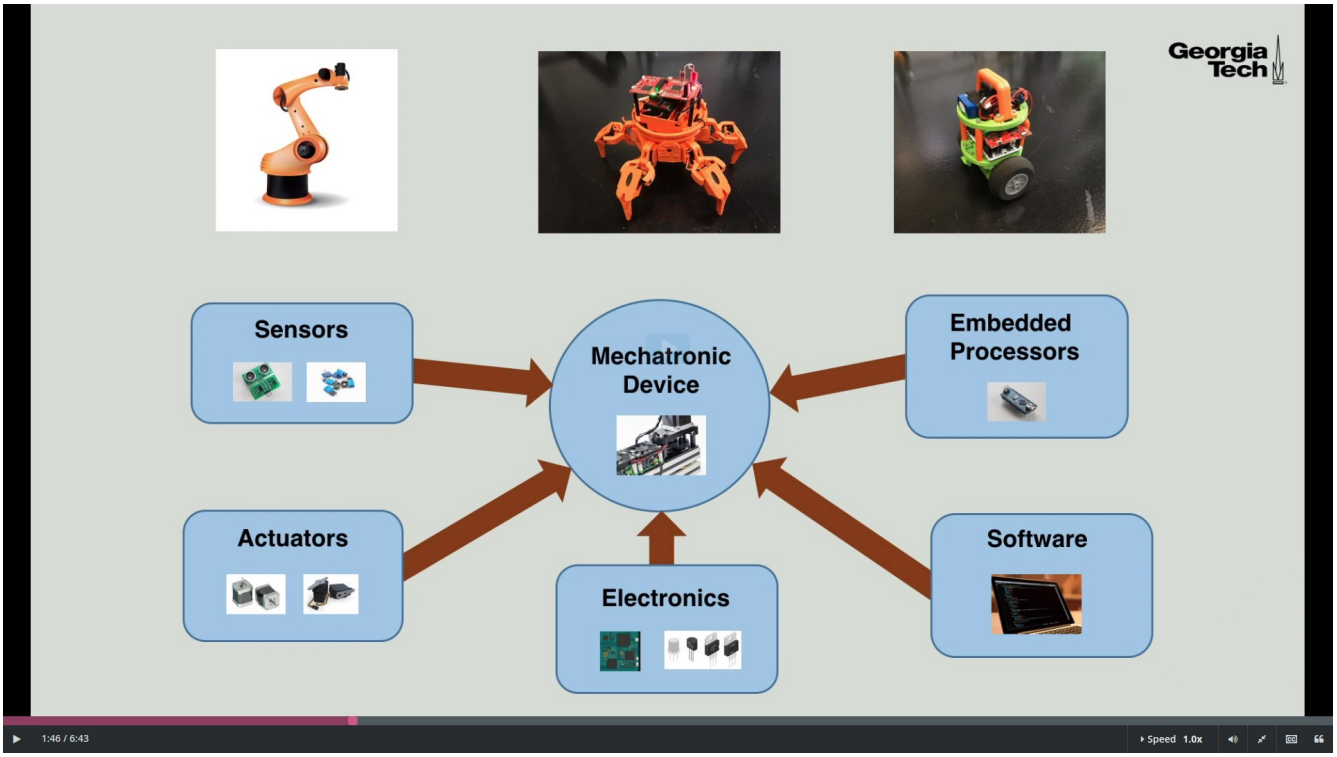
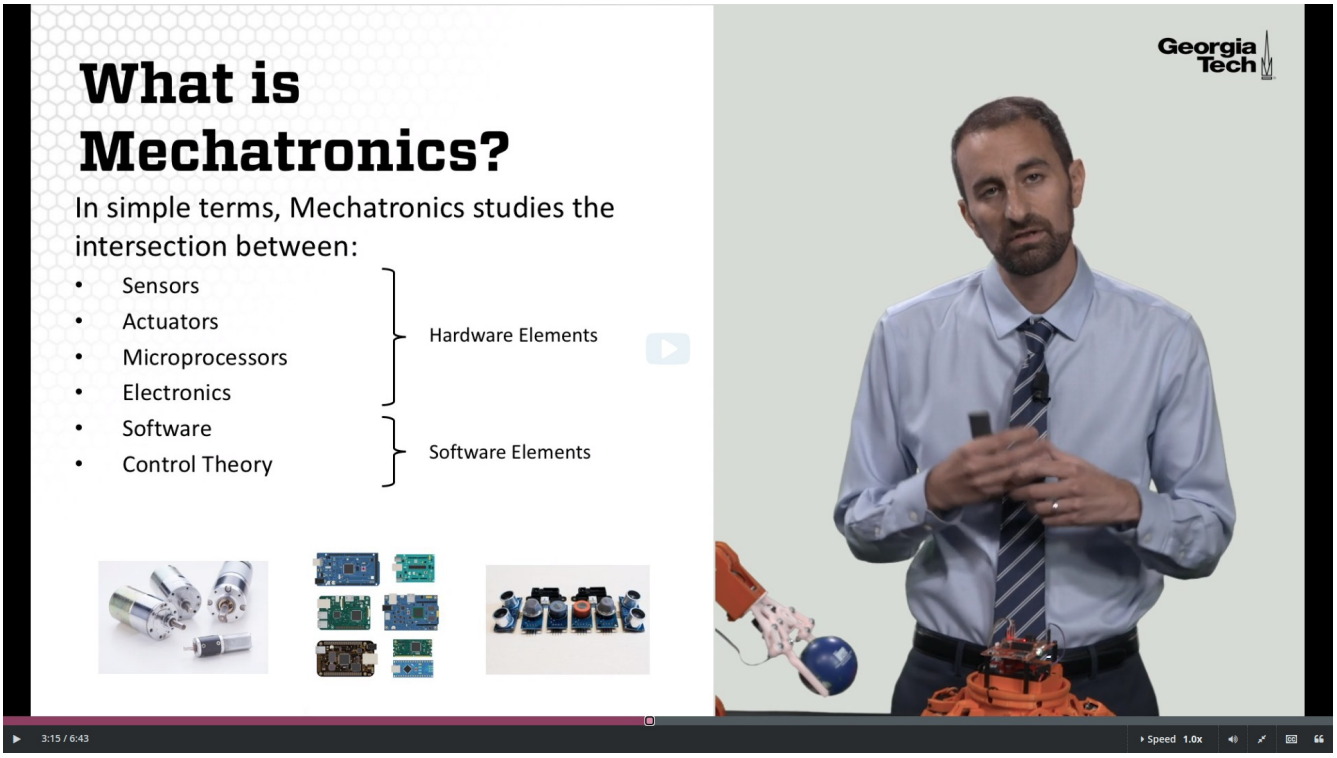
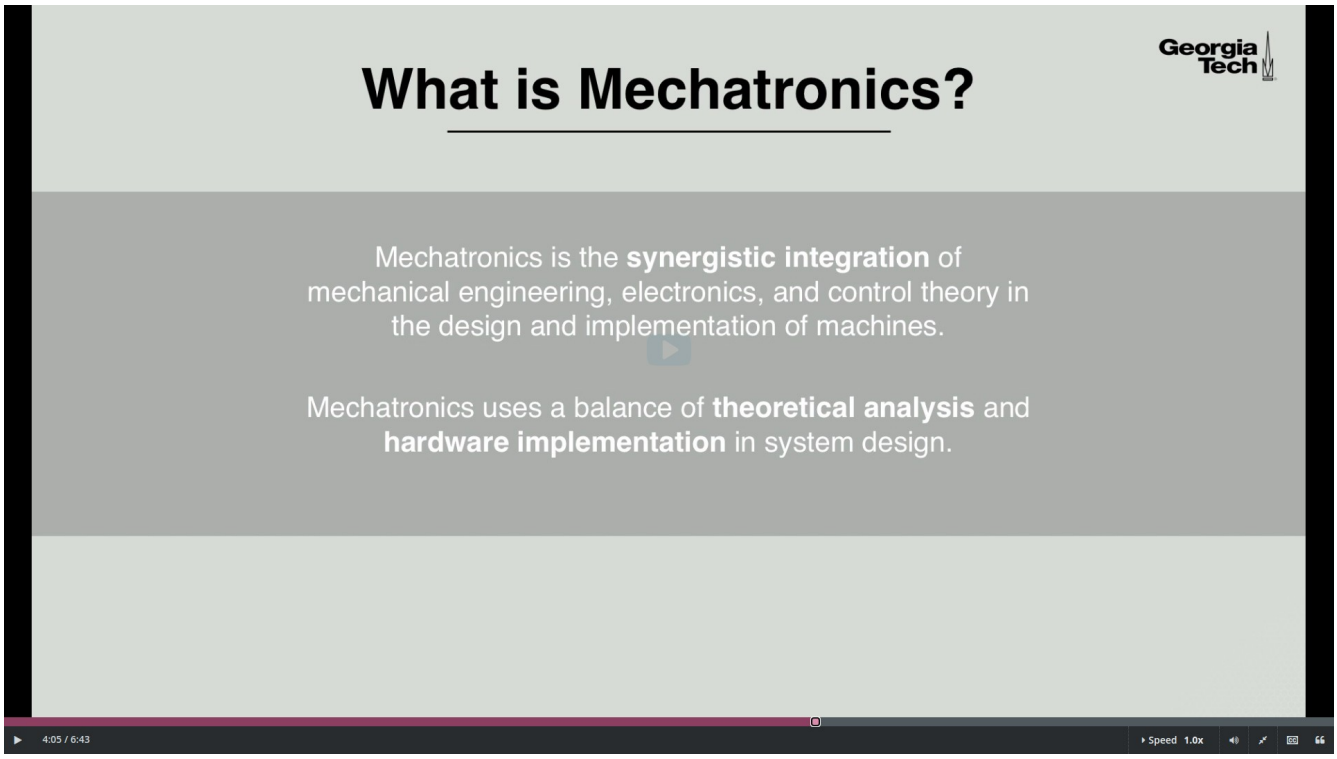
Topic 1 – What is Mechatronics?

