

Sistema de Rega

Grupo 1, Turno 5ªfeira 12h00 – 14h00

Tomás Nave, a22208623

André Jesus, a22207061

Arquitetura de Computadores | LEI | 23/03/2023

Índice

[Introdução 3](#_Toc97131069)

[Descrição do problema 4](#_Toc97131070)

[Fluxograma 5](#_Toc97131071)

[Tabela de Transições 7](#_Toc97131072)

[Implementação 9](#_Toc97131073)

[Conclusões 13](#_Toc97131074)

# **Introdução**

**Parte 1** – Microprocessador rudimentar

Com este trabalho pretendiamos desenvolver um automatismo de um sistema de rega tendo em atenção a fatores de avaria de sensores, avaria de mecanismos como a válvula do sistema e a outros fatores terçeiros ao sistema.

Para realizarmos este projeto de automação tivemos que construir um fluxograma, que tivesse em conta todos os imprevistos, uma tabela de estados com base no fluxograma que tivesse todos os testes que foram realizados e por fim a programação das ROMS do circuito em logisim de acordo com os resultados obtidos.

**Parte 2** – Microprocessador básico

Na segunda parte deste trabalho melhoramos o nosso automatismo passando de um microprocessador rudimentar para um microprocessador básico o que nos premitiu que retirássemos as limitações humanas que tinhamos ao utilizar um microprocessador rudimentar.

**Parte 3** – Microcontrolador Comercial

Na terceira parte deste trabalho tivemos como objetivo codificar o nosso automatismo, ou seja codificar o nosso fluxograma que já tinhamos da parte um em PIC16F628A. Tivemos que aprender a trabalhar no MPLAB e a passar o nosso automatismo para assembly

**Parte 1** – Microprocessador rudimentar

**Descrição do problema**

O problema que pretendemos resolver com este trabalho é um sistema de rega automático. Neste sistema temos um deposito de água que possui dois sensores S1 e S2 estando o sensor S1 um metro abaixo de S2. Este tanque possui uma válvula que é aberta quando o sinal está a 1, e essa válvula possui logo abaixo um sensor SH2 que tem como objetivo detetar se está a sair água ou não.

O incio de toda esta automação é o sensor SH1 que tem como objetivo detetar se a horta está regada ou não, se o sensor estiver com o sinal a 1 quer dizer que a horta está regada, se estiver a 0 quer dizer que não está regada e que precisa de ser regada, ou seja vai dar inicio á automação.

Mas antes do inicio da automação existe um sensor SC que é o sensor que deteta se está a chover, ou seja se estiver a chover estando o SC a 1 o automatismo nao liga pois se está a chover não é preciso gastar água do tanque para regar as plantas.

Agora se a horta precisa de ser regada ou seja SH1 a 0, e se não está a chover ai sim começa o automatismo.

Para a horta ser regada o deposito precisa estar cheio pelo menos até á marca do s2 ou seja a marca do s2 é a quantidade de água que é preciso para uma rega. O deposito é abastecido atravez de um furo que possui um motor para puxar a água,

Este tem também um sensor MSA que indica se o furo tem água ou não. Ou seja no automatismo primeiro é verificado se o tanque tem água para uma rega ou seja se está na marca de s2, se assim estiver o automatismo continua sem o motor funcionar pois não é preciso abastecer o tanque para fazer uma rega.

Se o deposito estiver abaixo da marca de S2 ou seja se S2 estiver a 0 primeiro vai ser verificado se o furo tem água ou não através do sensor MSA:

- se o furo não tiver água o sistema alerta que não existe água no furo ou seja não vai ser possivel fazer a rega.

-se o furo tiver água o automatismo continua com o motor a trabalhar ou seja vai entrando água no deposito do furo e vai saindo ao mesmo tempo pela válvula pois vai estar a regar ao mesmo tempo.

Por fim estando o motor a trabalhar ou não é aberta a válvula essa válvula tem um sensor SH2 e um timer. Este timer tem programado já de origem o tempo que a água demora a chegar da válvula ao sensor pois esse tempo não é imediato.

Este timer T2 serve para distinguir se a válvula está estragada ou se é o sensor que está estragado, por outras palavras, se o SH2 estiver a 0 ou seja se o sensor da valvula nao tiver detetado água e o SH1 estiver a zero significa que o terreno não está regado e que a válvula está avariada(ST2-1|ST1-0|ST0-1), mas se SH1 estiver a 1 significa que a horta está regada e que o SH2 está avariado(ST2-1|ST1-1|ST0-0).

