

EA079 Laboratório de Micro e Minicomputadores: Hardware Laboratório IV

Projeto de uma Expansão de Memória

1 Objetivos do Laboratório IV

Os objetivos deste laboratório são

1. Entender as diferenças entre os diversos CIs de memória, suas tecnologias e temporizações e sua adequabilidade a diferentes arquiteturas de hardware.
2. Projetar e implementar um módulo de expansão de memória de 8K x 8 a ser acoplado ao MC68HC11, bem como sua interface de acesso.

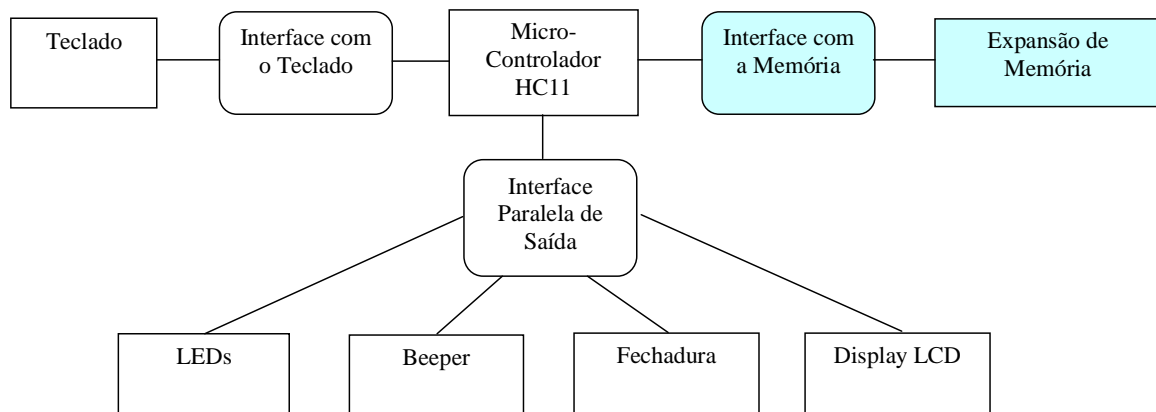


Figura 1 - Arquitetura Global do Sistema e Componentes sendo Desenvolvidos neste experimento

2 Material

- Placa PDHC11-FEEC
- Placa de montagem wire-wrap
- Componentes disponíveis no almoxarifado da FEEC (p.ex.: soquetes, resistores, rolo de fio para wire-wrap, CI's como 74373 e 74138/74139)
- Ferramentas de montagem: kit de wire-wrap
- Ferramentas de depuração: osciloscópio e multímetro

3 Cronograma de Atividades

Este laboratório deve estender-se por 3 aulas. Previamente ao início do laboratório, cada grupo deve elaborar um esquemático para o circuito de expansão de memória, que deve ser discutido com o professor na aula imediatamente anterior ao início do laboratório. Na primeira aula do laboratório, serão iniciadas as atividades de montagem do circuito. Para isso, cada grupo deve ter um esboço dos esquemáticos do seu projeto já validado pelo professor e providenciar junto com o almoxarifado da FEEC os componentes necessários para a montagem. O cronograma de trabalho deve portanto seguir a seguinte sequência:

1a aula: montagem/teste de interface de memória

2a aula: montagem/teste de interface de memória

3a aula: finalizar testes e demos.

4 Projeto Conceitual

Podemos dividir esta parte do projeto em dois subprojetos envolvendo decisões em relação à escolha de memória e ao projeto da interface MCU/Memória [1, 3, 4].

4.1 Escolha de Memória

Nesta etapa, iremos implementar uma expansão de memória de 8k x 8 bits no espaço de endereços \$8000 - \$9FFF.

1. No almoxarifado da FEEC dispõem-se três tipos de memórias: 2114A-5, TMS4256 e 6264. Qual delas você utilizaria para o projeto de expansão da unidade de memória da placa HC11-FEEC?
2. Em termos de circuitaria, quais são as principais diferenças entre a interface com uma memória DRAM e a interface com uma memória SRAM? Quais das memórias disponíveis no almoxarifado são DRAMs e quais são SRAMs? Qual delas é mais rápida, SRAM ou DRAM? Quais seriam as vantagens/desvantagens da escolha por memórias SRAM ou DRAM?
3. Qual é o diagrama de tempo do ciclo de barramento do MC68HC11?
4. Quais são os principais parâmetros de tempo tanto do ciclo de leitura quanto do ciclo de escrita de uma memória SRAM?
5. Analise a memória TMS2114: Qual é a sua capacidade de armazenamento? Qual é a sua pinagem? Quais são as suas principais características elétricas? Qual é o seu diagrama de tempo do ciclo de leitura/escrita da memória? Avalie, de forma estimada, a compatibilidade temporal desta memória com o MC68HC11.
6. Analise a memória HM6224: Qual é a sua capacidade de armazenamento? Qual é a sua pinagem? Quais são as suas principais características elétricas? Qual é o seu diagrama de tempo do ciclo de leitura/escrita da memória?

