



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE CRATEÚS
CIRCUITOS DIGITAIS – CRT0384

RELATÓRIO – PRÁTICA 06
FLIP FLOPS TIPO D E JK

MARLON GONÇALVES DUARTE

PROFESSORES: MARCIEL BARROS PEREIRA
RENNAN FERREIRA DANTAS

CRATEÚS-CEARÁ
SETEMBRO DE 2020

SUMÁRIO

	Pág.
INTRODUÇÃO	3
OBJETIVOS	6
MATERIAIS E MÉTODOS	7
RESULTADOS E DISCUSSÕES	8
CONCLUSÃO	14
REFERÊNCIAS	15

INTRODUÇÃO

A capacidade de recordar momentos passados ou informações recebidas em outros instantes da vida, distinguem os humanos de outras tantas espécies de mamíferos presentes na natureza. O ser humano desenvolveu a matemática e a uniu com a memorização, permitindo o desenvolvimento da obra humana e de suas relações. Através da mnemônica, tradições das mais diversas culturas que povoam o solo terrestre, são repassadas e ecoadas ao longo de milhares de anos. Por exemplo, por volta de 2000 a.C. as civilizações egípcias já criavam gatos dentro do ambiente doméstico semelhante ao praticado atualmente. Esse costume de criar animais e os domesticar é ainda mais antigo, quando se estuda outras civilizações como as que habitavam a atual região da Ásia Meridional.

O desenvolvimento da tecnologia realizado pela humanidade, teve como um dos desafios, a criação de sistemas capazes de armazenar informações e poder utilizá-las na posterioridade como um dado necessário à realização de determinada tarefa, tal qual o homem é capaz de fazer. O conhecimento construído pelos seres humanos sempre pode ser armazenado em materiais como o papel, tecidos, cerâmicas, etc. Porém, o tempo e a interpretação, sempre foram inimigos destes métodos de registro de informação.

Ao passo que a ciência se desenvolvia, as folhas de papel não mais eram suficientes para realização de cálculos matemáticos complexos e que demandavam certa rapidez laborativa. Como o traçar de uma trajetória balística, de forma que o alvo seja atingido com precisão, levando em conta todas as ações naturais sofridas nos artefatos. É certo que essa utilidade da ciência não torna a humanidade boa, mas faz parte da história da humanidade e, até certo ponto, trouxe alguns benefícios às gerações mais recentes.

A criação de sistemas eletrônicos capazes de realizar cálculos complexos ajudou o homem a ganhar tempo na realização de atividades cognitivas. Contudo, a realização dos cálculos dependia de um enorme conjunto de válvulas e chaves que precisavam ser ligadas e desligadas várias vezes durante os cálculos, por exemplo, de equações diferenciais. Essa tarefa era realizada por humanos, na maioria das vezes mulheres, que precisavam ter um domínio muito preciso das funções de cada chave e acioná-la ou não a depender dos dados de entrada dos cálculos, dados esses que consistiam nos famosos 0 e 1. O desenvolvimento dos circuitos *Flip-Flop* possibilitou fazer essas trocas de estados de forma automática e mais rápida, além de possibilitar a memorização do estado lógico das chaves.

Não existe um único tipo de flip-flop, entretanto, será dado foco aos tipos JK e D. O último, possui esta nomenclatura, pelo fato de ter saída fiel a entrada, ou seja, o dado de saída é correspondente ao dado que está sendo inserido. Ele assumirá o valor 1 se D for igual a 1 e 0 quando o D for igual a 0, porém a mudança se dará sempre na alteração do estado do clock. O sinal

de *clock* será fornecido por um circuito oscilador que alterna sua saída entre os sinais *1* e *0*, ou *alta* e *baixa*, respectivamente.