Lesson 1: Mechatronics as the Interface of Actuators, Sensors, and Computers

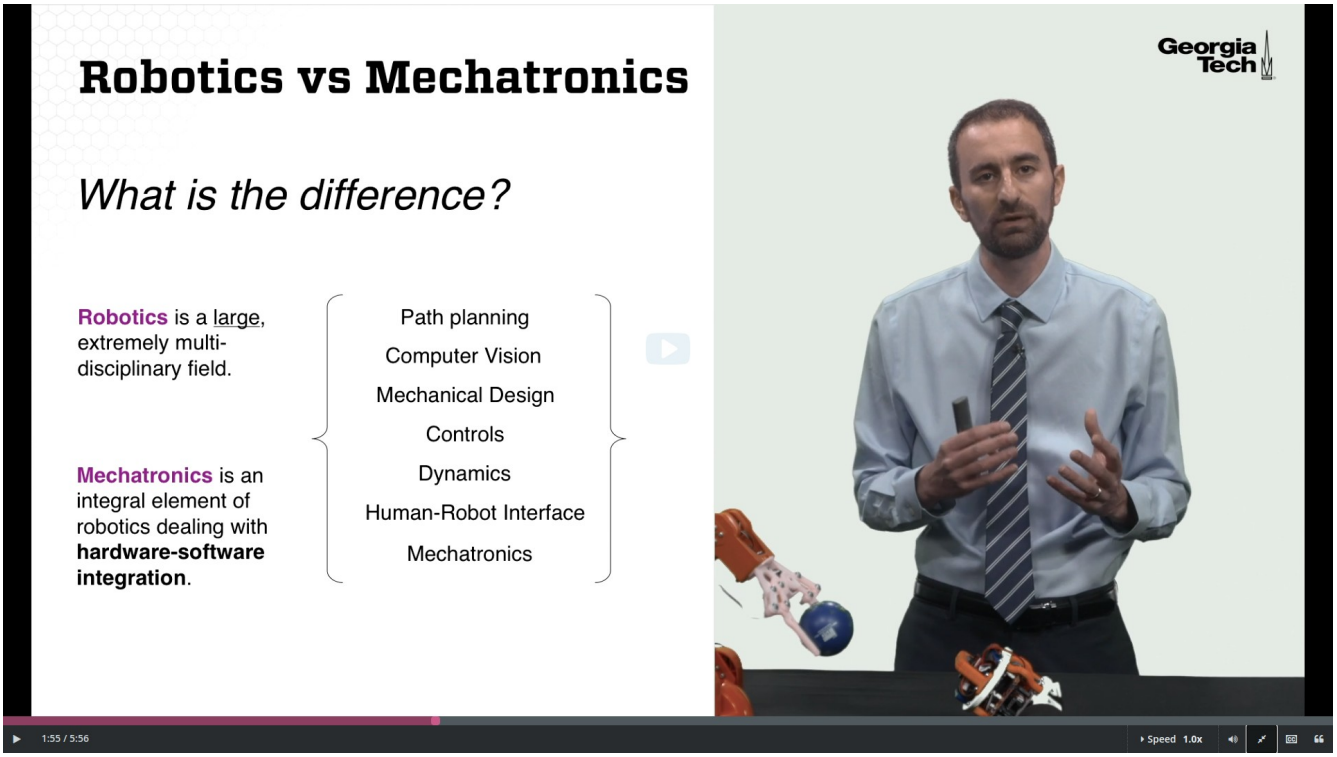




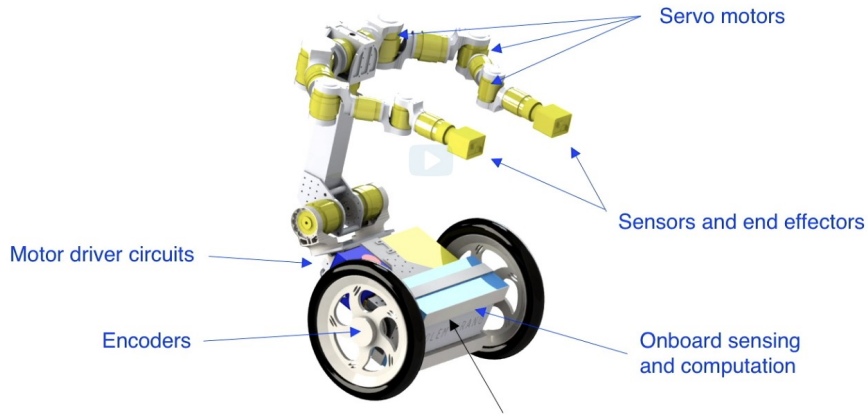


Kl

Lesson 2: The Role of Mechatronics in Robotics



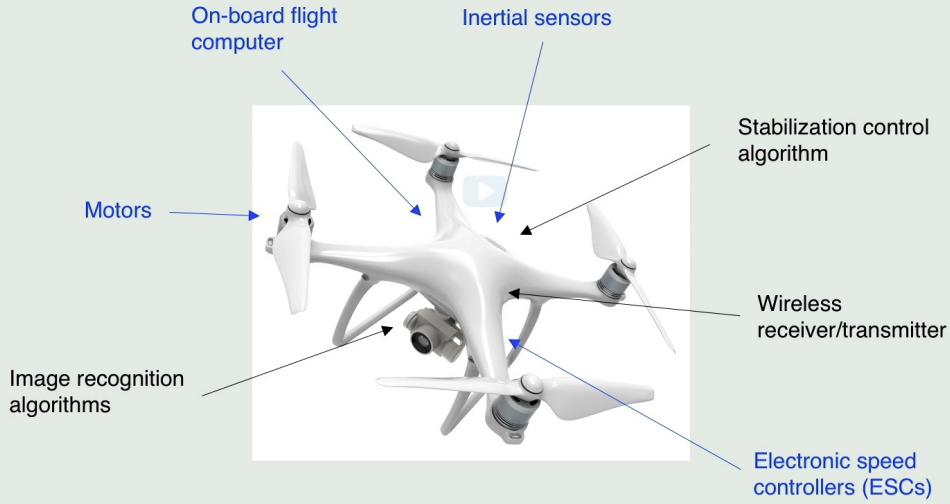
Georgia Tech Krang Robot



Mechatronics elements

2:25 / 5:56 Speed 1.0x

Quadrotor Drone



Mechatronics elements

3:15 / 5:56 Speed 1.0x

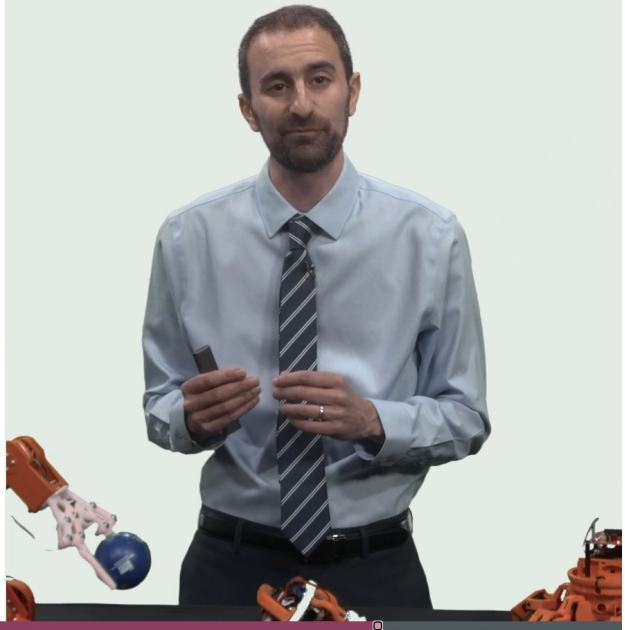
