

Disciplina: ELE1717 - Sistemas Digitais
Aluno:

Período: 2020.2
Data: 19/10/2020

1- Desenvolva um circuito baseado em um uC AVR (ATMega328P) para implementar um programador horário. O programador horário irá controlar o funcionamento da saída **Out** de acordo com um agendamento definido pelo usuário. Dispositivos como o programador horário são bastante utilizados em aplicações industriais e até em aplicações residenciais. O código fonte que será carregado no uC AVR deverá estar em *Assembly*, o sistema digital deverá possuir aparência conforme a Figura 1 e a descrição de seus elementos é apresentada na Tabela 1.

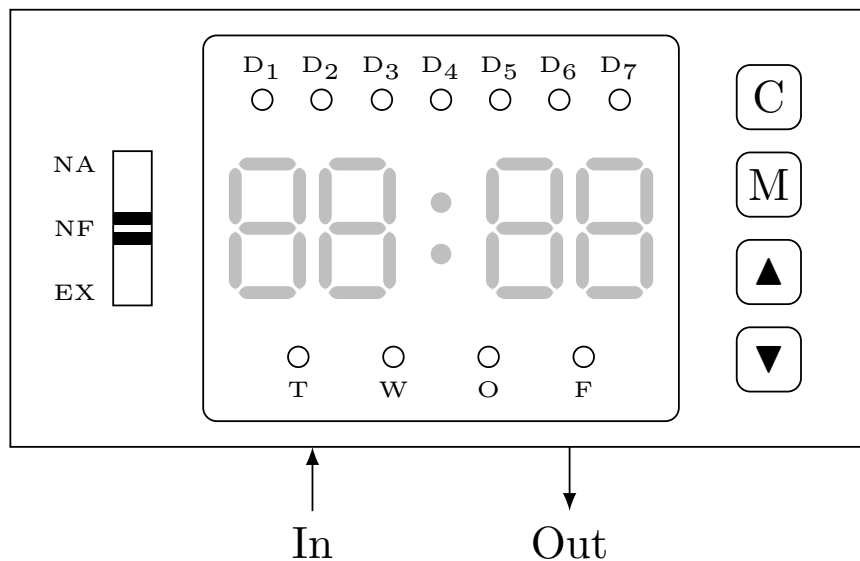


Figure 1: Aparência da interface homem-máquina do programador horário

| Elemento | Descrição |
|----------|--|
| | Display para exibição da hora |
| | Chave seletora de três estados |
| | Painel de LEDs da sinalização do dia da semana |
| | Painel de LEDs da sinalização dos modos de operação |
| | Botão para ajuste do relógio (<i>Pushbutton</i>) |
| | Botão para ajuste da programação (<i>Pushbutton</i>) |
| | Botão de incremento (<i>Pushbutton</i>) |
| | Botão de decremento (<i>Pushbutton</i>) |

Table 1: Elementos da interface homem-máquina do programador horário

Funcionamento do sistema:

O sistema digital do programador horário funciona como um relógio com calendário semanal, adicionado de um sistema de agendamento para permitir um usuário qualquer programar o funcionamento da saída **Out**, de acordo com condições pré-definidas. A saída **Out** funciona em três condições distintas determinadas pela posição da chave seletora. Se a chave seletora se encontra na posição NA, o **Out** é 1 quando a hora atual estiver no interior do intervalo agendado pelo usuário e 0 para as demais situações. Se a chave seletora se encontra na posição NF, a saída **Out** se comporta de forma oposta ao caso da NA. Se a chave seletora se encontra na posição EX, a saída **Out** apresentará o mesmo valor da entrada **In** quando a hora atual estiver no interior do intervalo agendado pelo usuário e 0 para as demais situações. O programador horário utiliza um circuito integrado RTC (*Real Time Clock*) que, através de comunicação SPI, fornece data e hora. Por fim, a configuração do relógio será realizada ao se pressionar o botão **R** e o ajuste do agendamento será realizado ao se pressionar o botão **A**.