(Usámos ST2-1|ST1-1|ST0-0 para indicar que o sensor de humidade, SH2, está avariado)

OUTPUTS:

ST – Start timer

NC - Não usado

T2 – Timer2

M – Motor de tirar a água para o depósito

V – Válvula

ST2 ST1 ST0 - Status

INPUTS:

S1, S2 – Sensores de água no depósito

SH1, SH2 – Sensores de Humidade

MSA – Motor sem água no Furo

SC – Sensor de Chuva

TOUT, TOUT2 - Timerout

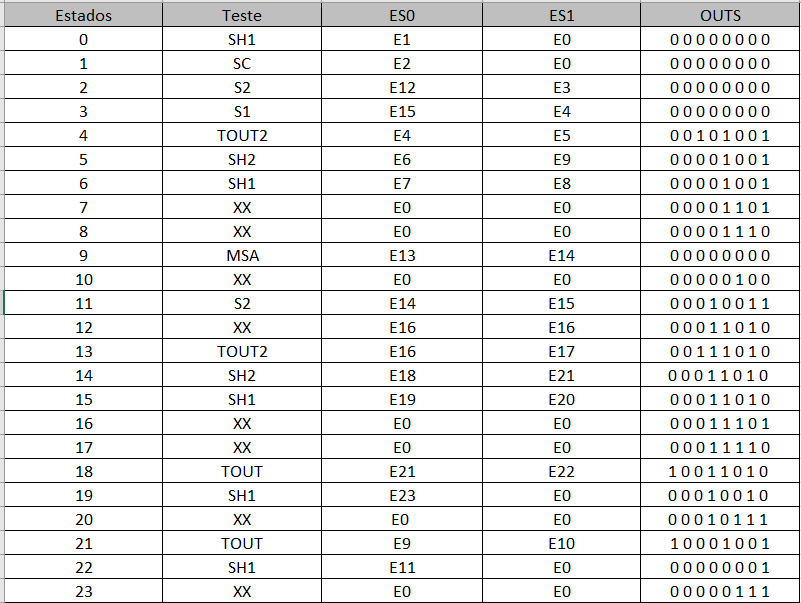
**Fluxograma**

Diagrama, Esquemático

Descrição gerada automaticamente

# Começamos o fluxograma com a verificação se a horta está ou não regada, se estiver SH1(1) não é necessário regar a horta, se não estiver regada, SH1(0), temos de verificar se está ou não a chover, se estiver SC(1), não é necessário regar a horta, se não estiver a chover, SC(0), é preciso. Se S2(1) então significa que temos água necessária para fazer uma rega. Se S1(1) significa que o sistema de rega pode estar a trabalhar sem a ajuda do motor, pois não precisa de mais água no depósito, então ativamos um TOUT2 para que a água tenha tempo de chegar ao SH2, se SH2 se mantiver a 0, significa que algo correu mal, ou seja, ou a válvula está a avariada e não deixa passar a água, ou o SH2 está avariado, para isso temos de verificar se o terreno foi ou não regado, se tiver sido regado então o SH2 está avariado, se o terreno estiver seco então a válvula está a avariada.   Se SH2(1) então está a passar água por isso ativamos um TOUT para que dê tempo para a horta ficar regada, se ao fim do timer SH1(1) então a horta está regada, se SH(0) então houver uma falha grave e é necessário intervenção. Voltando ao teste S2, se S2 estiver a 0 então temos que verificar se existe água no furo, através do sensor MSA, se não houver MSA(0) então não é possivel regar a horta, se MSA(1) então ligamos o motor e verificamos quando S2 fica ativo, quando estiver podemos começar a rega com o motor a trabalhar para que continue a puxar água para o depósito, de seguida efetuamos o mesmo processo já referido, ou seja ativamos um TOUT2 e quando este tiver terminado TOUT2(1) significa que a água já teve tempo suficiente para chegar ao SH2, se SH2 se mantiver a 0, significa que algo correu mal, ou seja, ou a válvula está a avariada e não deixa passar a água, ou o SH2 está avariado, para isso temos de verificar se o terreno foi ou não regado, se tiver sido regado então o SH2 está avariado, se o terreno estiver seco então a válvula está a avariada. Se SH2(1) então está a passar água por isso ativamos um TOUT para que dê tempo para a horta ficar regada, se ao fim do timer SH1(1) então a horta está regada, se SH(0) então houver uma falha grave e é necessário intervenção.