Agricultural Robot

Mechatronics elements

3:42 / 5:56 Speed 1.0x

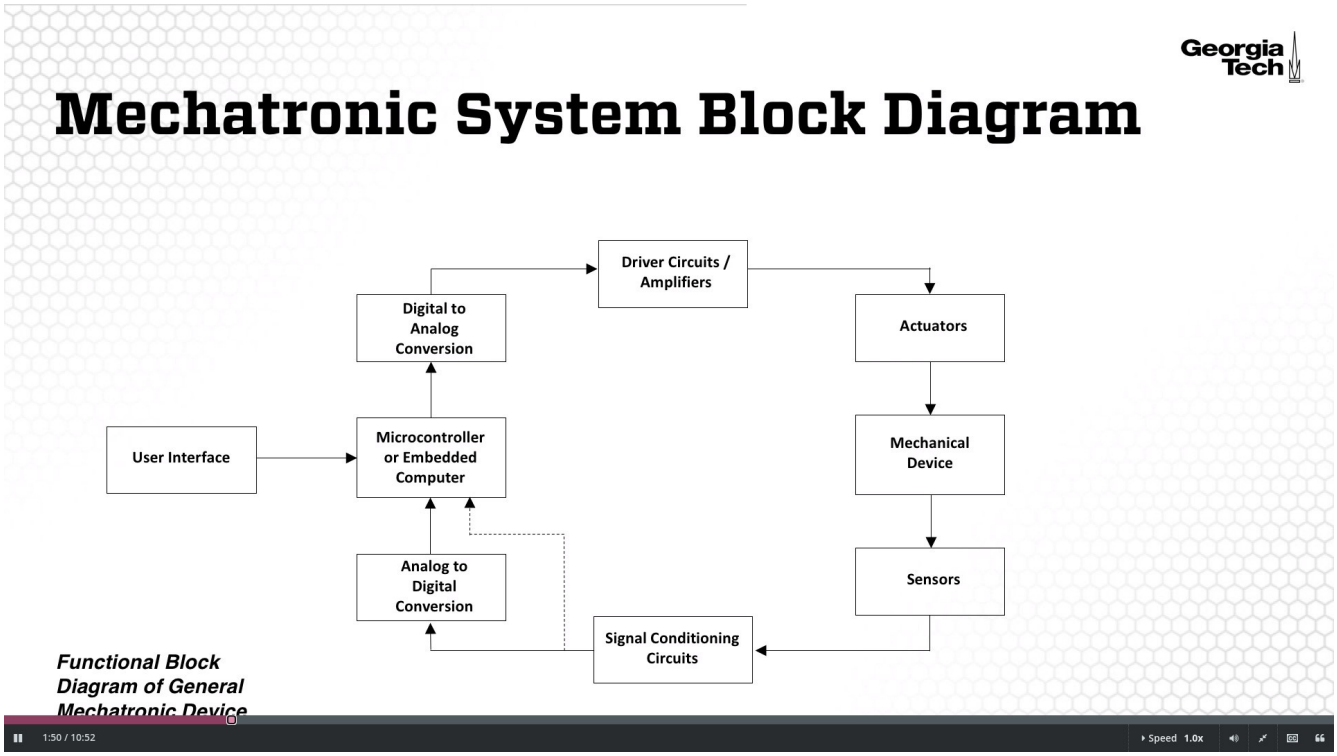
Mechatronics "Without" Robotics

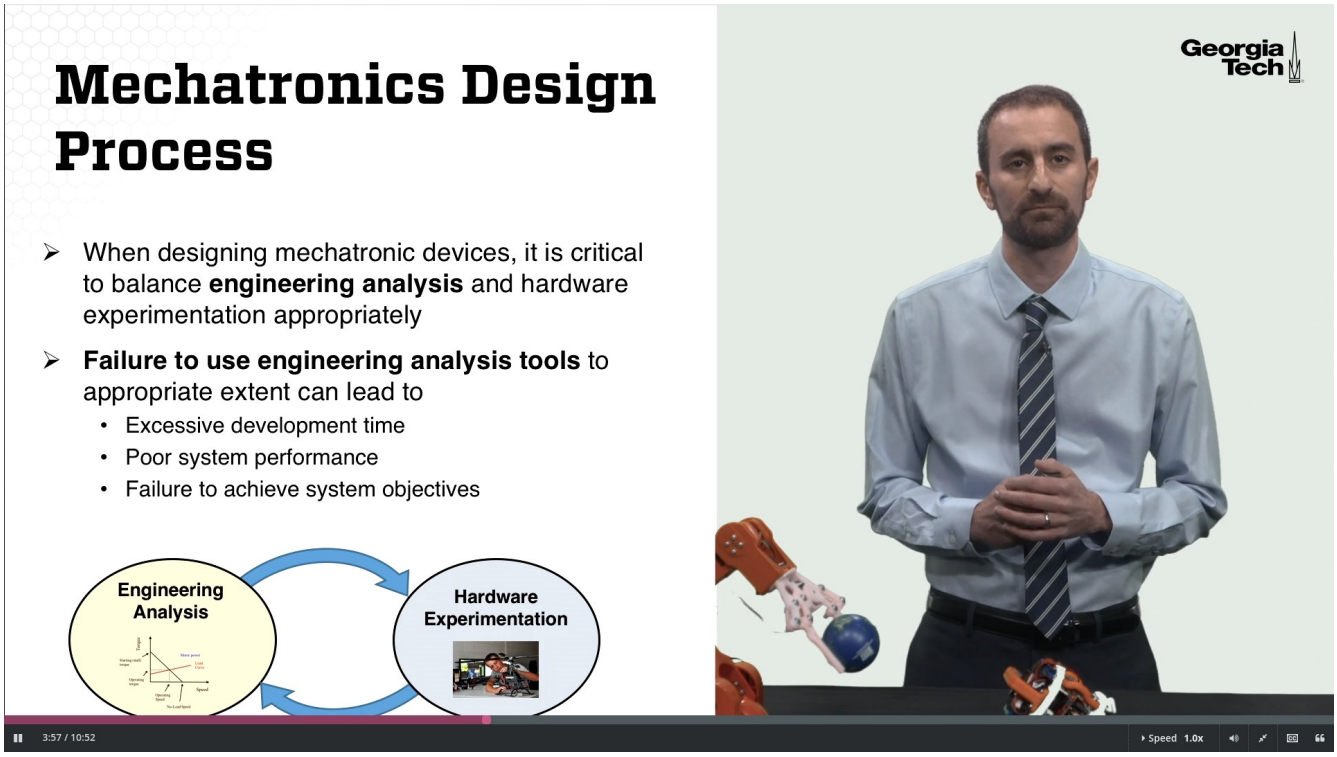
- While all robots are mechatronic devices in some form, *not all mechatronic devices are* robots.
- Many mechatronic devices lack the **intelligence**, **self-initiative**, **decision-making,** and **adaptability** inherent in many robotic devices.