Outro circuito *flip-flop*, é o do tipo *JK*, que possui um diferencial em relação ao tipo *D* na eliminação do estado proibido evitando saídas de mesmo valor lógico. Ele necessitará de *clock* para fazer as alterações dos estados das portas lógicas que compõe o circuito, permitindo que os valores sejam alterados de acordo com a oscilação do *clock*. As funções *preset* e *clear* atuam na saída *Q* independentemente do *clock*, ou seja, são assíncronas no circuito. Elas atuam, por exemplo, impedindo que, ao entrar dois sinais *altos* no *flip-flop*, as saídas *Q* e $\neg Q$ fiquem oscilando, bastando que se acione a porta *preset* para colocar em *alta* o *Q*, e/ou zerando se acionar a porta *clear*. As portas *J* e *K* possuem as letras que são as iniciais do nome do criador, Jack Kilby que inclusive ganhou Nobel em Física pela invenção do dispositivo. A entrada *J* coloca o valor de *Q* para alta respeitando a mudança de estado do *clock*, e *K* coloca o valor de *Q* para baixa, ou o $\neg Q$ para alta, já que este é um complemento da saída *Q*.

Mesmo sendo bastante úteis e largamente utilizados, os *flip-flops* possuem uma vulnerabilidade fatal na computação moderna com taxas de envio de dados altíssimas, que é a possibilidade de alteração de dados dentro do ciclo de mudança de estado do *clock*, o que, caso ocorra, pode produzir inconsistências nos sistemas mais complexos. Podem ser usados desde contadores, e até para armazenar um bit, ou um dígito binário. Com um conjunto considerável de *flip-flops*, é possível armazenar qualquer informação como o valor de um contador, um caractere ASCII em uma memória de um computador ou qualquer outra informação que esteja em formato binário.

Na prática 6, iniciam-se o trabalho com os circuitos sequenciais, que são diferente dos circuitos combinacionais, nos quais os dados de saída do circuito representam uma função única e exclusiva dos dados inseridos em suas entradas. Os circuitos sequenciais têm saídas e entradas que estão relacionadas a história passada do circuito, um dado inserido, porém que não se encontra mais na entrada, influencia nas saídas deste circuito. A ocorrência desses casos se dá em face da capacidade de memorização de estados de saída dos circuitos sequenciais, o que os inclui como dispositivos de armazenamento de informação.

Nesta prática, mas ainda em sua primeira parte, foram utilizados circuitos *flip-flop* do tipo *D*, que têm entradas chamadas *D* e *CLK*, que correspondem a entrada de dados e de *clock*, respectivamente. Bem como, duas saídas, que são relacionadas ao *Q* e seu complemento $\neg Q$. Por fim, uma entrada *EN*, que possui a função de *enable* (ativador). Aqui o dado que for inserido na entrada *D* será retornado na saída *Q* e seu complemento em $\neg Q$. Essa característica leva esse *flip-flop* a ser complementarmente chamado de: “retardo” ou “transparente”. Como etapa inicial desta parte da prática, deve-se implementar um contador de 0 a 7 utilizando *flip-flops* tipo *D*.

Na Segunda etapa, implementar o mesmo contador, porém com o circuito integrado correspondente a um flip-flop tipo D, ou seja, o flip-flop encapsulado, no caso o circuito 7474. A prática 6 também traz uma segunda parte constituída de mais duas etapas, sendo a primeira a exigir a representação da lógica combinacional necessária para fazer a contagem de 0 a 9 utilizando os flip-flops tipo JK.

OBJETIVOS

- Implementar um contador de 0 a 7 utilizando flip-flops tipo D;
- Implementar um contador de 0 a 7 utilizando o circuito integrado 7474 que é um duplo flip-flop tipo D;
- Encontrar a lógica necessária para implementar um contador assíncrono de 4 bits mod(x) com lógica combinacional necessária para fazer a contagem de 0 a 9;
- Implementar um contador de 0 a 9 utilizando flip-flops tipo JK cuja saída seja um display de 7 segmentos, com base no circuito TTL 7476;

MATERIAIS E MÉTODOS

Para a realização desta prática foram utilizadas algumas ferramentas bastante comuns a essas práticas, dentre elas estão: *Datasheet* (folha de informações) dos componentes; simulador de circuitos lógicos “Digital”, projetos iniciais fornecidos pelo professor. Durante a busca para as soluções dos problemas, foram priorizados alguns passos como uma forma de organizar o trabalho, bem como minimizar as falhas que viessem a ser sofridas, obtendo a seguinte sequência:

1. Fazer uma leitura reflexiva dos problemas;
2. Estudar as folhas de dados (*datasheet*) dos circuitos integrados 7474 e TTL 7476;
3. Refazer as ligações dos projetos pré-implementados pelo material de apoio no intuito de compreender as possibilidades de configurações;
4. Construir as tabelas verdade no caso do contador de 0 a 9 com lógica de reinício de contagem;
5. Implementar as expressões em forma de circuito no Digital (simulador);

RESULTADOS E DISCUSSÕES

O primeiro circuito implementado foi um contador de 0 a 7 com saída em um display de 7 segmentos que fora trabalhado algumas práticas atrás. O arquivo disponibilizado pelo professor para ser utilizado como um material de apoio na implementação deste projeto, tornou a elaboração do circuito bem mais claro. Na figura 1, é possível visualizar o arquivo enviado pelo docente, no qual se percebe as conexões necessária à implementação do novo circuito que deverá exibir não apenas dois bits de saída e sim três (1 1 1), resultando no decimal 7 (sete) como maior dígito de saída.

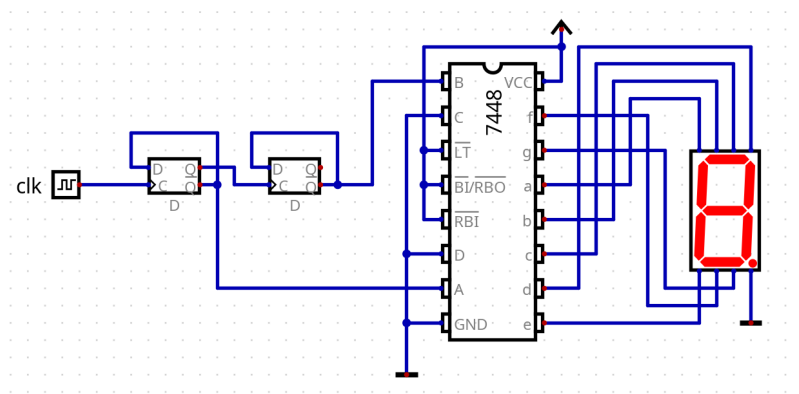


Figura 1: Circuito base enviado pelo professor.

Como resultado da implementação, foi obtido um circuito bastante parecido com o da figura 1, porém com mais um flip-flop adicionado na sequência. O projeto finalizado é visto na figura 2, na qual consta o circuito em funcionamento exibindo o decimal 6 que corresponde ao binário 110. Se observado com atenção, fica transparente a saída binária nos barramentos de entrada do CI 7448, em que se observa as entradas A, B e C recebendo 011 respectivamente. Sendo que a ordem de significância obedece ao alfabeto tendo A o menos significativo e Z o mais significativo. Portanto, colocando na ordem correta fica: C, B e A respectivamente 110.

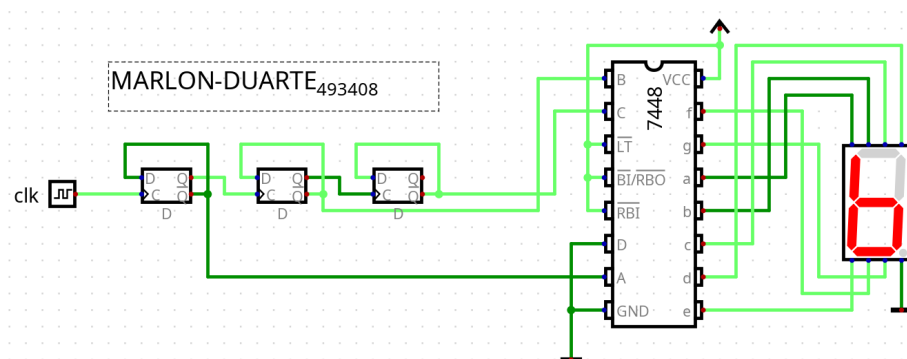


Figura 2: Circuito implementado na parte 1 prática 6 com saída de display em 6 decimal.

