Al-Mustaqbal University
College of Sciences
Intelligent Medical System Department

Embedded systems

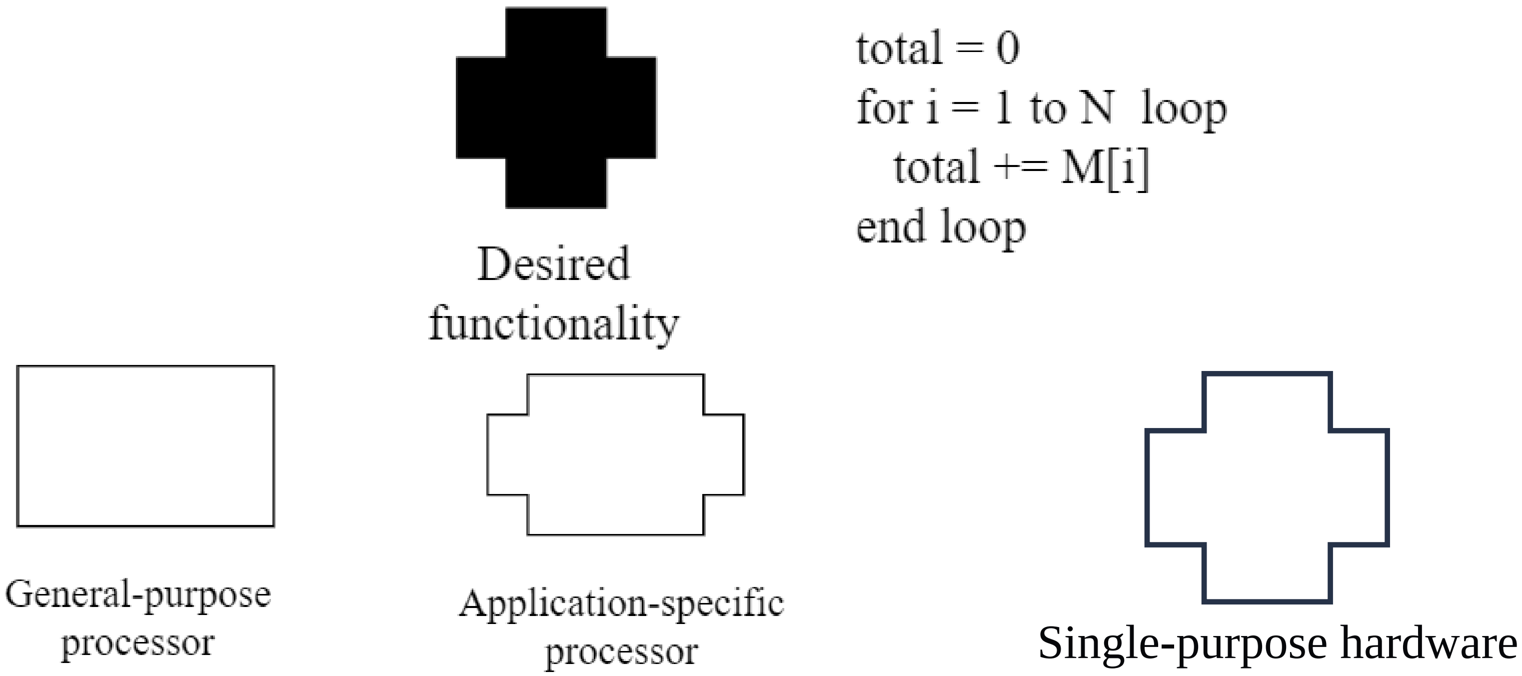
Lecture 2: Programmable Logic Devices

Prof.Dr. Mehdi Ebady Manaa

جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

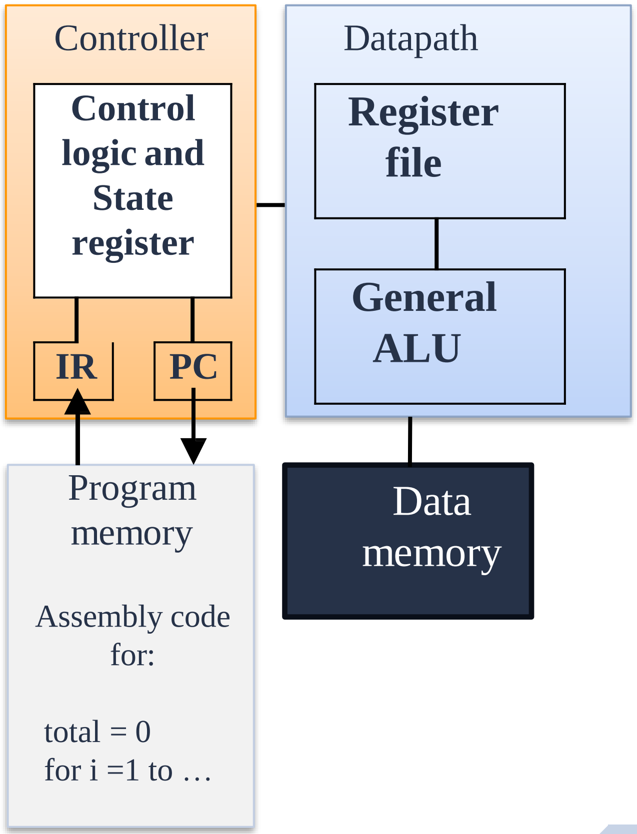
- Differ in their customization for the problem at hand

```
total = 0
for i = 1 to N  loop
   total += M[i]
end loop
```



