EEL3701: Digital Logic & Computer Systems

Menu

- Three classes of digital machines
- Three stages of digital design
- Logic Design
- Circuit Design

EEL3701: Digital Logic & Computer Systems

Classes of Digital Machines

- Three Classes of Digital Machines
  - >Combinational Circuits / Logic Circuits
  - >Sequential Logic
    - – Algorithmic State Machines (ASM)
  - >Microcomputers/Microprocessors
    - – Von-Neumann/Atanasoff Digital Computer Model

EEL3701: Digital Logic & Computer Systems

Combinatorial and Sequential Logic

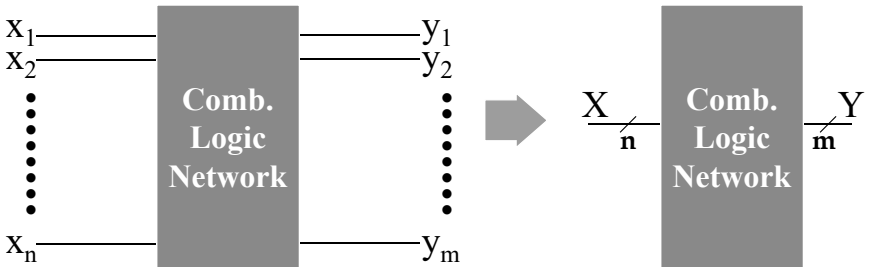
- Combinational Logic
  - >Machines that have zero memory
  - >Boolean Algebra & K-Maps
  - >Design of "glue" parts in larger digital applications
- Sequential Logic
  - >Finite Memory circuits
  - >Feedback & the concept of the ***state*** of a machine

EEL3701: Digital Logic & Computer Systems

Algorithmic State Machines

- Algorithmic State Machines (ASM or FSM)
  - >A ***modern*** approach to sequential logic design
  - >Has a ***programming flavor***, while allowing increased design complexity
- Microcomputers/Microprocessors
  - >Partially infinite memory
  - >General-purpose digital machines
  - >Architecture of a microprocessor
  - >Examples:
    - – Microchip'sAtmel AVR (EEL3923) and XMEGA (EEL3744)
    - – Microchip's PIC (EEL3923)
    - – Historic: 68HC11, 8051, TMS320F28335 (TI DSC)
    - – **G-CPU**

EEL3701: Digital Logic & Computer Systems
Model of **Combinational** Digital Machines

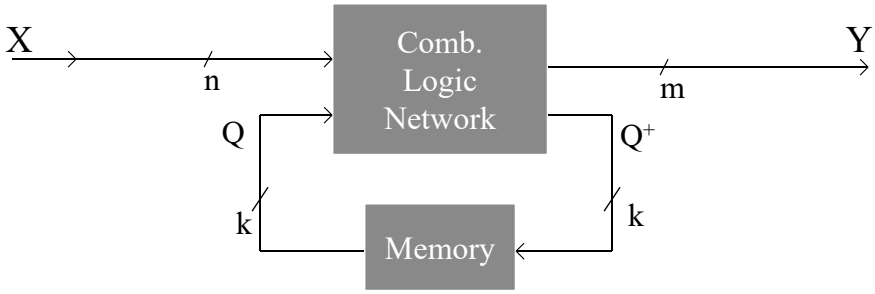


$X = [x_1, x_2, \ldots, x_n]^T$
$Y = [y_1, y_2, \ldots, y_m]^T$
$Y = F(X) = [f_1(X), f_2(X), \ldots, f_m(X)]^T$

Each output y_i can be computed if the inputs x_j are known.