Ainda na primeira parte da prática 6, também usando flip-flops do tipo D, foi mudado a montagem para a utilização dos Circuitos Integrados correspondentes aos flip-flops do tipo D. O CI escolhido para esta etapa foi o 7474 que consta nas bibliotecas do Digital. Para compreender as funcionalidades do CI, o software editor de circuitos utilizado oferece a possibilidade de analisar a construção do componente, bastando para isso, clicar com o botão direito do mouse e em seguida clicando em *abrir circuito*.

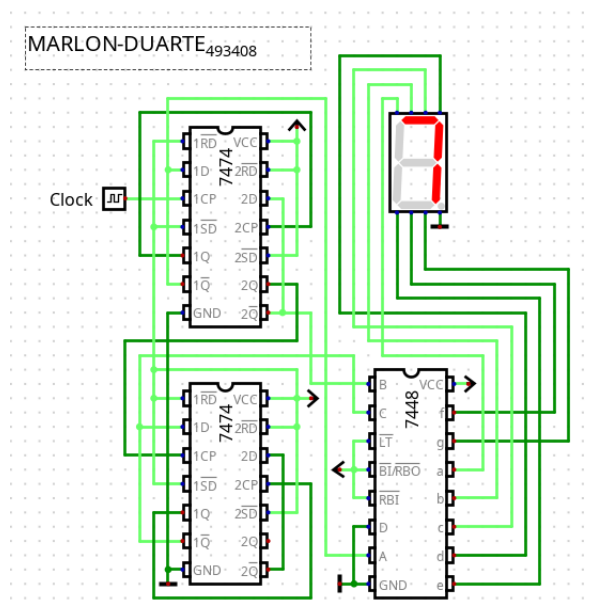


Figura 3: Circuito contador de 0 a 7 com encapsulados de flip-flop tipo D exibindo o decimal 7 como último estado de contagem.

A figura 3, apresenta o projeto em funcionamento exibindo no display o decimal 7. Como citado anteriormente, o valor binário inserido no CI gerenciador do display pode ser conferido analisando os pinos A, B, C e D do 7448. Onde se tem o D=0, C=1, B=1 E A=0. A ordem de significância dos pinos já foi comentada anteriormente. Um dado interessante de se relatar é que a implementação deste projeto ficou bastante semelhante à implementação do anterior bastando apenas atentar às entradas corretas e às ligações dos barramentos com os cuidados de sempre, de forma a evitar curto-circuitos e/ou sinais de entradas em pinos de saída (que neste caso o Digital reconhece como curto-circuito, o que na realidade nem sempre o é).

Na segunda parte da prática 6, foram utilizados circuitos flip-flop do tipo JK. Esse tipo de flip-flop possui entradas J e K que determinam a saída Q e seu complemento $\neg Q$. Sendo que, quando há um sinal alto na entrada J, o Q será alto também, logicamente seu complemento $\neg Q$ será baixo. Se alterar o J para baixa o Q continuará com sinal alto e só terá alguma mudança na inserção de sinal no K, quando então o Q passará a ter sinal baixo e $\neg Q$ sinal alto. O grande diferencial desse flip-flop é não permitir sinais iguais no Q e $\neg Q$, que poderia acontecer, por exemplo, quando as

entradas J e K estivessem setadas em alta e alta, respectivamente. Porém, neste caso, o flip-flop JK ficará alternado entre alta e baixa no Q e em seu complemento.