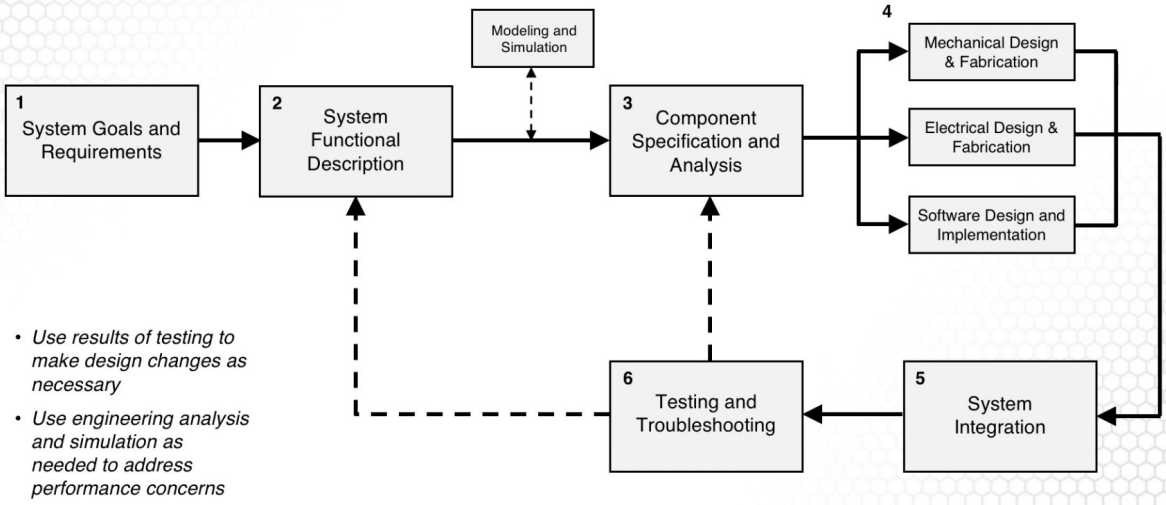
4:52 / 5:56 Speed 1.0x

Lesson 3: The Mechatronics Design Process





Mechatronics Design Process

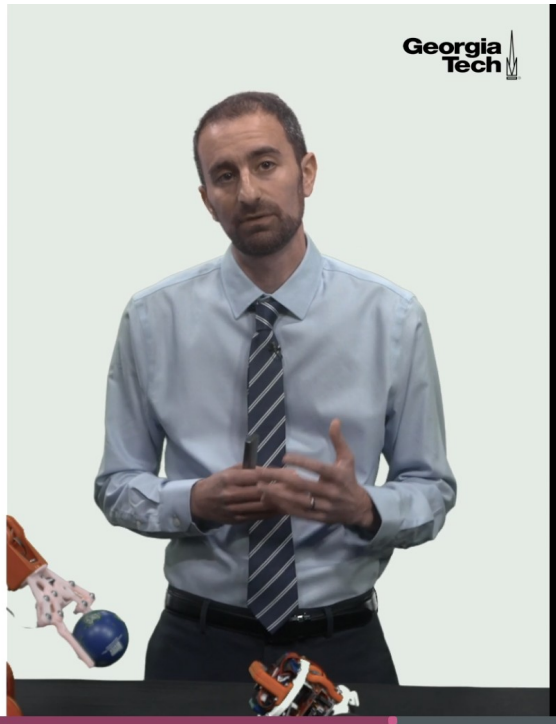


- *Use results of testing to make design changes as necessary*
- *Use engineering analysis and simulation as needed to address performance concerns*

9:08 / 10:52 Speed 1.0x

Mechatronics Design Process

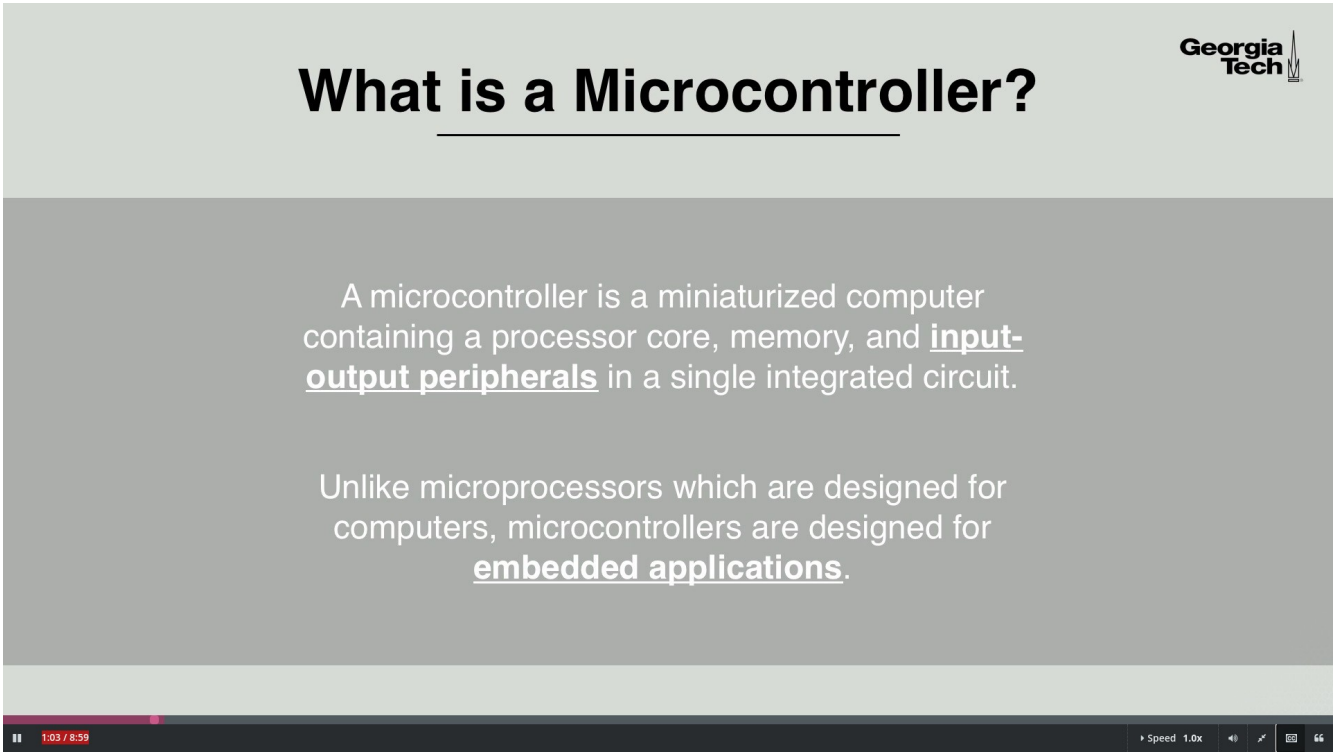
- **Mechatronics design process is important to follow**, even for simple systems
- Can save time, money, and effort
- "Trial and error" methods used often by beginners may work for simple systems but will be **unacceptably costly** when developing complex devices

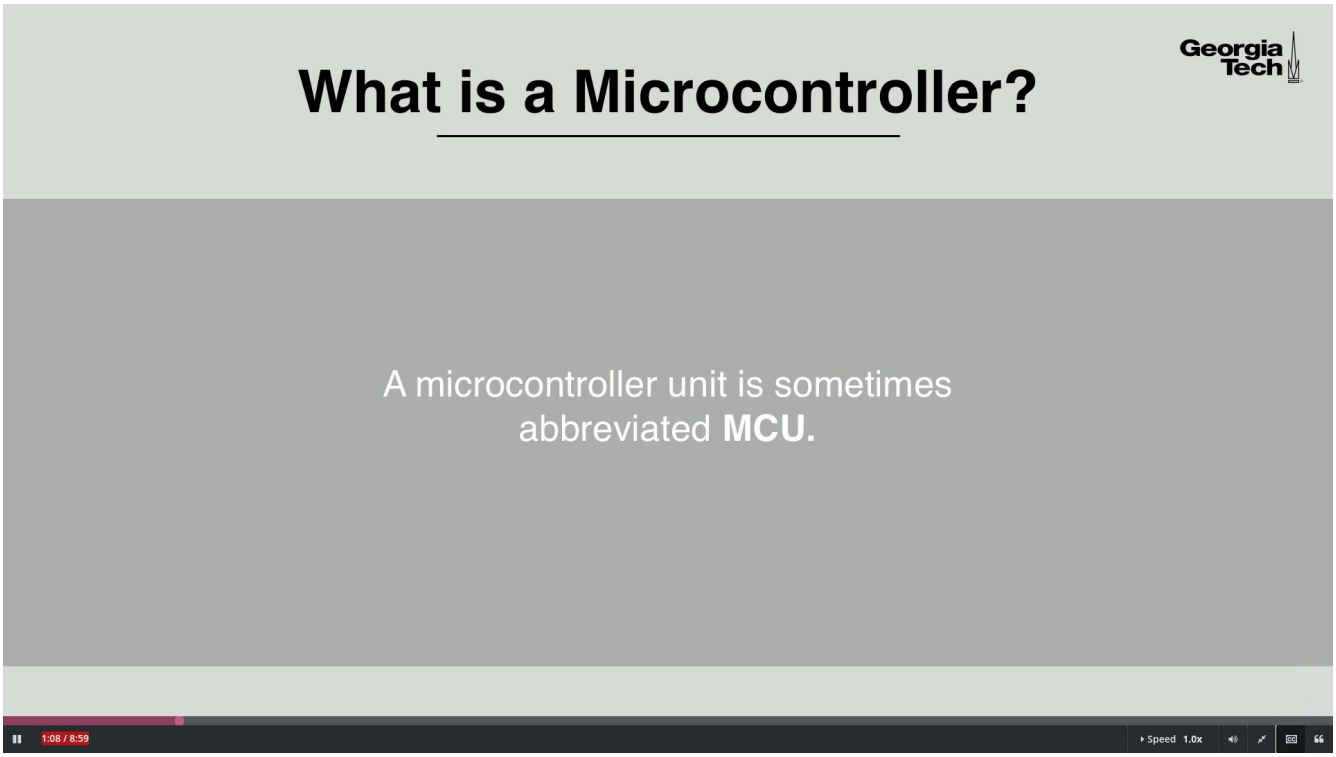


9:22 / 10:52 Speed 1.0x

Topic 2 – Introduction to Microcontroller Technology

Lesson 1: **Microcontrollers, Microprocessors, and Embedded Computers**





What are Embedded Applications?

