

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

SECRETÁRIA ELETRÔNICA

ELE1717 - Grupo 02 - Projeto - Problema 01

ALISSON GABRIEL LUCAS DA SILVA
LUCAS AUGUSTO MACIEL DA SILVA
LUCAS GUALBERTO SANTOS RIBEIRO
SAMARA REVOREDO DA SILVA
WILLIAN MOURA GONDIM DE FREITAS

NATAL/RN
2021

PROBLEMA 01 - SECRETÁRIA ELETRÔNICA

Relatório apresentado à componente curricular
ELE1717 - Sistemas Digitais sob a orientação do
professor Samaherni Moraes Dias.

RESUMO

Este relatório tem como objetivo encontrar uma possível solução para o problema 1 (secretária eletrônica) da disciplina de Sistemas Digitais (ELE1717). Primeiramente iremos projetar uma máquina de estados de alto nível e em seguida criar um projeto RTL estruturado, onde teremos um bloco de controle, com a MDE em baixo nível, e um bloco de dados. Ao decorrer do trabalho mostraremos cada funcionalidade da secretária eletrônica separadamente, com seus respectivos trechos da máquina de estados.

Palavras-chaves: Sistemas digitais, Circuitos digitais, MDE, RTL, Memórias.

SUMÁRIO

1 INTRODUÇÃO	5
2 ELABORAÇÃO DA SECRETÁRIA ELETRÔNICA	5
2.1 MÁQUINA DE ESTADOS DE ALTO NÍVEL	6
2.1.1 Inicialização	6
2.1.2 Repouso	8
2.1.3 Memória	9
2.1.3.1 Gravar	10
2.1.3.2 Reproduzir Tudo	10
2.1.3.3 Reproduzir única faixa	11
2.1.3.4 Pausar	12
2.1.4 Botão Next e Back	12
2.1.5 Resetar e Desligar	13
2.2 NÍVEL DE TRANSFERÊNCIA ENTRE REGISTRADORES (RTL)	14
3 CONCLUSÃO	23
REFERÊNCIAS	23
ANEXO A	24
ANEXO B	25
ANEXO C	26

1 INTRODUÇÃO

A revolução tecnológica nos circuitos elétricos, iniciada por Claude Elwood Shannon em 1936, após a publicação de “A Symbolic Analysis of Relay and Switching Circuits” iniciou uma nova categoria de circuitos denominada circuitos digitais. Desde então a sua aplicação vem sendo cada vez mais difundida até chegarmos nos dias de hoje cercados por diversos aparelhos com sistemas digitais, desempenhando funções específicas, como por exemplo, localização em tempo real, registro de velocidade e aceleração, controle de processos industriais, secretária eletrônica.







Dentro dessas funções específicas este trabalho tem como objetivo detalhar a elaboração de um sistema digital de uma secretária eletrônica. Esse sistema inclui um display; 4 botões sendo: 1 de Pause/Play, 1 de Next, 1 de Back e 1 On; além de 5 leds informativos que indicam: O (indicação de ligado), R (indicação de gravação de mensagem), P (indicação de reprodução de mensagens), F (indicação de memória cheia), M (indicação de mensagem na secretária eletrônica).

A fase de projeto, na qual esse trabalho consiste, tem como objetivos específicos de especificação de uma Máquina de Estados (MDE) de alto nível, do projeto em Nível de Transferência entre Registradores (RTL) estruturado que contém o bloco de controle com MDE de baixo nível e bloco de dados.

2 ELABORAÇÃO DA SECRETÁRIA ELETRÔNICA

Para o projeto o primeiro passo foi a elaboração da MDE de alto nível e consequentemente o projeto RTL. Os próximos subtópicos irão descrever detalhadamente cada passo da elaboração de cada elemento supracitado. Ambas estão disponíveis de forma completa nos anexos.

Figura 01 - Display completo da secretária eletrônica.

Elemento	Descrição
	Display principal (exibição de valor entre 0 e 25)
	Botão ON (inicializar/repouso/reset - tipo <i>pushbutton</i>)
	Botão PLAY (reprodução/pausa - tipo <i>pushbutton</i>)
	Botão BACK (navegação decremental - tipo <i>pushbutton</i>)
	Botão NEXT (navegação incremental - tipo <i>pushbutton</i>)
	Led (O: em funcionamento; R: gravando; P: reproduzindo; F: cheio; M: tem mensagem)

Fonte: Material do Problema 01 elaborado pelo professor do componente curricular ELE1717 (2021).

2.1 MÁQUINA DE ESTADOS DE ALTO NÍVEL

O desenvolvimento da MDE de alto nível se iniciou com a análise do problema proposto para podermos criar os estados em cima das funcionalidades especificadas.

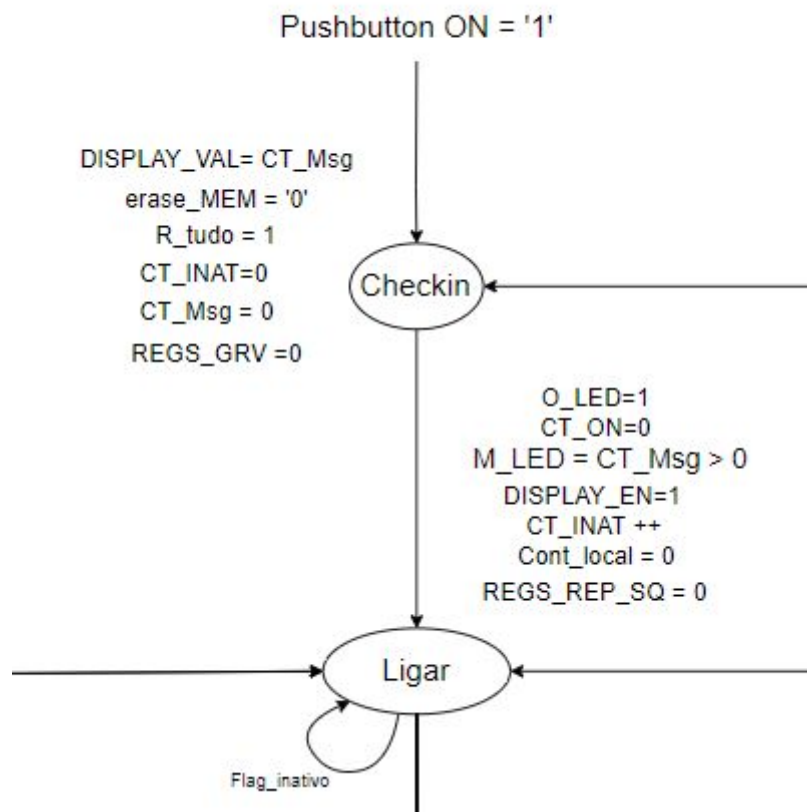
Como a máquina de estados ficou complexa, devido ao funcionamento do sistema, vamos abordar ela nos próximos subtópicos de acordo com essas funções, separadamente para um melhor entendimento e consequentemente mais facilidade para a implementação do sistema.