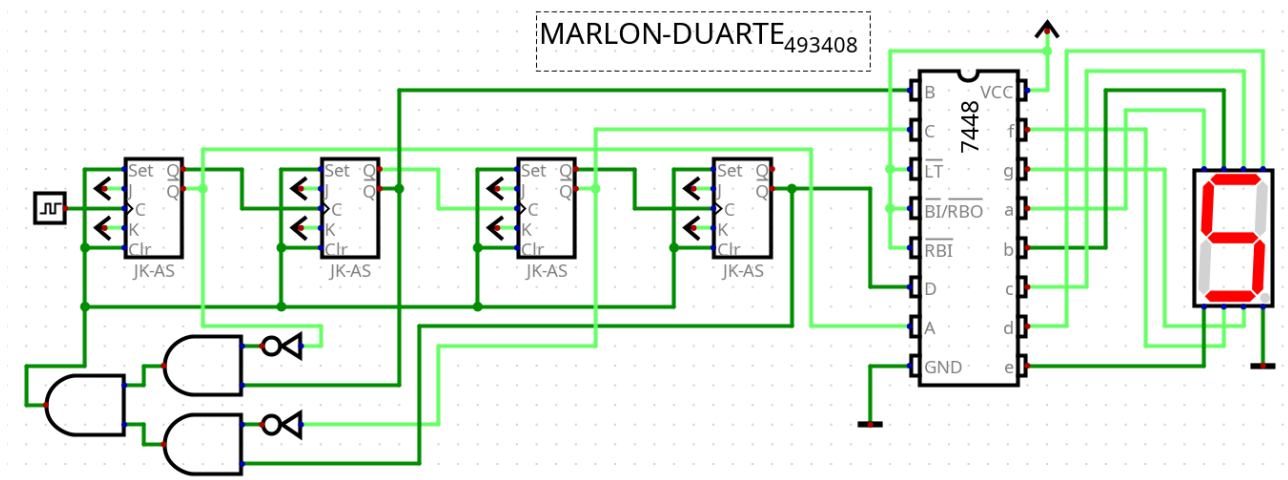


Figura 4: Circuito contador de 0 a 9 utilizando uma lógica para limitar a contagem em 9 decimal.

Na figura 4, encontra-se a implementação do circuito contador de 0 a 9 solicitado na segunda parte da prática 6. Como o sistema dispõe de um único display de 7 segmentos, o que resulta em apenas um dígito de saída, limitou-se a trabalhar com a contagem de 0 a 9. Porém, há no sistema de flip-flops, a capacidade de contar até 15, que, inclusive, poderia ser implementado em hexadecimal, porém o projeto que foi utilizado desde a prática 4, somente é capaz de exibir até o decimal 9. Como uma forma de expandir o aprendizado dentro do estudo de circuitos digitais, foi criado um CI novo para a biblioteca do Digital, o qual recebeu o nome de 74M48, em uma alusão ao circuito a que se assemelha. A tarefa deste CI é exibir na saída do contador a representação hexadecimal que equivale aos números de 0 a 15 decimais. Na figura 5, está o circuito 74M48 no contador exibindo no display o dígito E hexadecimal que equivale ao 14 decimal.

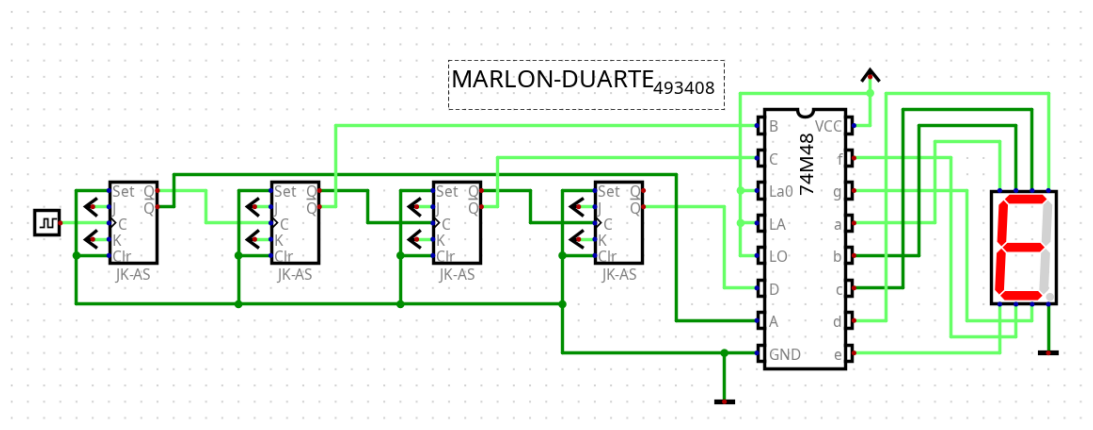


Figura 5: Contador de 0 a 15 com representação de saída hexadecimal.