Embedded Applications Use **Microcontrollers**

Computers Use **Microprocessors**

Microcontrollers vs Microprocessors

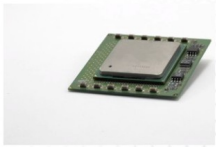
Microcontrollers

- Includes input and output devices (serial communications, analog-to-digital converters, GPIO digital logic pins, etc.)
- Slower clock rate
- Smaller memory (kB's)
- A lot cheaper (~$10's)
- Does not run a classical operating system
- Low power

Courtesy of Texas Instruments

Microprocessors

- Input-output limited to standard buses for memory and peripherals
- Very high clock rates (low latency operations)
- Large memory spaces (GB's)
- More expensive (~$100's)
- Runs operating system
- Higher power



Microcontrollers vs Embedded Computers

Georgia Tech

Embedded computers are “middle ground” - part microcontroller, part microprocessor

Embedded computers have some of the same input/output peripherals as a microcontroller, but with a **faster processor** and **more memory**

Embedded computers are designed for real-time applications where more **advanced software/processing** is needed (such as computer vision). They run an **operating system**.

Embedded computers are usually **more expensive** than microcontrollers

Raspberry Pi single-board computer

Odroid XU4 single-board computer

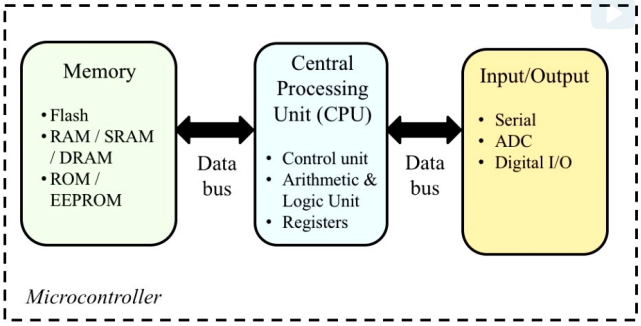
*Embedded computers are also called **single-board computers.***

4:03 / 9:00 Speed 1.0x

Elements of a Microcontroller

Georgia Tech

Microcontrollers are integrated circuits that incorporate memory and input/output components on **a single chip**.



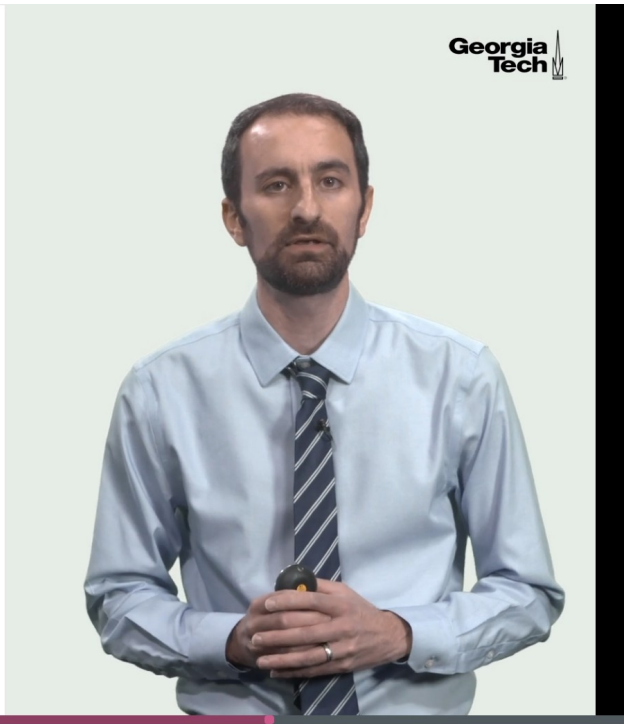
5:40 / 9:00 Speed 1.0x

Elements of a Microcontroller

Microcontrollers have a **central processing unit** (CPU) that executes program instructions

CPU's contain three main elements:

- **Control Unit:** Generates timing signal used to fetch program instruction from memory and execute it
- **Arithmetic and Logic Unit:** Performs actual instruction on data (e.g., addition of two numbers)
- **Registers:** Memory locations inside CPU that hold internal data while instructions are being executed



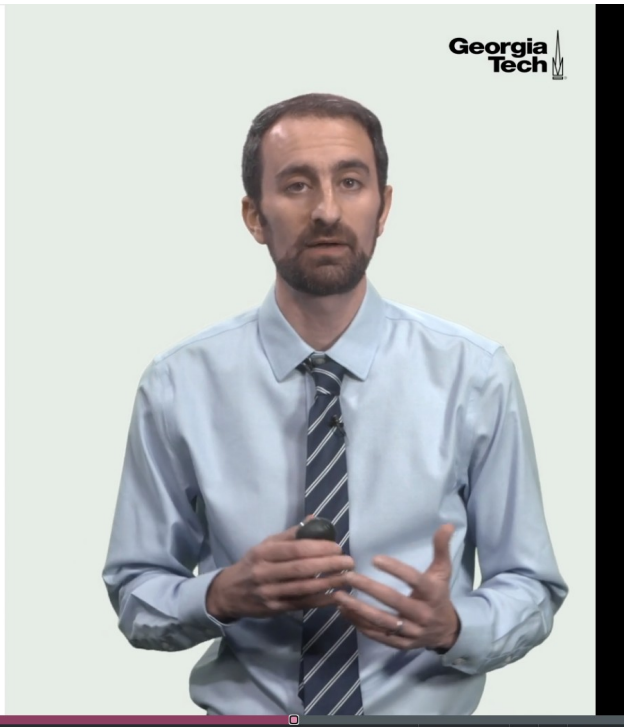
6:35 / 9:00 Speed 1.0x

Elements of a Microcontroller

Microcontrollers have both **volatile** and **non-volatile** memory onboard.

Volatile Memory: Data is maintained in memory as long as power is applied. If power is removed, data will be lost.

Non-Volatile Memory: Data is maintained even when power is removed.



6:45 / 9:00 Speed 1.0x

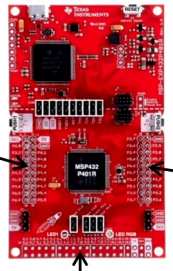
Microcontroller Memory Types

Memory Type	Description	
ROM	Read-Only Memory	Nonvolatile memory that is programmed during manufacture of chip. Data can be read but cannot be written during use. Used for fixed programs (pre-installed libraries, pre-installed programs, etc).
PROM	Programmable ROM	Same as ROM but can be programmed once by user with no further changes allowed.
EPROM	Erasable PROM	Can be programmed more than once. Contents can be erased by shining UV light through window on top of device.
EEPROM or Flash memory	Electrically-Erasable PROM	Similar to EPROM, but contents can be erased by applying high-voltage signal rather than UV light.
RAM	Random Access Memory	Volatile memory that requires power to operate. Access time for data is constant and is not dependent on physical location of data.
DRAM	Dynamic RAM	RAM that uses capacitors to store data. Data must be refreshed periodically (rewritten) due to charge leakage.
SRAM	Static RAM	RAM in which data does not need to be refreshed as long as power is applied. Faster than DRAM but more expensive.

7:15 / 9:00 Speed 1.0x

Elements of a Microcontroller

Microcontrollers have general purpose input-output (GPIO) pins that enable interfacing with onboard peripherals.



