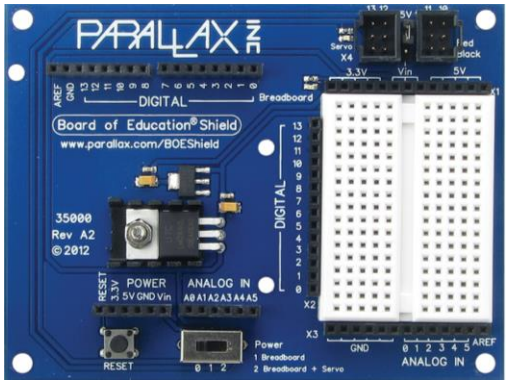
Arduino BOE kit and Raspibot Board

Class 1: Mobile Robot Development Platforms



August 7, 2017

Don Wilcher

Presented by:

DesignNews CEC CONTINUING EDUCATION CENTER Digi-Key ELECTRONICS ROHDE&SCHWARZ

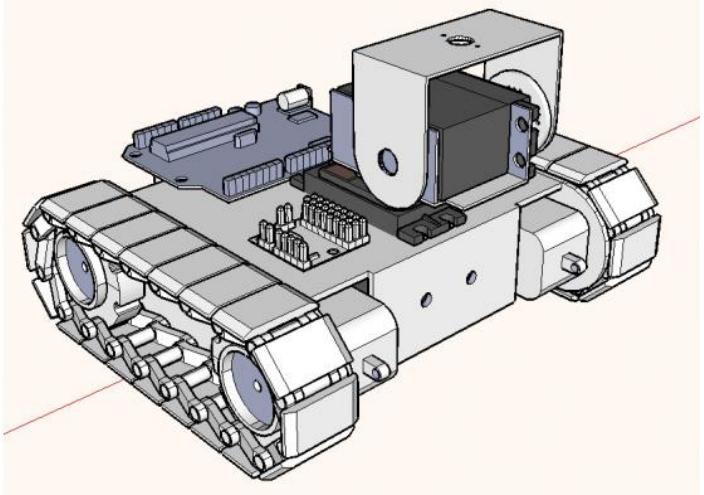
Class 1: Mobile Robot Development Platforms

Agenda

- What Are Mobile Robot Development Platforms?
- Exploring Mobile Robot Development Platform Examples
- The Parallax Arduino BOE Shield
- Hands-On Labs: Coding Examples

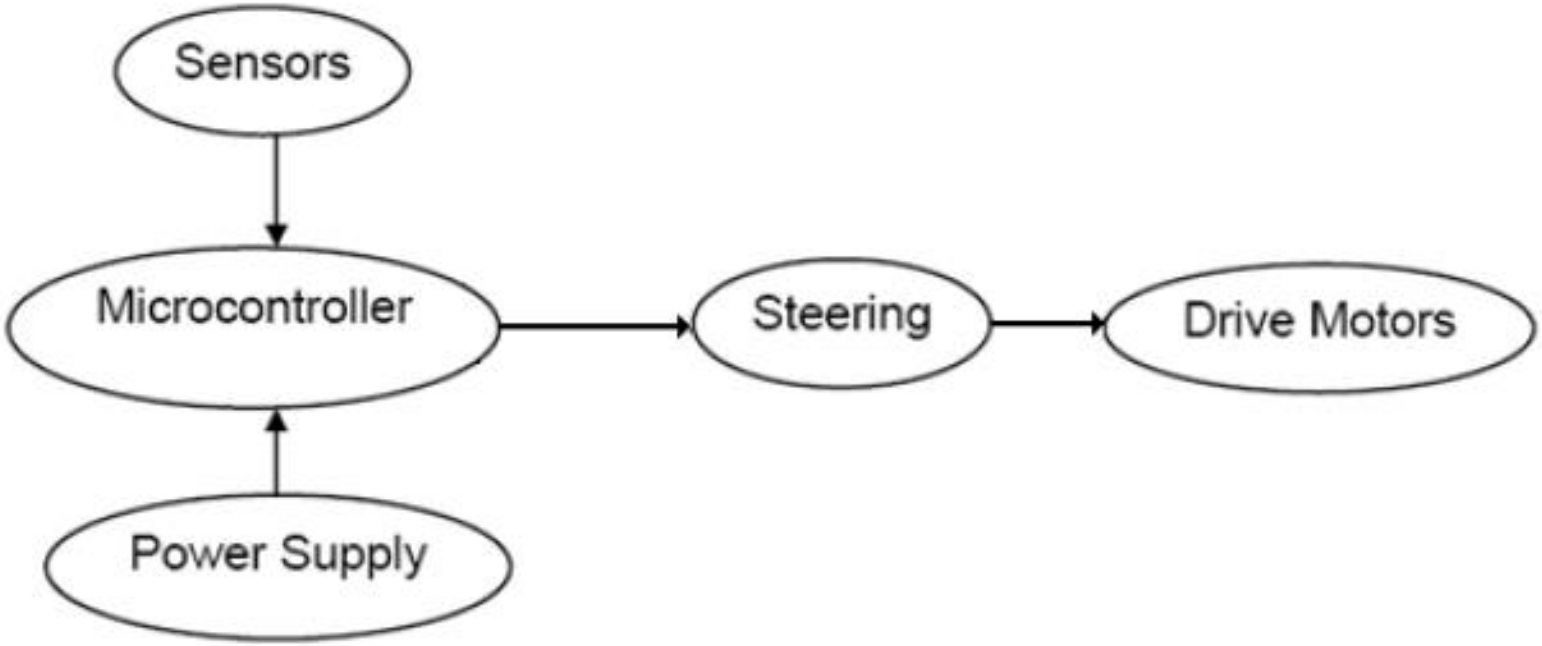
What Are Mobile Robot Development Platforms?

Engineering infrastructure that allows rapid build and test of automatic systems capable of locomotion.



What Are Mobile Robot Development Platforms?...