Modos de operação:

O programador possui 10 modos de operação. O primeiro é o modo *Run* no qual o display principal apresenta a hora atual, o conjunto de leds acima do display principal indica o dia da semana (Se domingo o $D_1=1$, se segunda o $D_2=1$ e assim por diante) e o conjunto de leds abaixo do display principal indica o modo em que o programador se encontra (no caso do modo *Run* nenhum destes leds devem ficar em nível lógico alto). O segundo modo é o *Timer_H* sinalizado por um nível lógico alto no led **T**. O modo *Timer_H* é utilizado para ajustar a hora atual do programador. O terceiro modo é o *Timer_M* sinalizado pelo led **T** piscando^a. O modo *Timer_M* é utilizado para ajustar o minuto atual do programador. O quarto modo é o *Week* sinalizado por um nível lógico alto nos leds **T**, D_1 e pelo led **W** piscando^a. O modo *Week* é utilizado para ajustar o dia atual do programador. O quinto modo é o *On_H* sinalizado por um nível lógico alto no led **O**. O modo *On_H* é utilizado para ajustar a hora inicial do intervalo no qual a saída do programador será ativada. O sexto modo é o *On_M* sinalizado pelo led **O** piscando^a. O modo *On_M* é utilizado para ajustar o minuto inicial do intervalo no qual a saída do programador será ativada. O sétimo é o modo *Week_On* sinalizado por um nível lógico alto nos leds **O**, D_1 e pelo led **W** piscando^a. O modo *Week_On* é utilizado para ajustar o dia da semana da hora inicial do intervalo. O oitavo modo é o *Off_H* sinalizado por um nível lógico alto no led **F**. O modo *Off_H* é utilizado para ajustar a hora final do intervalo no qual a saída do programador será ativada. O nono modo é o *Off_M* sinalizado pelo led **F** piscando^a. O modo *Off_M* é utilizado para ajustar o minuto final do intervalo no qual a saída do programador será ativada. O décimo e último modo é o *Week_Off* sinalizado por um nível lógico alto nos leds **F**, D_1 e pelo led **W** piscando^a. O modo *Week_Off* é utilizado para ajustar o dia da semana da hora final do intervalo.

^aPiscando com uma frequência de 2 Hz.

Funcionamento do sistema (Ajuste do relógio):

O infográfico da Figura 2 apresenta o detalhamento do procedimento para ajuste do relógio do programador horário. Estando no modo *Run* (1), o usuário deverá pressionar o botão **R** para inicializar o processo de ajuste do relógio, o que resultará na mudança para o modo *Timer_H* (2). Para definir a hora basta pressionar o botão de incremento ▲ ou decremento ▼ (3). Após definir a hora, o usuário deverá pressionar o botão **R** para mudar para o modo *Timer_M* (4). Neste momento o usuário poderá definir o minuto pressionando o botão de incremento ▲ ou decremento ▼ (5). Com a hora definida, o usuário deverá mais uma vez pressionar o botão **R** para mudar para o modo *Week* (6). Neste modo, basta o usuário pressionar o botão de incremento ▲ ou decremento ▼ (7) para definir o dia da semana. Por fim, deve-se pressionar o botão **R** para finalizar o processo e retornar o programador horário para o modo *Run* (8).