O contador funcionou perfeitamente, porém o solicitado na prática 6 era um contador de 0 a 9 com sistema de reinício exatamente após o 9 decimal. Como visto na figura 4, foi então desenvolvido um limitador que reinicia a contagem na entrada do 10 decimal, através da criação de uma lógica para acionar a porta clear dos flip-flops o que resulta em zerar as memórias deles contribuindo para o reinício de suas atividades.

A prática 6 ainda possui, na sua segunda parte, uma outra etapa na qual deveria ser implementado o mesmo projeto da figura 4 porém com os encapsulados correspondentes as portas lógicas utilizadas em todo o projeto. Foi necessário então utilizar 4 circuitos integrados para realizar todas as tarefas desta implementação. O CI 7448, que gerencia o display de 7 seguimentos, também foi necessário aqueles correspondentes aos flip-flops utilizados, neste caso seria o 7476 para FF tipo JK em conjunto com mais outro do mesmo para compor 4 dígitos de contagem, haja vista que um somente poderá contar até 3 pois cada um só possui dois flip-flops em sua construção. Por fim, foi utilizado também um CI 7408 que é um quádruplo AND de 2 entradas cada. O resultado pode ser visto na figura 6.

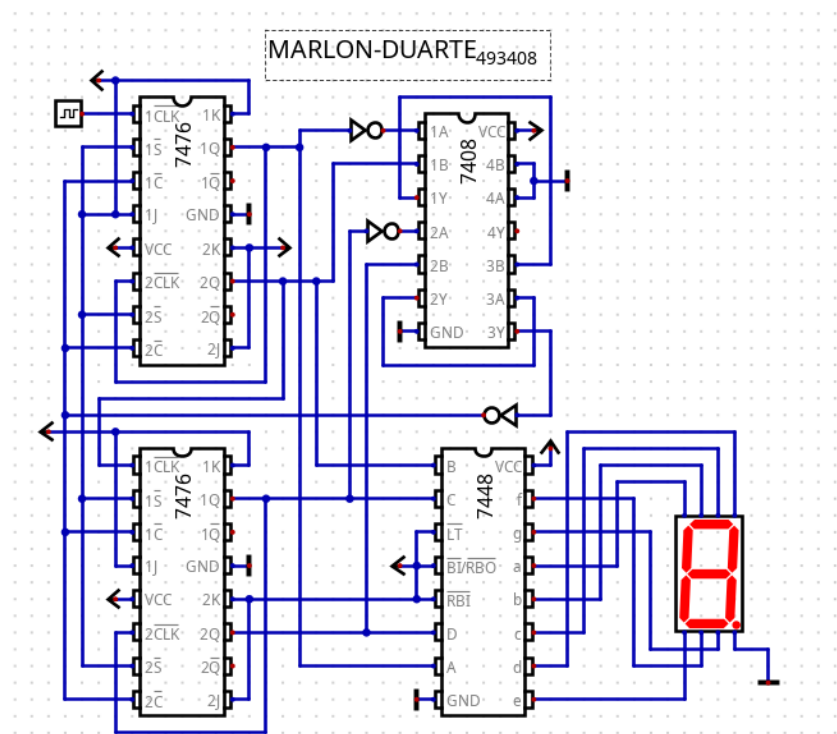


Figura 6: Circuito contador de 0 a 9 com portas AND limitando até entradas binárias 1001.

O circuito da figura 5 funciona perfeitamente, desde que não seja interrompida sua execução na mudança de estado do clock, ou seja, no momento em que os valores são alterados nos flip-flops. Caso isso ocorra, a execução ficará comprometida pois o sistema pulou a contagem do binário 1010 e avançou para 1011 na memorização. Este valor não foi implementado para acionar o sistema de reinício do CI 7408, ação que só ocorreria ao passo que este recebesse 1010. Desta forma, o sistema

continua a sequência 1100, 1101, 1110 e 1111, valores estes que não são comportados pelo circuito do display de 7 segmentos que, por causa disso, passou a exibir formas diferentes, porém que constam no seu *datasheet*. Na figura 7 é possível ver um dos Algarismos mostrados no display quando o erro acontece.

