Introduction to the Microcontroller

John R. Wright, Jr., PhD, CSTM, CLSSGB, CSCE, F.ATMAE
AENG 467, Mobile Robotics
Millersville University
What is a Microcontroller?

http://avrlab.net/atmel-atmega128-datasheet-download
Brief History of the Microcontroller

- In 1971, the first microcontroller was invented by two engineers at Texas Instruments, according to the Smithsonian Institution.
  - Gary Boone and Michael Cochran created the TMS 1000, which was a 4-bit microcontroller with built-in ROM and RAM.
  - The same year that the microprocessor was invented at Intel
- The microcontroller was used internally at TI in its calculator products from 1972 until 1974, and was refined over the years.
- In 1974, TI offered the TMS 1000 for sale to the electronics industry. The TMS 1000 was available in various configurations of RAM and ROM sizes.

http://www.ehow.com/info_10018768_history-microcontroller.html
Brief History of the Microcontroller

- During the 1990s, microcontrollers with electrically erasable and programmable ROM (EEPROM) memories, such as flash memory, became available.
  - These microcontrollers could be programmed, erased and reprogrammed using only electrical signals.
  - Prior to the electrically reprogrammable devices, microcontrollers often required specialized programming and erasing hardware, which required that the device be removed from its circuit, slowing software development and making the effort more expensive.
  - With this limitation removed, microcontrollers were able to be programmed and reprogrammed while in a circuit so devices with microcontrollers could be upgraded with new software without having to be returned to the manufacturer. Many current microcontrollers, such as those available from Microchip and Atmel, incorporate flash memory technology.

http://www.ehow.com/info_10018768_history-microcontroller.html
Some Popular Microcontrollers

- BS2
  - www.parallax.com
- BX-24p
  - www.basicx.com
- Arduino Atmega32u4
  - http://pattonrobotics.com/products/teensy-3-2
  - www.radioshack.com
<table>
<thead>
<tr>
<th>Feature</th>
<th>Teensy 3.0</th>
<th>Teensy 3.2</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Price</strong></td>
<td>19.00</td>
<td>19.80</td>
<td>US Dollars</td>
</tr>
<tr>
<td><strong>Processor Core</strong></td>
<td>MK20DX128VLH5 Cortex-M4</td>
<td>MK20DX256VLH7 Cortex-M4</td>
<td>MHz</td>
</tr>
<tr>
<td><strong>Rated Speed</strong></td>
<td>48</td>
<td>72</td>
<td>MHz</td>
</tr>
<tr>
<td><strong>Overclockable</strong></td>
<td>96</td>
<td>96</td>
<td>MHz</td>
</tr>
<tr>
<td><strong>Flash Memory Bandwidth</strong></td>
<td>128</td>
<td>256</td>
<td>kbytes</td>
</tr>
<tr>
<td><strong>Cache</strong></td>
<td>96</td>
<td>192</td>
<td>Mbytes/sec</td>
</tr>
<tr>
<td><strong>RAM</strong></td>
<td>32</td>
<td>256</td>
<td>Bytes</td>
</tr>
<tr>
<td><strong>EEPROM</strong></td>
<td>16</td>
<td>64</td>
<td>kbytes</td>
</tr>
<tr>
<td><strong>Direct Memory Access</strong></td>
<td>2</td>
<td>2</td>
<td>kbytes</td>
</tr>
<tr>
<td><strong>Digital I/O Voltage Output</strong></td>
<td>34 3.3V Only</td>
<td>34 3.3V Only</td>
<td>3.3V Volts</td>
</tr>
<tr>
<td><strong>Voltage Input</strong></td>
<td>5V Tolerant</td>
<td>Pins Volts</td>
<td></td>
</tr>
<tr>
<td><strong>Analog Input Converters</strong></td>
<td>14</td>
<td>21</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Resolution</strong></td>
<td>1</td>
<td>2</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Usable</strong></td>
<td>16</td>
<td>16</td>
<td>Bits</td>
</tr>
<tr>
<td><strong>Prog Gain Amp</strong></td>
<td>13</td>
<td>13</td>
<td>Bits</td>
</tr>
<tr>
<td><strong>Touch Sensing</strong></td>
<td>0</td>
<td>2</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Comparators</strong></td>
<td>12</td>
<td>12</td>
<td>Bits</td>
</tr>
<tr>
<td><strong>Analog Output DAC Resolution</strong></td>
<td>2</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Timers</strong></td>
<td>2</td>
<td>3</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>FTM Type</strong></td>
<td>2</td>
<td>3</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>PWM Outputs</strong></td>
<td>10</td>
<td>12</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>PDB Type</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>GMT (infrared) Type</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>LPTMR Type</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>PIT (interval) Type</strong></td>
<td>4</td>
<td>4</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Systick</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>RTC (date/time)</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Communication</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>USB</strong></td>
<td>3</td>
<td>3</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Serial</strong></td>
<td>1</td>
<td>2</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>With FIFOs</strong></td>
<td>1</td>
<td>3</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>High Res Baud</strong></td>
<td>3</td>
<td>3</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>Fast Clock</strong></td>
<td>2</td>
<td>2</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>SPI</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>With FIFOs</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>I2C</strong></td>
<td>1</td>
<td>2</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>CAN Bus</strong></td>
<td>0</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>I2S Audio</strong></td>
<td>1</td>
<td>1</td>
<td>Pins</td>
</tr>
<tr>
<td><strong>FIFO Size</strong></td>
<td>4</td>
<td>8</td>
<td>Pins</td>
</tr>
</tbody>
</table>