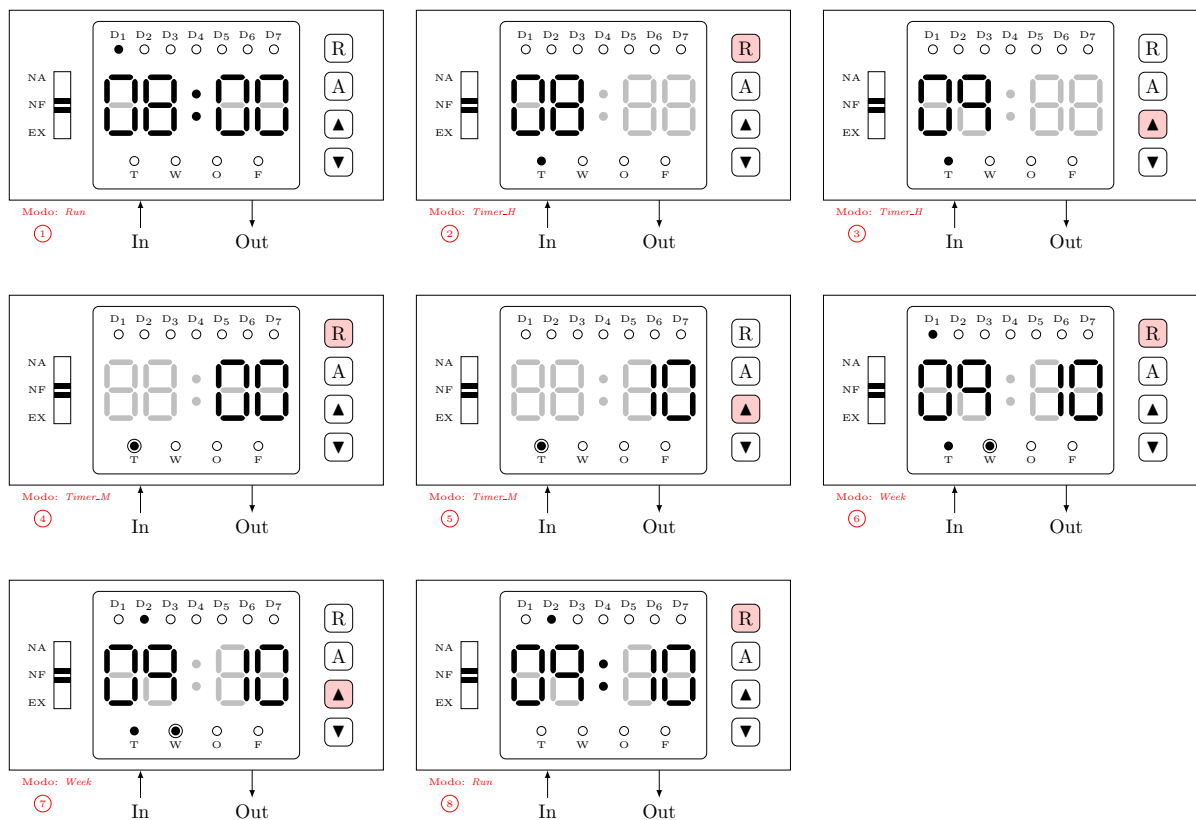


Figure 2: Infográfico do ajuste do programador horário para hora atual 09:10 de uma segunda.

Funcionamento do sistema (Ajuste do agendamento):

O infográfico da Figura 3 apresenta o detalhamento do procedimento para ajuste do agendamento do programador horário. Estando no modo *Run* (1), o usuário deverá pressionar o botão **A** para inicializar o processo de agendamento, o que resultará na mudança para o modo *On_H* (2). Para definir a hora de início do agendamento basta pressionar o botão de incremento ▲ ou decremento ▼ (3). Após definir a hora de início do agendamento o usuário deverá pressionar o botão **A** para mudar para o modo *On_M* (4). Neste momento o usuário poderá definir o minuto de início do agendamento pressionando o botão de incremento ▲ ou decremento ▼ (5). Com o horário de início do agendamento definido, o usuário deverá mais uma vez pressionar o botão **A** para mudar para o modo *Week_On* (6). Neste modo, basta o usuário pressionar o botão de incremento ▲ ou decremento ▼ (7) para definir o dia da semana. A definição do horário de fim do agendamento seguirá o mesmo procedimento realizado para a definição do horário de início do agendamento. Após definir o dia da semana do fim do agendamento, modo *Week_Off* (11), basta o usuário pressionar o botão **A** finalizando o processo e retornando o programador horário para o modo *Run* (12).