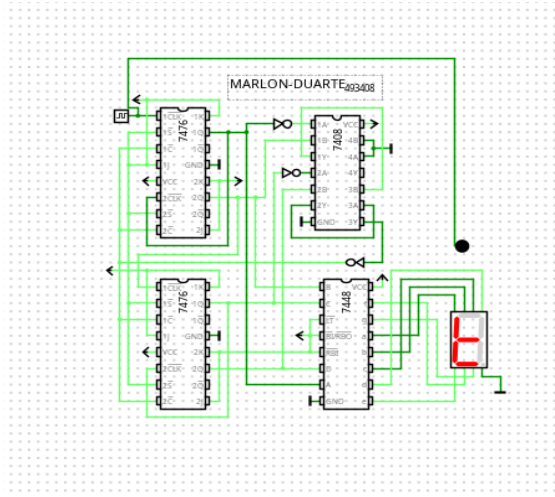


Figura 7: Erro na execução do contador de 0 a 9.

Para resolver o problema apresentado na figura 7, foi necessário a implementação de um reiniciador mais robusto, que impedisse a contagem de qualquer valor maior que 1001. Foi então cabível a construção de uma tabela verdade e um mapa-K que gerasse um circuito que impedisse que os flip-flops, mesmo interrompidos na alteração de estado do clock, continuassem a contagem depois do 1011.

Prática 06 - Marlon Duarte - 493408

A	B	C	D	S
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

AB \ CD	00	01	11	10
00				
01				
11	1	1	1	1
10			1	1

AB \ AC	00	01	10	11
00				
01				
10	1	1		
11	1	1	1	1

Figura 8: Construção da tabela verdade e do mapa de Karnaugh resultado na expressão lógica: $A \wedge B + A \wedge C$.

É importante observar que a expressão encontrada para o circuito responsável por reiniciar a contagem dos flip-flops tem o A como o bit mais significativo. Essa informação é imprescindível na hora de fazer as ligações dos barramentos, pois o CI 7448 tem como bit mais significativo o D e isso pode gerar uma confusão que resultaria no fracasso do projeto. Feito toda essa parte teórica, passando para a prática em si, foi necessário buscar outro CI na biblioteca do Digital, pois a expressão $A \wedge B \vee A \wedge C$ utiliza além do AND um OR. Um dos circuitos integrados que implementam portas lógicas OR, é o 7432 e está na biblioteca do Digital. Ao final das ligações e observadas todas as possibilidades de erros, o circuito resultante pode ser visto na figura 9.

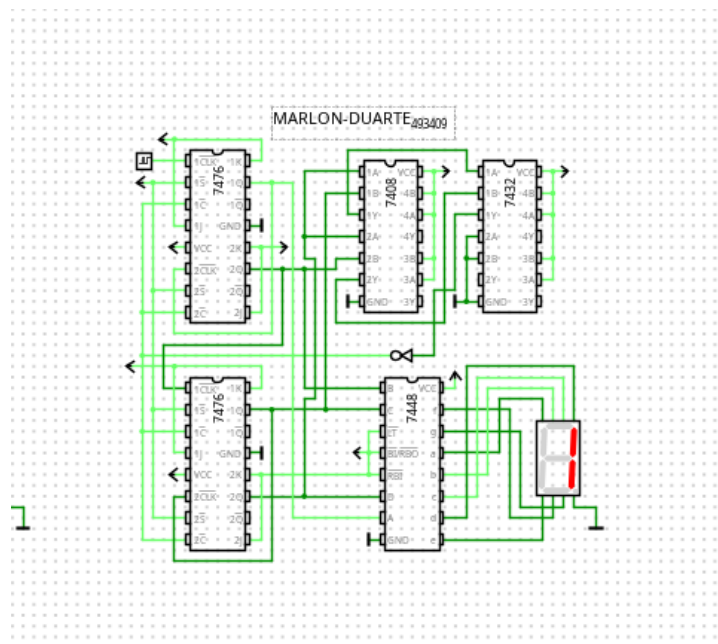


Figura 9: Circuito contador de 0 a 9 com sistema de reinício de contagem melhorado.