https://www.pjrc.com/teensy/teensy31.html
Teensy 3.2

• 32 bit ARM processor
  • ARM = Advanced RISC Machine
    • RISC = Reduced Instruction Set Computer

• 72MHz Cortex-M4 Technology (Can Overclock at 96MHz)

• 64K RAM
News Flash!
Introducing the Teensy 4.0 (Teensy 3.2 Everything Killer)

- 32 bit ARM processor
  - ARM = Advanced RISC Machine
    - RISC = Reduced Instruction Set Computer
- 600MHz Cortex-M7 Technology
- 1024K RAM
- HACKADAY ARTICLE 8/7/2019
- [https://www pjrc.com/store/teensy40.html](https://www pjrc.com/store/teensy40.html)
What is an IDE?

IDE = Integrated Development Environment

- An integrated development environment (IDE) is a software suite that consolidates the basic tools developers need to write and test software.
- Typically, an IDE contains a code editor, a compiler or interpreter and a debugger that the developer accesses through a single graphical user interface (GUI).
- An IDE may be a standalone application, or it may be included as part of one or more existing and compatible applications.

http://searchsoftwarequality.techtarget.co/definition/integrated-development-environment

- JAVA uses Eclipse as its IDE
- We use C++ to Program the Teensy
  - via a patch called “Teensyduino”
    - utilizing the Arduino IDE

https://www pjrc.com/teensy/teensyduino.html
Teensy Vs. Arduino

• **Comparison**
  • https://www.youtube.com/watch?v=rklfsYRshRQ

• **Getting Setup with Teensy Software! (14:36)**
  FYI:
  Brian is the inventor of the Teensy Motherboard (Carrier Board) that we use and used to be with RoboOdyssey - maker of the BX24p (BasicX) microcontroller.

  • Please be very careful when using the microcontroller without the carrier board - pins might short out the device if they should touch metal (use a wood table or use on top of a piece of paper)! Or one may plug it into a bread/proto board.

  Step 1 - Download Arduino IDE
  Step 2 - Download Teensyduino
  Follow instructions and make sure this is installed in same location as the Arduino software.
  Step 3 - Open the Arduino program once and then close it.
  Step 4 - Open Arduino software, connect microcontroller to USB of computer
  Step 5 - Under Tools...Board Menu select the microcontroller you are using
  Step 6 - Write a short program and try to connect. You may need to be patient the first time as it takes a little time the very first time we launch the Teensyduino.
Intro to ARM Technology!

- [Link to ARM's YouTube video](https://www.youtube.com/watch?v=_R06xdksYTI)

Cortex-A
Highest performance
Optimized for rich operating systems

Learn more about the Cortex-A series processors

Cortex-R
Fast response
Optimized for high-performance, hard real-time applications

Learn more about the Cortex-R series processors

Cortex-M
Smallest/lowest power
Optimized for discrete processing and microcontroller

Learn more about the Cortex-M series processors

SecurCore
Tamper resistant
Optimized for security applications

Learn more about SecurCore processors

[Link to ARM's processors page](https://www.arm.com/products/processors)
What is a Carrier Board?

PRT Motherboard tutorial 1 (12:08)
Intro to hardware and USB Voltage Isolator
https://www.youtube.com/watch?v=9in-VuT08q0&index=10&list=PLY12Xb6BPCrriVQfwxfCJdSI F6Chkh6R&t=73s
Training Platform

http://www.robodyssey.com/mouse-basic/

http://pattonrobotics.com/products/onebot-basic-complete
Our R&D Application
Responsibilities
(Senior Capstone R&D)

- To learn the code on your own using the provided resources. You will likely learn as you go - as you need the capability. There are multiple ways to approach the challenges and R&D project. Each developer may have a different approach so teaching will be done side-by-side as a result. You must research the code, however.

- You are responsible for the technology you check out! Replacement sensors and microcontrollers can be found here:
  - www.pattonrobotics.com
  - https://www.pjrc.com/teensy/
  - https://openmv.io/products/openmv-cam-h7

- You are expected to spend a considerable amount of time outside of class working on this R&D project. 75-100 hours of development time is typical and expected of a capstone experience.
I Code Every Night...

• All Night Longer - this semester!
• https://www.youtube.com/watch?v=z5_2xkOpVHU