2.1.1 Inicialização

No primeiro trecho da máquina de estados encontramos dois estados iniciais, o Checkin, responsável pela inicialização de grande parte das variáveis e aplicação da quantidade de mensagens no registrador do valor do display; e o Ligar, responsável por ligar o LED O (indicativo de sistema em funcionamento), exibição no

display numérico do número de mensagens, inicialização de outras variáveis e flags, como poderá ser visto, e início da contagem do tempo inativo.

Figura 02 - Display completo da secretária eletrônica.



Fonte: Elaborado pelos autores(2021).

Como pode ser observado na Figura 02 existem duas formas de entrar no estado Checkin. Na primeira execução quando o botão ON é pressionado ou quando o botão ON é pressionado depois de o sistema ter sido desligado.

Na tabela 01 abaixo podemos ver as variáveis utilizadas nesse estado.

Tabela 01. Variáveis do estado Checkin

Variáveis:	Descrição:
DISPLAY_VAL	Armazena o valor a ser mostrado no display
erase_MEM	Variável responsável por apagar os dados da memória
R_tudo	Variável que dirá se a máquina vai reproduzir todas as faixas ou uma única faixa

CT_INAT	Contador de inatividade
CT_Msg	Contador da quantidade de mensagens gravadas
REGS_GRV	Ponteiro da memória de gravação

Fonte: Elaborado pelos autores (2021).

A saída do estado Checkin leva ao estado Ligar. Esse estado é composto por quatro entradas, além da que vem do Checkin mais três, uma que vem do estado Resetar, outra do estado de Repouso e a última do estado “++”. Todos os estados citados serão explicados detalhadamente nos próximos subtópicos. Além dessas quatro entradas existe um loop que permite a permanência nele enquanto todos os botões estiverem negados, representado pela variável Flag_inativo.

Na tabela 02 abaixo podemos ver as variáveis utilizadas nesse estado.

Tabela 02. Variáveis do estado Ligar

Variáveis:	Descrição:
O_LED	Variável que representa o estado do LED O
CT_ON	Contador de tempo do botão ON pressionado
DISPLAY_EN	Variável responsável pela ativação/desativação do display
CT_INAT	Contador de inatividade
Cont_Local	Contador de amostras de uma mensagem
REGS_REP_SQ	Ponteiro da memória de reprodução sequencial
Flag_inativo	Quando nenhum botão é pressionado

Fonte: Elaborado pelos autores (2021).

2.1.2 Repouso

O estado “Repouso” é iniciado quando o estado “Ligar” fica inativo por um minuto, ou quando o botão ON é pressionado por menos de 2 segundos. Esse estado é responsável por detectar mensagens a serem gravadas e por isso é necessário zerar algumas variáveis antes da gravação.

Na tabela 03 abaixo podemos ver as variáveis utilizadas nesse estado.

Tabela 03. Variáveis do estado Repouso.

Variáveis:	Descrição:
R_LED	Variável que representa o estado do LED R
P_LED	Variável que representa o estado do LED P

2.1.3.1 Gravar

Primeiro estado, quando há uma nova mensagem a ser gravada, é Gravar, onde é ligado o led R, do inglês *Record*, assim como é zerado o Cont_local para contar cada byte da mensagem a ser gravada. Logo após, há o estado addr, que aplica a opção de leitura ou não (variável r/w), dependendo do valor lógico do led R, ou seja, se estiver gravando, a escrita será feita na memória. Também no estado addr, há a escolha do endereço da memória que será aplicado, de reprodução ou gravação, através do multiplexador MUX, neste caso, a gravação é escolhida e então os dados do conversor são aplicados na memória, mediante o *enable* da memória (en), viabilizando a movimentação de dados. Quando o tamanho da mensagem é atingido, determinado pelo Cont_local atingindo o valor de 10 dados de 8 bits e é confirmado que não é um procedimento de reprodução através do sinal do led P, então o estado check é alcançado. O led M, que sinaliza se há mensagem na memória, é ativado e é realizada a checagem do número máximo de mensagens, com intuito de saber se a secretária atingiu o limite de mensagens. Caso o número de mensagens seja menor que o máximo, retorna-se ao estado de repouso; caso contrário, a máquina se direciona ao estado cheio, onde é ligado o led F, do inglês *FULL*, sinalizando que a memória não permite mais mensagens, e então, a máquina se direciona ao estado de repouso.

Tabela 04. Variáveis do estado Gravar.

Variáveis:	Descrição:
en	Viabiliza a manipulação da memória
r/w	Variável que define a leitura ou escrita na memória
addr	Variável que define o endereço da memória o qual será lido ou alterado
data	Variável responsável pelo que entra ou sai da memória