# Tabela de Transições



Por exemplo vamos ver o estado 5, realizando o teste SH2:

-se SH2 (estiver a 0) - Passa para o estado seguinte E6

-se SH2 (estiver a 1)- Passa para o estado seguinte E9

E o OUTS é o estado atual (0 0 0 0 1 0 0 1) , o 4 bite a 1 indicanos que a valvula está aberta

e os ultimos 3 bites (0 0 1) indicam nos que está a regar com o motor desligado

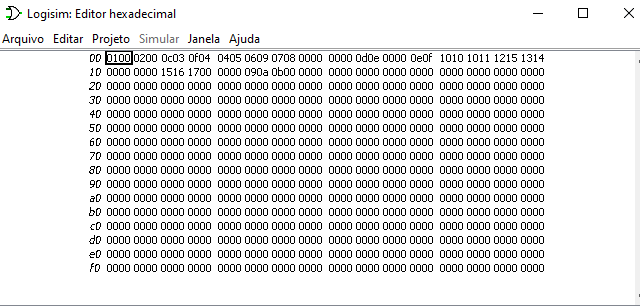
-

# Implementação

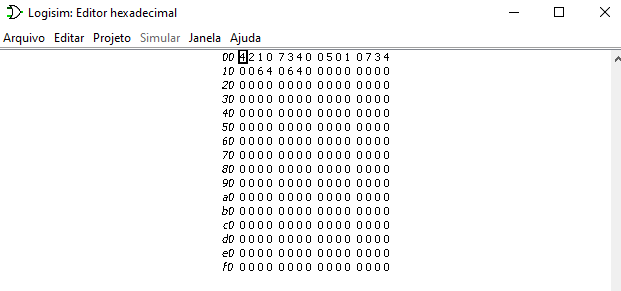
**Parte 1**

O conteúdo que colocamos na ROM-ES foram todos os ES0 e ES1 por esta mesma ordem , na ROM-TST colocamos todos os testes efetuados, e por fim na ROM-OUT colocamos os respetivos OUTS.

Na nossa máquina implementamos dois timers,um relativo ao tempo a que a água demora a sair da Válvula até chegar ao Sensor de Humidade 2, e outro relativo ao tempo que demora o Sensor de Humidade 1 a detetar que a horta está regada.

**ROM-ES:**

**ROM-TST:**

****

**ROM-OUT:**

Tabela

Descrição gerada automaticamente

**Parte 2** – Microprocessador básico

Utilizando o fluxograma do Trabalho 1 e a sua respetiva tabela de estados , fizemos o preenchimento da ROM da Memoria do programa.

Como podemos ver nas tabelas abaixo a tabela da direira é a nossa tabela de estados do fluxograma.   
A tabela da esquerda é a tabela que utiliza os valores da tabela do fluxograma e os coloca da estrutura necessaria para implementar na ROM do microprocessador básico.

Estrutura:

-Estado 0:

* Out - hexadecimal do out
* Test – hexadecimal do Test
* GOTOZ – Estado seguinte a 0 (Hex)
* GOTO – Estado seguinte a 1 (Hex)  
    
    
  A ROM memória vai ser preenchida de acordo com esta estrutura para cada estado, ou seja para cada estado vamos preencher 4 espaços da ROM.

Tabela

Descrição gerada automaticamente

**MEM Programa:**

Texto

Descrição gerada automaticamente

**Parte 3** – Microprocessador Comercial em Assembly

Tela de celular com texto preto sobre fundo branco

Descrição gerada automaticamenteCodificamos o nosso fluxograma da parte 1 do trabalho em PIC16F628A, onde tivemos que fazer a implementação do nosso fluxograma em Assembly.

Para fazer a implementacao do fluxograma utilzamos a estrutura que está descrita na imagem apresentada á direita, de maneira a implementar-mos as varia mudanca de estados do nosso fluxograma.  
  
Utilizamos:

PORTB: Para os INPUTS

S1 (BIT 0) S2 (BIT 1)

SC (BIT 2) SH1 (BIT 3)

SH2 (BIT 4) MSA (BIT 5)

TOUT2 (BIT 7) TOUT (BIT 6)

