último problema aumentou-se  $V_1$  para  $1\text{ V}$ , conforme ilustra a figura 3.3, e o valor de  $R_1$  foi aumentado de forma a que a subida de  $V_1$  não se refleti-se na tensão máxima de saída.

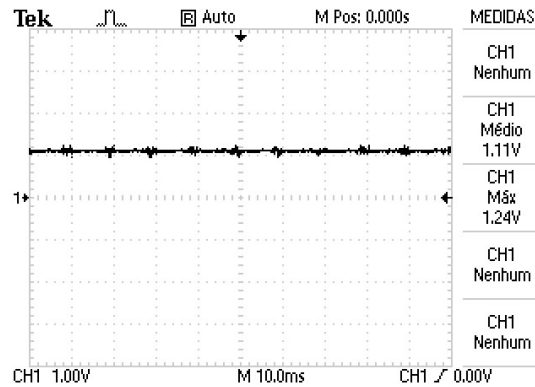


Figura 3.3.: Captura de osciloscópio de  $V_{ref}$

Os valores das resistências usadas na prática podem ser consultadas no circuito elétrico contido no anexo B.

## 3.2. Dimensionamento do regulador de tensão de referência

Para obter a tensão de referência necessária, recorreu-se ao regulador de tensão de referência paralelo *TL431*. A implementação deste componente é ilustrada na figura 3.4. Esta implementação foi baseada num exemplo referido no *datasheet* [1] deste componente.

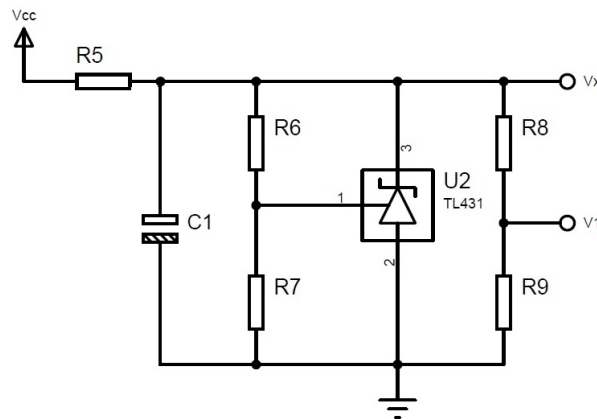


Figura 3.4.: Tensão de referência.

Segundo o *datasheet* deste componente a tensão de saída é dada pela expressão 3.2.1.

$$V_x = \left(1 + \frac{R_6}{R_7}\right) \cdot 2,5 \quad (3.2.1)$$

Pela expressão anterior verifica-se que apenas por este circuito é impossível obter a tensão de referência de  $0,8\text{ V}$ . Para resolver este último problema colocou-se um divisor resistivo

na saída, sucedido por um *buffer*, como se verifica na figura anterior e no circuito global contido no anexo B.

O dimensionamento deste circuito teve em conta as seguintes afirmações referidas no *datasheet* do *TL431* [1].

- Segundo o *datasheet* do *TL431* a corrente de cátodo do componente deve ser superior a  $1\text{ mA}$ .
- Este circuito integrado garante uma tensão aproximadamente de  $2,5\text{ V}$  no terminal 1 (neste caso aos terminais de  $R_7$ ).
- A corrente no pino 1 do *TL431* tem valores na ordem dos  $\mu\text{A}$  e assim esta é desprezada.
- A corrente de entrada no *buffer* é igualmente desprezada.

$R_5$  deve ter um valor que garanta a mínima corrente de cátodo. Para tal deve cumprir a equação 3.2.2.

$$\frac{V_{cc} - V_x}{R_5} - I_{R6} - I_{R8} \geq 1\text{ mA} \Leftrightarrow R_5 \leq \frac{V_{cc} - V_x}{I_{R6} + I_{R8} + 1} \quad (3.2.2)$$

Por fim, a tensão de saída é dada pela equação 3.2.3.

$$V_1 = R_9 \cdot I_{R9} \quad (3.2.3)$$

Na tabela 3.2, ilustram-se os parâmetros assumidos e considerados no dimensionamento do regulador de tensão.