Parts of a Typical Mobile Robot



Source:

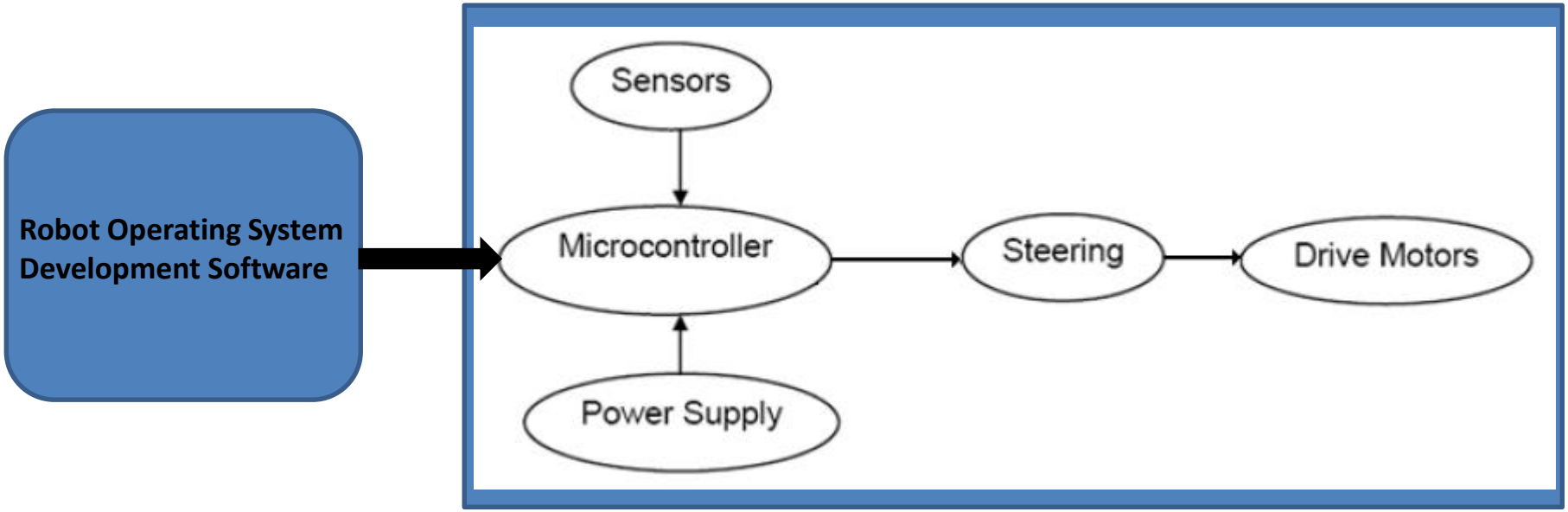
https://www.researchgate.net/publication/265266664_AC_2011-1253_AN_INTERDISCIPLINARY_TEAM-BASED_MOBILE_ROBOTS_DESIGN_COURSE_FOR_ENGINEERING_TECHNOLOGY_An_Interdisciplinary_Team-Based_Mobile_Robots_Design_Course_for_Engineering_Technology

Question 1

Name the five parts of a typical mobile robot.

What Are Mobile Robot Development Platforms?...

A Typical Mobile Robot Development Platform



What Are Mobile Robot Development Platforms?...

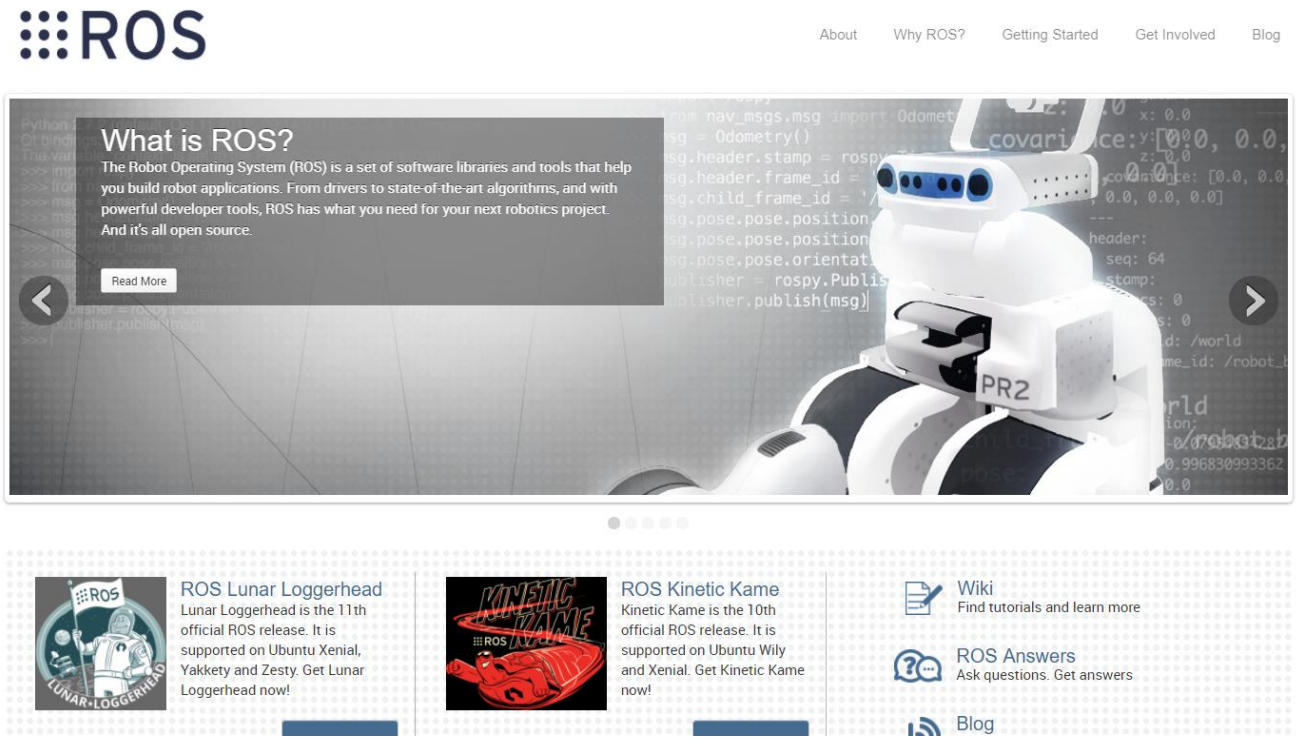
ROS(Robot Operating System):

- The Robot Operating System (ROS) is a set of software libraries and tools that help you build robot applications.
- From drivers to state-of-the-art algorithms, and with powerful developer tools, ROS has tools to assist in building a mobile robot.
- And it's all open source.