GPIO header

GPIO header

GPIO header

Courtesy of Texas Instruments

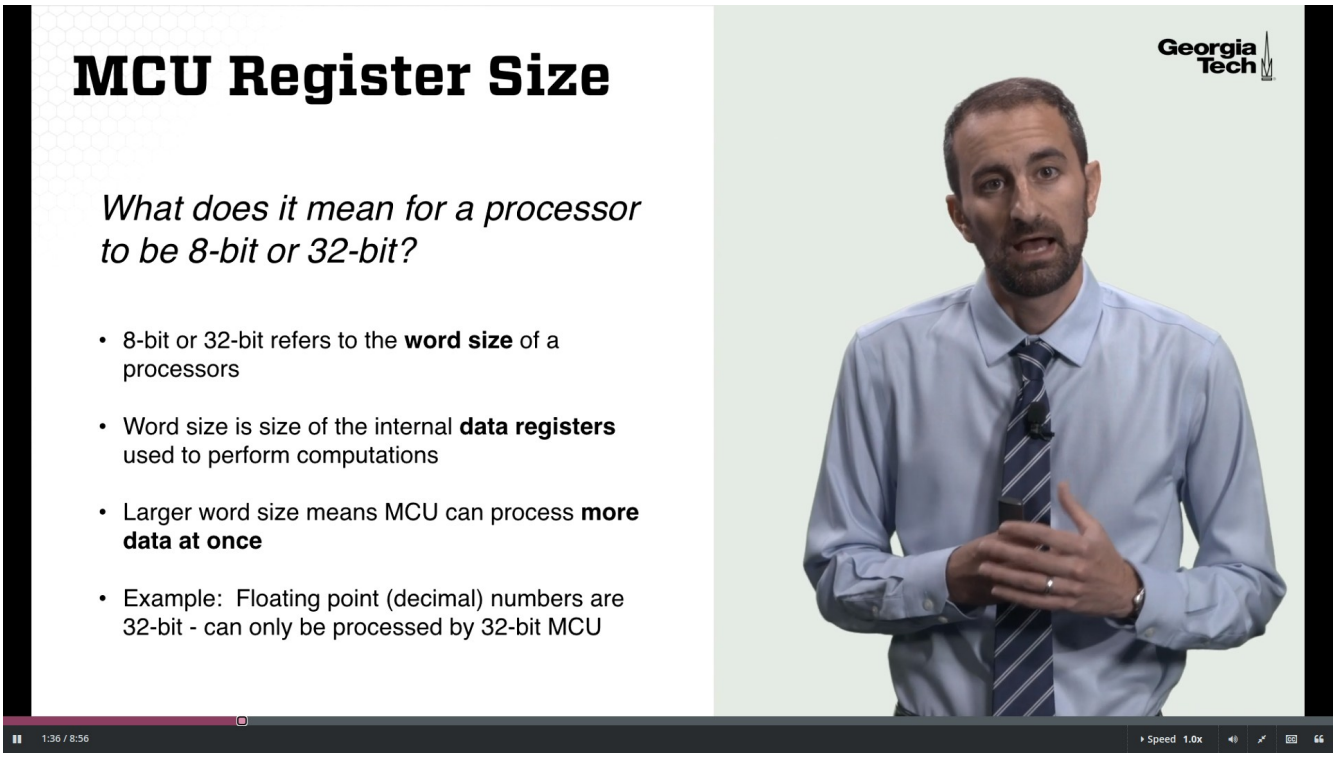
Courtesy of Texas Instruments

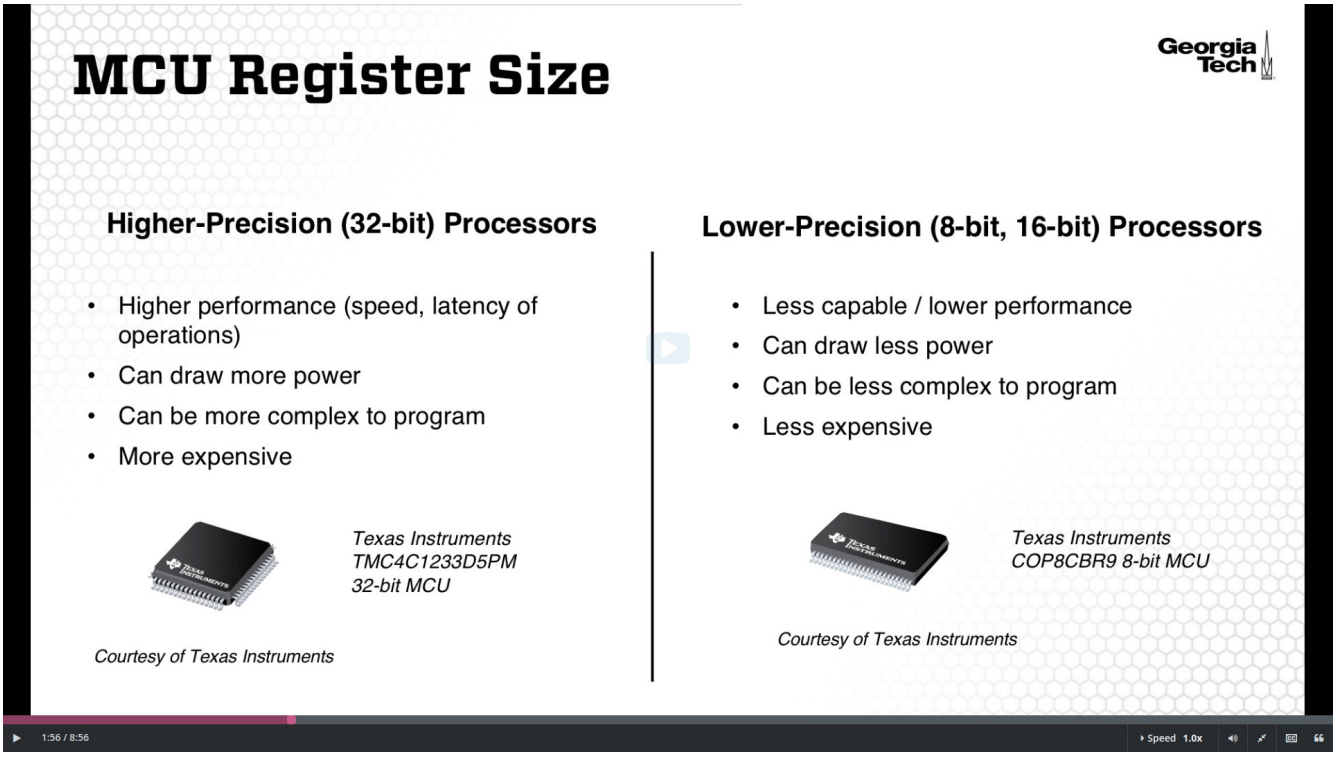
GPIO pins are a unique feature of MCUs which allow data to be sent between the processor and external devices

We will discuss GPIO in depth in the coming lessons.

8:05 / 9:00 Speed 1.0x

Lesson 2: Microcontroller Specifications





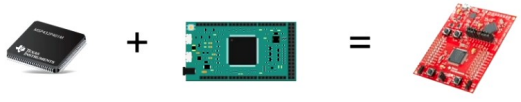
MCU Register Size

- A variety of microcontroller market analyses over the past decade have shown that **8-bit** MCUs capture roughly **40%** of the market share, **32-bit** MCUs capture **40%**, and **16-bit** MCUs capture **20%.**
- **Why are 8-bit MCUs equally as popular as 32-bit MCUs?**
- Many embedded applications are simple and require minimal computation. **Cost and power** considerations mean 8-bit MCU is more desirable.

MCU Development Kits

- Microcontroller is just a processor that looks like an integrated circuit with dozens of pins
- To use an MCU for prototyping, it needs to be paired with a breakout board

Microcontroller + Breakout Board = **Development Kit**

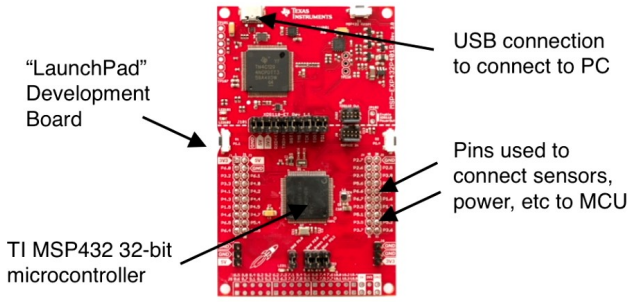


Courtesy of Texas Instruments

Texas Instruments' line of development kits are called ***LaunchPads.***

MCU Development Kits

When incorporated into a development kit, MCU becomes very useful for prototyping and experimentation.