Com o projeto da figura 9, mesmo que seja interrompida a contagem na mudança de estado, ou mesmo que os flip-flops memorizem um valor diferente de 1010, porém maior que 1001, o sistema de reinício impedirá que a contagem continue acionando a entrada CLEAR conforme implementado pelos circuitos 7408 e 7432.

CONCLUSÃO

Os projetos trabalhados na prática 6, inserem o estudante em um universo mais complexo e mais abrangente na eletrônica digital. Com o aprendizado adquirido até aqui e com um pouco mais de interesse de cada aluno, é possível desenvolver sistemas simples, mas que possuam aplicação prática no dia a dia. Através das descobertas possibilitadas pelo estudo de circuitos digitais, já é possível pensar projetos eletrônicos capazes de compor um sistema autômato voltado a realizar tarefas simples do cotidiano das pessoas, como: irrigar uma planta, acender luzes, acionar eletrodomésticos, etc. Atividades essas que compõe, mais ou menos, uma área da computação chamada de internet das coisas ou IoT (*internet of things*).

Com a prática 6, a interpretação do funcionamento dos sistemas computacionais fica cada vez mais claro. É possível compreender como funciona, pelo menos basicamente, o sistema de gerenciamento de memória nas linguagens baixas ou linguagens de máquina. Entretanto, é preciso saber que os componentes eletrônicos que atuam nas tarefas de armazenamento de informação são bem mais complexos pelo número absurdo de dados que são calculados atualmente. Circuitos como os contadores que foram utilizados nesta prática, trabalham com *nibble* que são uma minúscula fração do que os computadores pessoais contemporâneos comportam. Ao mesmo tempo, simploriamente falando, a complexidade dos sistemas na verdade são grandes concatenações dos sistemas estudados nas aulas de circuitos digitais.

A realização desta prática, demonstrou-se fundamental nos aspectos citados acima, vide o domínio que o cientista da computação necessita ter em relação a construção de sua máquina de trabalho. Existem profissionais que defendem que o estudante de ciência da computação deveria estudar, introdutoriamente, linguagem de máquina para compreender como os sistemas atuam, além de perceber os reais motivos pelos quais são exigidos cuidados na hora de manipulação de variáveis e alocação de memória. Quando se trabalha com Assembly, por exemplo, o programador deve observar com cuidado o uso da memória, sobretudo quando se trata de dispositivos que possuem um espaço limitado de armazenamento de dados. Tendo consciência deste tratamento, na hora de implementar códigos em linguagens mais altas, o estudante de ciência da computação teria mais cuidado quanto a escolher a maneira de uso dos dados que está manipulando.

REFERÊNCIAS

- CHASE, Otavio; ALMEIDA, F. Sistemas embarcados. Mídia Eletrônica. Página na internet:< www.sbjovem.org/chase>, capturado em, v. 10, n. 11, p. 13, 2007.
- DIAMOND, J.; PEDRYCZ, W.; MCLEOD, D. Fuzzy JK flip-flops as computational structures: Design and implementation. **IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing**, v. 41, n. 3, p. 215-226, 1994.
- DIODO EMISSOR DE LUZ. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2020. Disponível em: <https://pt.wikipedia.org/w/index.php?title=Diodo_emissor_de_luz&oldid=59315460>. Acesso em: 10 set. 2020.
- DISPLAY. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2020. Disponível em: <<https://pt.wikipedia.org/w/index.php?title=Display&oldid=58656332>>. Acesso em: 1 jul. 2020.
- FLIP-FLOP. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2020. Disponível em: <<https://pt.wikipedia.org/w/index.php?title=Flip-flop&oldid=58541950>>. Acesso em: 18 jun. 2020.
- TOCCI, Ronald J.; WIDMER, Neal S.; MOSS, Gregory L. **Sistemas digitais**. Pearson Educación, 2010.