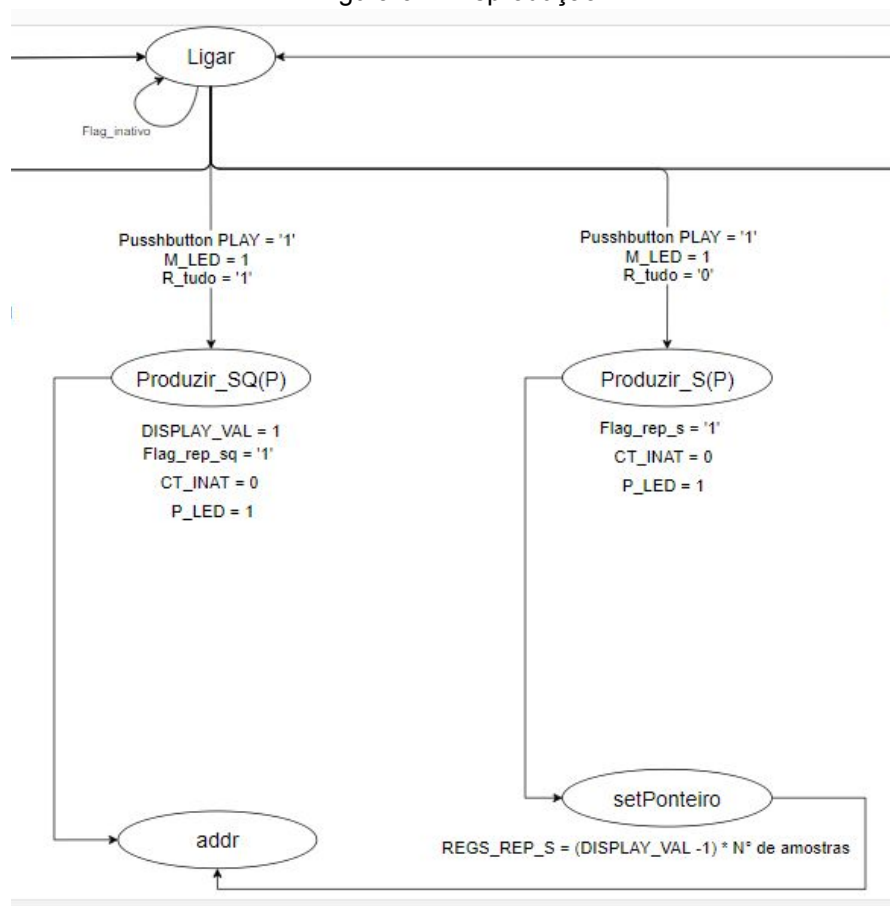
2.1.3.2 Reproduzir Tudo

Neste procedimento, o estado em que a máquina é direcionado é o Produzir_SQ, no qual a reprodução será completa, isto é, todas as mensagens na secretária eletrônica serão reproduzidas, se houver mensagem; caso contrário, a máquina não entra no estado.

No estado, o display é forçado a ir para a primeira mensagem e acompanhar qual mensagem está sendo reproduzida, devido ao estado Display_val ++ que aumenta o valor do display a cada mensagem. No entanto, primeiramente o contador de inatividade é zerado

(CT_INAT) e o led de reprodução é ligado (led P), consequentemente, o sistema passa ao estado de acesso à memória, utilizando a mesma máquina usada pela gravação, mas com seletores diferentes para o endereço da memória. Na saída do é utilizado um Conversor D/A, onde esse dispositivo receberá os dados da memória, caso o led de reprodução esteja ligado, funcionando como um *enable* para a saída. Quando uma mensagem é completada o display aumenta seu valor no estado Display_val ++ e retorna ao estado addr para um nova mensagem, no entanto, se o registrador que guarda a posição da memória estiver igual o da gravação significa que não há mais mensagens a serem lidas, retornando ao estado de Ligar, zerando o ponteiro, caso haja outra reprodução. O esquema da máquina de estados é mostrado na Figura 04.

Figura 04 - Reprodução



Fonte: Elaborado pelos autores (2021).

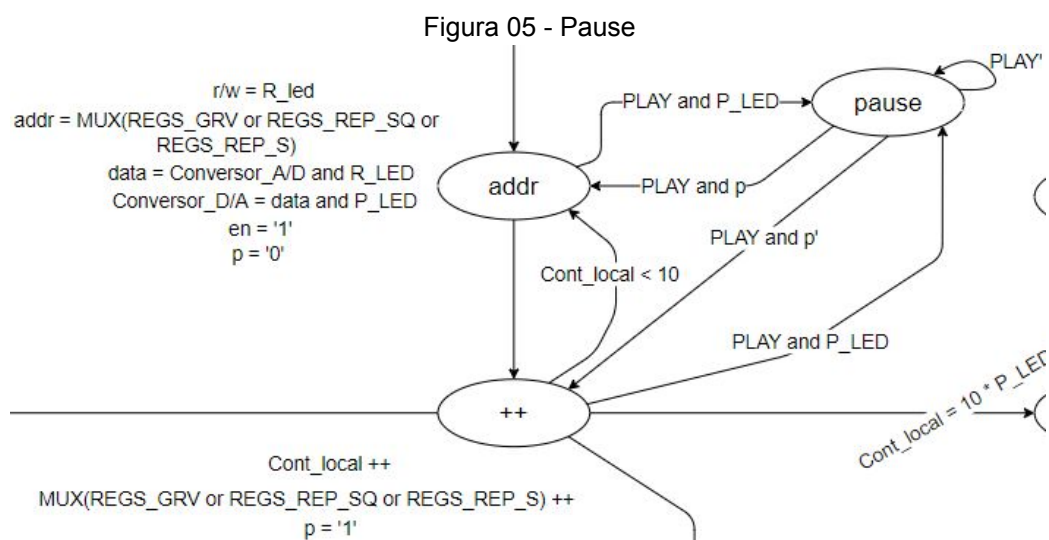
2.1.3.3 Reproduzir única faixa

É possível que o usuário queira reproduzir apenas uma única faixa da memória, para isso, é necessário apertar o botão NEXT e BACK pelo menos uma vez, quando isto ocorre, a máquina vai para o estado "Produzir _S(P)" ilustrado na

figura 4. O ponteiro REGS_REP_S, então é atualizado com o valor exato da memória onde inicia esta faixa, e o um processo parecido com a reprodução total inicia. A diferença é que quando uma faixa é atualizada o sistema já retorna para o estado ligar conforme pode-se observar na figura 3.

2.1.3.4 Pausar

A pausa só pode ser realizada durante a reprodução de uma mensagem, como mostrado na Figura 5. Podemos chegar no estado “pause” partindo de “addr” ou “++”, em ambos os casos, caso seja pressionado o botão Play durante uma reprodução (P_LED) iremos para o “pause”, enquanto o botão de Play não for acionado novamente ficaremos preso nesse estado, caso o botão seja apertado voltamos para um dos dois estágios, quem vai controlar o caminho de retorno é a variável “p”, ao observar a Figura 05 vemos que no estado “addr” a variável possui valor 0, já em “++” seu valor é 1, com isso, conseguimos controlar a volta dependendo do valor de “p”.



Fonte: Elaborado pelos autores (2021).