Courtesy of Texas Instruments

5:57 / 8:56 Speed 1.0x

Example MCU Development Kit Specifications

Arduino Due

Manufacturer	Arduino (board) / Atmel (MCU)
Microcontroller	AT91SAM3X8E
Architecture	ARM 32Bit Cortex-M3
Speed	84 Mhz Max
SRAM	96 KBytes
Flash	512 KBytes
Cost	$36-$50
Software Development	ATmel Studio 6 (free)

6:18 / 8:56 Speed 1.0x

Example MCU Development Kit Specifications

STMicro STM32F4 Discovery

Manufacturer	STMicro (board + MCU)
Microcontroller	STM32F407
Architecture	ARM 32Bit Cortex-M4
Speed	180 Mhz Max
SRAM	192 KBytes
Flash	1 MByte
Cost	$15
Software Development	Keil

6:41 / 8:56 Speed 1.0x

Example MCU Development Kit Specifications

Clicker 2 PIC18FJ Development Kit

Manufacturer	MicroElektronica (board), Microchip (MCU)
Microcontroller	PIC18FJ
Architecture	8-bit PIC
Speed	48 Mhz
SRAM	3 KBytes
Flash	128 KBytes
Cost	$40
Software Development	MPLAB IDE

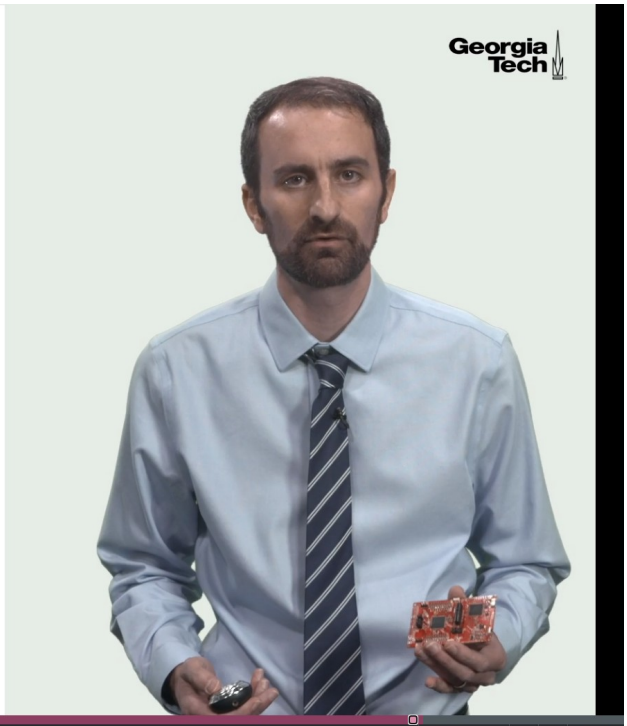
7:06 / 8:56 Speed 1.0x

Example MCU Development Kit Specifications

TI MSP432 LaunchPad

Courtesy of Texas Instruments

Manufacturer	TI (board + MCU)
Microcontroller	MSP432P401R
Architecture	ARM 32Bit Cortex-M4F
Speed	48 Mhz Max
SRAM	64 KBytes
Flash	256 KBytes
Cost	$13
Software Development	Code Composer Studio (CCS)

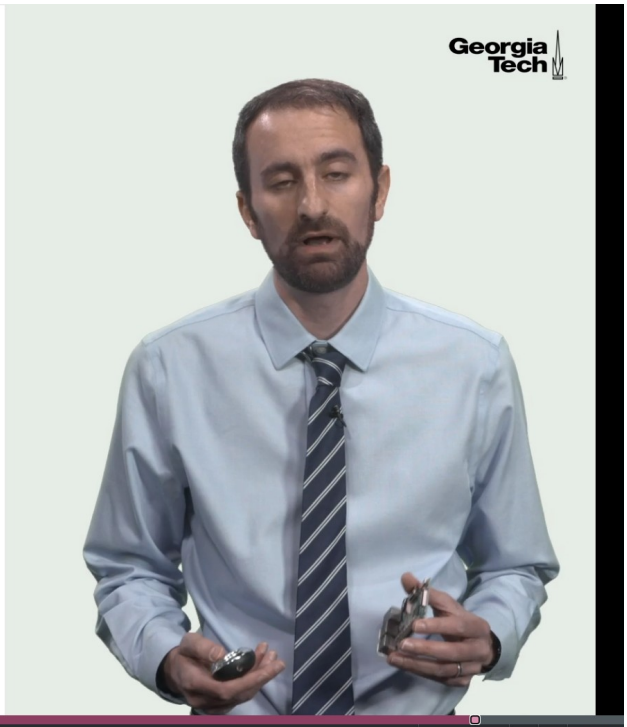


7:33 / 8:56 Speed 1.0x

Example MCU Development Kit Specifications

Raspberry Pi 3 Model B

Manufacturer	Raspberry Pi Foundation
Processor	Broadcom Quad Core BCM2837
Architecture	64 bit ARMv8
Speed	1.2 Ghz Max
SRAM	1 GByte
Flash	None - SD card needed
Cost	$35
Operating System	Raspbian



7:56 / 8:56 Speed 1.0x

Lesson 3: The MSP432 Microcontroller

MSP432 Specifications

MSP432P401R Microcontroller

- Up to 48 MHz clock
- 64 KB SRAM
- 256 KB flash (non-volatile storage)
- Emphasis on compactness, low cost, low power

Courtesy of Texas Instruments

Lenovo T61 Laptop (Intel Core I5)

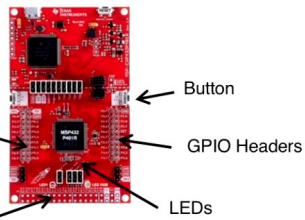
- 2.4 GHz clock
- 8 GB SRAM
- 300 GB solid state hard drive
- Emphasis on low latency calculation

1:22 / 12:06 Speed 1.0x

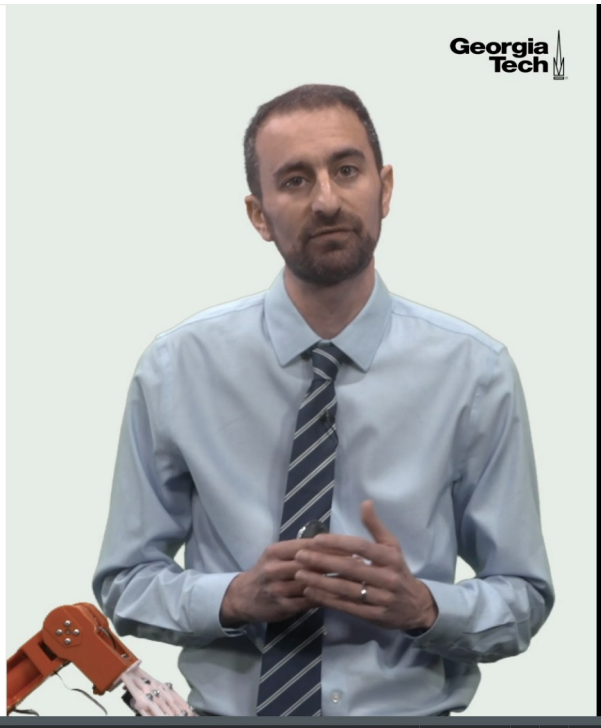
General Purpose Input-Output

Microcontrollers contain numerous general purpose input-output (GPIO) pins that are used to interface with external devices

These pins are exposed on development kits (or Evaluation Modules) as headers, buttons, or LEDs



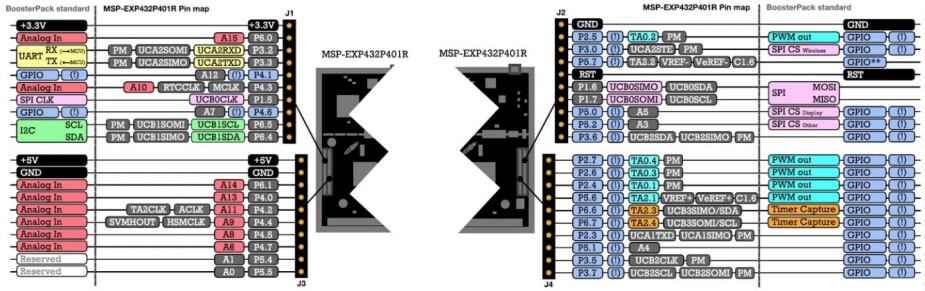
Courtesy of Texas Instruments



2:31 / 12:06 Speed 1.0x

General Purpose Input-Output

- What do all these pins do? Documentation provides pin functionality.
- Many pins can serve multiple possible purposes. Pin functionality is configured (and can be changed) during program execution.

