**Outline**

Planned register usage:

REG B will hold operand 2 (control)

REG C will hold the bit mask

REGS DE will hold Operand 1 that cab shift to 16-bits

REGs HL will hold the 16-bit sum

**In Memory:**

100 is the address of Operand 1

102 is the address of Operand 2

104 & 105 is where the 16-bit product is to be stored

Load address of operand 1 into HL

Load operand 1 into register E

Zero Register D

move 1 into Reg C for the mask

Load address of Operand 2 into HL

Load Operand 2 into register B

Zero REGs HL for summing

Now we test, add, and shift

1) Move REB B to REG A

AND with REG C (Mask)

Branch if zero to 2)

DAD (Double add DE to HL)

## Now shift Operand 1 in REGs DE

## Rotate REG C (Mask) left through Carry bit

## Branch if no carry back to 1)

2) MOV REG E to REG A

Clear carry bit

Rotate left through carry

Move REG A back to REG E

Move REG D to REG A

Rotate left through carry

Move REG A back to REG D

Move Reg C to A

Clear carry bit

Rotate REG A left through carry

Move REG A to REG C

Branch if no carry to 1)

**Store REGs HL in memory**

Halt

**In Memory:**

100 is the address of Operand 1

102 is the address of Operand 2

104 & 105 is where the 16-bit product is to be stored

Load address of operand 1 into HL

LXI H, 100

Load operand 1 into register E

MOV E,M

Zero Register D

MVI D,0

move 1 into Reg C for the mask

MVI C,1

Load address of Operand 2 into HL

LXI H,102

Load Operand 2 into register B

MOV B,M

Zero REGs HL for summing

LXI H,0

## Now we test, add, and shift

1. Move REB B to REG A

MOV A,B

AND with REG C (Mask)

ANA C

Branch if zero to 2)

JZ 2)

DAD (Double add DE to HL)

DAD

## Now shift Operand 1 in REGs DE

## Rotate REG C (Mask) left through Carry bit

## Branch if no carry back to 1)

2) ## MOV REG E to REG A

MOV A,E

## Clear carry bit by setting to 1 and then complement

STC

CMC

## Rotate left through carry

RAL

## Move REG A back to REG E

MOV E,A

## Move REG D to REG A

MOV A,D

## Rotate left through carry

RAL

## Move REG A back to REG D

MOV D,A

## Move Reg C to A

MOV A,C

## Clear carry bit

STC

CMC

## Rotate REG A left through carry

RAL

## Move REG A to REG C

MOV C,A

## Branch if no carry to 1)

JNC 1)

## Store REGs HL in memory

SHLD 104

Or

**MOV A,L**

**STA 104**

**MOV A, H**

**STA 105**

HLT

**Homework**:

Come up with a sequence of 8080/8085 assembly language instructions that will

store the contents of register pair HL into memory locations 104 and 105.

This is the 16-bit result of multiplying two 8-bit numbers together.

Due Wednesday October 21.

Example from intel document:

LXI H,OOH ;CLEAR H&L TO ZEROS

DAD SP ;GET SP INTO H&L

SHLD SAVSP ;STORE SP IN MEMORY

Example 2:

 DAD D : Add DE and HL

 SHLD 4004H : Store I6-bit result in memory locations 4004H and 4005H.

 HLT : Terminate program execution

From web:

*The 8085 has seven internal 8-bit registers for data, called A, B, C, D, E, H, and L. Six of these registers can also be used in pairs to hold 16-bit quantities. These pairs are BC, DE and HL, where B, D and H hold the MSB of the value and C, E and L hold the LSB, respectively.*

*When these register pairs are transferred to/from memory, the LSB is always stored in the lower address and the MSB in the upper. The SHLD and LHLD instructions can be used to transfer the HL pair to/from an arbitrary address, while the PUSH and POP instructions can transfer any of the three pairs to/from the stack.*

*I hope this answers your question. I'm not sure what you mean when you say L is "used first".*