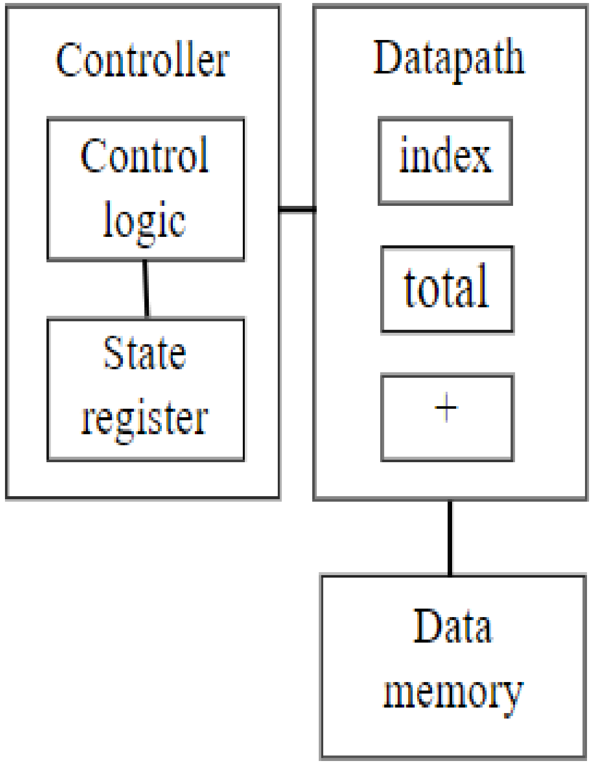
General-purpose processors

- Programmable device used in a variety of applications
  - Also known as “microprocessor”
- Features
  - Program memory
  - General datapath with large register file and general ALU
- User benefits
  - Low time-to-market and NRE costs
  - High flexibility
- Examples
  - Pentium, Athlon, PowerPC , ARM ,…



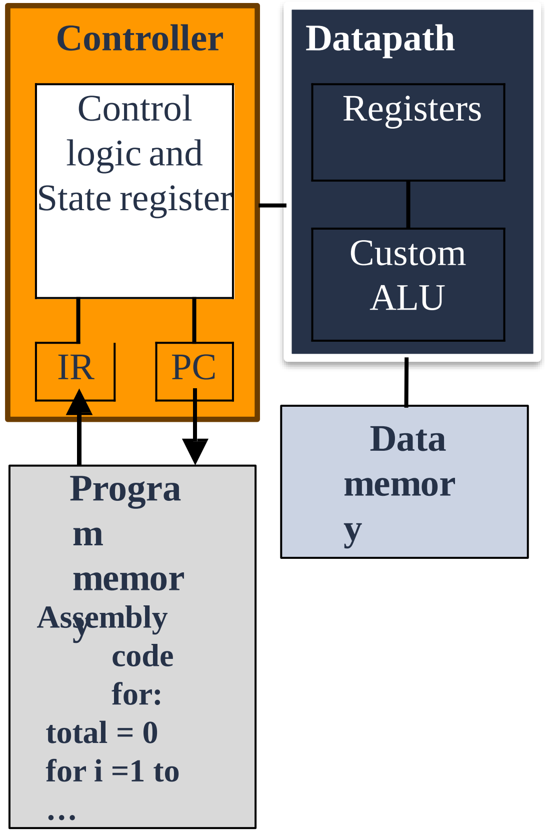
Application-Specific ICs (ASICs)