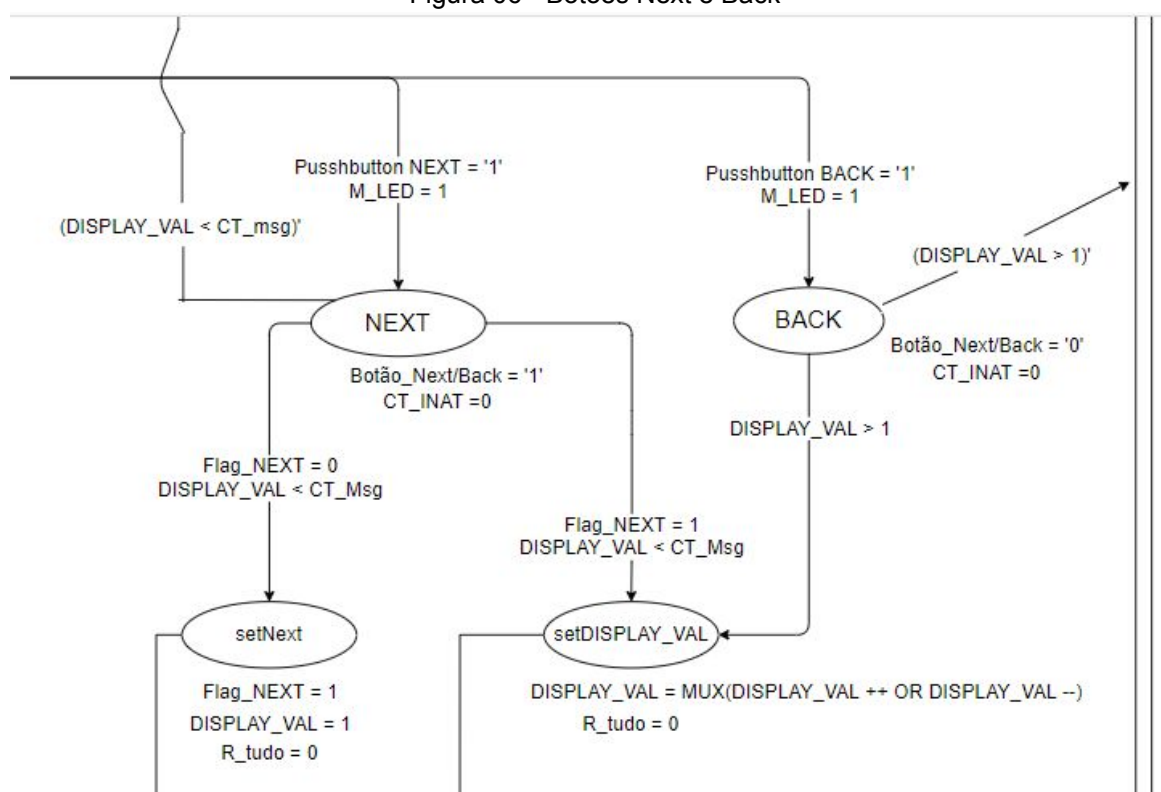
2.1.4 Botão Next e Back

Conforme ilustrado na Figura 6, caso o cliente não queira reproduzir todas as faixas de áudio gravadas na memória ele pode optar por escutar uma única faixa. Para isso, no estado ligar, basta apertar o botão Next ou Back pelo menos uma vez, e então a variável de reprodução (R_tudo) muda de valor, assim, na próxima vez

que o usuário apertar o botão Play, o sistema irá reproduzir apenas aquela mensagem especificamente.

O estado SetNext, Mostrado na Figura 6, coloca o valor do display em 1 quando é a primeira vez que o cliente aperta o botão Next, em casos onde o display está no limite máximo de mensagens e o mesmo botão é apertado, a máquina retorna para o estado inicial (Ligar). Quando o botão Back é apertado e o display encontra-se na primeira faixa, a máquina também efetua a mesma operação.

Figura 06 - Botões Next e Back

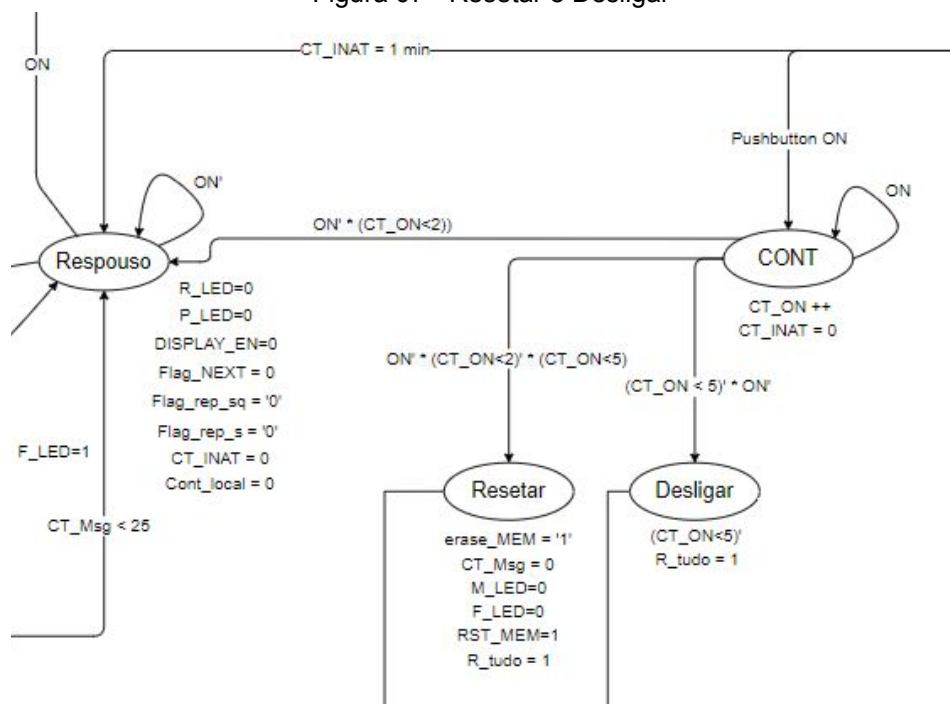


Fonte: Elaborado pelos autores (2021).

2.1.5 Resetar e Desligar

Por se tratar de uma máquina que utiliza memória Flash, é possível desligar a máquina sem resetar a memória e as mensagens gravadas, para tal, deve-se segurar o botão ON por 5 ciclos de clock ou mais. O estado “cont” faz a contagem de quanto tempo o usuário segurou o botão de ligar, no caso de menos de 2 ciclos, a máquina entra em repouso, quando está entre 2 e 4, o sistema apaga os endereços de memória onde estão gravadas as mensagens e retorna para o estado ligar.

Figura 07 - Resetar e Desligar



Fonte: Elaborado pelos autores (2021).

2.2 NÍVEL DE TRANSFERÊNCIA ENTRE REGISTRADORES (RTL)

Para o desenvolvimento do projeto RTL, primeiro, decidiu-se produzir um bloco de operação com todas as especificações realizadas na máquina de estados de alto nível. E então fazer as modificações para melhor o fluxo de dados, sem adicionar muita complexidade na estrutura do bloco de operação. A estratégia de resolução utilizada pelo grupo se usufrui muito de lógica com pouca aritmética, ou seja, são poucos os elementos de soma, subtração e multiplicação.

