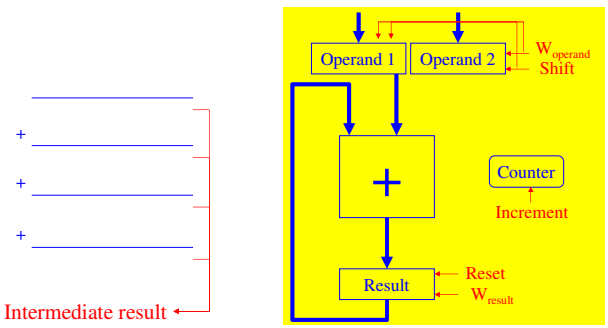
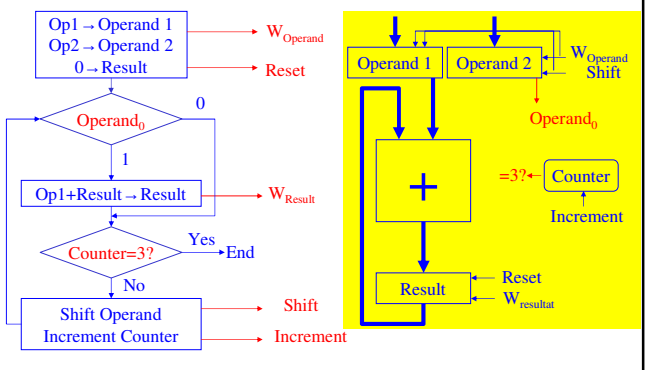
Control

Control

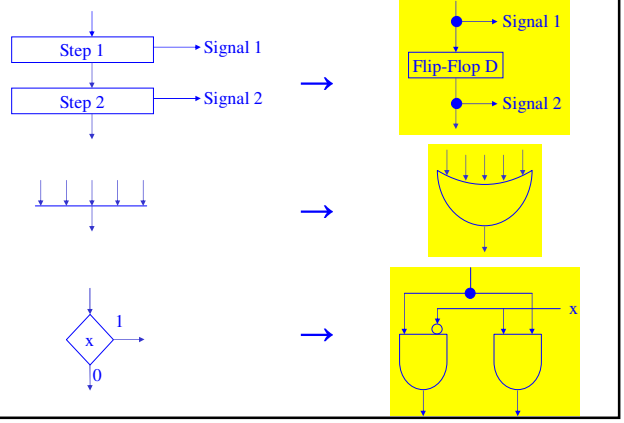


- Example with sequential multiplication

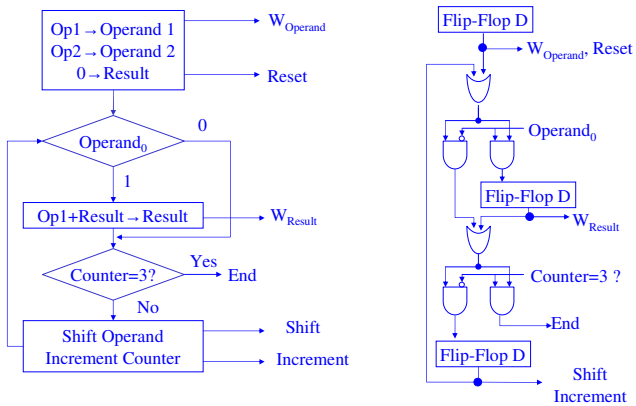
Multiplier Control



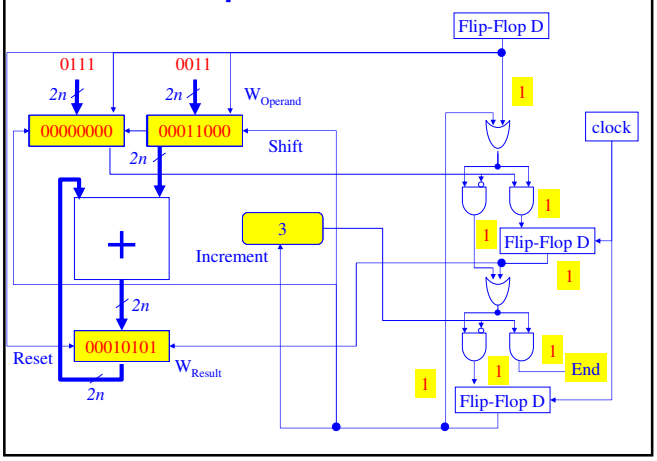
Converting a Flowchart into a Control Circuit