Source:
http://www.ros.org/

What Are Mobile Robot Development Platforms?...

ROS(Robot Operating System): Website



Source:

http://www.ros.org/

What Are Mobile Robot Development Platforms?...

The Microcontroller:

- At the heart of a mobile robot development platform is a microcontroller.
- A microcontroller is a computing device capable of executing a program
  a) (i.e. a sequence of instructions)
  b) often referred to as the “brain” or “control center” in a robot since it is usually responsible for all computations, decision making, and communications.

Source:
http://www.robotshop.com/blog/en/how-to-make-a-robot-lesson-4-understanding-microcontrollers-2-3700

Question 2

At the heart of a mobile robot development platform is a _____.

a) a heart

b) lungs

c) microprocessor

d) microcontroller

What Are Mobile Robot Development Platforms?...

The Microcontroller Development Platform:

- An embedded hardware ecosystem that allows rapid creation of intelligent devices and system.
- A microcontroller development platform is consists of:

  a) target microcontroller (mcu).

  b) regulated power supplies (+5V and 3.3V).

  c) on board crystal oscillator.

  d) accessible GPIO (General Purpose Input-Output pins)

What Are Mobile Robot Development Platforms?...

The Microcontroller Development Platform:

e) supported I/O features.

i. HDMI

ii. audio

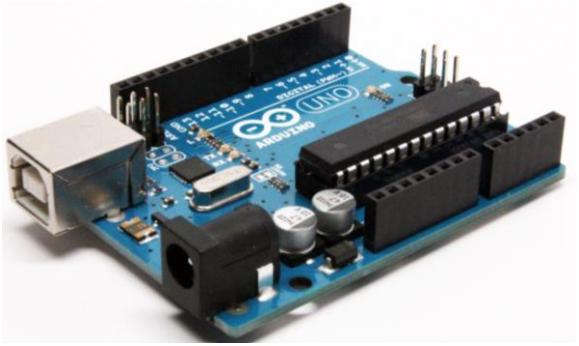
iii. potentiometers.

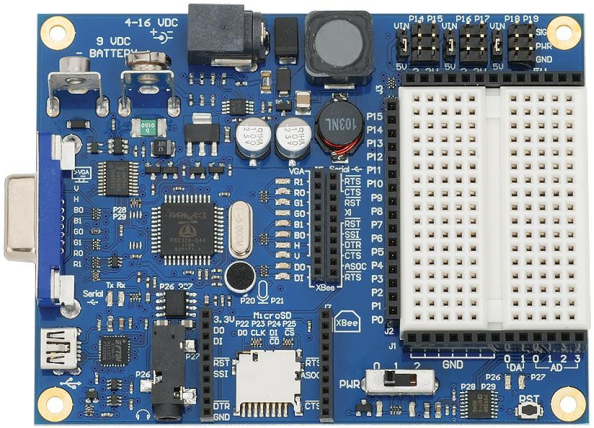
iv. pushbutton switches.

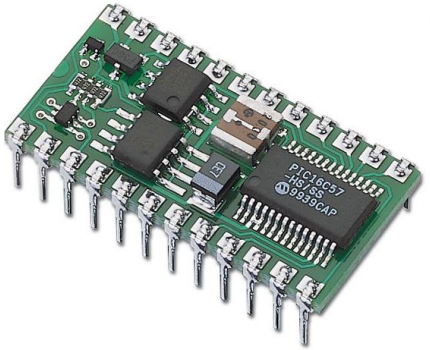
v. LEDs.

vi. VGA.

What Are Mobile Robot Development Platforms?...





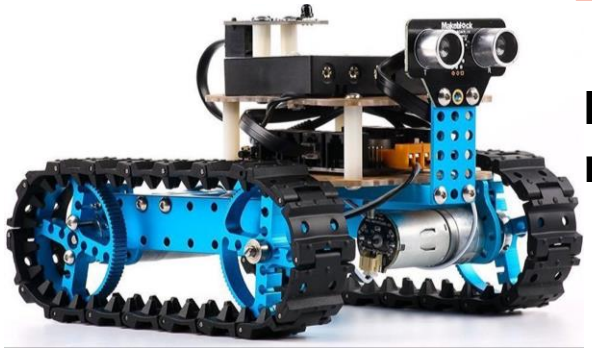


Examples of Microcontroller Development Platforms:

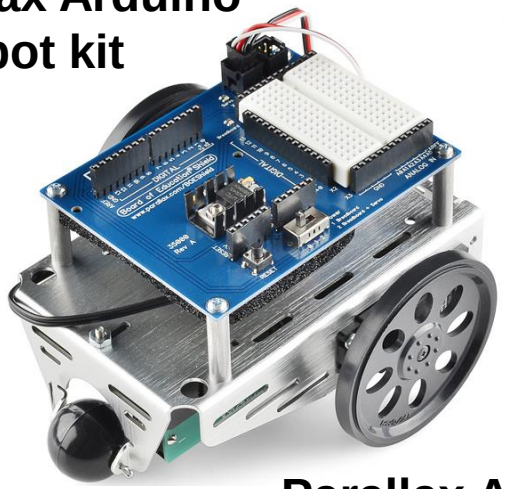
Presented by:

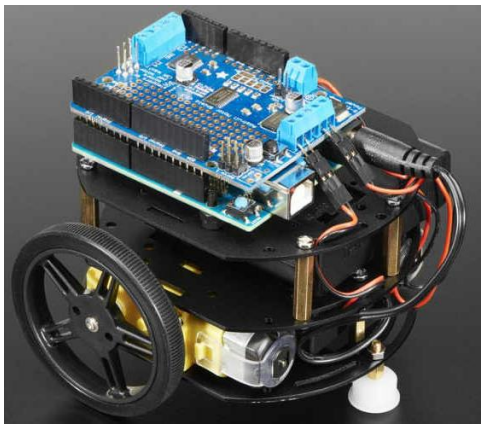
Exploring Mobile Robot Development Platform Examples

Parallax Arduino BOE bot kit



Makeblock mobile robot kit





Dexter Industries GoPiGo bot

Mini 3Layer Round Robot kit

Parallax Arlo

Presented by:

Question 3

The Basic Stamp BOE bot is the precursor to what contemporary mobile robot development platform.

a) The GoPiGo mobile robot

b) The Arlo

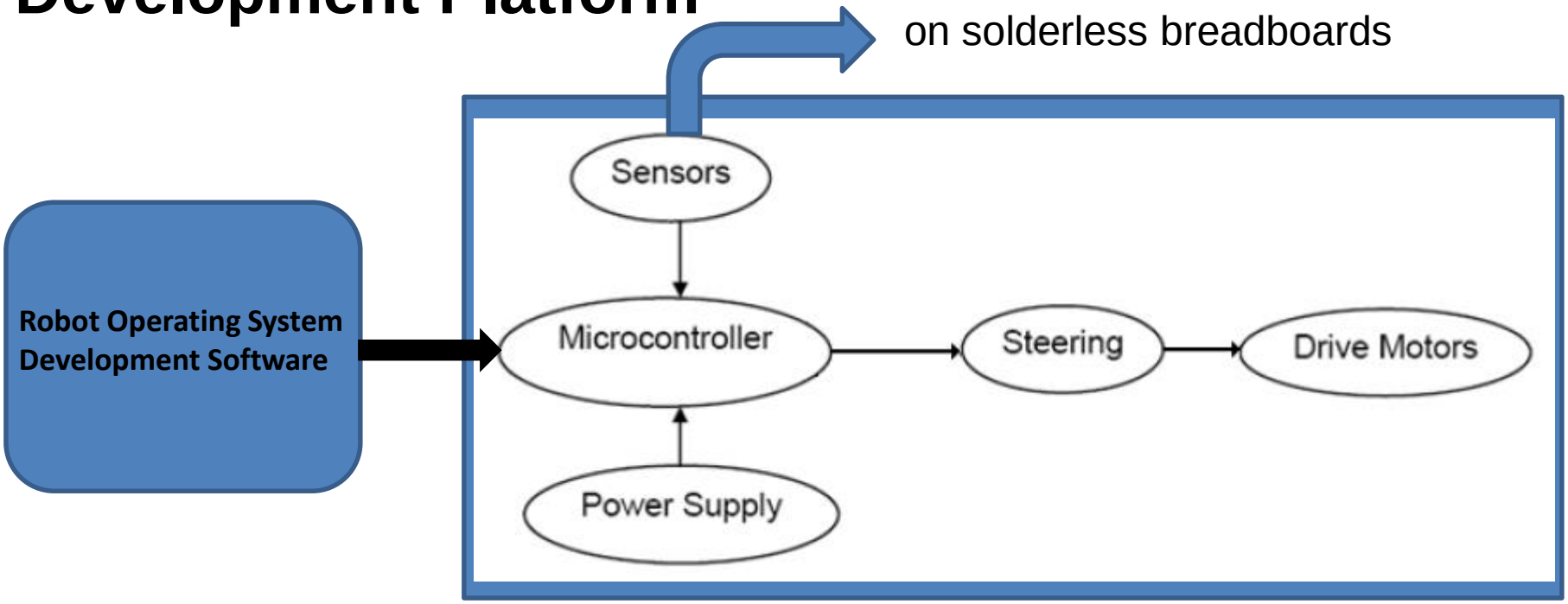
c) The Arduino BOE bot kit

d) Makeblock robot kit

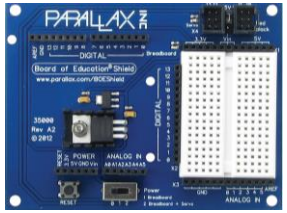
Exploring Mobile Robot Development Platform Examples…

A Typical Mobile Robot Development Platform

Some sensor circuits are built on solderless breadboards



The Parallax Arduino BOE Shield…



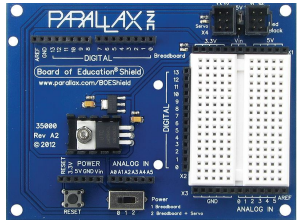
What is a Parallax Arduino BOE Shield?