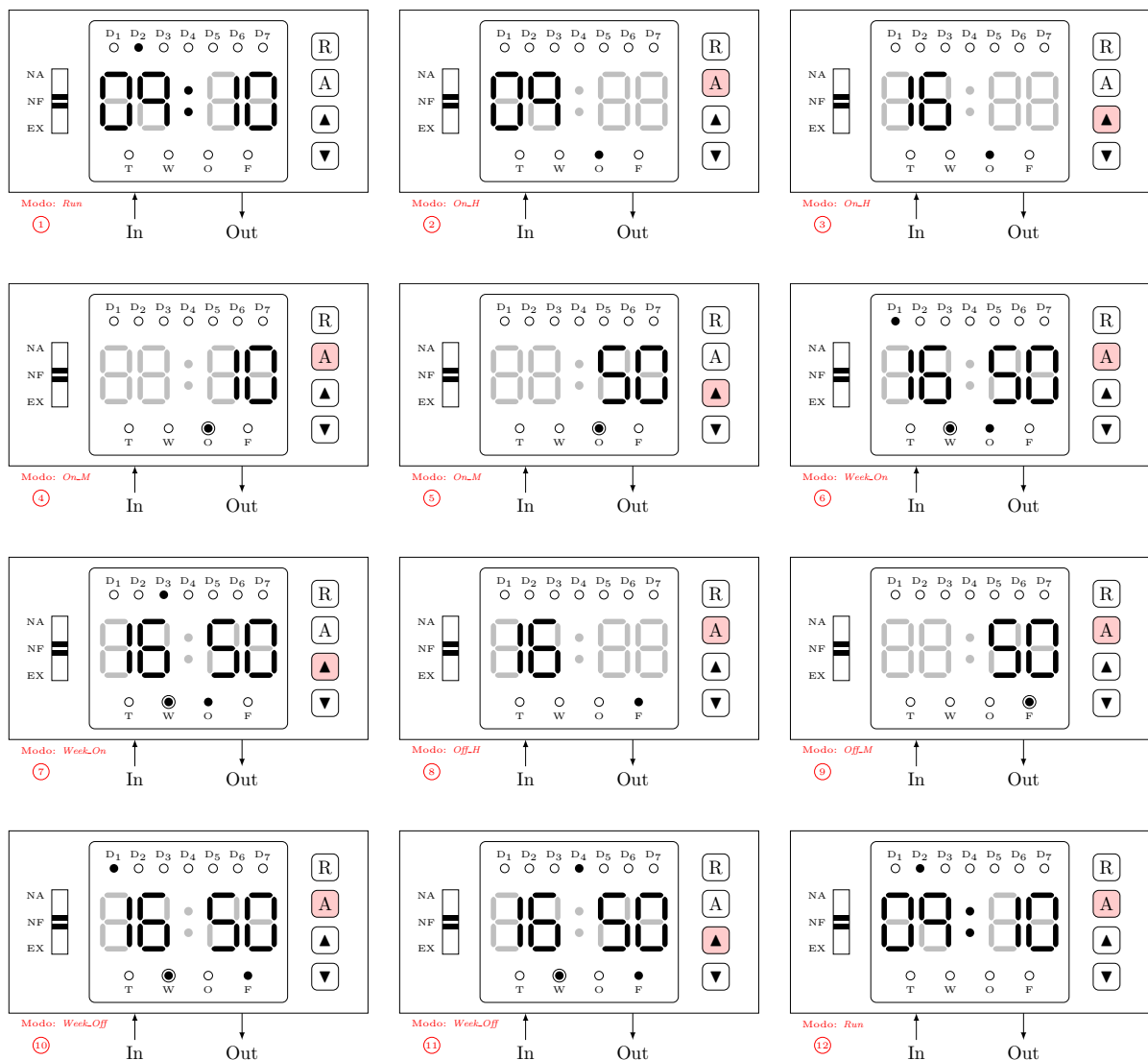


Figure 3: Infográfico do ajuste de agendamento do programador horário para início do agendamento às 16:50 de uma terça e final do agendamento às 16:50 de uma quarta.

Algumas observações sobre o funcionamento do programador horário

- O dia e a hora exibidos no modo *Run* são obtidos de um CI de RTC via SPI. O projetista deve definir como será realizada essa coleta, tratamento e exibição dos dados do RTC;
- Pode ser utilizado como CI RTC um dos seguintes modelos: MAX6902, DS3234 ou PCF2123;
- Todos os leds quando estão piscando o fazem com uma frequência de 2 Hz;
- Ao se pressionar os *pushbuttons* ▲ ou ▼ o valor será incrementado ou decrementado com uma frequência de 2 Hz;
- O usuário poderá alterar a chave seletora sem a obrigação de reajustar o programador horário pois a chave só afeta o funcionamento da saída **Out**;
- Uma vez que o usuário saia do modo *Run* a única forma de retornar para ele é passando por todos os outros modos do ajuste do relógio ou do ajuste do agendamento.

É importante no projeto:

- Na semana de projeto é importante estudar o microcontrolador e estudar os periféricos que serão necessários;
- O projeto será realizado através de MDE de alto nível, diagramas necessários e definição dos periféricos necessários;
- Na semana de projeto não é necessário desenhar o circuito e nem elaborar o código fonte;
- Todos os detalhes (definição de *clocks*, de atividades em paralelo, de uso de interrupções e etc) necessários para a implementação devem ser definidos no projeto;

É importante na implementação:

- Na semana de implementação são necessários desenvolver o código fonte e todos os diagrama esquemáticos do circuito;
- Todos os projetos devem conter os diagramas esquemáticos dos circuitos eletrônicos em .pdf em folhas A4 com legenda e seguindo as normas de desenho técnico (pode utilizar software para isso, Ex. Programas de desenho de PCB);
- Deve ser implementado o projeto recebido, são apenas permitidas alterações no projeto quando o mesmo está errado e, deverá ser apontado no relatório, o erro identificado e a solução adotada;
- Para comprovar o funcionamento podem ser elaboradas simulações, as quais devem estar detalhadas no relatório e em vídeo;

Referências:

1. Livros de arquitetura de computadores;
2. Datasheet do microcontrolador AVR ATMega328P, MAX6902, DS3234 e PCF2123;;
3. Livros de projetos com microcontroladores;