Na otimização do bloco operacional foram adicionados 6 multiplexadores para o controle das entradas do comparador e registradores que em momentos diferentes recebem entradas diferentes. Como pode ser visto na Figura 08.

Os multiplexadores são denominados MUX, MUX_SUM, MUX_S, MUX_Display, MUX1_comp1 e MUX2_comp1. MUX seleciona qual será a entrada para o registrador que guarda o valor do endereço da memória e sua saída é determinada pelos sinais de controle como descrito na Tabela 5.

Tabela 05. Dados MUX

Sinais de controle		Saídas
Flag_rep_s	Flag_rep_sq	
0	0	REGS_GRV
0	1	REGS_REP_SQ
1	0	REGS_REP_S

MUX_SUM seleciona qual será a entrada para o somador que incrementa uma unidade no valor do dado e sua saída é determinada pelos sinais de controle descritos na Tabela 6.

Tabela 06. Dados MUX_SUM

Sinais de controle		Saídas
Flag_rep_s	Flag_rep_sq	
0	0	REGS_GRV
0	1	REGS_REP_SQ
1	0	REGS_REP_S
1	1	DISPLAY_VAL

MUX_S seleciona qual será a entrada do registrador REGS_REP_S e sua saída é determinada pelos sinal de controle sum/display que sinaliza quando ele está incrementando e quando ele está recebendo o valor do display.

MUX_Display seleciona qual será a entrada do registrador Display_VAL e sua saída é determinada pelos sinais de controle sum_Display, sub_Display que sinalizam quando deve entrar um incrementando ou decremento, assim como setUni que sinaliza quando o valor do display deve mudado para 1 e tem a maior prioridade. Sinais de controle e saídas estão descritos na Tabela 7.

Tabela 07. Dados MUX_Display.

Sinais de controle			Saídas
setUni	sum_Display	sub_Display	
0	0	0	CT_Msg
0	1	0	DISPLAY_VAL +1
0	0	1	DISPLAY_VAL -1
1	1	0	1
1	0	1	1

.MUX1_comp1 determina qual valor vai para comparação com a saída do multiplexador MUX2_comp1 que seleciona a outra comparação. As saídas em virtude de seus seletores podem ser vistas nas Tabelas 8 e 9.

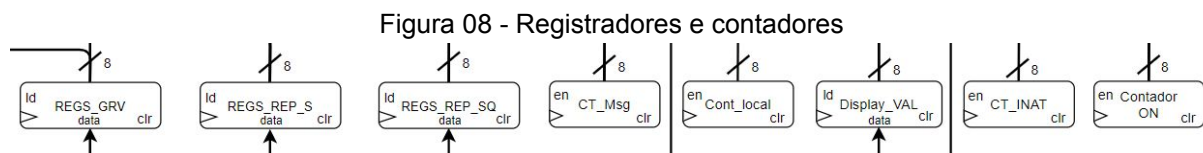
Tabela 08. Dados MUX1_comp1.

Sinais de controle - MUX1			Saídas
s_comp1	s1_comp1	s2_comp1	
0	0	0	CT_Msg
0	0	1	REGS_GRV
0	1	0	CT_ON
0	1	1	REGS_REP_SQ
1	0	0	DISPLAY_VAL
1	0	1	REGS_REP_S
1	1	0	REGS_REP_SQ
1	1	1	0

Tabela 09. Dados MUX2_comp1.

Sinais de controle - MUX2			Saídas
s_comp1	s1_comp1	s2_comp1	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	REGS_GRV
1	0	0	(Display * 10) - 1
1	0	1	25
1	1	0	60
1	1	1	CT_Msg

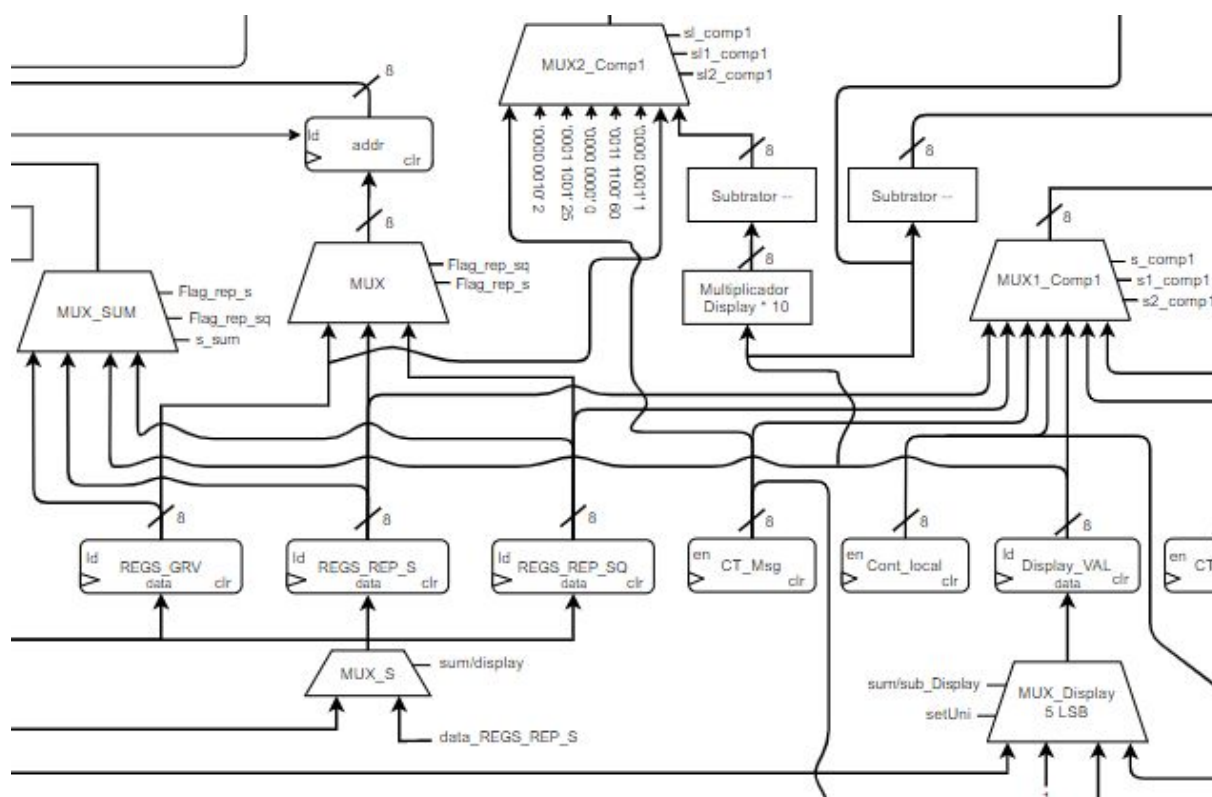
Na Figura 08, são mostrados os registradores e contadores. Os dispositivos com *ld* (*load*) são registradores; já os com *en* (*enable*) são contadores e ambos com o *clr* (*clear*). Os *lds* são utilizados para permitir a entrada, ou não, dos dados (*data*) que se encontram direcionados a cada registrador; enquanto os *ens* são utilizados para permitir que o contador possa incrementar seu valor. O *clr* funciona para apagar os dados existentes.