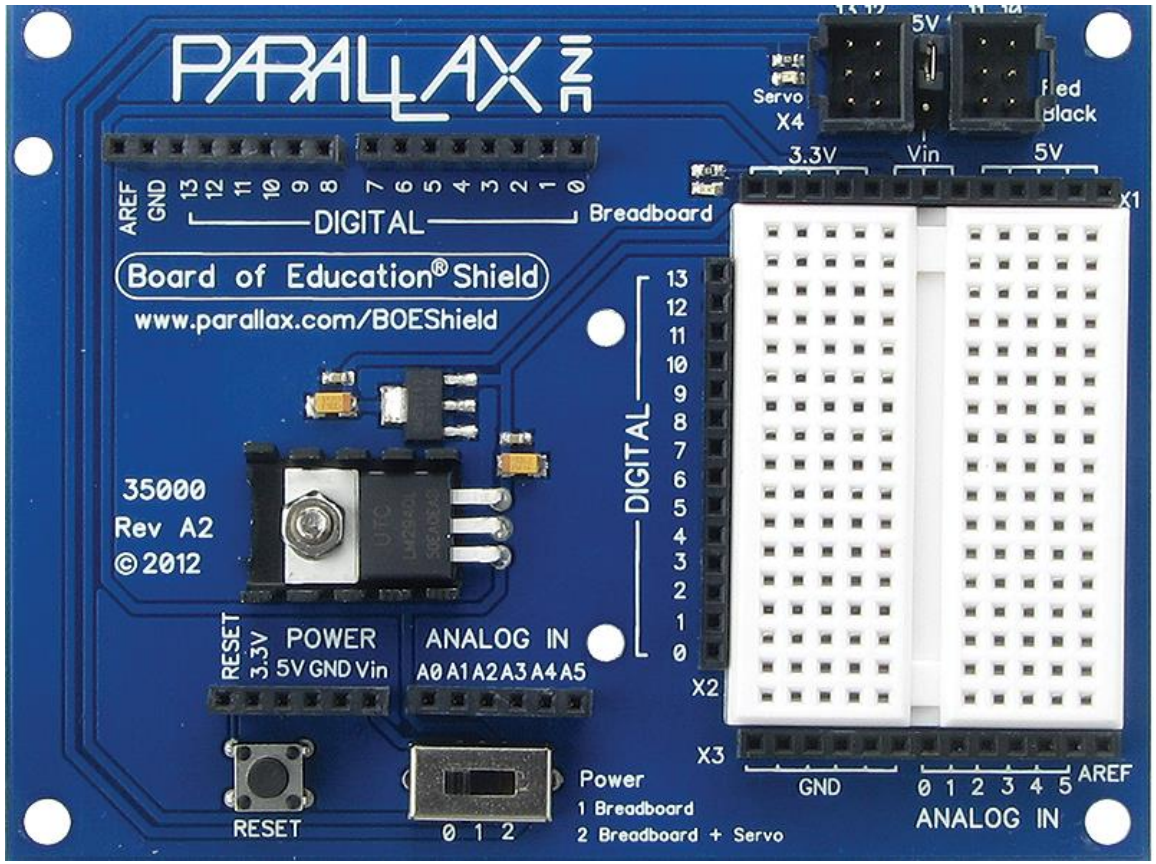
a) An electronics platform for building robotics devices and machines

b) Provides accessibility to GPIO and Analog pins using single inline female header connectors.

c) Provides accessibly to (2) 2x3 male pins for servo motor attachment.

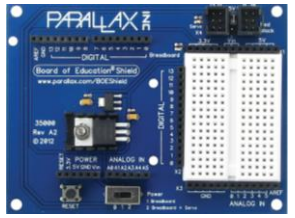
The Parallax Arduino BOE Shield

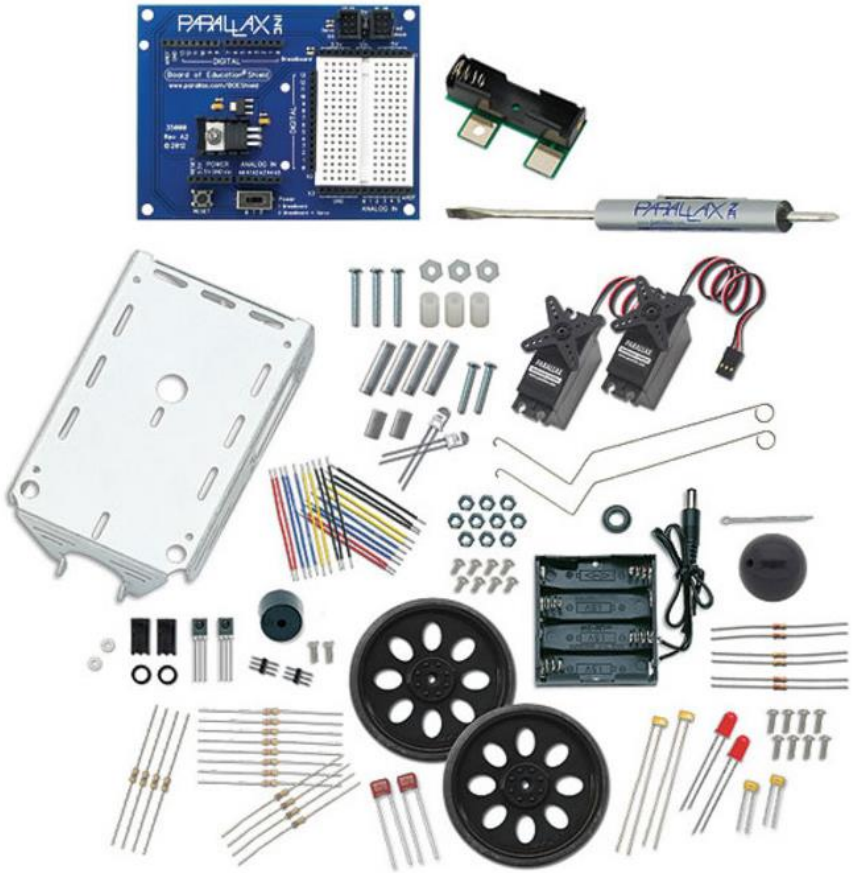




Up close view of the Parallax Arduino BOE Shield.

