### Trabalho Prático 2: Ligador

Software Básico - DCC008

Guilherme Resende Vieira - 2015004178 / Lucas Fonseca Mundim - 2015042134 / Pedro Nascimento Costa - 2015083388

# 1 Introdução

O trabalho tem por objetivo construir um Ligador para máquina 'Swombat', alinhado ao Montador préviamente construído. Para tal, deve-se usar linguagem C ou C++, optou-se por C pelo costume maior em utilizá-la. Para a construção do ligador, foi necessária uma modificação no código do montador para adaptá-lo ao novo propósito.

# 2 Solução do Problema

Para a solução do problema, inicialmente foram estipuladas algumas regras a serem seguidas pelos códigos em *assembly* de forma que o programa seja executado de maneira correta. São elas:

- Todas as linhas de .data devem ser iniciadas com um label;
- Caso exista uma linha com .extern, nada pode vir anteriormente na linha (a existência de um .extern é opcional);
- As *Labels* criadas são globais.

Feito isso, o programa segue dois processos para sua execução: Montagem e Ligação.

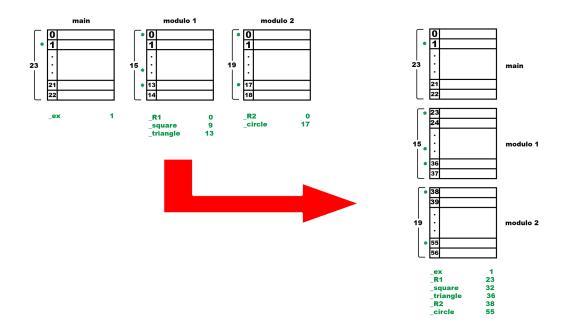
#### 2.1 Montagem

O processo de montagem foi alterado a fim de adaptar o arquivo de saída, gerando um arquivo que contém o número de posições na memória necessárias para o programa de entrada e a lista das *labels* declaradas e sua respectiva posição na memória, além do código em binário do programa dividido de 8 em 8 bits. No caso de uma chamada declarada por uma label, a própria label é escrita na posição onde estariam os 8 bits de endereçamento, em vez de bits binários.

### 2.2 Ligação

O ligador recebe todos os arquivos necessários para o programa final. Em uma primeira etapa, os códigos dos arquivos de entrada são transcritos em um arquivo temporário de forma sequencial, onde seus endereços são atualizados com base na quantidade de posições na memória que já foram atribuídas, porém as labels ainda ocupam a posição de seus endereços. A cada arquivo lido, é formada umanova lista com todas as labels declaradas e suas posições de memória atualizadas. Em uma segunda etapa, o arquivo temporário é copiado para um arquivo de saída definitivo, desta vez checando a lista sempre que uma label é encontrada e, então, substituíndo no arquivo o endereço binário relativo à ela.

Abaixo segue um esquema de como funciona o ligador:



Esquematização de um Ligador

# 3 Avaliação Experimental

Para a avaliação experimental, foi criado um programa simples em Assembly de forma que ele fosse modularizado e chamasse outros arquivos ".a" para testar o ligador, para que fossem gerados os seus códigos em linguagem de máquina, e então testados em um ambiente de simulação da CPU Swombat pelo CPUSim 4.0.9. Para o teste a seguir, o arquivo .mif do programa em Assembly, montados através do montador e do ligador desenvolvido na execução deste trabalho:

## 3.1 tst/teste

O *tst/teste* contém um programa simples, conforme pedido na especificação do trabalho. O código *main.a* contém um programa que recebe quatro parâmetros e, de acordo com o primeiro, chama um dos *moduloN.a* externos para a execução do programa. Suas funções são as seguintes:

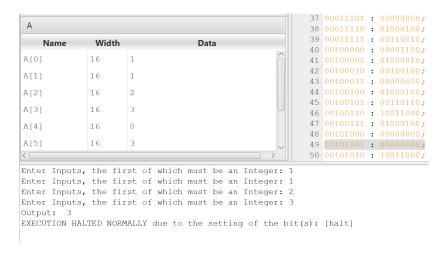
• modulo1.a: retorna o maior valor;

• modulo2.a: retorna o menor valor;

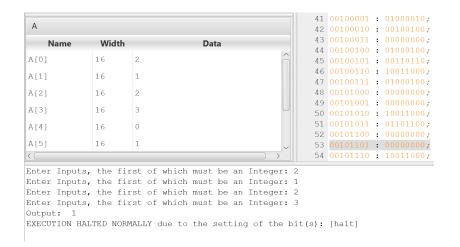
• modulo3.a: retorna o somatório dos valores;

• modulo4.a: retorna o produto dos valores;

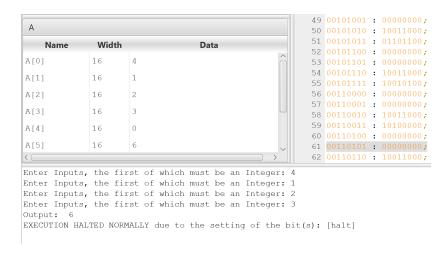
• modulo5.a: retorna a média dos valores.



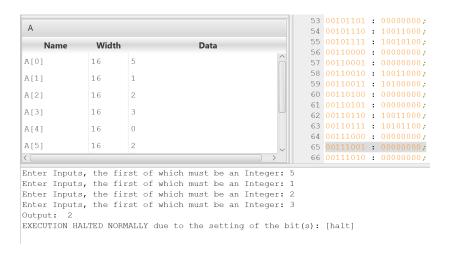
CPUSim: OP = 1



#### CPUSim: OP = 2



**CPUSim:** OP = 4



CPUSim: OP = 5

Como pode ser observado, o arquivo de saída obtido pelo montador desenvolvido é executado corretamente pelo simulador, onde todos os registradores se comportaram como previsto. As imagens em resolução original encontramse em doc/ScreenShots.

# 4 Conclusão

Este trabalho possibilitou o desenvolvimento de um ligador atrelado ao montador anterior para a máquina *Swombat*. O principal benefício proporcionado pelo projeto foi o aprendizado de forma prática de como a ligação na compilação funciona, junto com a montagem.