

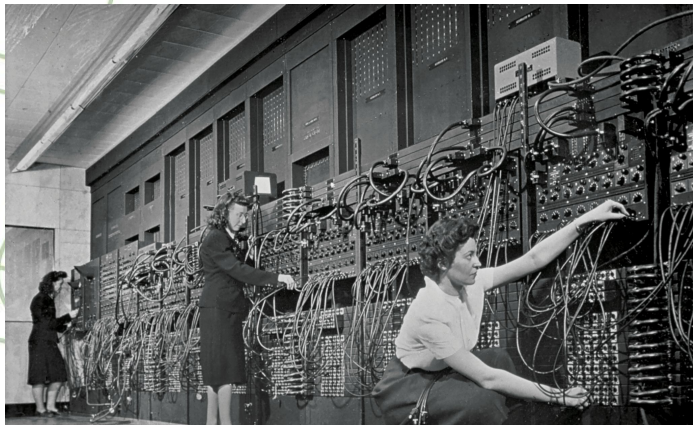
CST Análise e Desenvolvimento de Sistemas AOC786201 - Fundamentos de Arquitetura e Organização de Computadores

Calculadora Rudi
(Material do Prof. Roberto de Matos)

A evolução das máquinas de calcular



Calculadora SHARP com as 4 operações básicas, % e memória



O ENIAC, produzido em 1945, foi a primeira máquina Turing-completa (devendo ter capacidade de armazenamento e manipulação de dados, estruturas de controle e podendo simular qualquer outra máquina de Turing), porém sua “memória de programa” era dada pela disposição de cabos, por conta disso, normalmente não é reconhecido como um computador.

A evolução das máquinas de calcular



O NPL Pilot Ace, produzido em 1950, pode ter sido o primeiro computador de propósito geral. Juntamente com este projeto britânico o qual Alan Turing colaborou, há outros projetos da época que competem como sendo o primeiro computador de propósito geral



No final da década de 1970 e início da década de 1980 computadores pessoais como Apple-II e IBM PC passam a serem produzidos em escala e ganham espaço nos escritórios e casas.

Diferença: Calculadora vs Computador

✗ Uma calculadora apenas executa operações matemáticas pré-definidas

✓ Um computador pode ser programado para fazer qualquer tarefa computacional

Da calculadora de propósito específico a um quase computador

- Será apresentada e implementada uma calculadora com somas e subtrações apenas
- Primeiramente, uma calculadora fixa (propósito específico)
 - Realiza um cálculo pré-determinado (equação fixa)
 - Aceita como entrada apenas os valores das variáveis
- Em seguida, uma calculadora de propósito geral
 - Aceita como entrada operações e valores de variáveis.
- Por fim, uma calculadora programável de computação automática
 - Lê um programa definido pelo programador e executa uma série de passos (limitados a sua capacidade de memória).
 - Não é um computador no sentido de uma Máquina de Turing, mas serve para compreensão de conceitos e implementação típica

Calculadora fixa

- Primeiramente, imagine que é necessário realizar uma sequência de operações, expressas pela seguinte equação:

$$Y = H + I + J + K + L - N - O$$

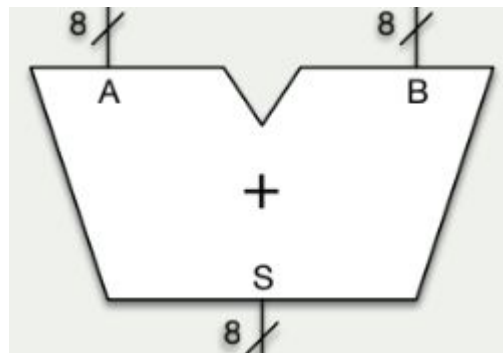
- Neste caso, o valor de cada incógnita desta equação pode variar. Digamos que em um determinado momento os valores sejam: $H=5$, $I=3$, $J=7$, $K=8$, $L=16$, $N=2$ e $O=1$, fica, portanto:

$$Y = 5 + 3 + 7 + 8 + 16 - 2 - 1 = 36$$

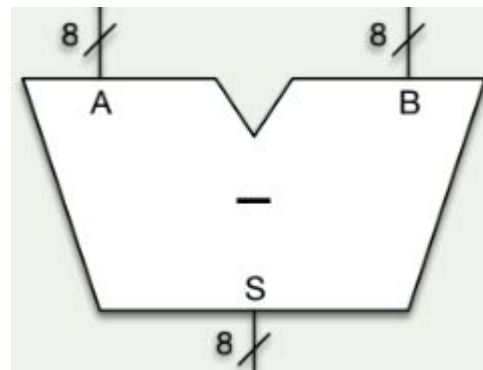
Calculadora fixa

- Se estamos trabalhando com algarismos de 8 bits, podemos realizar estas operações com somadores e subtratores de 8 bits.

Somador de 8 bits:

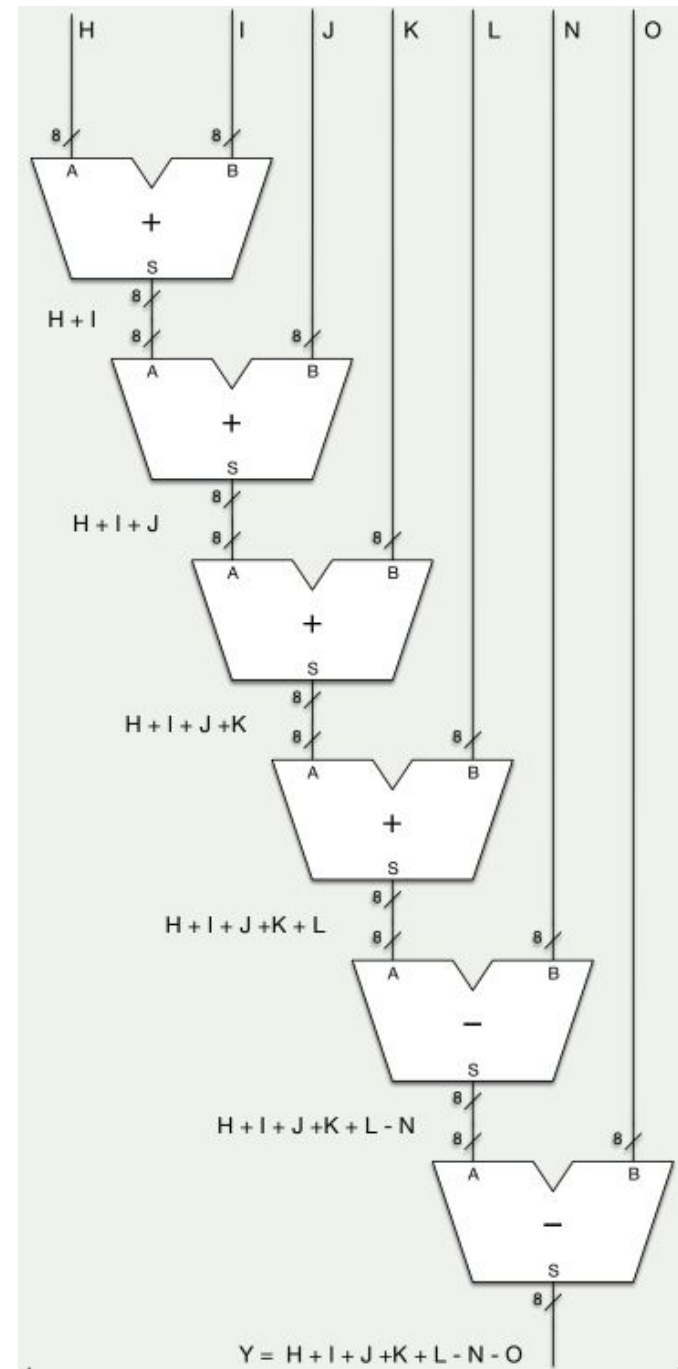


Subtrator de 8 bits:



Calculadora fixa

A implementação desta calculadora fixa que calcula $Y = H + I + J + K + L - N - O$ está ilustrada ao lado



Calculadora fixa

- Esta calculadora seria útil se estivesse resolvendo algum problema em particular. Por exemplo, digamos que se tratasse de um problema de física e H, I, J, K e L são forças positivas a um movimento e N e O são forças opostas ao movimento.

$$Y = H + I + J + K + L - N - O$$

- Mas, na realidade os problemas variam bastante e a própria ordem de solução também pode variar. Por exemplo, se quiséssemos primeiro subtrair as forças negativas e depois somar as positivas, essa calculadora já não serviria, precisaria se reorganizada.
- Os computadores resolve este problema, pois eles executam comandos que estão escritos em uma memória. Para mudar seu comportamento, basta mudar o conteúdo da memória

Calculadora fixa

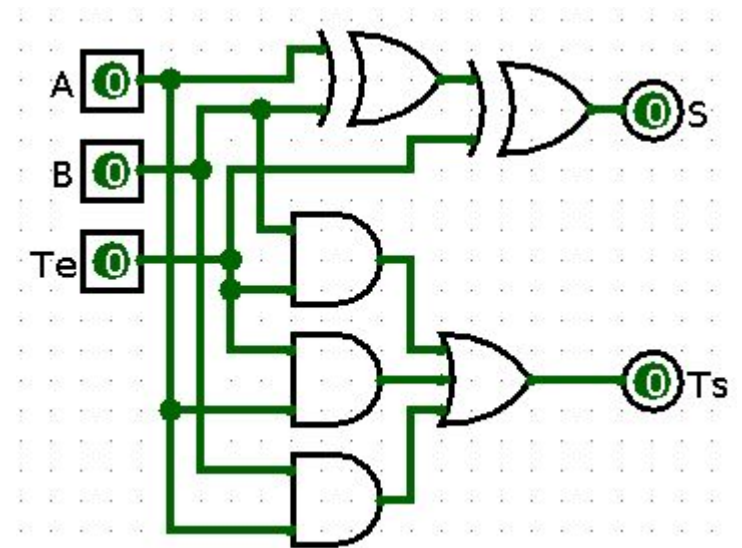
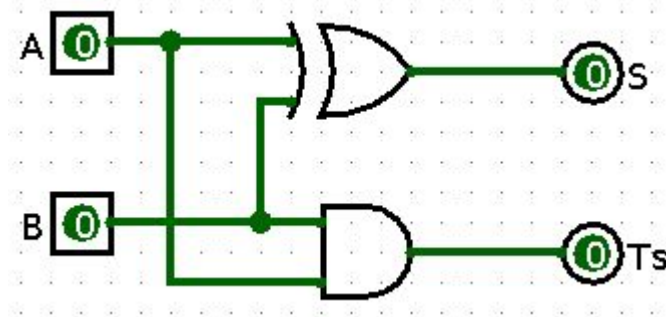
- Esta calculadora seria útil se estivesse resolvendo algum problema em particular. Por exemplo, digamos que se tratasse de um problema de física e H, I, J, K e L são forças positivas a um movimento e N e O são forças opostas ao movimento.