Tabela 3.2.: Parâmetros de entrada para dimensionamento do regulador de tensão.

$I_{R6} [\text{mA}]$	0,29
$I_{R8} [\text{mA}]$	0,13
$V_{cc} [\text{V}]$	8
$V_x [\text{V}]$	3
$V_1 [\text{V}]$	0,8

Tendo em conta as equações 3.2.1, 3.2.2 e 3.2.3 e as considerações da tabela 3.2, obtêm-se os valores das resistências que estão registados na tabela 3.3.

Tabela 3.3.: Valores de resistências calculadas para o regulador de tensão.

$R_5(\text{máximo}) [\text{k}\Omega]$	1,72
$R_6 [\text{k}\Omega]$	0,14
$R_7 [\text{k}\Omega]$	8,62
$R_8 [\text{k}\Omega]$	16,9
$R_9 [\text{k}\Omega]$	6,15

Os valores das resistências usadas na prática podem ser consultadas no anexo B. Este últimos valores diferem dos calculados devido à associação de resistências que se realizou

é impossível conseguir valores resistivos iguais ao necessário. Outro motivo é que a tensão de referência  $V_1$  tinha de ser superior ao calculado, como já referido anteriormente.

Com este circuito conseguiu-se ampliar a gama conforme ilustra a figura 3.5.

No entanto salienta-se que o valor de tensão máximo está um pouco alto, porque os sensores são influenciados pela luminosidade e assim medem-se valores diferentes em cada ensaio. Acabou-se por concluir que o valor apresentado não é prejudicial. Outro facto que infere perturbações no sinal são as oscilações da calha.

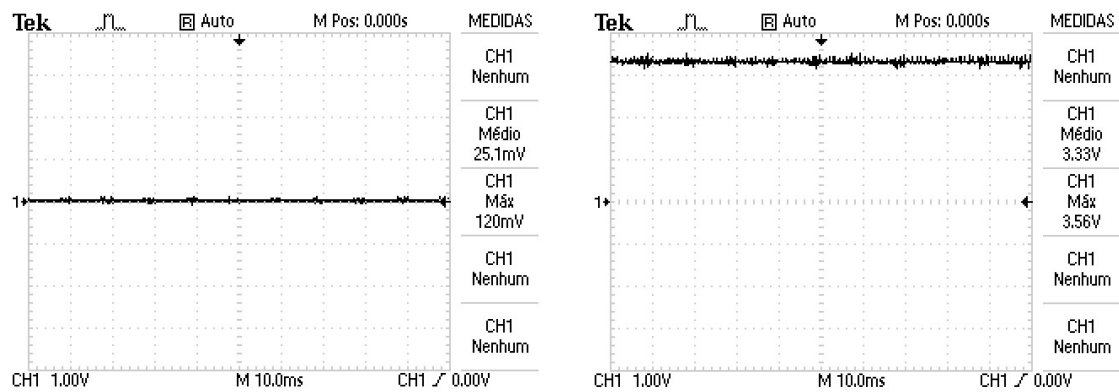


Figura 3.5.: Capturas de osciloscópio com a bola no extremo longínquo (esq.) e próximo (dir.) .

## 4. Estruturação e desenvolvimento de código

Na elaboração do trabalho em *C* usou-se o compilador XC32. O trabalho tem diversos aspetos que devem ter especial atenção. A não linearidade dos valores dos sensores o ruído existente na leitura e a utilização de temporizador para aquisição do valor instantâneo de cada sensor, são alguns dos pontos com os quais teve-se especial atenção.

### 4.1. Aquisição de pontos

Na aquisição de pontos deparou-se com algum ruído existente, para que tal seja minimizado, optou-se por, em cada ponto obtido, o *PIC* adquirir 10 valores seguidos de tensão, e será feita a respetiva média. Com isto minimizou-se o ruído existente no valor lido.