$y_i(t) = f_m(X(t))$ {We often omit the (t) notation.}

EEL3701: Digital Logic & Computer Systems
Model of **Sequential** Digital Machines



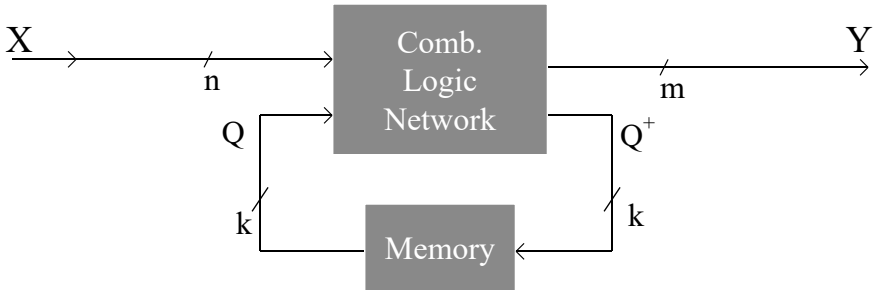
Each Q is called a **state** ≡ a summary of the past or historical behavior.

$Y = F(Q,X)$ There are m equations or m scalar functions.

$Q^+ = G(Q,X)$ There are k equations or k scalar functions.

EEL3701: Digital Logic & Computer Systems

Algorithmic State Machine (ASM) Design

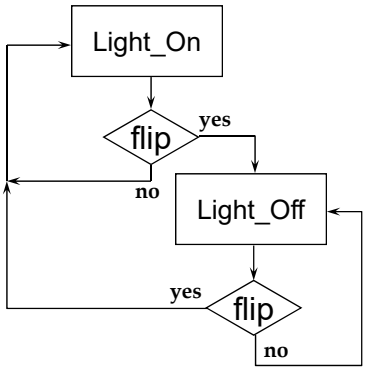


Example: Add 1 to the number I am thinking of ?

EEL3701: Digital Logic & Computer Systems

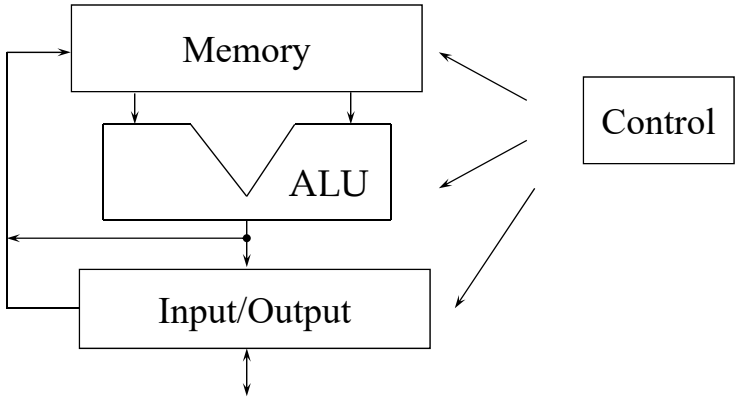
Algorithmic State Machine (ASM) Design

- The "modern model" for designing state machines
  - >It is about 37 year old (created in mid-1970's)
- Has a programming flavor
- ASM does not introduce a new class of machines



EEL3701: Digital Logic & Computer Systems

Von-Neumann/Atanasoff Digital Computer Model



EEL3701: Digital Logic & Computer Systems

Three Stages in the Design of Digital Systems

- System Design or System Specifications:
  - >Break the overall system into subsystems
  - >Specify the characteristics of each subsystem
- Example: Designing a digital computer involves specifying the number of bits per word, size of memory, buses, etc.

EEL3701: Digital Logic & Computer Systems

Logic Design

- **Logic Design:** How to interconnect the basic logic building blocks to perform specific functions
- Example: In building an arithmetic logic unit (ALU), you needs to specify the logic gates and flip-flops that will give the unit the capability to manipulate n-bit binary numbers

EEL3701: Digital Logic & Computer Systems

Circuit Design

- **Circuit Design:** How to interconnect specific components, e.g., ICs, resistors, switches, LEDs, etc.
- *The lectures of this course deal primarily with logic design.*
- *The laboratory deals with logic circuit design, circuit constructions and debugging, and implementation.*

EEL3701: Digital Logic & Computer Systems

The End!