$$Y = H + I + J + K + L - N - O$$

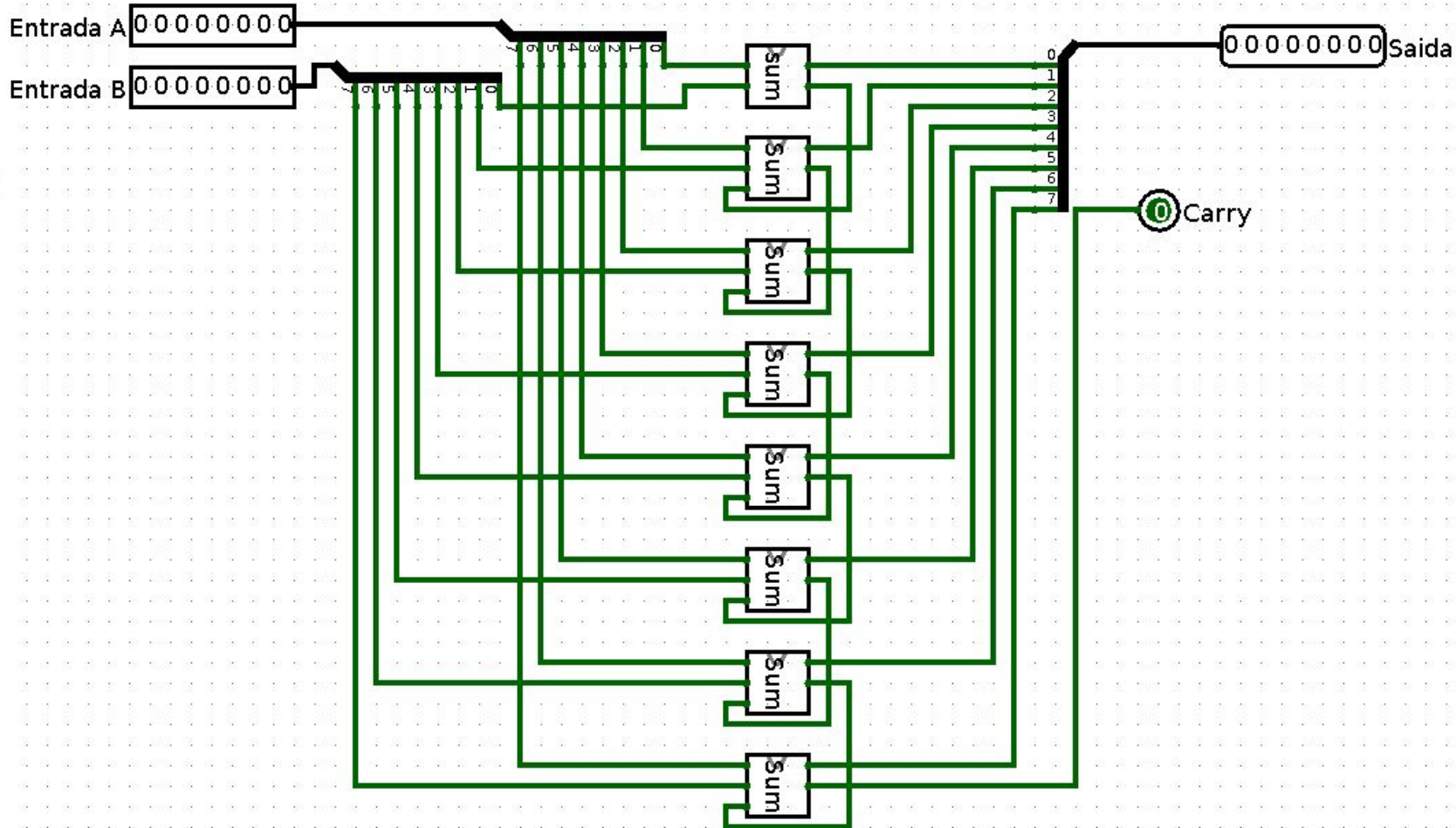
- Mas, na realidade os problemas variam bastante e a própria ordem de solução também pode variar. Por exemplo, se quiséssemos primeiro subtrair as forças negativas e depois somar as positivas, essa calculadora já não serviria, precisaria se reorganizada.
- Os computadores resolve este problema, pois eles executam comandos que estão escritos em uma memória. Para mudar seu comportamento, basta mudar o conteúdo da memória
- Para entender um computador, vamos fazer uma calculadora aprimorada que funciona de forma programável.

Calculadora fixa: Implementação

Meio somador / somador completo

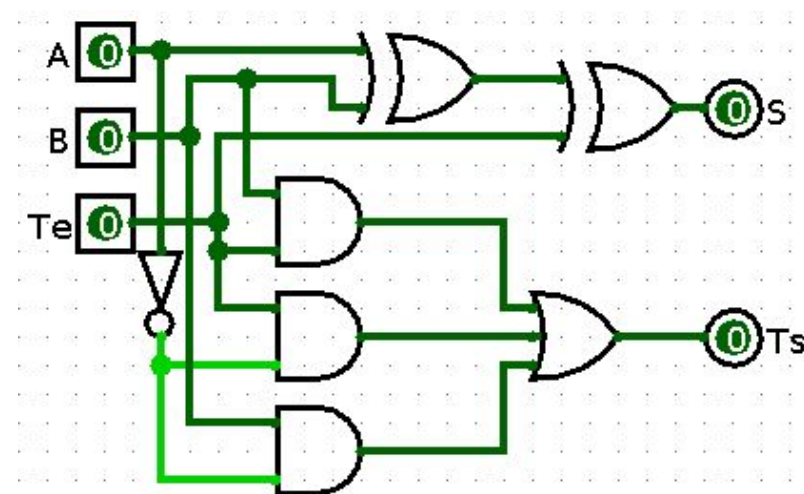
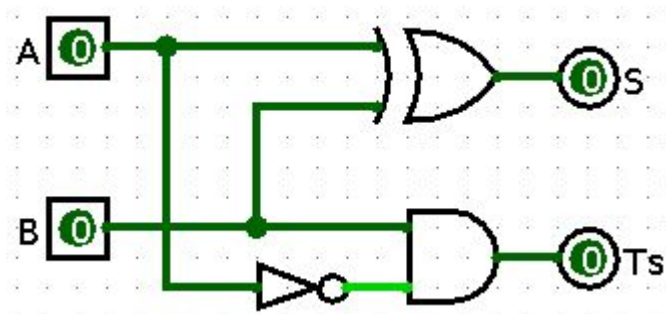