- Digital circuit designed to execute exactly one program
  - coprocessor, hardware accelerator
- Features
  - Contains only the components needed to execute a single program
  - No program memory
- Benefits
  - Fast
  - Low power
  - Small size



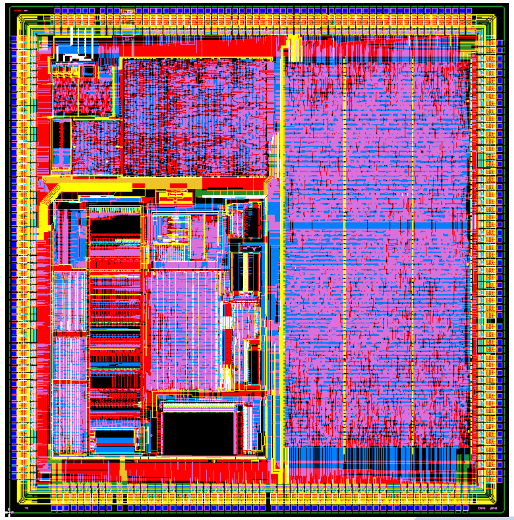
Application-specific IS processors (ASIPs)

- Programmable processor optimized for a particular class of applications having common characteristics
  - Compromise between general-purpose and ASIC (custom hardware)
- Features
  - Program memory
  - Optimized datapath
  - Special functional units
- Benefits
  - Some flexibility, good performance, size and power
- Examples
  - DSPs, Video Signal Processors, Network Processors,..