Esta aquisição torna-se necessária, porque a variação da tensão ao longo da “régua” não é linear. Assim sendo, optou-se por fazer aproximações lineares de primeiro grau. Para que tal seja possível, é necessário, sempre que o *PIC* é ligada, obter-se determinados pontos e, com estes, seja possível calcular 6 retas, com intervalo entre cada ponto medido de 50 *mm*, num total de 7 pontos.

Sendo dois sensores, iria obter-se dois gráficos. Simplificando um pouco o código, e deste modo tentar diminuir o erro obtido. Optou-se por fazer a leitura dos dois sensores, e subtrair o valor de tensão lido pelo sensor da direita, pelo da esquerda. Com isto obteve-se uma só reta.

### 4.2. Temporizador do *PID*

Usar um temporizador também se torna importante, com valor muito elevado, faz com que o funcionamento seja lento, difícil de coincidir com o ponto pretendido, e um valor baixo demais pode tornar o funcionamento do *PIC* muito “pesado”. Por isso optou-se por um valor de 10ms para o temporizador de aquisição de valores do sensor.

### 4.3. Comunicação Tensão e Posição

A leitura é feita por uma função denominada *mon\_getc* que utiliza *URXDA* para ler o valor recebido. Na comunicação e leitura dos valores *ADC*, usou-se um carácter auxiliar para as

diferentes *strings* enviadas para o *PIC*. Para analisar o que se pretende receber é comparado o primeiro carater com o “+” ou “p”, caso seja valor de gráfico ou posição de destino pretendida respetivamente. Usou-se a instrução *atoi* para converter o valor da *string* para inteiro.

## 5. Controlador *PID*

Inicialmente para elaborar o controlador para a nosso trabalho, optou-se por usar apenas a *PD*, sendo que por análise teórica seria possível. A variável *P* é responsável pela velocidade de resposta do sistema, enquanto que a componente *D* faz com que reaja mais fortemente à mudança no parâmetro do erro e por sua vez a velocidade de resposta seja maior. A primeira componente depende apenas do erro atual e da variável do processo, enquanto que a *D* depende da variável de processo, e da diferença do erro atual e o anterior. O controlador atua mediante a taxa de variação de erro, atua sobre transitórios, produzindo amortecimento no sistema.

No entanto com as experiências práticas acabou-se por concluir que a utilização da parte integrativa é importante para, em regime permanente, diminuir o erro e aproximar a bola da posição pretendida. O resultado mesmo com um pequeno erro, fará com que a componente integral aumente lentamente. Por fim limitou-se a variável *u* (*Output*), de tal forma a que nos extremos a calha não oscile excessivamente.

A figura 5.1 demonstra um diagrama de blocos de um controlador *PID*.

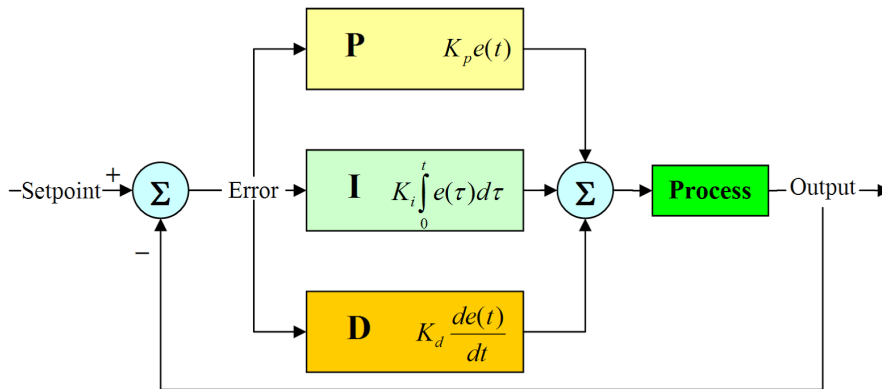


Figura 5.1.: Diagrama de blocos de *PID*[3].