Fonte: Elaborado pelos autores (2021).

Na Figura 09, há apresentação da disposição parcial do bloco operacional, onde se encontram os multiplexadores descritos.

Figura 09 - Multiplexadores do Bloco operacional

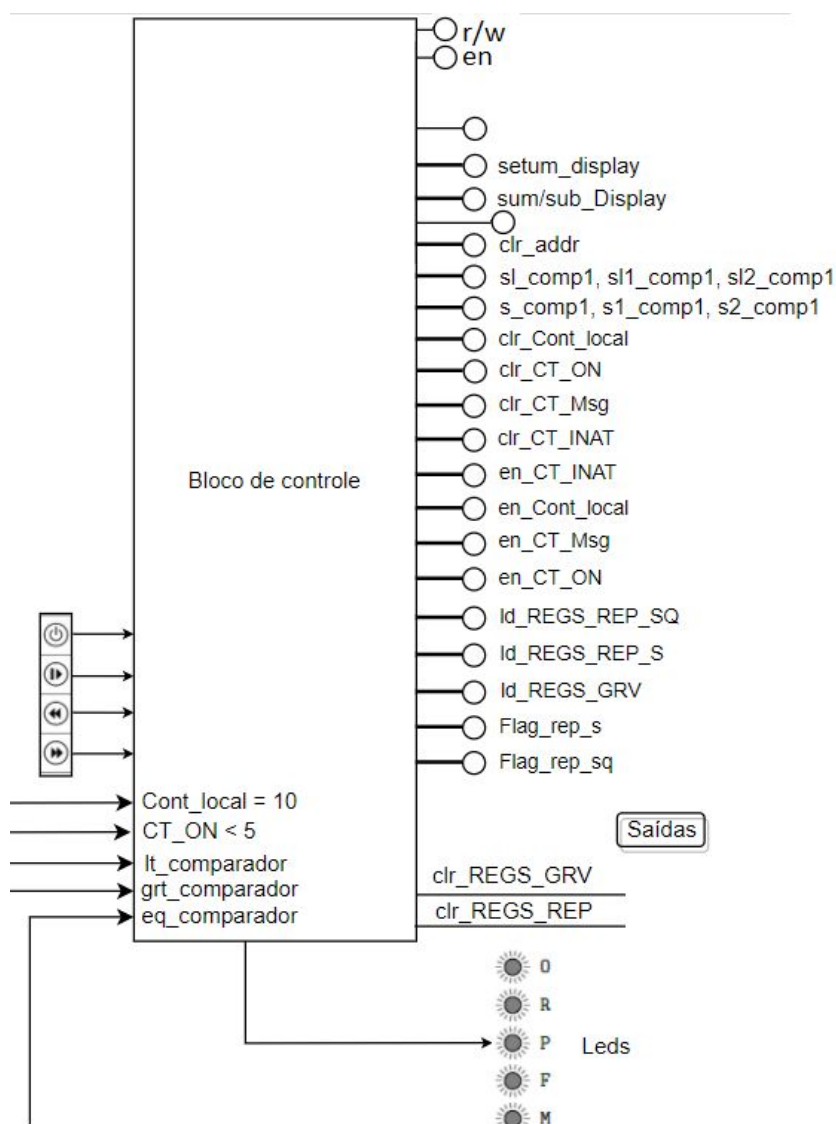


Fonte: Elaborado pelos autores (2021).

No projeto foram pensados três comparadores: Comparador, Comparador CT_ON < 5 e Comparador Cont_local = 10. São necessários os três, pois em alguns estados há duas comparações ao mesmo tempo, inviabilizando a aplicação de apenas um.

Na continuidade do projeto RTL, há a elaboração da máquina de estados de baixo nível, ou seja, a obtenção da FSM do bloco de controle. A aplicação dos sinais de controle à MDE de alto nível para que as manipulações envolvendo os registradores, contadores e outros componentes, onde residem os valores, possam efetuar a sua operação descrita em cada estado, portanto, gerando a máquina de baixo nível. O bloco de controle e seus sinais de entrada e saída estão dispostos na Figura 10.

Figura 10 - Bloco de Controle



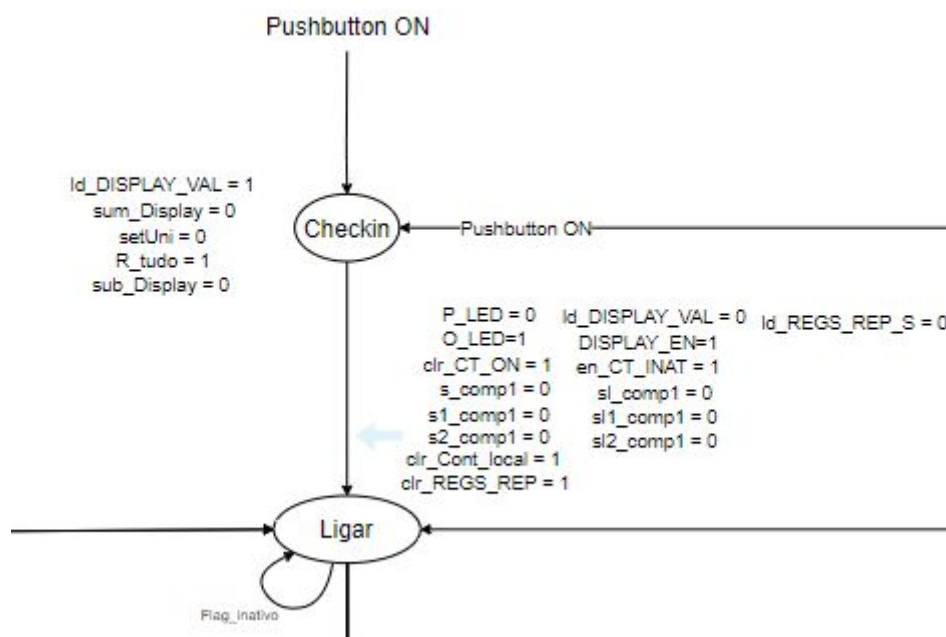
Fonte: Elaborado pelos autores (2021).