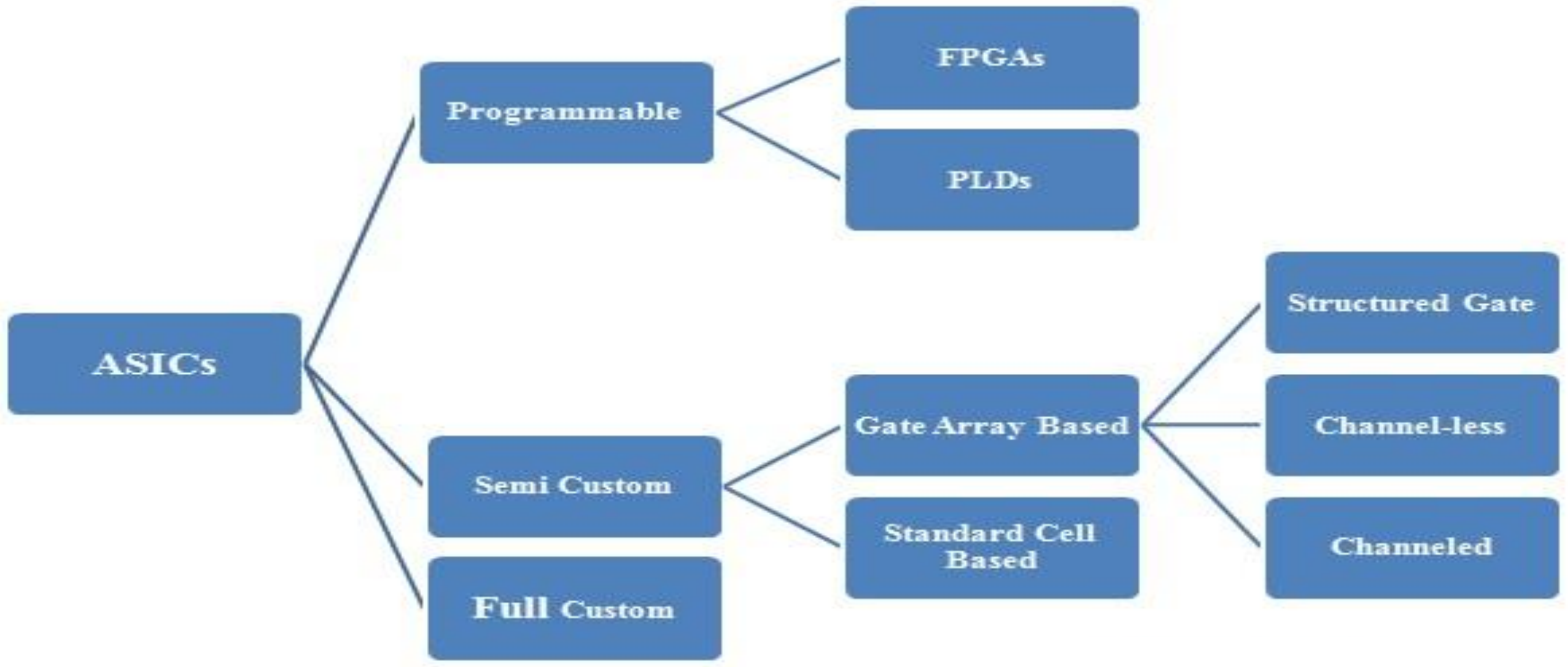


Application Specific Circuits (ASIC)

- Custom-designed circuits necessary if ultimate speed or energy efficiency is the goal and large numbers can be sold.
- Approach suffers from long design times and high costs.



Application Specific Circuits (ASIC)



Comparing ASIP with general processors and ASIC.

General Purpose Processors

- Such as x86, ARM
- High-end processors consume thousands of designer-years.
- Aim to MAX flexibility for all applications
- Compiler and OS must be designed for all applications, entry level is too high
- x86 price is high

ASIP

- Is designed for a domain of applications
- Its assembly instruction set is designed to accelerate most appearing function and critical functions.
- The hardware cost and power consumption are relatively much lower. The price can be very low under volume sales.
- It is usually for predictable computing

ASIC

- Non-programmable, usually can reach the lowest power and silicon cost for only one application.
- ASIC was a dominant solution when the level of integration was limited.
- Because of the high NRE cost, it will be gradually less popular

NRE and unit cost metrics

- Unit cost
  - the monetary cost of manufacturing each copy of the system, excluding NRE cost
- NRE cost (Non- Recurring Engineering cost)
  - The one-time monetary cost of designing the system
- total cost = NRE cost + unit cost * # of units
- per-product cost = total cost / # of units

  = (NRE cost / # of units) + unit cost 5

Storage

- What is a memory?
  - Artifact that stores bits
  - Storage fabric and access logic
- Write-ability
  - Manner and speed a memory can be written
- Storage-permanence
  - ability of memory to hold stored bits after they are written
- Many different types of memories
  - Flash, SRAM, DRAM, etc.
- Common to compose memories