Multiplier Control Circuit

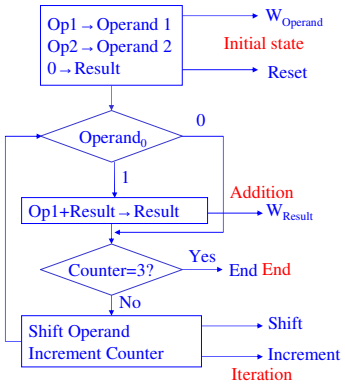


Multiplier Control Circuit



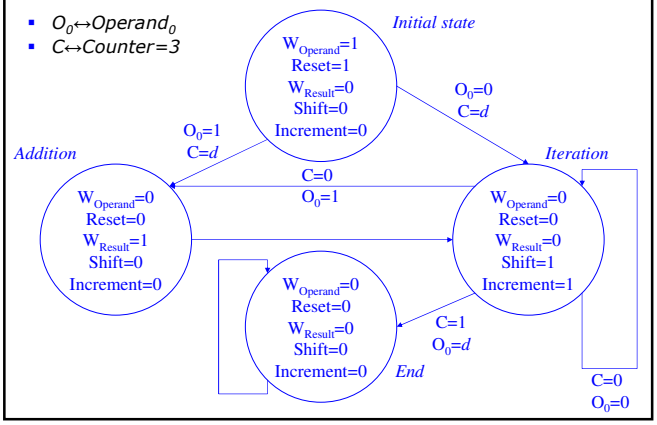
A Systematic Approach for Designing a Sequential Circuit

- Approach: convert a finite-state automaton into a circuit
- Multiplier:
  - states
  - Inputs: *Operand*$_0$ et *Counter=3*.
  - Outputs: $W_{Operand}$, Reset, W_{Result}, Shift, Increment, End.



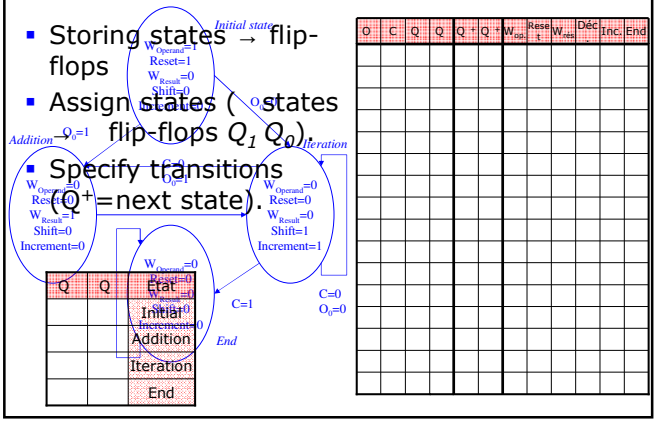
Sequential Circuit Design - Automaton

- $O_0 \leftrightarrow Operand_0$
- $C \leftrightarrow Counter=3$

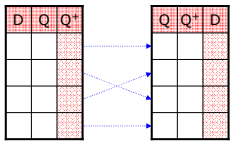


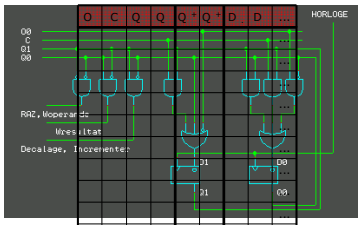
Sequential Circuit Design – Transition Table

- Storing states → flip-flops
- Assign states ($Q_1 Q_0$) to states
- Specify transitions (Q^+=next state).



Sequential Circuit Design – Final Circuit





- Using D flip-flops
- Set D_1, D_0 so as to impose desired transitions
- $Q^+=D$ → transition table = truth table of D_1, D_0.
- Outputs correspond to current states

$$D_1 = \overline{Q_1}.Q_0 + Q_1.\overline{Q_0} + C.Q_1$$

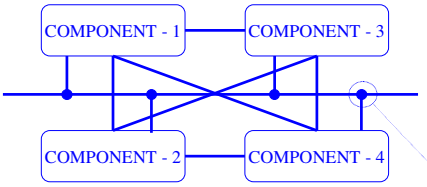
$$D_0 = Q_1.\overline{Q_0} + Q_1.Q_0 + C.\overline{Q_1}$$

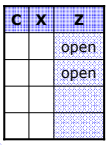
$$W_{operand} = \text{Reset} = \overline{Q_1}.\overline{Q_0}$$

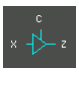
$$W_{res} = \overline{Q_1}.Q_0$$

$$Shift = Increment = Q_1.\overline{Q_0}$$

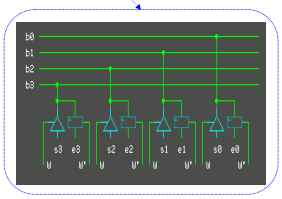
Data Paths - Bus



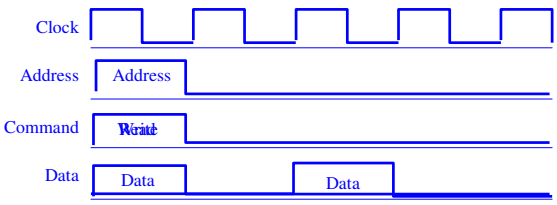




- Connecting a large number of components: **bus**.
- Bus =a set of *n* -bit wires
- Very cheap, but not efficient with large number of components



Bus



- Synchronous buses:
  - All components timed with same clock
  - Links indicate **address, data, command**
  - Internal buses or connection to memory
- Access protocol (priorities,…) implemented in control circuit