Calculadora fixa: Implementação Somador de 8 bits

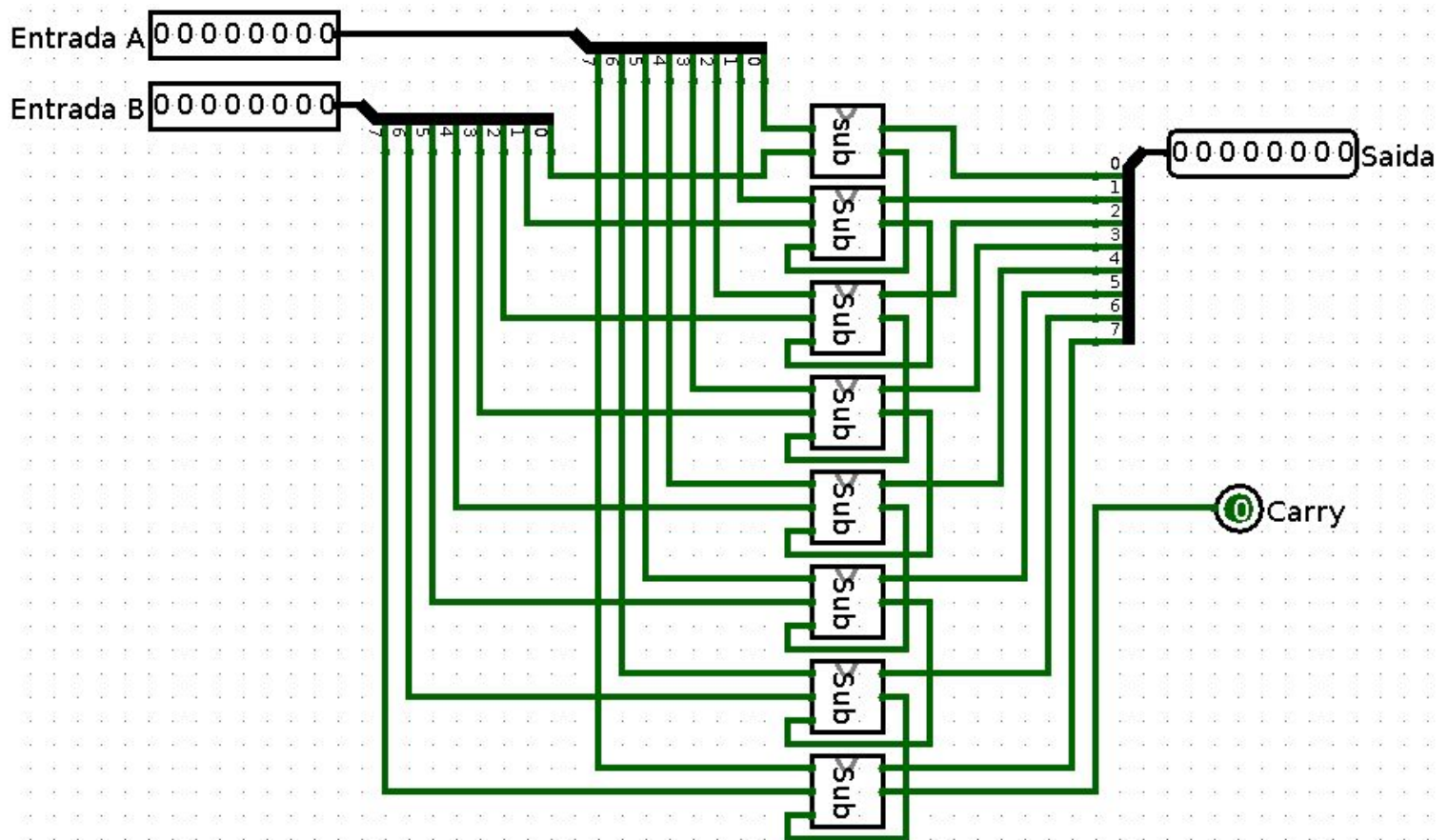


Calculadora fixa: Implementação

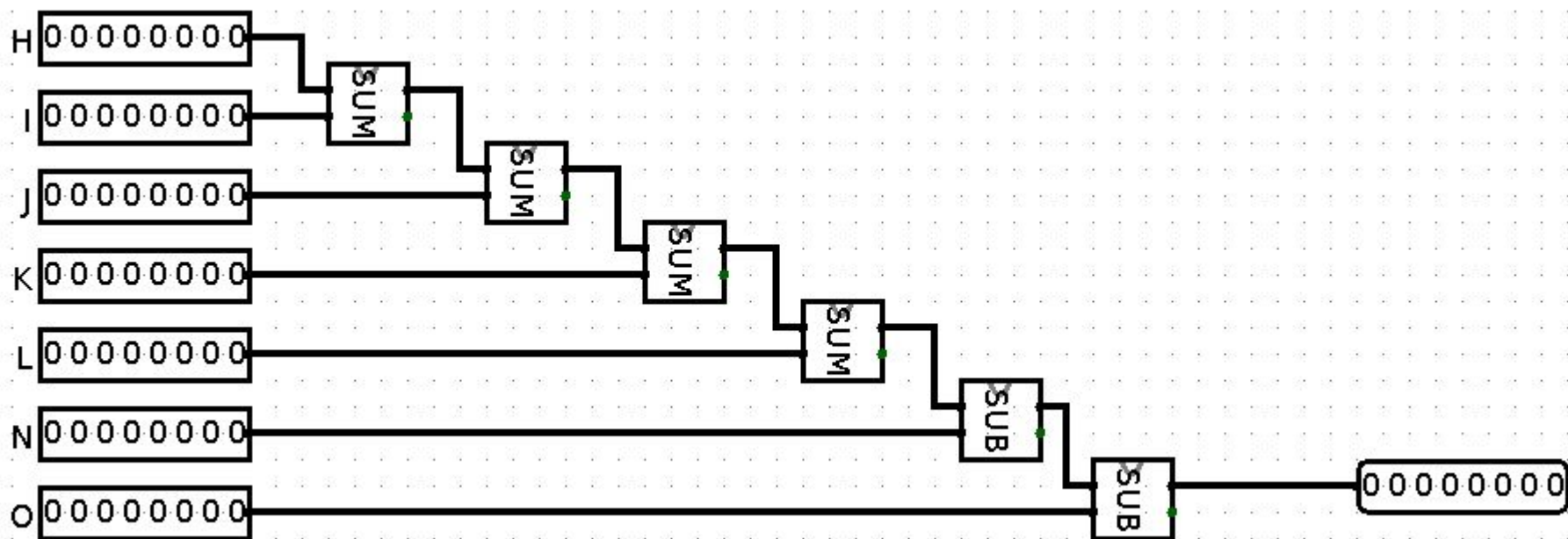
Meio subtrator / subtrator completo



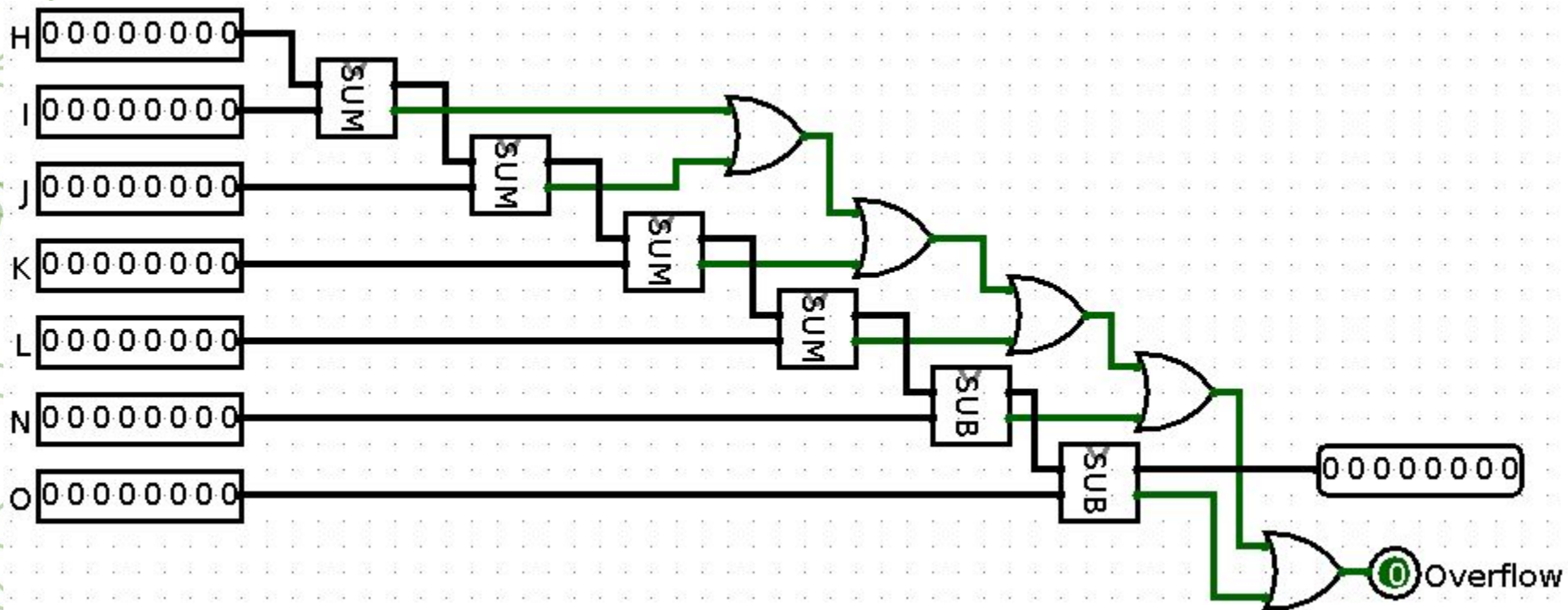
Calculadora fixa: Implementação Subtrator de 8 bits



Calculadora fixa: Implementação

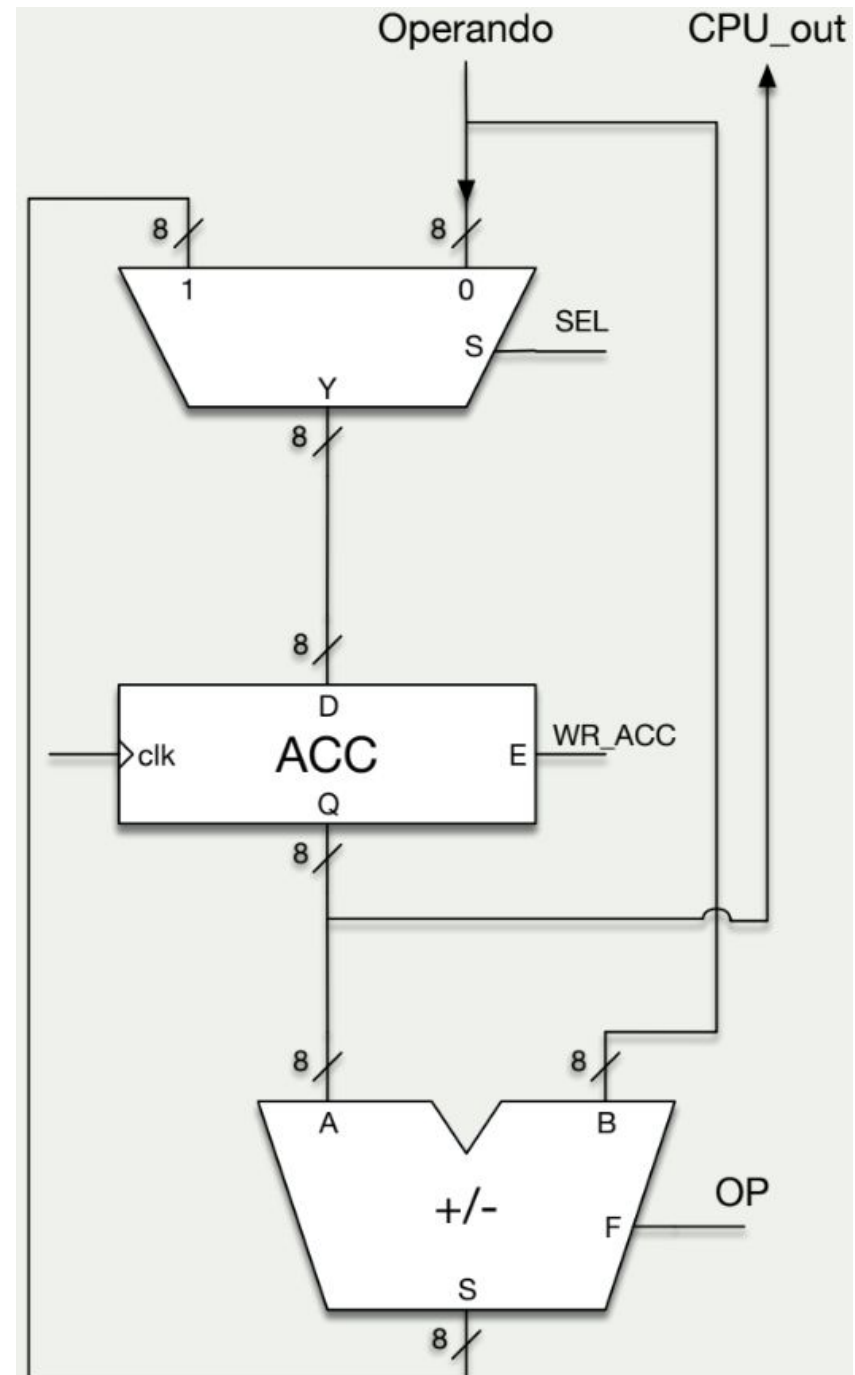


Calculadora fixa: Implementação com overflow



Calculadora de propósito geral

- Uma calculadora programável poderia realiza a operação $Y = H + I + J + K + L - N - O$, ou qualquer outra sequência.
- Para isso a calculadora deve ser capaz de receber a operação (OP) que deve ser realizada (neste caso de soma ou subtração)
- Também é preciso que a calculadora tenha um registrador (ACC) e reaproveite os circuitos internos, podendo utilizar o mesmo somador e o mesmo subtrator múltiplas vezes, conforme demandado pelo programa, assim como uma forma de habilitar a escrita deste registrador (WR_ACC)
- Deve ter uma entrada (Operando) que permita que os numerais que devem ser somados ou subtraídos sejam informados.
- Deve ter uma saída (CPU_out) que apresenta o resultado (parcial ou final).
- Deve também ter uma forma de informar se o Operando deve ser carregado em memória ou ser objeto de operação aritmética (SEL).
- Por fim, uma entrada também de sincronismo (CLK)



Calculadora de propósito geral

Comandos:

$ACC \leftarrow Operando$

$CPU_{out} \leftarrow ACC$

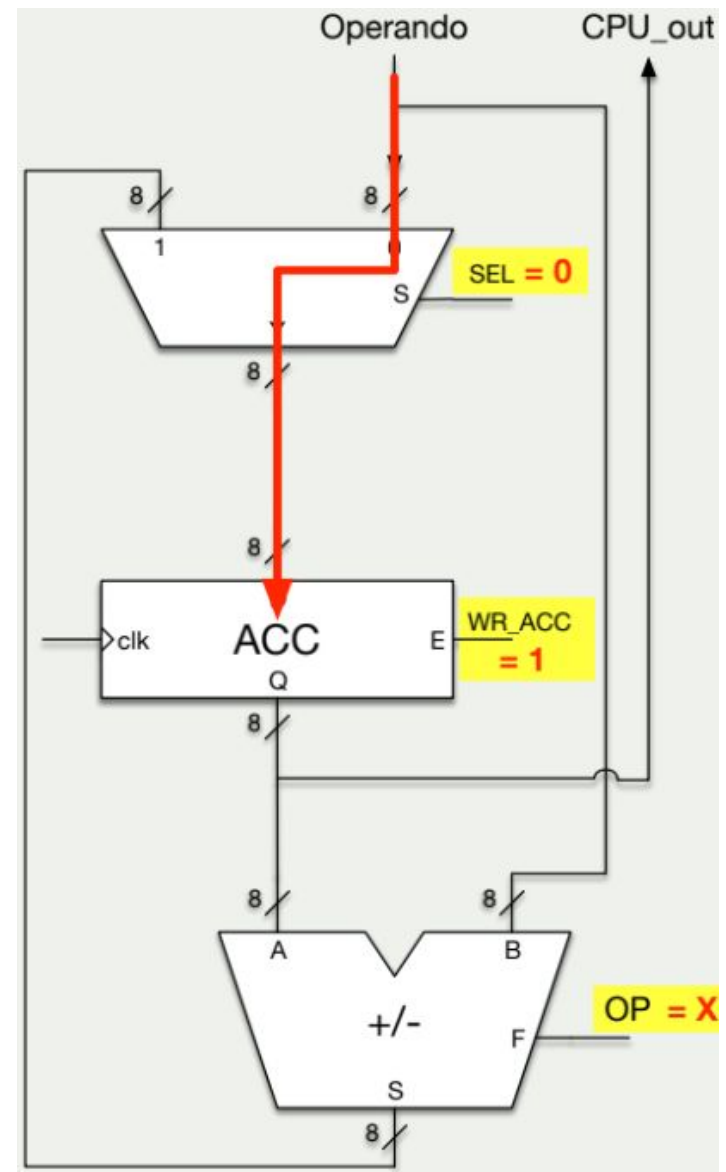
$ACC \leftarrow ACC + Operando$

$ACC \leftarrow ACC - Operando$

Calculadora de propósito geral

Comandos:

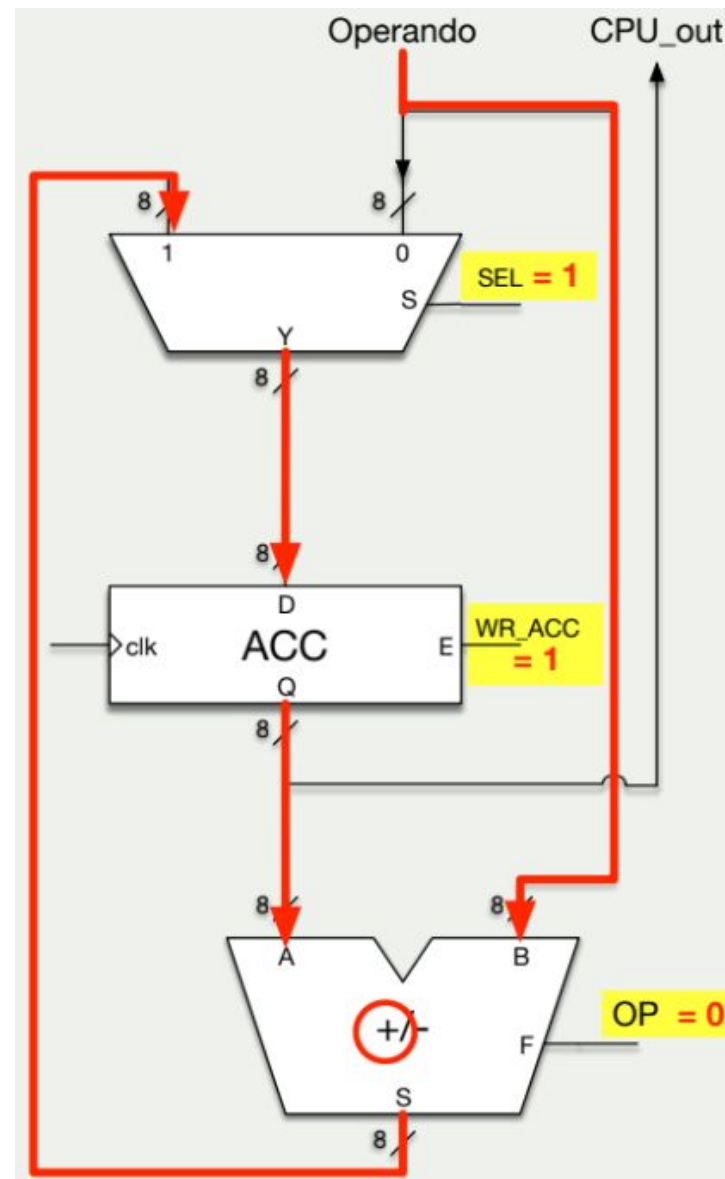
$ACC \leftarrow Operando$



Calculadora de propósito geral

Comandos:

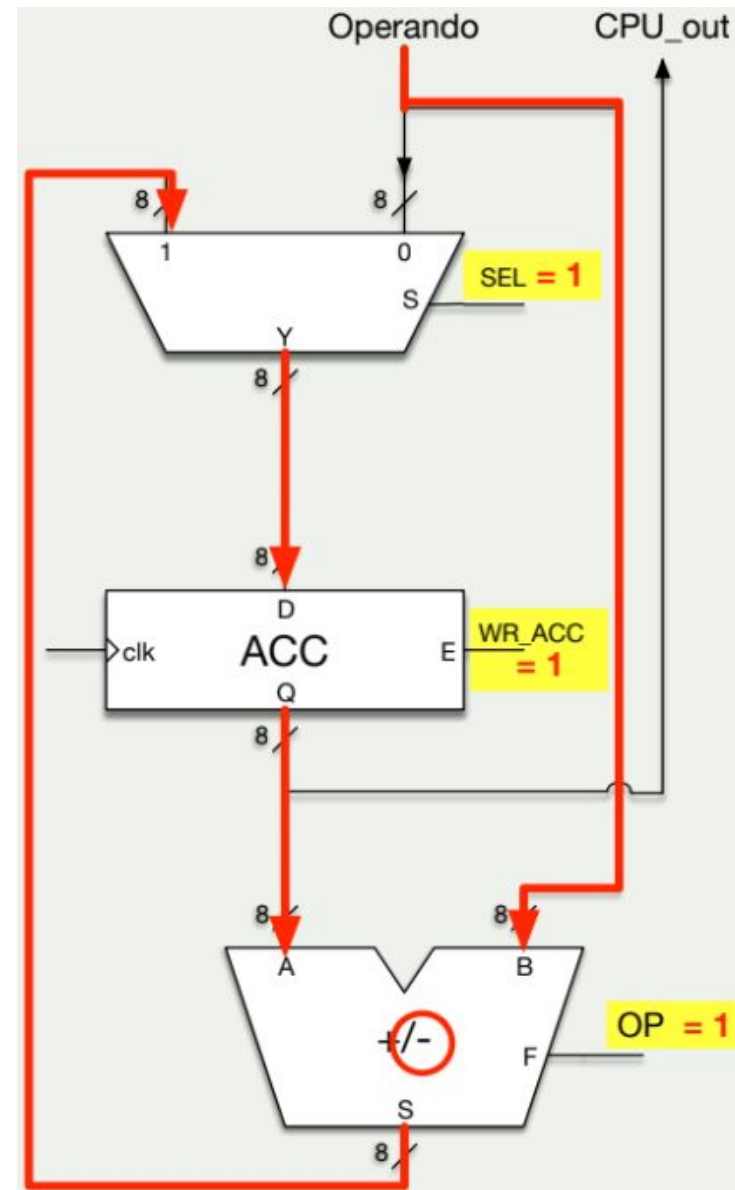
$$ACC \leftarrow ACC + Operando$$



Calculadora de propósito geral

Comandos:

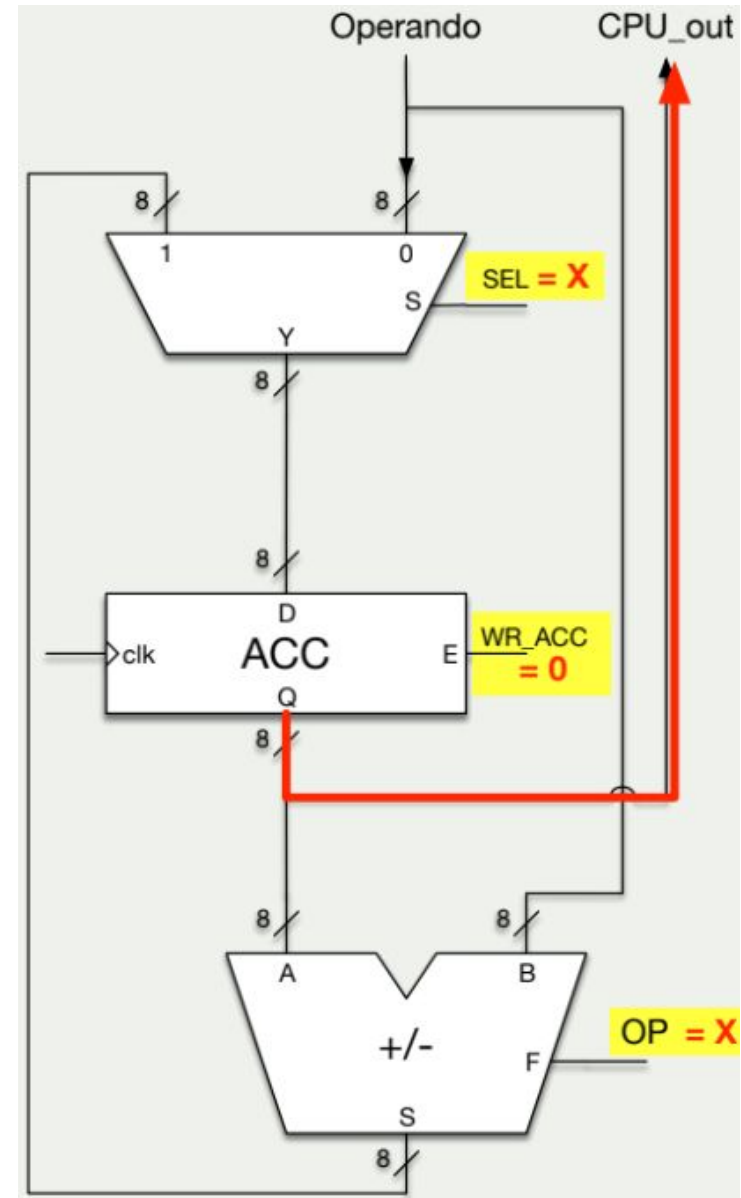
$$ACC \leftarrow ACC - Operando$$



Calculadora de propósito geral

Comandos:

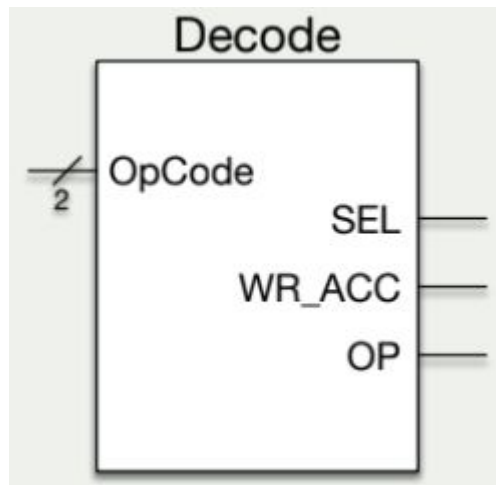
$CPU_out \leftarrow ACC$



Calculadora de propósito geral

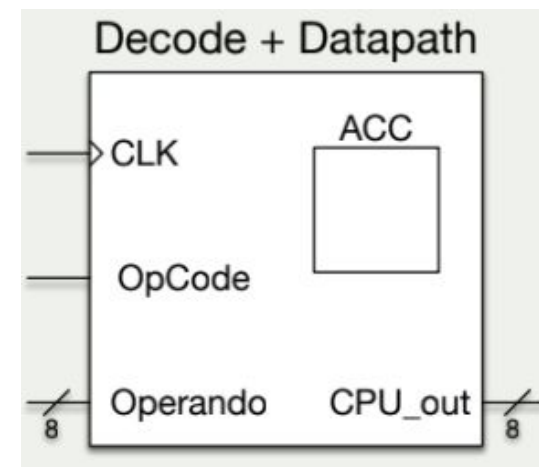
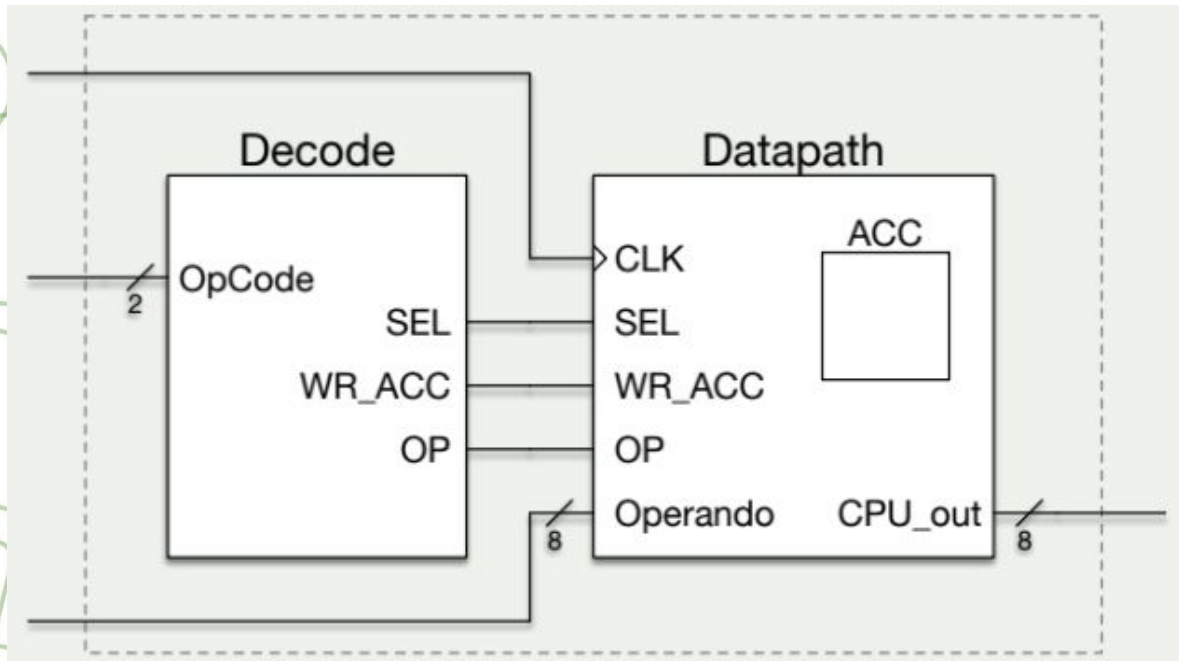
Comandos:

Operação	OpCode	SEL	WR_ACC	OP
$CPU_{out} \leftarrow ACC$	00	X	0	X
$ACC \leftarrow Operando$	01	0	1	X
$ACC \leftarrow ACC + Operando$	10	1	1	0
$ACC \leftarrow ACC - Operando$	11	1	1	1

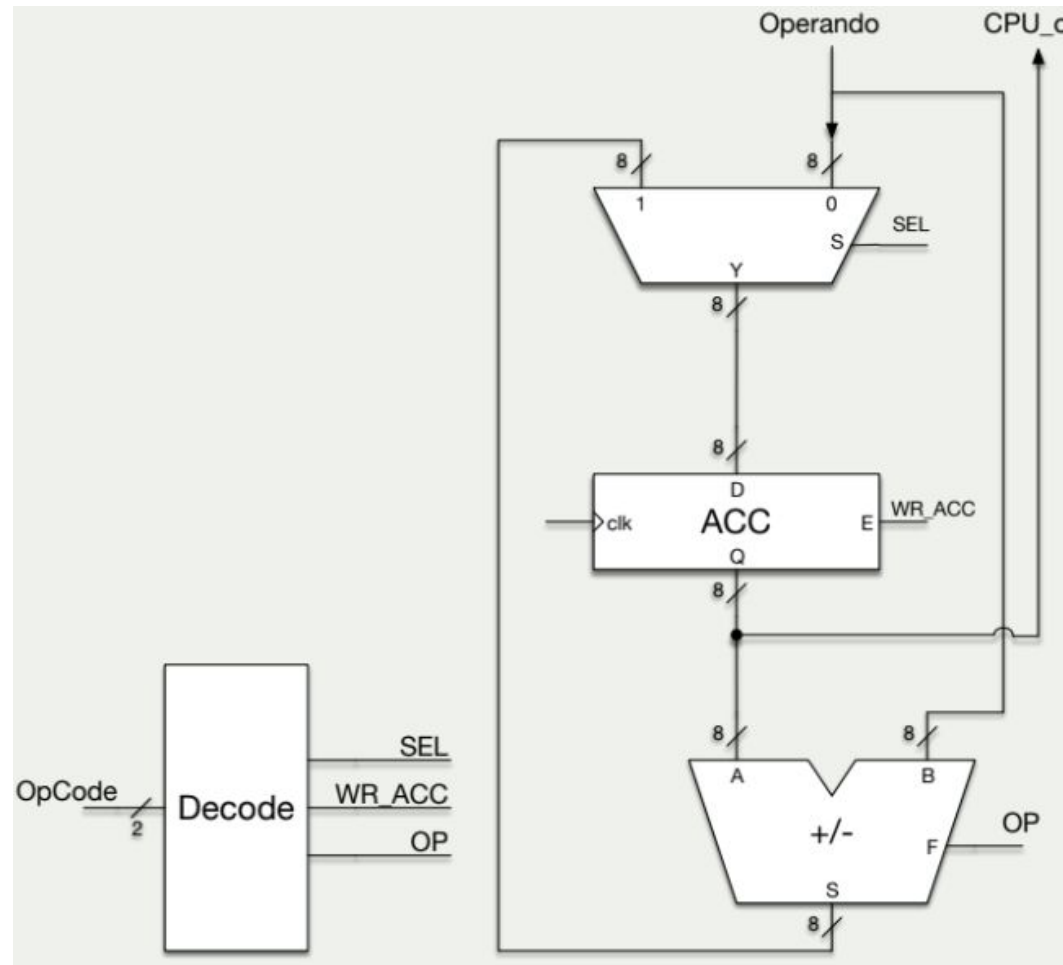
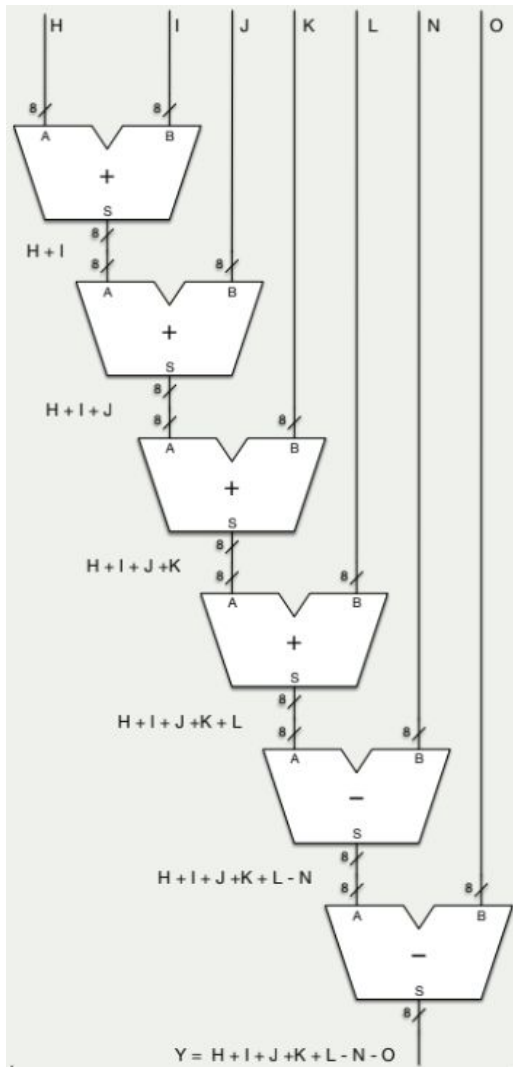


Calculadora de propósito geral

Decode + Datapath



Fixa versus de propósito geral



Calculadora de propósito geral

Exemplo

$$H + I + J + K + L - N - O$$
$$5 + 3 + 7 + 8 + 16 - 2 - 1$$

Operação	OpCode	Operando
$ACC \leftarrow 5$	01	00000101
$ACC \leftarrow ACC + 3$	10	00000011
$ACC \leftarrow ACC + 7$	10	00000111
$ACC \leftarrow ACC + 8$	10	00001000
$ACC \leftarrow ACC + 16$	10	00010000
$ACC \leftarrow ACC - 2$	11	00000010
$ACC \leftarrow ACC - 1$	11	00000001
$CPU_out \leftarrow ACC$	00	XXXXXXXX

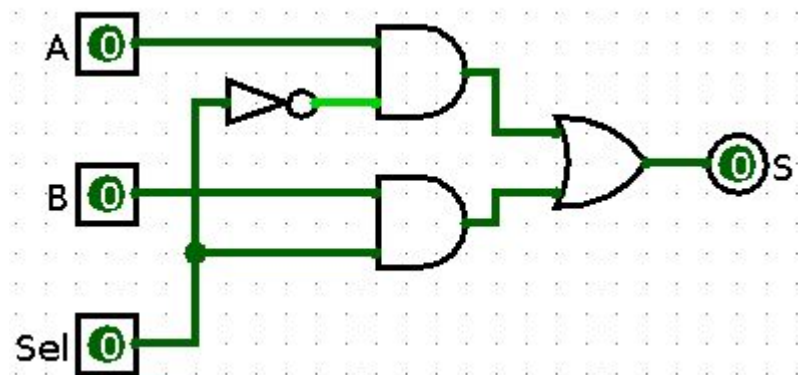
Calculadora de propósito geral

Exemplo

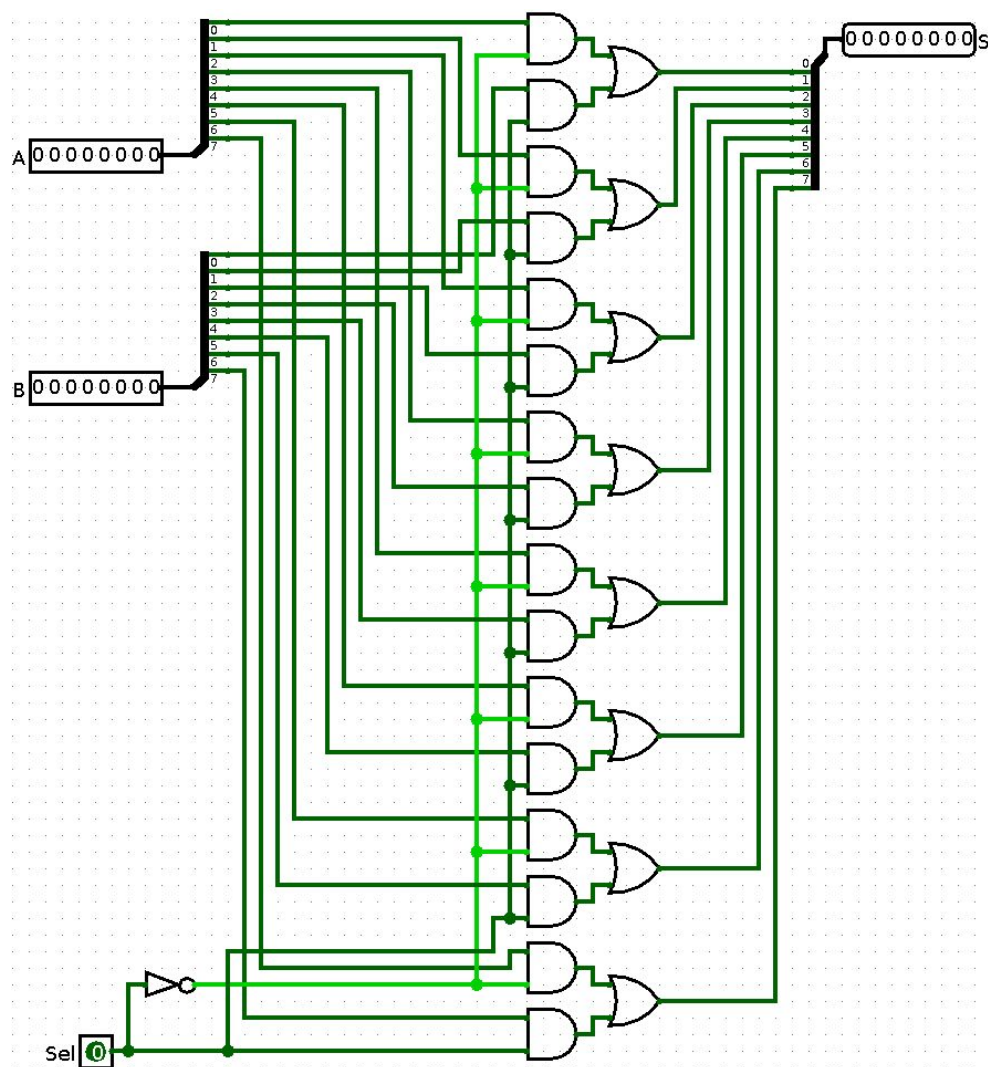
$$H + I + J + K + L - N - O$$
$$5 + 3 + 7 + 8 + 16 - 2 - 1$$

Operação	OpCode	Operando
$ACC \leftarrow 5$	01	00000101
$ACC \leftarrow ACC + 3$	10	00000011
$ACC \leftarrow ACC + 7$	10	00000111
$ACC \leftarrow ACC + 8$	10	00001000
$ACC \leftarrow ACC + 16$	10	00010000
$ACC \leftarrow ACC - 2$	11	00000010
$ACC \leftarrow ACC - 1$	11	00000001
$CPU_out \leftarrow ACC$	00	XXXXXXXX