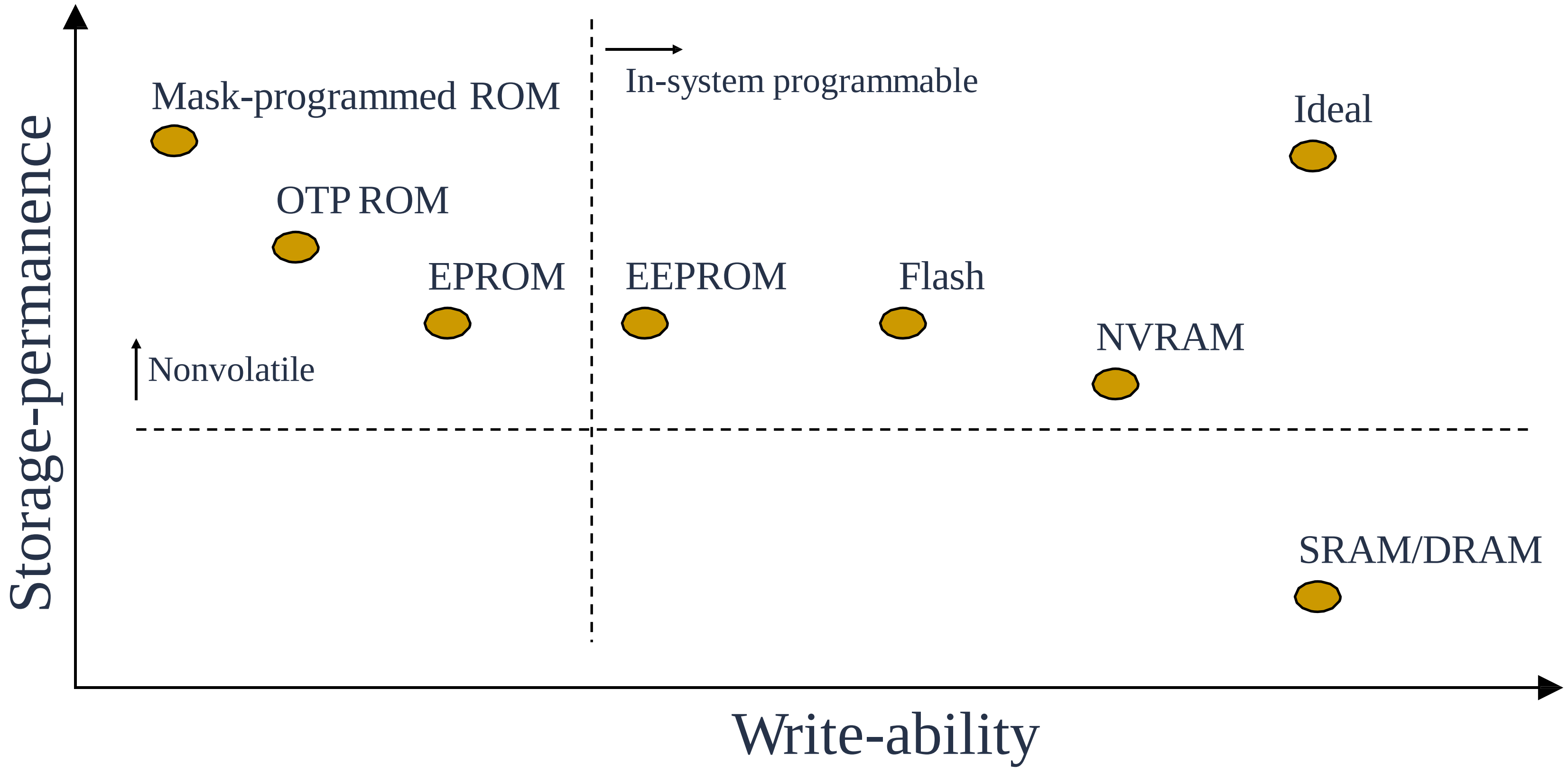
Write-ability

- Ranges of write ability
- High end
  - Processor writes to memory simply and quickly
  - E.g., RAM
- Middle range
  - Processor writes to memory, but slower
  - E.g., FLASH, EEPROM
- Lower range
  - Special equipment, “programmer”, must be used to write to memory
  - E.g., EPROM, OTP ROM
- Low end
  - Bits stored only during fabrication
  - E.g., Mask-programmed ROM

Storage-permanence

- Range of storage permanence
- High end
  - Essentially never loses bits
  - E.g., mask-programmed ROM
- Middle range
  - Holds bits days/months/years after memory's power source
  - turned off
  - E.g., NVRAM
- Lower range
  - Holds bits as long as power supplied to memory
  - E.g., SRAM
- Low end
  - Begins to lose bits almost immediately after written
  - E.g., DRAM

Memory Types



Communication

- What is a bus?
  - An artifact that transfers bits
  - Wires, air, or fiber and interface logic
- Associated with a bus, we have:
  - Connectivity scheme
    - Serial Communication
    - Parallel Communication
    - Wireless Communication
  - Protocol
    - Ports
    - Timing Diagrams
    - Read and write cycles
  - Arbitration scheme, error detection/correction, DMA, etc.

Serial Communication

- A single wire used for data transfer
- One or more additional wires used for control (but, some protocols may not use additional control wires)
- Higher throughput for long distance communication
  - Often across processing node
- Lower cost in terms of wires (cable)
- E.g., USB, Ethernet, RS232, I2C, etc.