The Parallax Arduino BOE Shield…

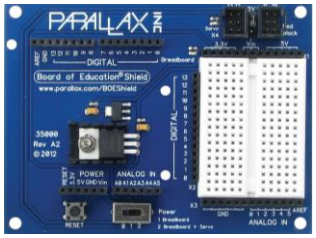




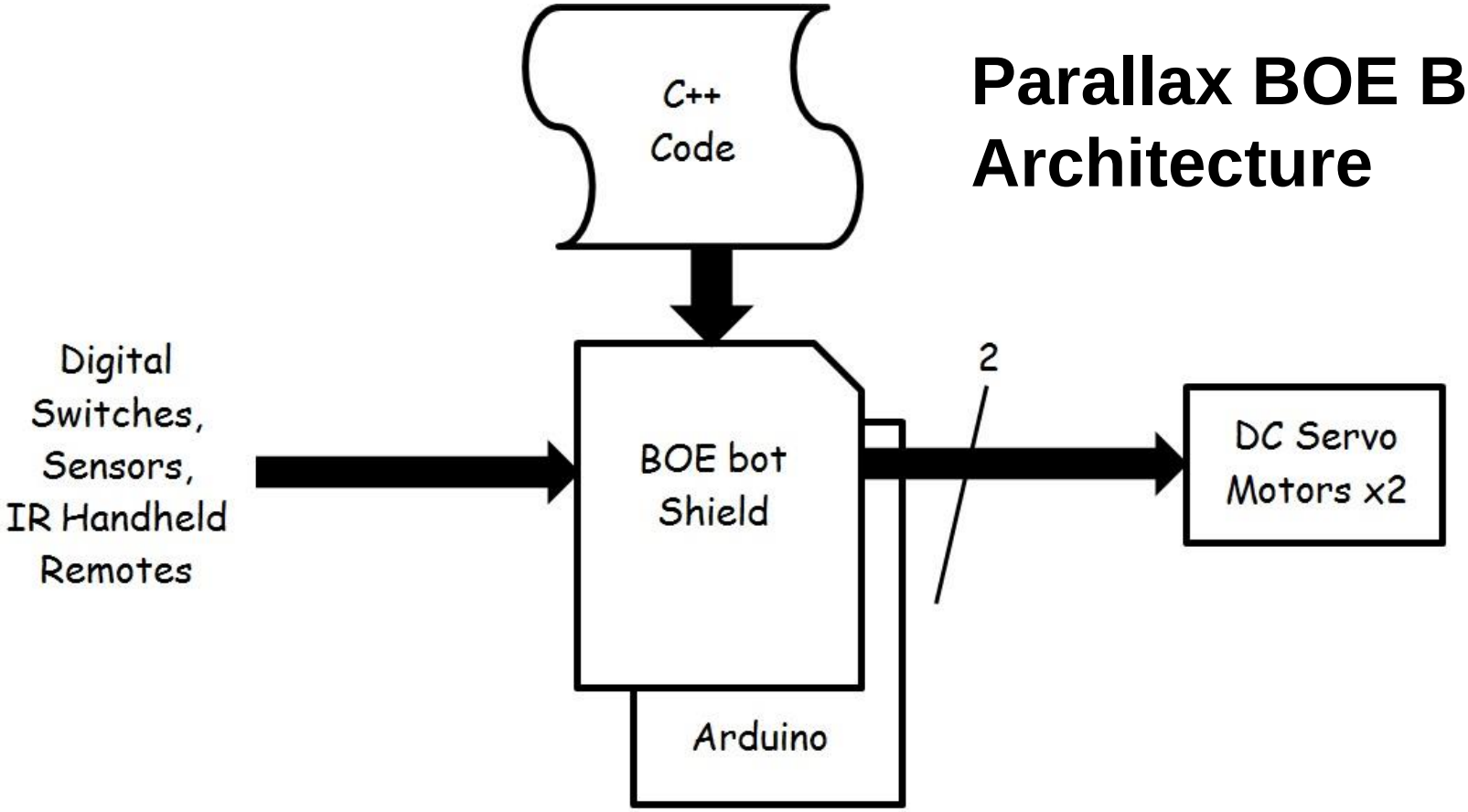
Parallax Arduino BOE Shield is packaged as a robotics kit.

Source:
https://www.digikey.com/product-detail/en/parallax-inc/32335/32335-ND/6009017

The Parallax Arduino BOE Shield…



Parallax BOE Bot Architecture



Question 4

The Arduino BOE bot architecture supports how many servo motors?