A maioria dos sinais encontrados na saída do bloco de controle são para manipulação de registradores e contadores. Os demais já foram apresentados e descritos na máquina de estado de alto nível.

O bloco operacional desenvolvido para essa aplicação está disponível no anexo C deste documento por motivos de melhor visualização das variáveis e sub-blocos (MUX, COMPARADORES, etc).

Como a máquina de estados já foi explicitada, apenas é demonstrada a aplicação dos sinais de controle do bloco de controle na MDE. Na inicialização, tem-se:

Figura 11 - Inicialização na MDE baixo nível

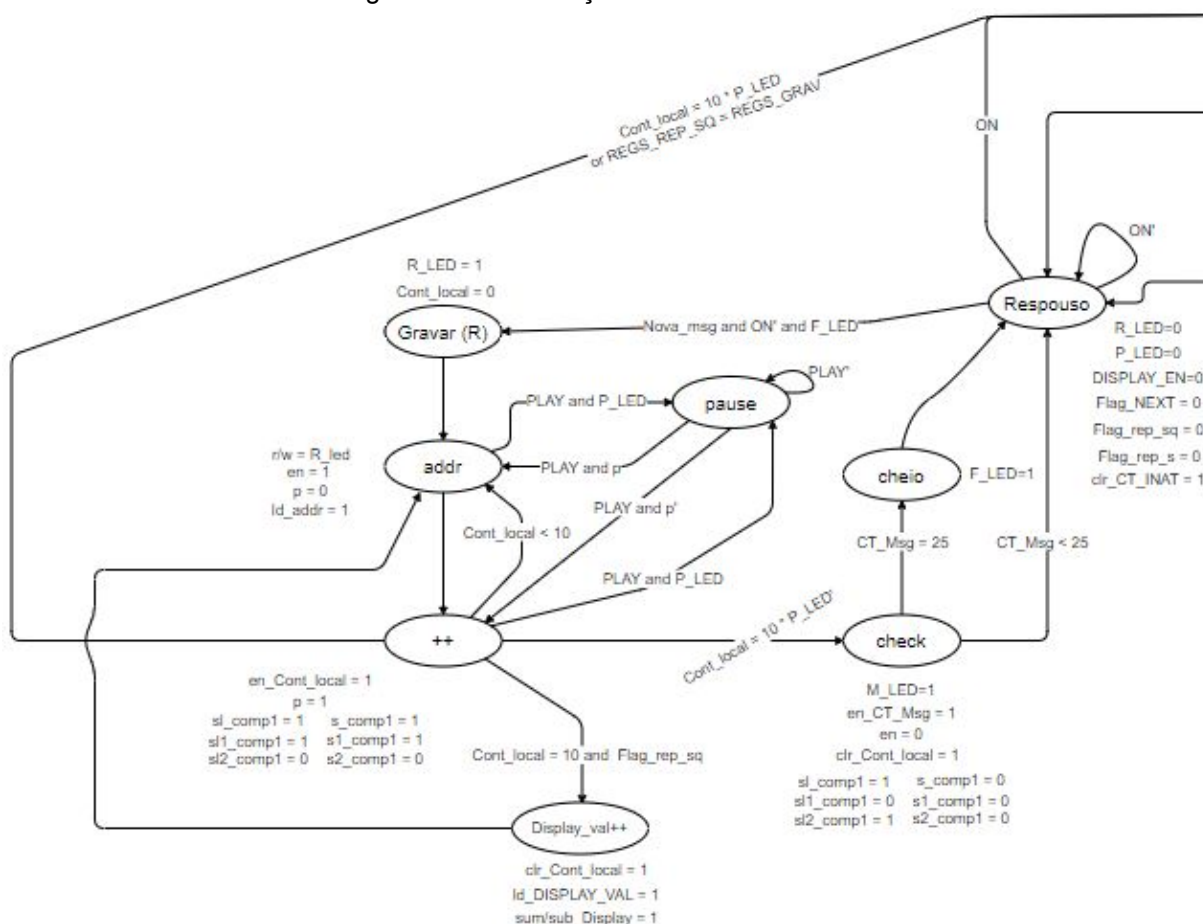


Fonte: Elaborado pelos autores (2021).

Há uma significativa mudança na quantidade de itens, visto que para uma comparação são necessários os seletores para os multiplexadores e os sinais de controle dos dados que entram e saem de cada registrador.

No Chekin, os sinais Id_DISPLAY_VAL, sum_Display, sub_Display e setUni são aplicados de maneira que o display receberá o valor da quantidade de mensagem guardado no CT_Msg. Como indicado na máquina de estados anterior. A configuração dos seletores dos MUX1_comp1 ('000') e MUX2_comp1 ('000') estão de forma que CT_Msg está sendo comparado com o número 0, definindo se há mensagem para poder ligar o led M.