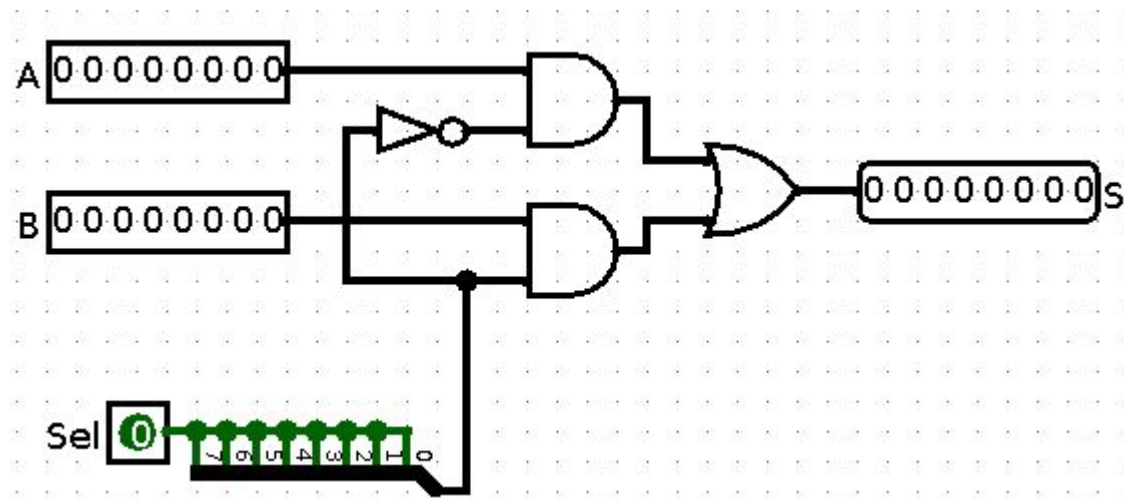
Calculadora de propósito geral: Implementação do Mux 1 bit



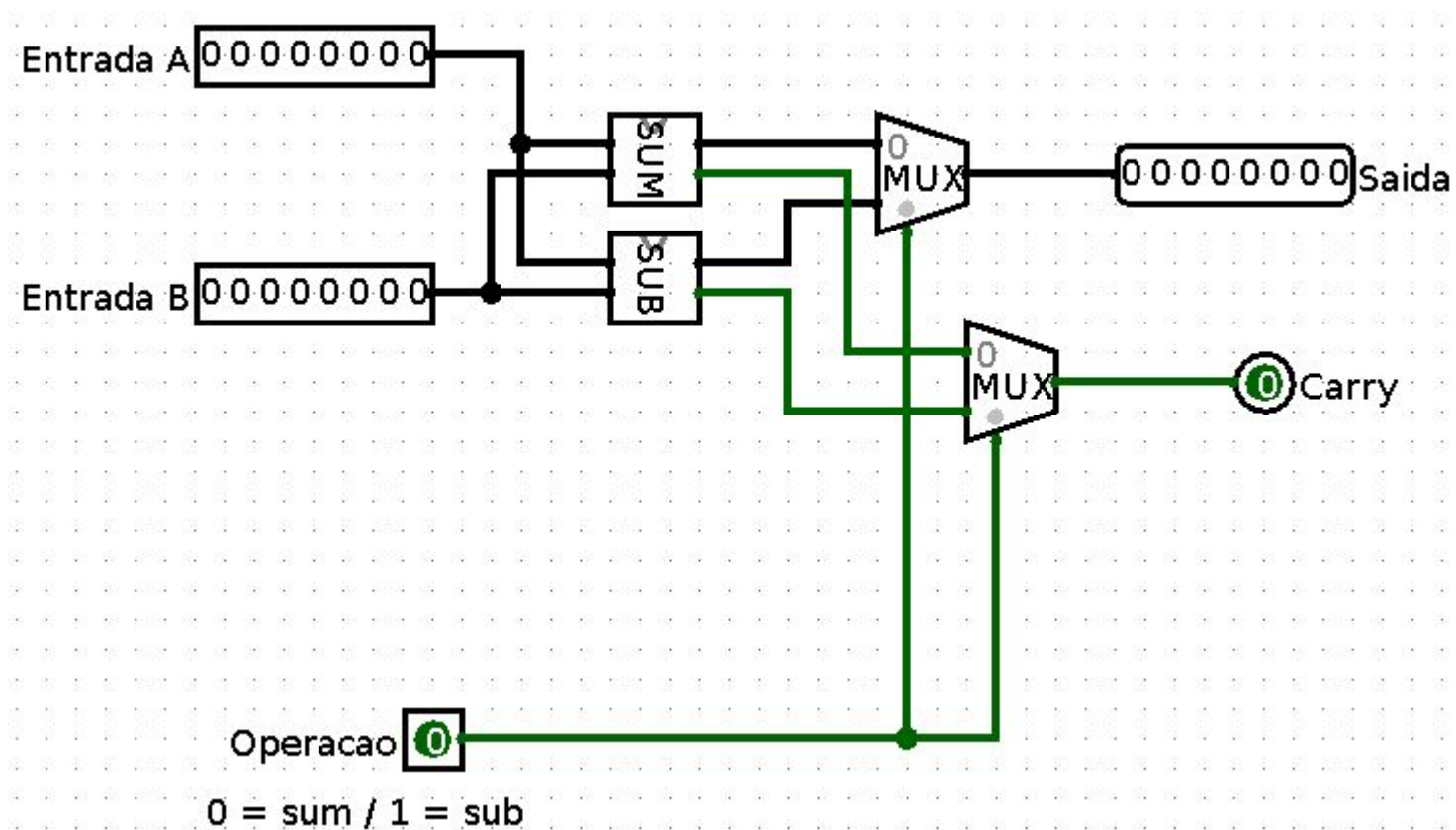
Calculadora de propósito geral: Implementação do Mux 8 bits