Avalie, de forma estimada, a compatibilidade temporal desta memória com o MC68HC11.

7. Com base nas suas análises, escolha o componente que você utilizará no seu projeto e justifique a sua escolha.

4.2 Interface MCU/Memória

Temos disponíveis na placa PDHC11 o sinal nSELMX0, que é a saída do decodificador da placa (ativa em baixo) para o espaço de memória \$8000 - \$9FFF (8kBytes), além dos sinais R/nW (já explorado anteriormente) e nLEIA e nESCR (ativos em baixo), que indicam respectivamente operações de leitura e escrita externas à CPU. Na etapa final, esta expansão irá conter o programa principal, e deverá emular uma EPROM. Assim, deveremos implementar um *jumper* que, quando colocado, impedirá a escrita nesta expansão de memória, emulando assim o comportamento da EPROM de programa. Quando o *jumper* não estiver colocado, a memória permitirá a escrita, o que possibilitará a carga do programa a ser executado.

1. O que acontece se os pinos de dados do CI de memória forem conectados de forma trocada (por exemplo, o D0 da MCU for ligado no D1 da memória e vice-versa)? E se os pinos de endereço da memória estiverem trocados? Existe algum problema em termos do aplicativo utilizá-lo para armazenar os dados?
2. Analise a compatibilidade dos parâmetros de tempo/funcionais/elétricos entre a memória e o MC68HC11 com base nas especificações técnicas que você obteve. Circuitos adicionais são necessários para adequar as características das duas unidades? Caso sim, faça um esboço do esquemático para este circuito de interface utilizando os componentes disponíveis no almoxarifado e reavalie a compatibilidade dos parâmetros.
3. Desenhe o esquemático da interface e liste todos os componentes necessários.

5 Testes

Ao contrário do que poderíamos imaginar, efetuar testes em uma unidade de memória não é uma tarefa trivial [2]. Uma vez que o circuito esteja montado e funcionando, diversos defeitos de montagem podem existir, sem que o funcionamento normal do micro-controlador seja afetado. Por exemplo, podem ocorrer curtos entre os bits de endereços e bits de dados, que são difíceis de serem identificados. O resultado, nestes casos, seria escrever numa posição e afetar outra, caracterizando-se um erro na montagem do circuito.

Sugerimos um primeiro teste de memória, com o uso de quatro padrões

01010101 (\$55),
00001111(\$0F),
00110011 (\$33),
10101010 (\$AA):

Escreva inicialmente em todas as posições da memória o padrão 01010101;

varra a memória, comparando o conteúdo de cada posição com 01010101 antes de escrever o segundo padrão 00001111;
e assim sucessivamente até testar todos os padrões.

Um segundo teste será desenvolver um programa que escreve 00000000 (\$00) em todas as posições da expansão, e depois aguarda a colocação do *jumper*. Após a colocação, o pressionar de uma tecla ou do *push-button* da porta E inicia a segunda fase, que é tentar escrever 11111111 (\$FF) em todas as posições da expansão, e após isto verificar se todas as posições preservam o conteúdo \$00. Caso alguma posição não retenha o valor inicial, o fato deverá ser informado pelo *display*.

O teste final será simplesmente recompilar o programa usado no laboratório 3 (teclado) para rodar a partir da posição \$8000 e carregá-lo na expansão. Com o *jumper* colocado, roda-se o programa.

Referências

[1] Alan Clements, Microprocessor Systems Design: 6800 Hardware, Software and Interfacing, 3rd edition, PWS Publishing Company, 1997 (ISBN: 0-534-94822-7)

[2] Ronald J. Tocci and Neal S. Widmer, Digital Systems: Principles and Applications, 7th edition, Prentice Hall, 1998 (ISBN: 0-13-700510-5)

[3] HC11 M68HC11A8 Technical Data. Motorola.

[4] HC11 M68HC11 Reference Manual. Motorola.