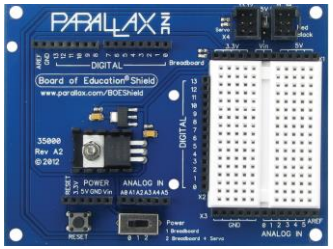
a) 4 servo motors

b) 3 servo motors

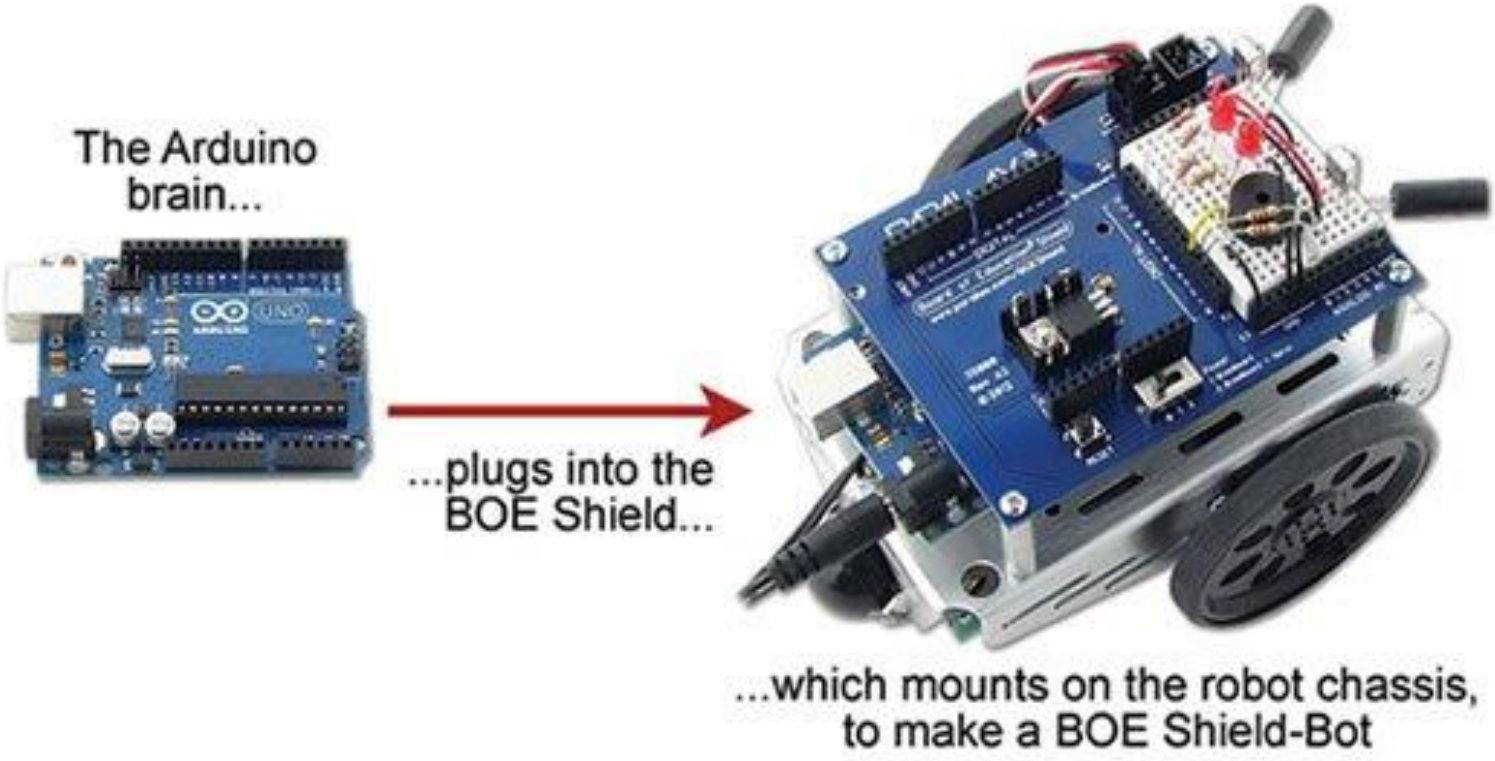
c) none of the above

d) 2 servo motors

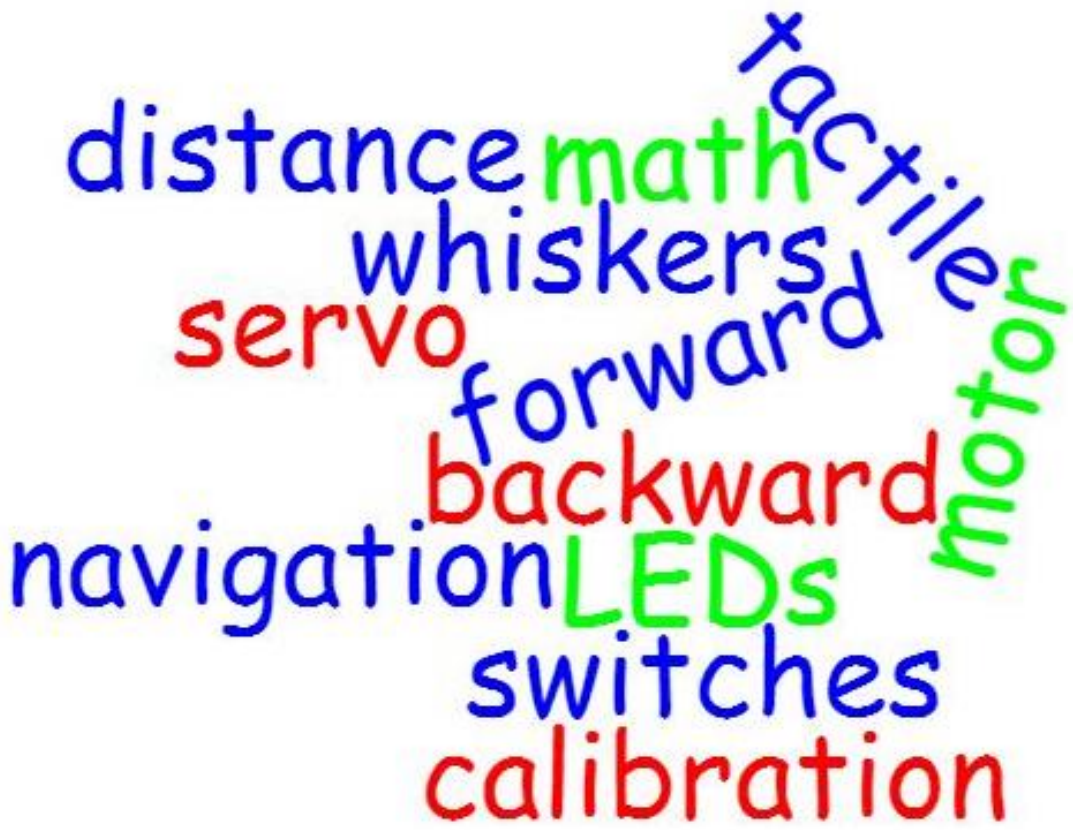
The Parallax Arduino BOE Shield…



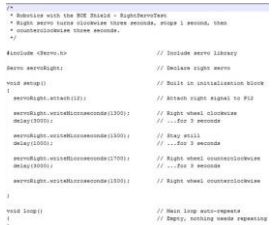
Assembly of the BOE Shield to the Arduino



Hands-On Labs: Coding Examples



Hands-On Labs: Coding Examples…



Objectives of Coding Labs

- To insure the Arduino IDE is installed correctly.
- To explore the Arduino IDE's programming environment.
- To explore turning the Arduino into a programmable calculator.

Note: We'll be using examples from Robotics with Board of Education Shield for Arduino Book by Andy Lindsay, version 1.0

Book Source:

https://www.parallax.com/sites/default/files/downloads/122-32335-Robotics-BOE-Shield-Bot-Arduino-v1.0.pdf