Parallel Communication

- Multiple buses used for data transfer
- One or more additional wires used for control
- Higher throughput for short distance communication
  - Data misalignment problem
  - Often used within a processing node
- Higher cost in terms of wires (cable)
- E.g., ISA, AMBA, PCI, etc.

Wireless Communication

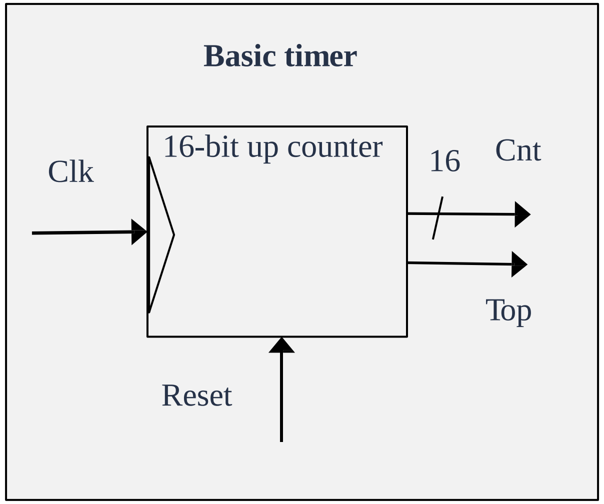
- Infrared (IR)
  - Electronic wave frequencies just below visible light spectrum
  - Diode emits infrared light to generate signal
  - Infrared transistor detects signal, conducts when exposed to infrared light
  - Cheap to build
  - Need line of sight, limited range
- Radio frequency (RF)
  - Electromagnetic wave frequencies in radio spectrum
  - Analog circuitry and antenna needed on both sides of transmission
  - Line of sight not needed, transmitter power determines range

Peripherals

- Perform specific computation task
- Custom single-purpose processors
  - Designed by us for a unique task
- Standard single-purpose processors
  - “Off-the-shelf”
  - pre-designed for a common task

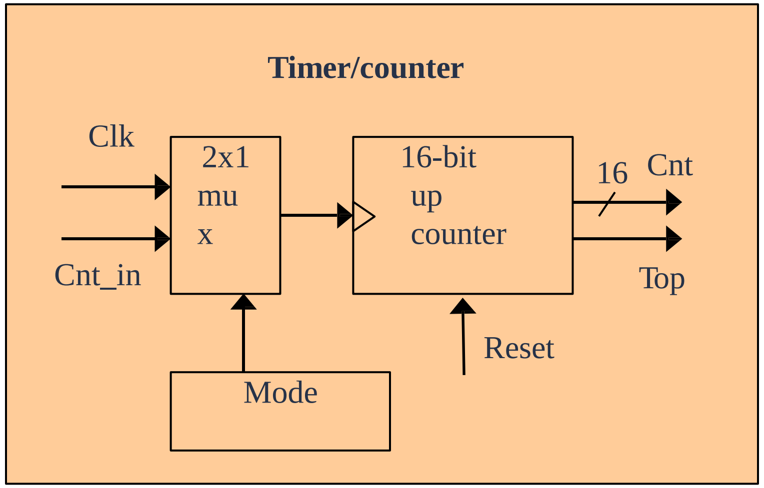
Timers

- Timers: measure time intervals
  - To generate timed output events
  - To measure input events
  - Top: max count reached
- Range and resolution



Counters and Watchdog Timer

- **Counter**: like a timer, but counts pulses on a general input signal rather than clock
  - e.g., count cars passing over a sensor
  - Can often configure device as either a timer or counter



- **Watchdog Timer**
- Must reset timer every X time unit, else timer generates a signal
- Common use: detect failure, self-reset