Verifica-se que a variável de entrada do controlador é o erro (diferença entre *Setpoint* e variável de processo ).

As componentes *P*, *I* e *D* são as componentes, proporcional, integrativa e derivativa respectivamente.  $k_p$ ,  $k_d$  e  $k_i$  são o ganho proporcional, derivativo e integrativo do controlador.

$k_p$  é ajustado diretamente, enquanto que o ganho derivativo  $k_d$  depende de  $T_d$  e  $T_s$  e  $k_p$ . Esta última relação é explícita na equação 5.0.1. O mesmo se passa com  $k_i$  que depende de  $k_p$ ,  $T_s$  e  $T_i$ , com a relação dada pela equação 5.0.2.

$$k_d = k_p \cdot \frac{T_d}{T_s} \quad (5.0.1)$$

$$k_i = k_p \cdot \frac{T_s}{T_i} \quad (5.0.2)$$

- $T_s$  é o valor do temporizador *PID*.
- $T_d$  é o tempo de diferenciação.
- $T_i$  é o tempo de integração.

Desenvolvendo e simplificando por aproximações das equações da figura 5.1, adquiri-se a equação 5.0.3 para  $P, I$  e  $D$ .

$$u(t) = K_p \left( e(t) + \frac{1}{T_i} \int e(\sigma) \cdot d\sigma + T_d \cdot \frac{de(t)}{dt} \right) \quad (5.0.3)$$

Simplificando adquiri-se a equação 5.0.4:

$$u(n) = K_p \cdot e(n) + k_i \cdot \sum e(k) + k_d \cdot (e(n) - e(n-1)) \quad (5.0.4)$$



## 6. Conclusão

O diagrama de blocos elaborou-se corretamente, sendo esta uma fase inicial importante para todo o desenrolar do trabalho.

Verificou-se a importância do acondicionamento de sinal, para aumentar o intervalo de valores lidos, que inicialmente não abrangia toda a gama do pino do *PIC* (*AN0,AN1*).

A utilização do *interface* foi importante, porque simplificou-se a construção do gráfico, e envio da posição de destino, bem como a leitura de alguns valores (posição de bola estimada e variável de saída do controlador) em tempo real.

Verificou-se que a subtração dos valores de tensão atenuou um pouco o erro, sendo que na posição extrema o sensor mais distante tem pouca influência no valor, enquanto que na proximidade da posição central ambos contribuem de forma significativa, e nessa posição o erro é menor. No entanto ter usado um algoritmo para determinar qual o sensor a usar também seria uma boa escolha.

A calha oscila, tenta posicionar a bola na posição pretendida, apesar de não conseguir parar nesta. Conseguiu-se perceber que face a análise teórica, em que seria plausível funcionar sem a parte integrativa, acabou-se por verificar que esta é importante, tendo sido acrescentada na parte final. No entanto a bola não consegue parar na posição destino. Isto deve-se aos valores da *PID* usados não estarem corretamente calibrados. A possibilidade de variar estes valores em tempo real iria facilitar a calibração final, porque estes valores são sensíveis, e alterar em código prolongou o tempo de calibração.

Para melhorar o interface e percepção do sistema de medida de posição da bola, poderia-se visualizar no *LabView* o gráfico da diferença de tensão dos sensores em função da posição.