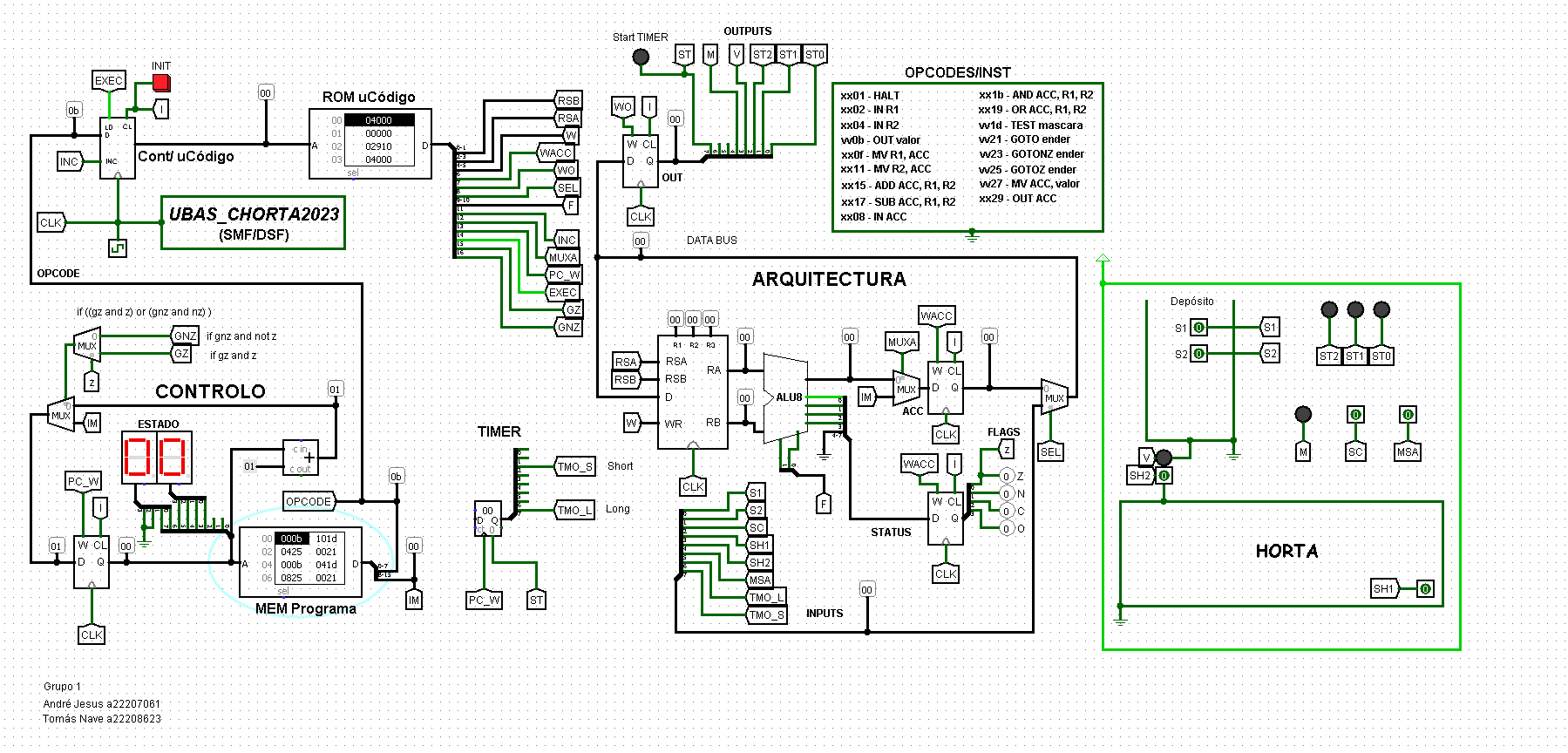
PORTA: Para os OUTPUTS

Parte 1 – Microprocessador Rudimentar

Diagrama, Esquemático

Descrição gerada automaticamente

Parte 2 – Microprocessador Básico



**Parte 3** – Microcontrolador Comercial

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Link para descarregar a implementação no MPLAB:

<https://drive.google.com/file/d/11SmLcgDPbqMibJALQR74JNQmJM6S-c0p/view?usp=drive_link>

**Desvantagens e Vantagens dos difrentes metodos de codificacao**

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

**Conclusões**

A conclusão alcançada com este trabalho é que até um simples automatismo como um sistema de rega possui inumeras condicionantes como cadeias enormes de funções que precisam de outras funções para só assim poderem ser realizadas, como enumeros fatores de avaria onde temos que prevelos e preparar o sistema para qualquer se seja o problema ou acontecimento, ou seja chegamos á conclusão que quando falamos de automações mesmo sendo para fazer uma função tao simples como regar um campo agricola a automação que está por trás é muito mais complexo do que aparenta.

Parte 1 – Parte 2

O maior problema que sentimos durante a realização deste trabalho foi a implementação da automação no circuito no logisim devido a não estarmos totalmente confortáveis a trabalhar com ROMS.

Parte 3  
O maior problema que sentimos durante a realização deste trabalho foi a implementação do timer em assembly e a dificuldade em compreender a linguagem assembly visto que ainda não temos muita prática na mesma.