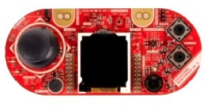


Courtesy of Texas Instruments

3:17 / 12:06 Speed 1.0x

Shields and BoosterPacks

- Shields (or TI BoosterPacks) are often available for MCU development kits that enable easy integration between peripherals and GPIO pins
- Designed to fit on top of development kits without need for wires



TI BOOSTXL-EDUMKII Educational BoosterPack

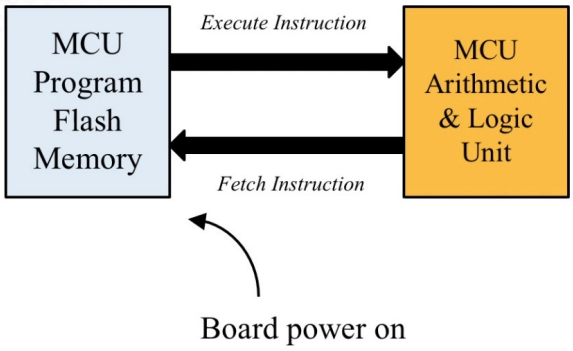
TI BOOSTXL-XLN2003 Dual Stepper Motor Driver BoosterPack

Courtesy of Texas Instruments

3:21 / 12:06 Speed 1.0x

Programming and Executing Code

Workflow for microcontroller operation



- As soon as MCU boots up, it reaches into program memory (flash) and loads the first instruction you programmed it with
- Thus your program executes automatically every time *on start up*
- Your program continues to run until board powered off
- Unlike typical PC, MCU does not run other background process / programs - *it does nothing except what you program it to do!*

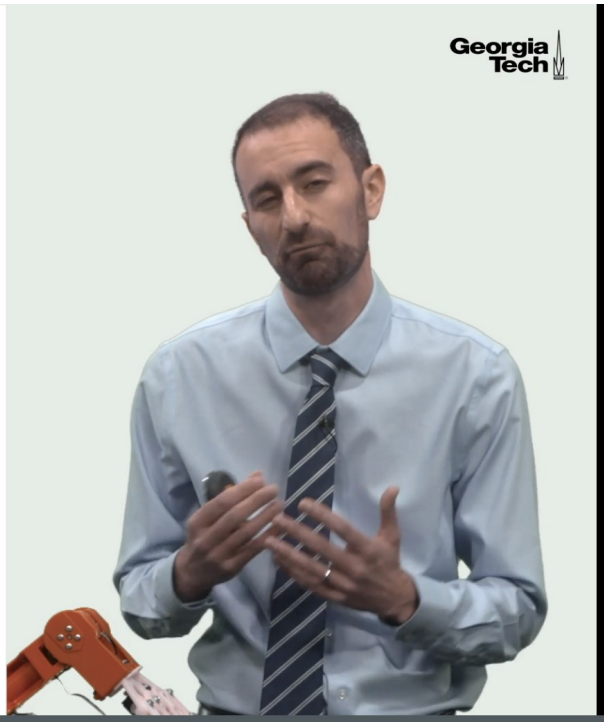
4:26 / 12:06 Speed 1.0x

Programming a Microcontroller

- Unlike PCs, you cannot write code directly on the microcontroller
- Instead, you write code on PC and then program (write) it into the MCU flash memory
- MCU must be connected to an external programmer for this to happen

TI MSP Gang Programmer - Programs up to 8 MSP chips at once

Courtesy of Texas Instruments



5:01 / 12:06 Speed 1.0x

Programming a Microcontroller

- Many development kits (or evaluation modules, EVMs) have on-board programming capability, meaning they can be programmed from a computer
- MSP432 LaunchPad has an onboard programmer (XDS-110-ET onboard emulator)

XDS110 On-board emulator

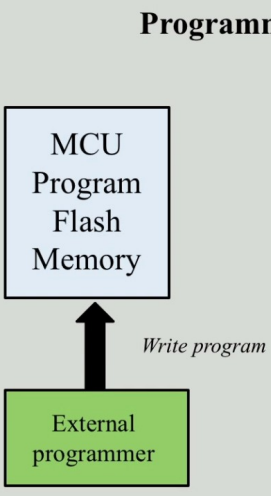
Courtesy of Texas Instruments

Purpose of on-board emulator:

- Provide USB connection to PC
- Program MCU from PC
- Enable PC-based debugging
- Provide power to MCU from USB connection

6:45 / 12:06 Speed 1.0x

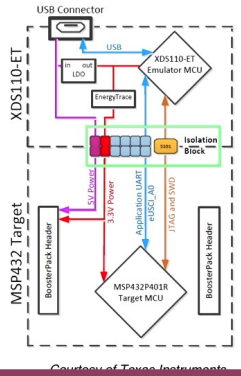
Programming a Microcontroller



7:15 / 12:06 Speed 1.0x

Powering the MSP432

- The MSP432 can be powered using the Launchpad USB interface with a computer
- MSP is powered at 3.3V. Jumper connections (red) provide power to MCU and header pins from USB connection across ”isolation block”
- Isolation block is group of jumpers that connect MCU portion of Launchpad to emulator portion

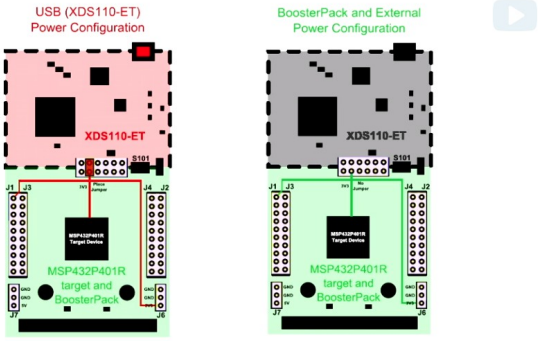


Courtesy of Texas Instruments

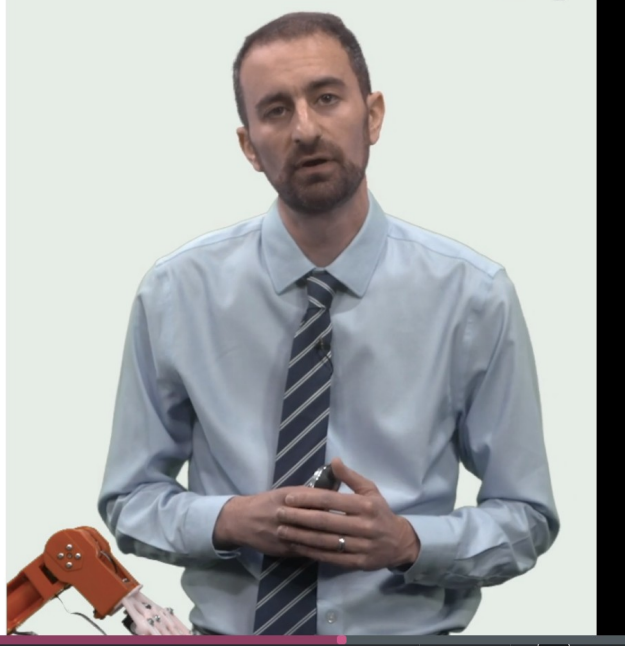
8:36 / 12:06 Speed 1.0x

Powering the MSP432

- To power the MSP432 from an external source (battery, power supply, etc.), remove 3.3V jumper in isolation block and provide 3.3V to J1.1 and ground to J3.22



Courtesy of Texas Instruments



9:30 / 12:06 Speed 1.0x

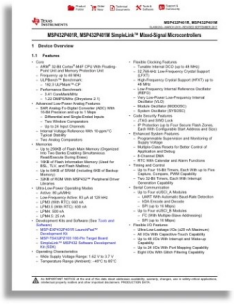
Documentation for the MSP432 LaunchPad

Georgia Tech

MSP-EXP432P401R LaunchPad User's Guide

- Technical specifications of LaunchPad dev kit

MSP432P401R Datasheet

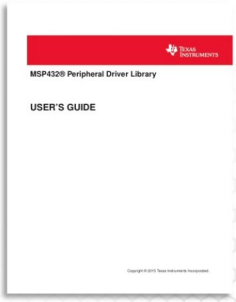


- Technical details of MSP432 hardware
- Pin diagrams of MSP432 integrated circuit

MSP432P401R Technical Reference Manual

- Details of MSP432 registers, configurations
- Main reference manual for embedded programming

MSP432P401R Driver Library User's Guide



- Definition of Driver Library API
- Documentation of all Driver Library (driverlib) functions

Courtesy of Texas Instruments