Calculadora de propósito geral: Implementação do Mux 8 bits v2

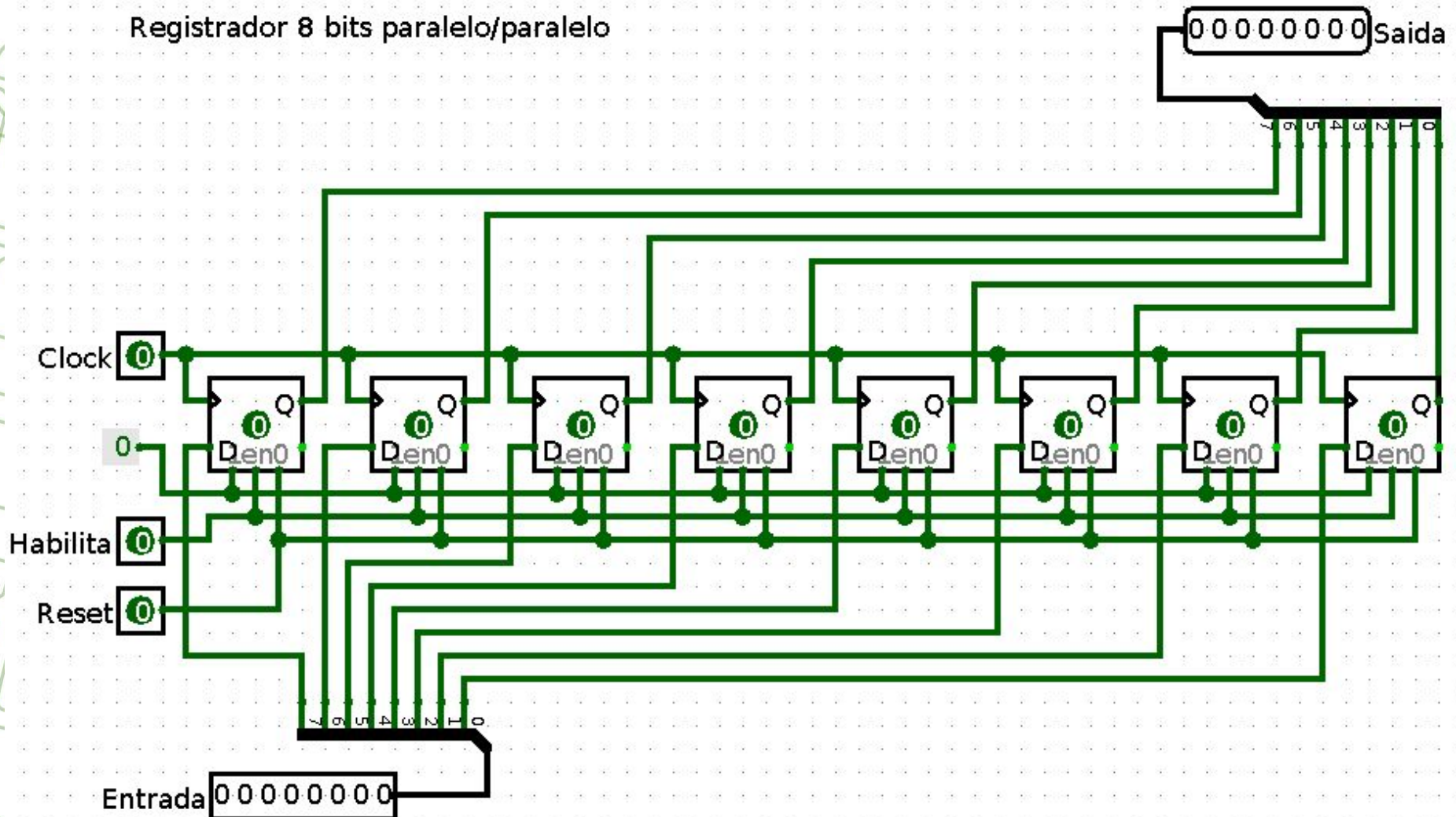


Calculadora de propósito geral: Implementação da ULA

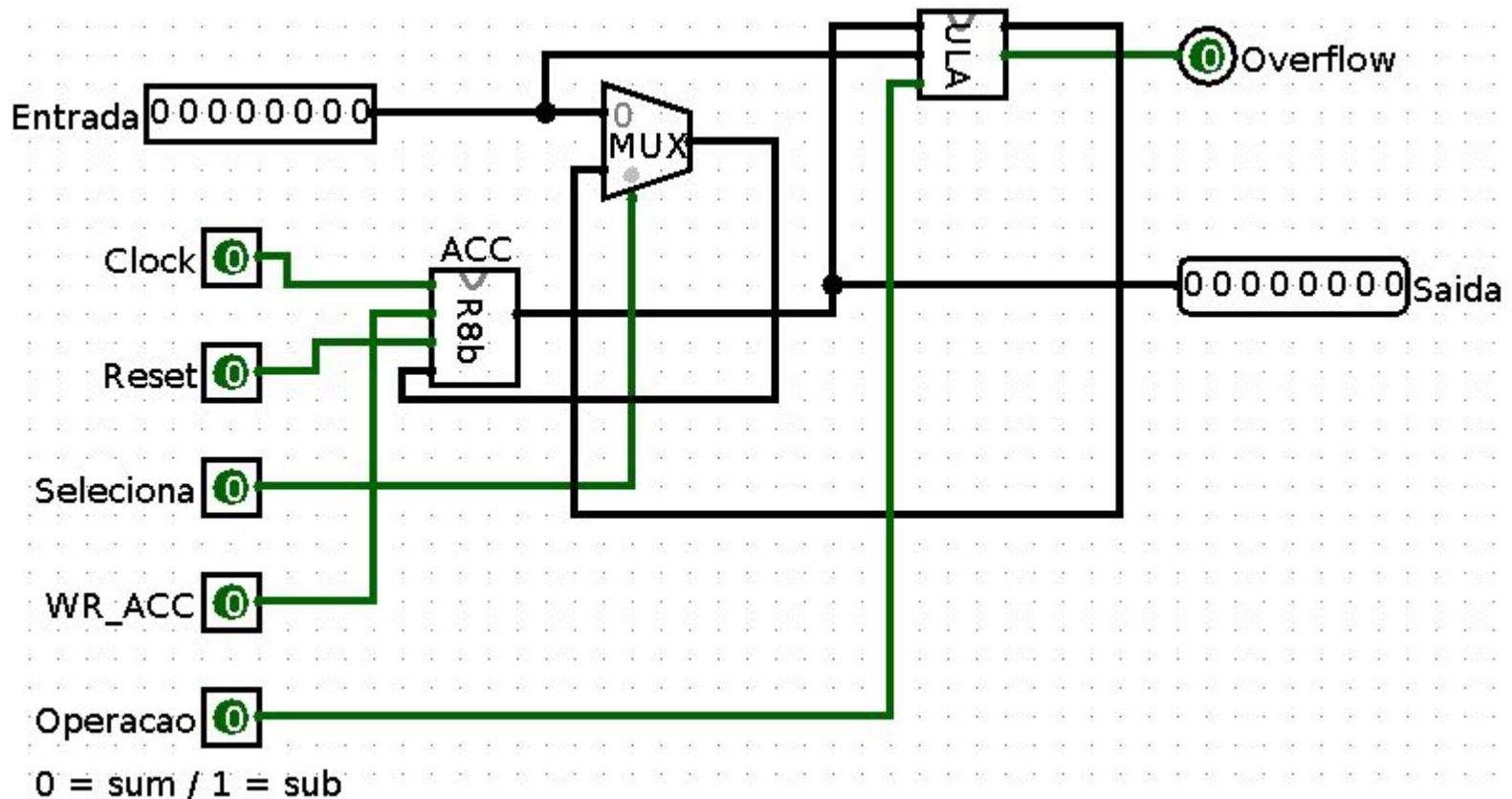


Calculadora de propósito geral: Implementação do registrador 8b

Registrador 8 bits paralelo/paralelo



Calculadora de propósito geral: Implementação do Datapath



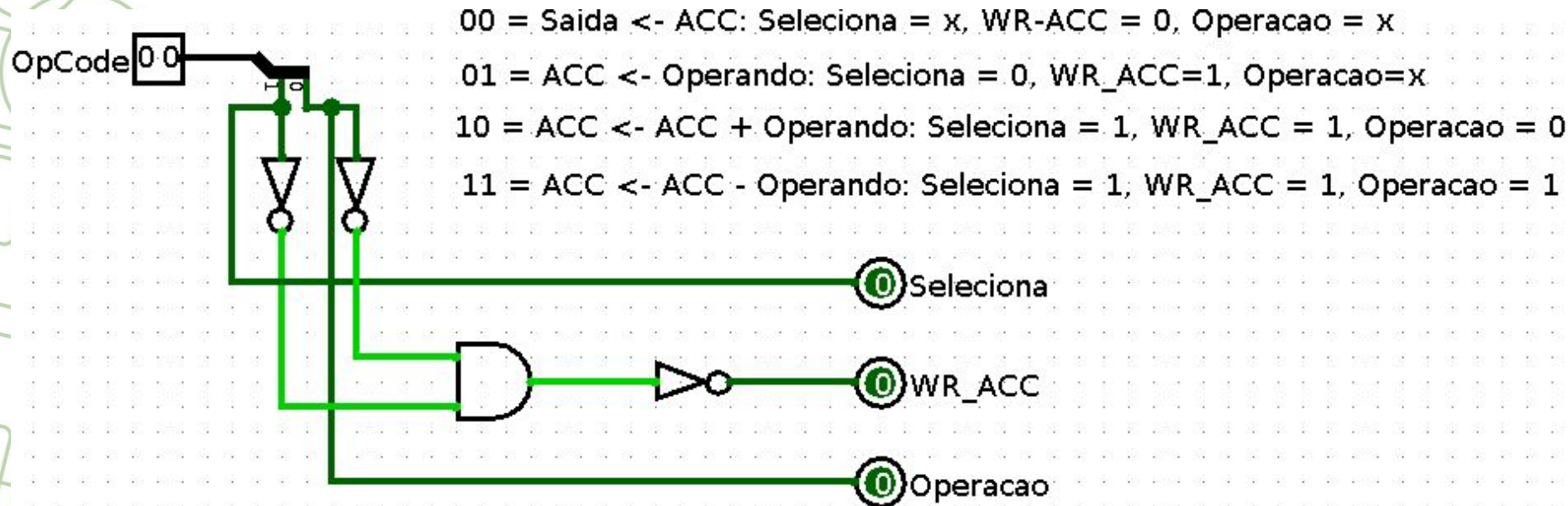
00 = Saída \leftarrow ACC: Seleciona = x, WR-ACC = 0, Operacao = x

01 = ACC \leftarrow Operando: Seleciona = 0, WR_ACC=1, Operacao=x

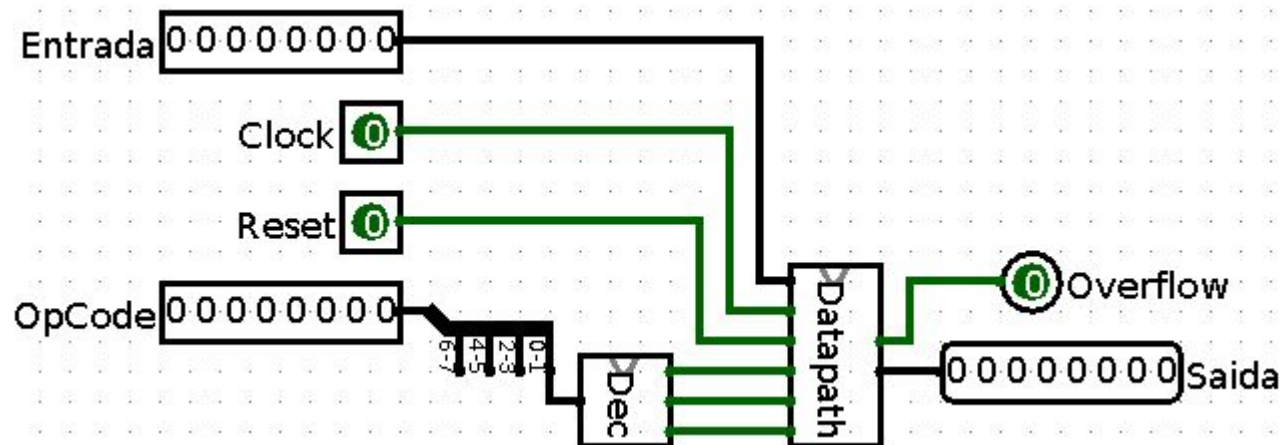
10 = ACC \leftarrow ACC + Operando: Seleciona = 1, WR_ACC = 1, Operacao = 0

11 = ACC \leftarrow ACC - Operando: Seleciona = 1, WR_ACC = 1, Operacao = 1

Calculadora de propósito geral: Implementação do Decoder



Calculadora de propósito geral: Implementação final



OpCode = Operação (na borda de subida do clock)