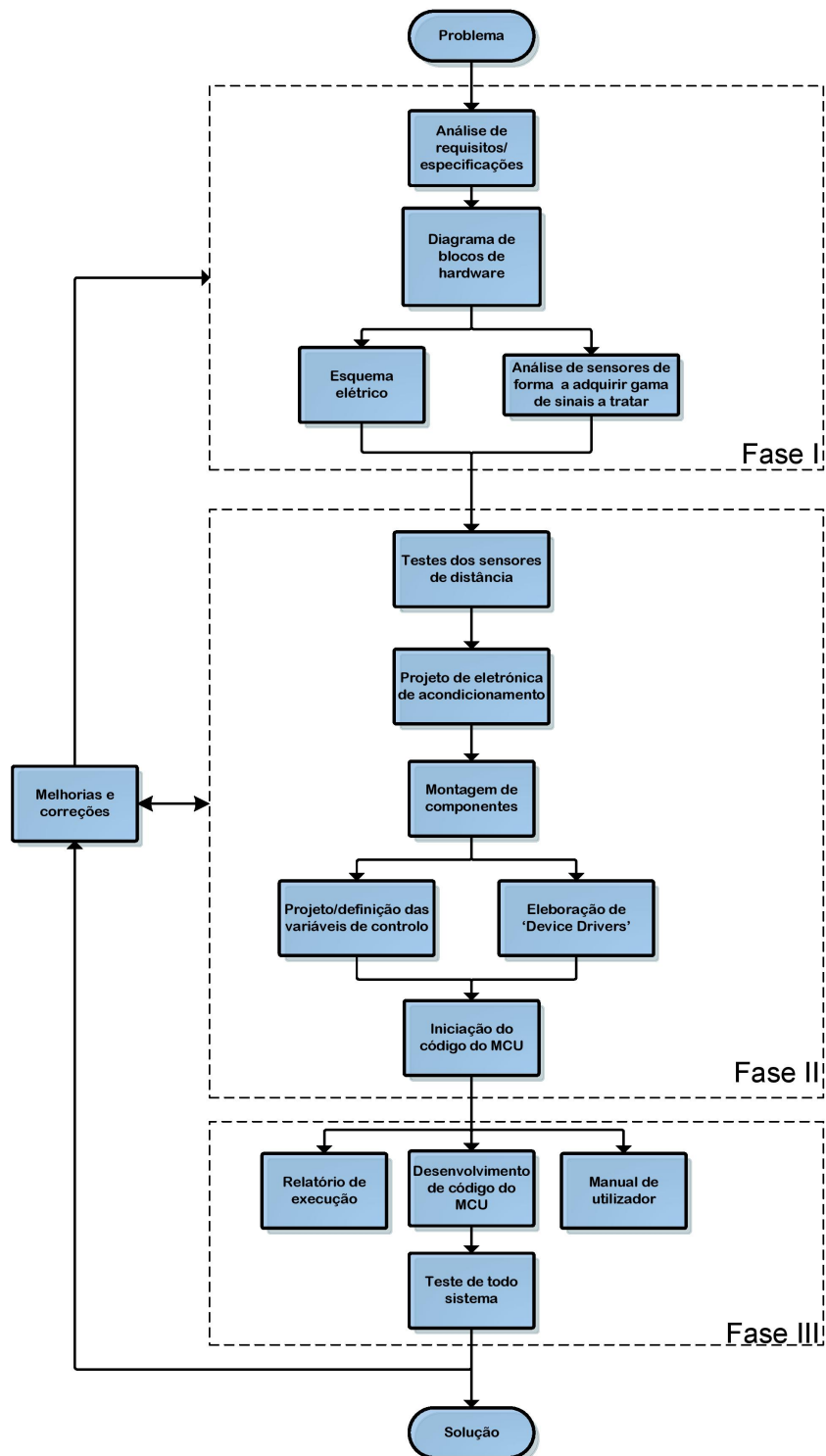
Por fim o trabalho foi benéfico para ganhar conhecimentos de programação para *PIC32*, e para além de perceber a importância do acondicionamento de sinal em um equipamento para, deste modo, diminuir o erro.

# Bibliografia

- [1] Texas Instruments Incorporated. Tl431 precision programmable reference, 2014.
- [2] Microchip. Mcp6021/1r/2/3/4, 2009.
- [3] Dr. Stienecker. 7: Pid control.

## **Anexos**

## A. Diagrama de blocos do projeto





## B. Circuito Eléctrico