Figura 12 - Inicialização na MDE baixo nível

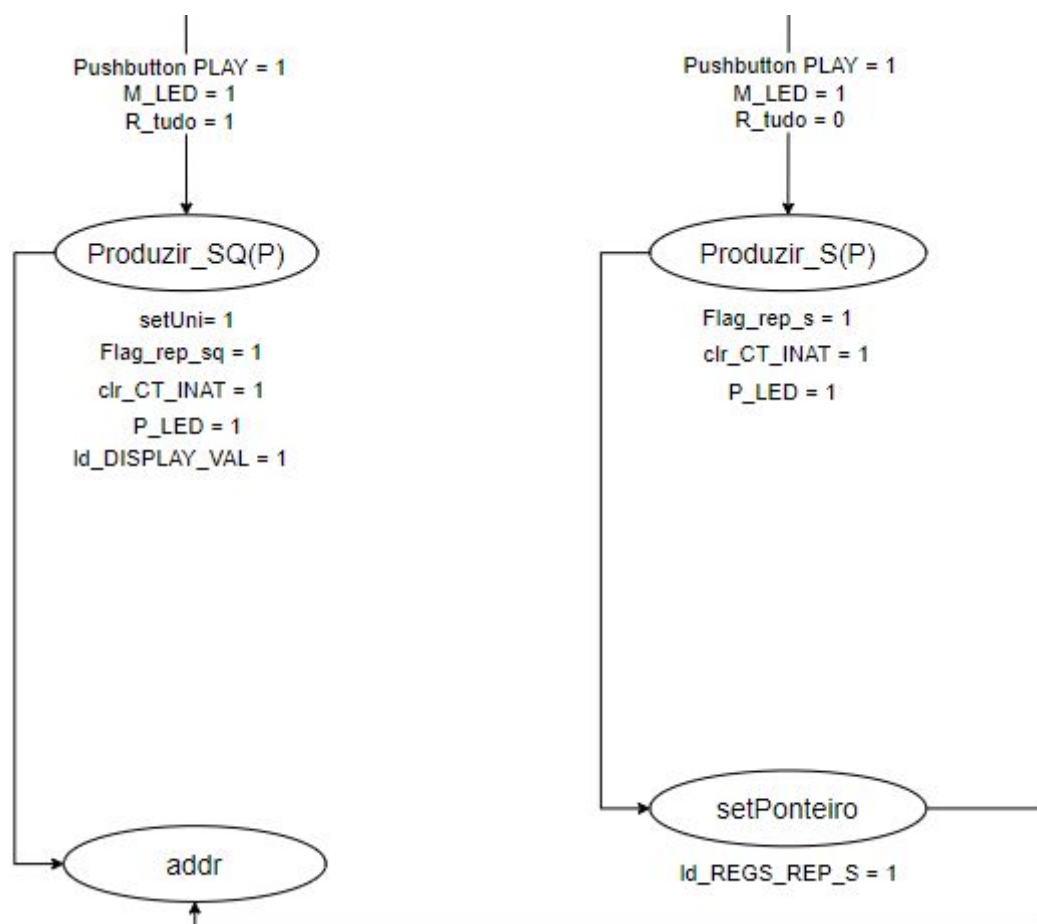


Fonte: Elaborado pelos autores (2021).

Nota-se na Figura 12, devido ao maior espaço, que o estado extra **addr** que se encontrava após **Display_val++**, na forma de retorno ao estado, foi retirado e a correta configuração foi implementada.

Vê-se que os seletores se configuram no estado **++** para comparar **REGS_REP_SQ** com **REGS_GRV** para saber quando a reprodução irá acabar, retornando ao estado **Ligar**.

Figura 13 - Inicialização na MDE baixo nível



Fonte: Elaborado pelos autores (2021).

Na Figura 13, observa-se, no estado `Produzir_SQ`, que os sinais `setUni` e `Id_DISPLAY_VAL` são ativados, significando que o display da secretária eletrônica irá para 1, pois a reprodução sequência irá começar na primeira mensagem. Também, os outros sinais como `P_LED` do led de reprodução, `clr_CT_INAT` e `Flag_rep_sq` são acionados, ligando o led, zerando o contador de inatividade e setando o seletor para os multiplexadores de soma e endereço da memória, respectivamente. Já na reprodução de mensagem única, `Produzir_S`, a outra bandeira é acionada para os multiplexadores, `Flag_rep_sq`, o `CT_INAT` é zerado e o led `P` é ligado também. E, em seguida, o `Id_REGS_REP_S` é acionado permitindo que o ponteiro da memória de reprodução de faixa única receba o valor da memória referente ao valor do display que o usuário colocou no momento. Ambos os estados, `Produzir_SQ` e `Produzir_S`, vão ao estado `addr` para o começar a reprodução.

3 CONCLUSÃO

Tendo em vista tudo o que foi abordado até aqui, é possível concluir que o objetivo principal do trabalho, que é projetar uma secretária eletrônica que grave e reproduza mensagens de voz, foi cumprido, uma vez que foram feitos a MDE e o RTL dessa máquina. Pode-se dizer que houve êxito na realização do trabalho pois, embora o projeto ainda não tenha sido implementado, foi criado um ambiente de aprendizagem que facilitou a fixação de todos os conteúdos vistos anteriormente em outras disciplinas, e, além disso, outros aprendizados também foram adquiridos, como a organização do tempo, o trabalho em grupo e o uso de ferramentas online.

REFERÊNCIAS

- BBC. **Encoding audio and video.** Disponível em: <https://www.bbc.co.uk/bitesize/guides/z7vc7ty/revision/2>. Acesso em: 22 jan. 2021.
- VAHID, Frank. **Sistemas Digitais: projeto, otimização e hdl's.** Porto Alegre: Bookman, 2008.

ANEXO A

Nome do líder: Alisson Gabriel Lucas da Silva

1. Equipe:

Redator: LUCAS GUALBERTO SANTOS RIBEIRO; Debatedor: LUCAS AUGUSTO MACIEL DA SILVA; Videomaker: WILLIAN MOURA GONDIM DE FREITAS; Auxiliar: SAMARA REVOREDO DA SILVA.

2. Problema:

Projetar uma secretária eletrônica capaz de gravar até 25 mensagens, contendo cada uma no mínimo 10 bytes, mostrar quantidade de mensagem em um display e a navegação do usuário entre elas, reproduzir todas as mensagens, reproduzir uma mensagem determinada pelo usuário, deletar todas as mensagens, reconhecer a atividade de botões periféricos de interação com o usuário e, a partir deles, viabilizar o desligamento, repouso e pausa da reprodução, além das funções já descritas.

3. Registro do Brainstorming:

Nos primeiros dias, tivemos que fazer uma pesquisa de como resolver, procurando exemplos e conteúdos sobre projeto RTL e máquina de estados de baixo e alto nível. A maioria, senão todos, havia passado por Circuitos Digitais há alguns semestres, ou seja, precisava-se relembrar conteúdos, especialmente, com relação à memória e termos técnicos utilizados nos livros base. Na terceira reunião, já tínhamos definido memória Flash por não resetar quando desligada, isto é, não volátil e então os integrantes pensaram em maneiras de produzir a secretária eletrônica, aplicando as ideias em uma máquina de estados de alto nível e nas reuniões posteriores, nós coletamos todas as informações, aplicamos em uma máquina de estados e prosseguimos ao projeto RTL com o bloco operacional e bloco de controle e então, a produção da máquina de estados de baixo nível.

4. Pontos-Chaves:

Utilização da memória, elementos síncronos e assíncronos que podem confundir ao projetar e causar erros na implementação, quantidade de sinais de controle que podem confundir, dependendo da quantidade, e neste projeto foram muitos.

5. Questões de pesquisa:

Para o problema proposto: Pesquisas sobre projeto RTL, MDE, Sistemas embarcados, sobre memórias, sobre projetos e artigos de circuitos elaborados para secretárias eletrônicas.

6. Planejamento de pesquisa: Através de artigos, pesquisas científicas, livros, professores, vídeos, palestras, amigos.

ANEXO B

MDE: <https://app.diagrams.net/#G1AhRslSmG4mQuYS8WxIOHZfsaWeRS6ufq>

RTL: <https://app.diagrams.net/#G1Y1LSDza1C9qkl1Gkj5dv5GtUwoROpw8r>

ANEXO C