Watchdog Timer

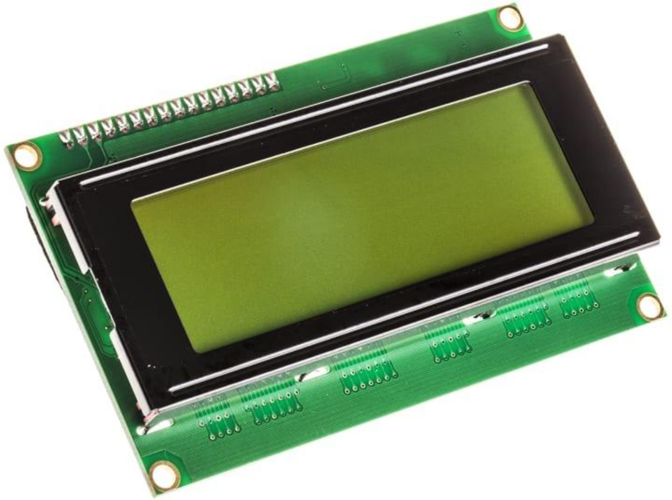
- UART: Universal Asynchronous Receiver Transmitter
  - Takes parallel data and transmits serially
  - Receives serial data and converts to parallel
- Parity: extra bit for simple error checking
- Start bit, stop bit
- Baud rate
  - Signal changes per second
  - Bit rate, sometimes different

Pulse Width Modulator (PWM)

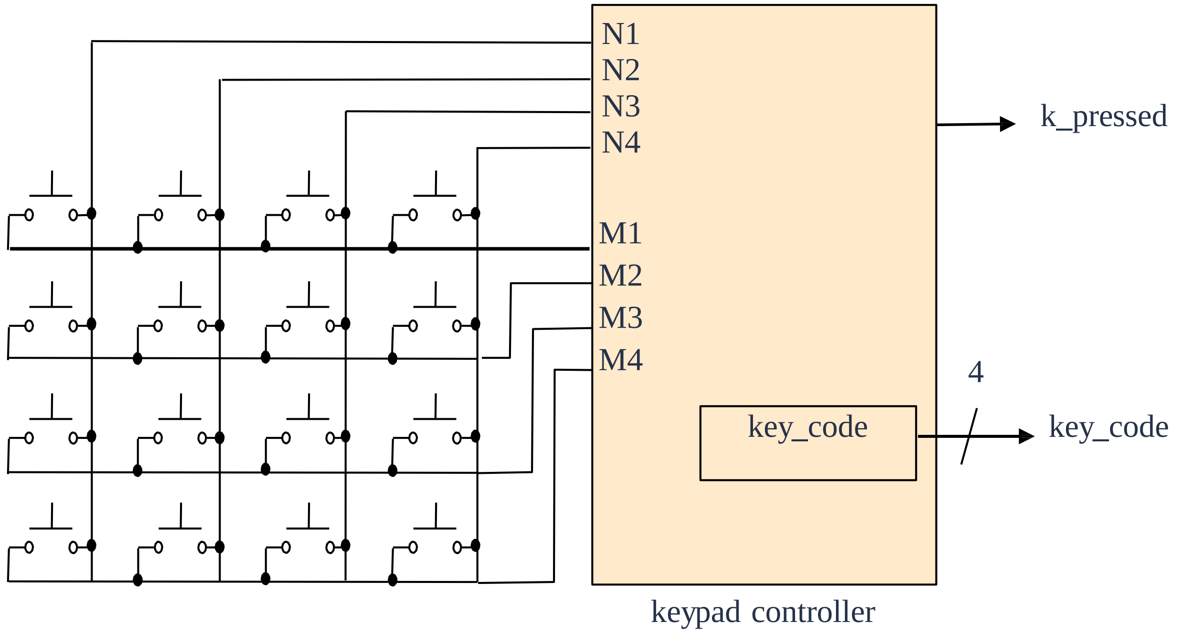
- Generates pulses with specific high/low times
- Duty cycle: % time high
  - Square wave: 50% duty cycle
- Common use: control average voltage to electric device
  - Simpler than DC-DC converter or digital- analog converter
  - DC motor speed, dimmer lights

LCD

- Liquid Crystal Display
- N rows by M columns
- Controller build into the LCD module
- Simple microprocessor interface using ports
- Software controlled



Keypad



N1
N2
N3
N4
M1
M2
M3
M4
k_pressed
4
key_code
key_code
keypad controller

N=4, M=4

Stepper Motor Controller

- Stepper motor: rotates fixed number of degrees when given a “step” signal
  - In contrast, DC motor just rotates when power applied, coasts to stop
- Rotation achieved by applying specific voltage sequence to coils
- Controller greatly simplifies this

Thank You