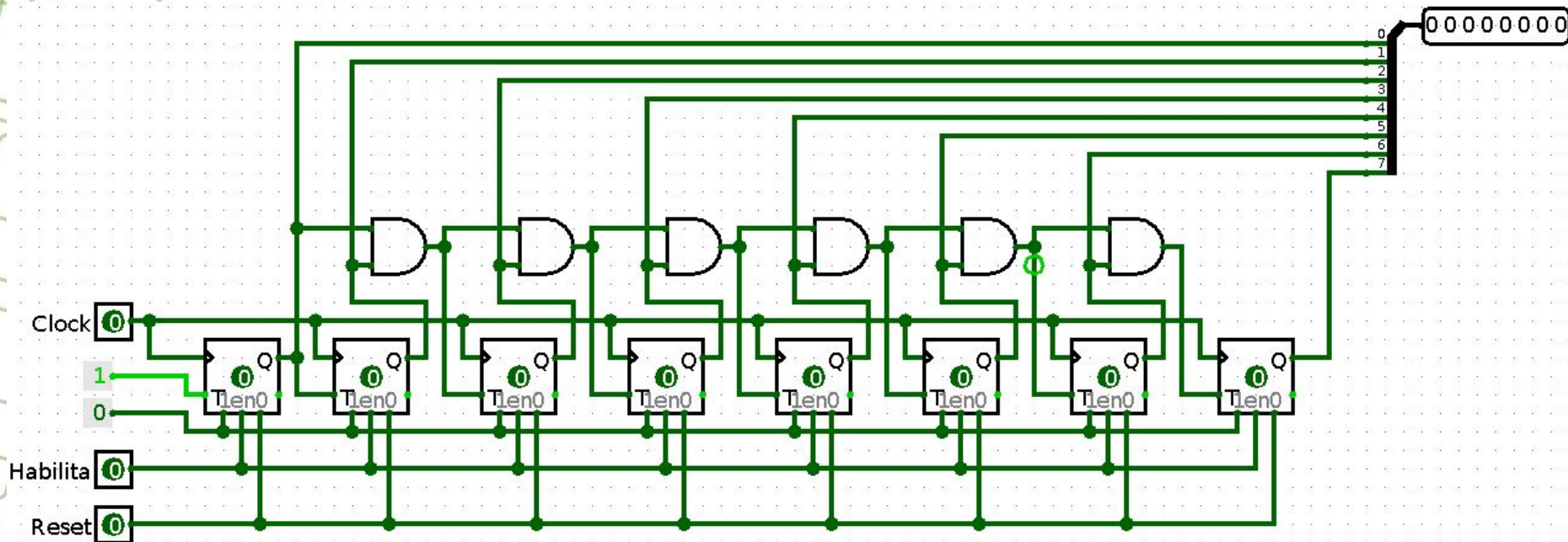
00 = Saida \leftarrow ACC

01 = ACC \leftarrow Operando

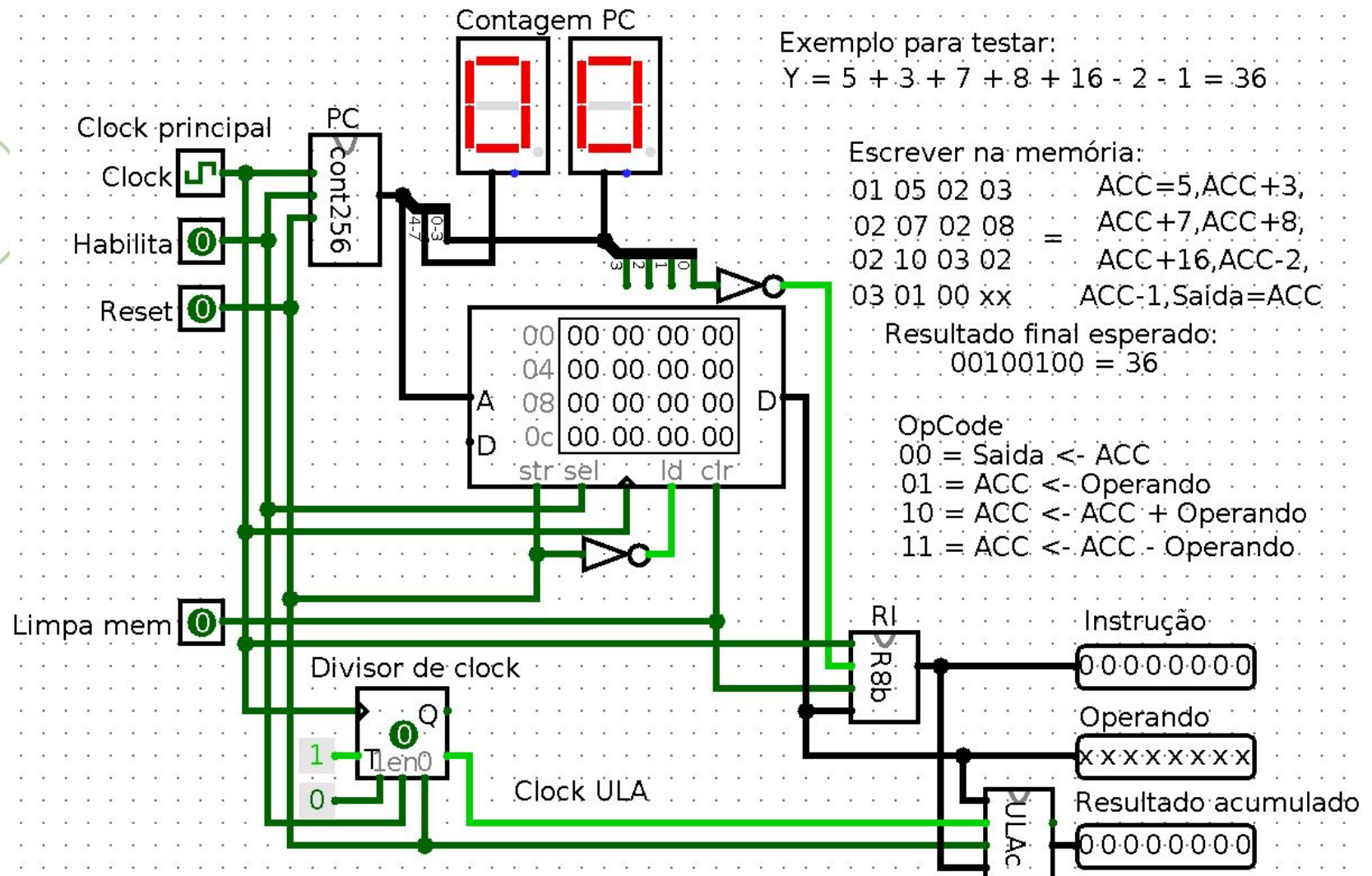
10 = ACC \leftarrow ACC + Operando

11 = ACC \leftarrow ACC - Operando

Calculadora programável: Computação automática



Calculadora programável: Computação automática



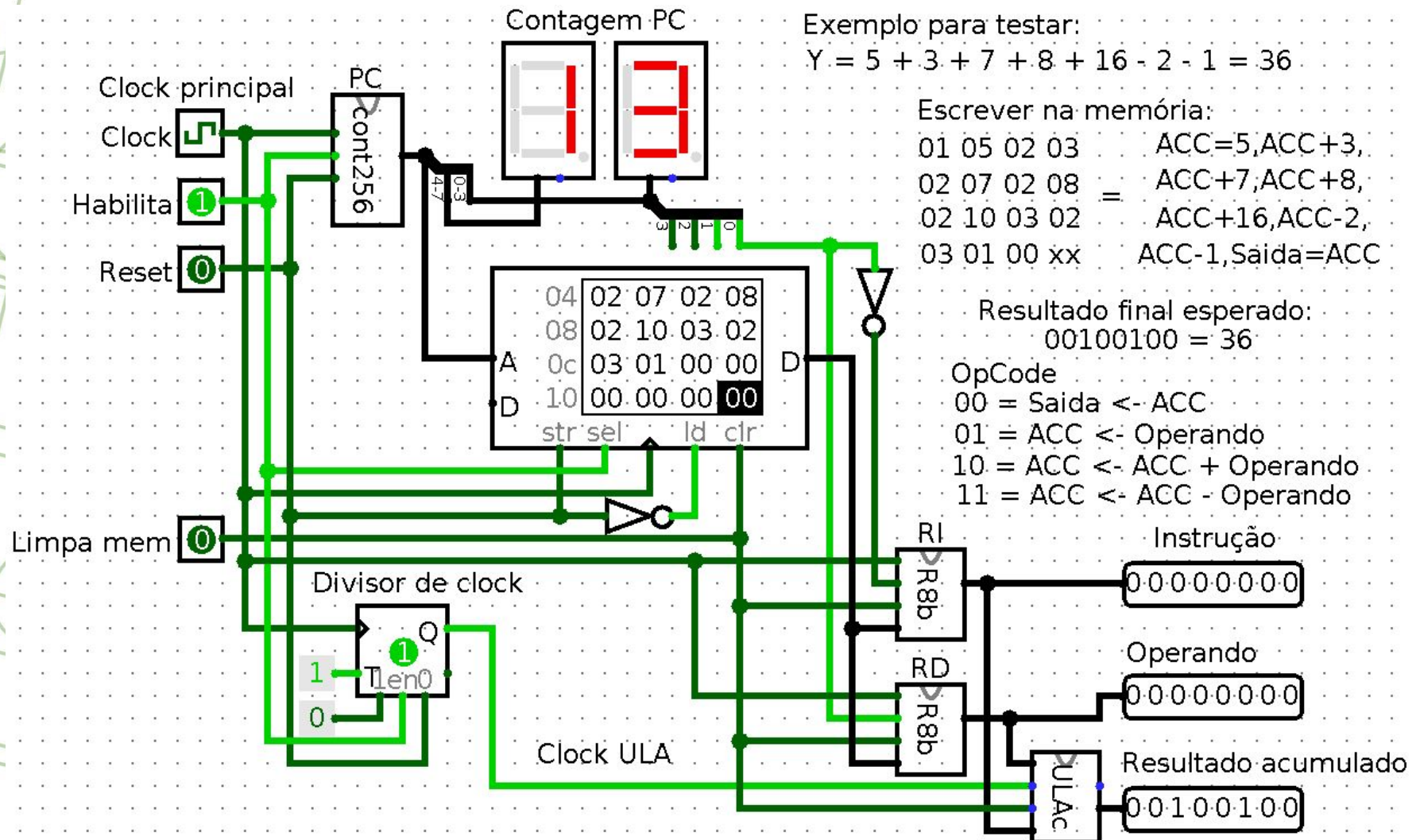
Calculadora programável: Elementos adicionados

- PC (Contador de Programa): Normalmente é um registrador da unidade de controle que armazena o endereço da próxima instrução a ser buscada e executada pelo processador. Durante a execução do programa, o PC é incrementado automaticamente após cada instrução, garantindo a execução sequencial do código.
 - Se este projeto fosse mais elaborado, deveria haver a possibilidade de fazer desvios para um endereço de memória arbitrário.
- RI (Registrador de Instrução): Registrador da unidade de controle da CPU que armazena a instrução em execução que será interpretada pela unidade de controle.
 - Neste projeto o Decoder faz a função de unidade de controle

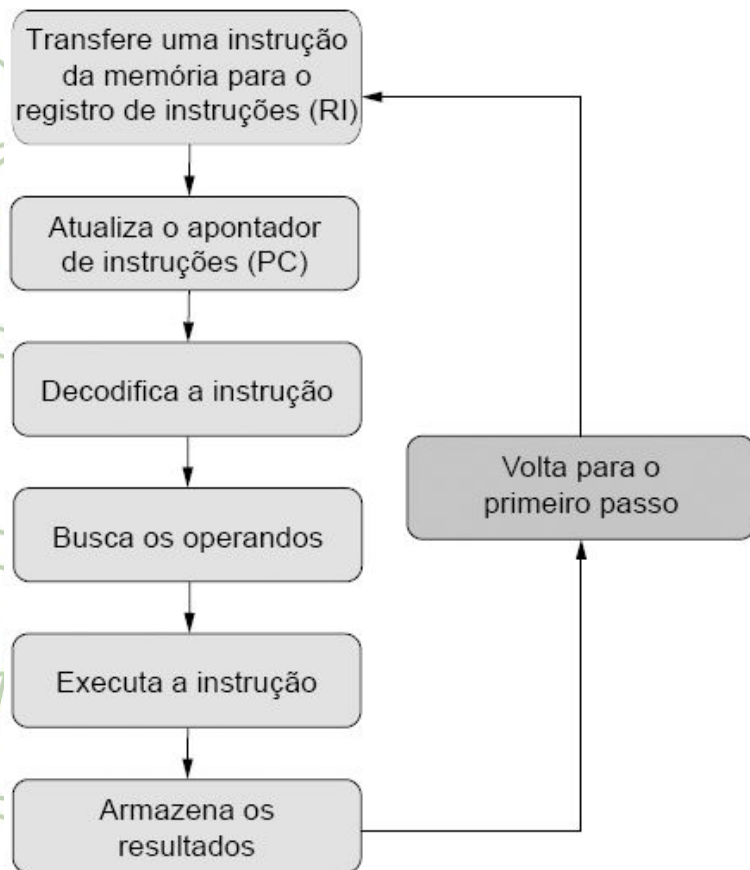
Calculadora programável: Elementos adicionados

- RD (Registrador de Dados ou Memory Data Register), é um registrador normalmente presente na unidade de controle ou na unidade de memória de uma CPU usado para armazenar temporariamente os dados que estão sendo lidos ou escritos na memória principal (RAM).
 - Neste projeto este registrador está sendo utilizado apenas como buffer de leitura.
- Divisor de clock: Aqui é utilizado um flip-flop tipo T (toggle) que muda de estado a cada pulso de clock, dividindo a frequência do clock por 2. Foi aplicado para que no primeiro ciclo de clock seja obtida a instrução e no segundo ciclo o operando (argumento).
 - Nesta implementação, todas as instruções recebem até um argumento, possibilitando realizar a leitura da instrução e do argumento em 2 ciclos do clock principal.

Calculadora programável: Computação automática v2

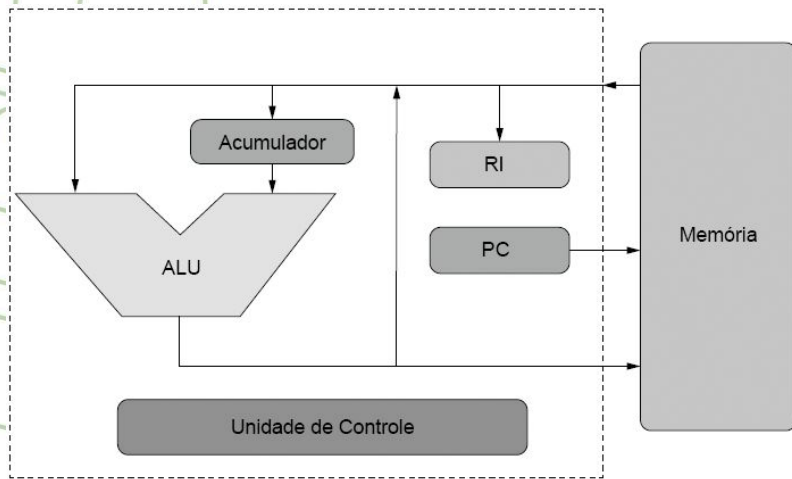


Calculadora programável: Descrição do funcionamento



- O circuito implementado faz a função de um processador, componente responsável pela interpretação das instruções em linguagem de máquina e controle dos demais dispositivos do computador.
- Para executar um programa armazenado na memória, o processador realiza buscas de instruções no endereço de memória indicado pelo Program Counter (PC).
- A instrução é transferida para o registrador de instrução (RI) e o valor do apontador de instruções é automaticamente incrementado para apontar para a próxima instrução na memória.

Calculadora programável: Descrição do funcionamento



- A organização interna de um processador para a execução de suas instruções pode variar bastante. Enquanto os processadores mais básicos contam com pelo menos um acumulador (AC), os modelos mais avançados podem ter dezenas ou até mesmo centenas de registradores para armazenar os resultados temporários das operações.
- Além das instruções aritméticas e lógicas, um processador pode ter vários outros tipos de instruções. Por exemplo, em uma instrução de desvio incondicional, o valor do apontador de instruções (PC) é alterado para o endereço especificado nessa instrução.