Multiplication of two binary integers

- Consider multiplying two 4 bit integers in a 4 bit machine

\[ \begin{array}{c}
\text{1011} & \text{multiplier (Q)} \\
\times & \text{1101} \\
\text{1000} & \text{result (R)}
\end{array} \]

Result is 16 bits resulting in overflow condition!

Design Representation

Initial Conditions
Steps to Follow

Initial Step: A = to MDR

- Test Q0 for 1 or 0
  - a. If Q0 = 1; C, A = A + MDR. Move bits to right. (shift_right C-A-Q)
  - b. If Q0 = 0; (shift_right C-A-Q)
- Do step 1 until all bits in Q are tested

Step-by-Step

- Fill initial values

- Set A = MDR

- Q0 = 1
  - C = A + MDR
  - Shift bits to the right

- Q0 = 0
  - C = A + MDR
  - Shift bits to the right
Step-by-Step

- Q1 = 1
- C = A + MDR

- Shift bits to the right

Addressing Modes

- Direct
- Immediate
- Indirect
- Register Direct
- Register Indirect
- Register Indirect plus Offset

Direct Addressing

Used for manipulating an absolute address.
Example: LOAD R1, <ADDR>
will load the contents of <ADDR> into R1

Immediate Addressing

Used when dealing with constants.
Example: LOAD R1, #value
will load value into R1.
Register Direct Addressing

Used to copy the contents of one register into another.
Example: COPY R2,R1 or MOVE R2,R1
will copy contents of R1 into R2.

Example:
COPY R2,R1

or
MOVE R2,R1
will copy contents of R1 into R2.

CPU

Register Indirect Addressing

Typically used for accessing a list of consecutive memory locations.
Example: LOAD R2,<R1>
will load the contents of address stored in R1 into R2

CPU

Register Indirect Addressing plus Offset

Typically used when accessing array and structures.
Example: LOAD R2,R1,offset
will load the contents of address stored in R1+offset into R2

CPU