Hands-On Labs: Coding Examples…

Activity 1: Download and Install the Software

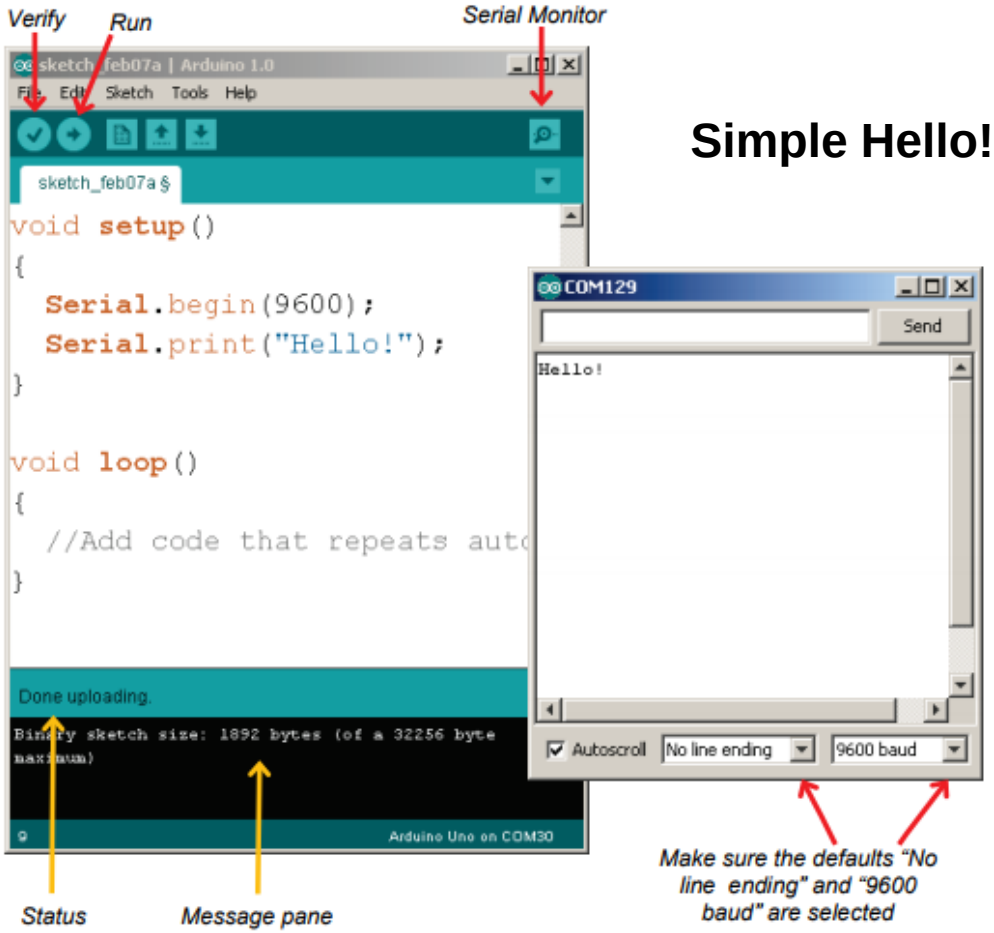
If this is your first time working with the Arduino system, you will need to set up a software option to write programs, called *sketches*. Two choices are Codebender and Arduino IDE.

Getting Started with the Arduino IDE Software

Arduino IDE software and drivers install on your Windows, Mac, or Linux computer. You do not need to be online to use it.

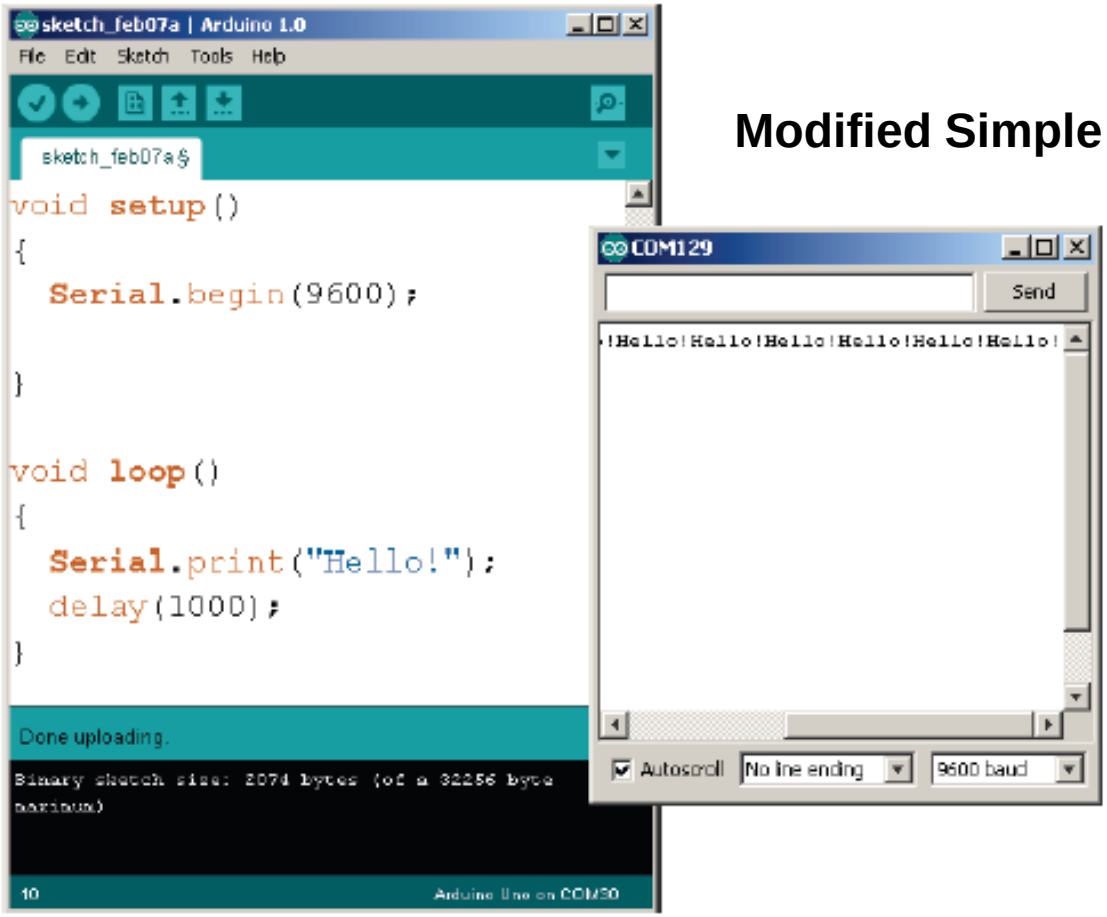
- ✓ Go to www.arduino.cc and click on the Getting Started link.
- ✓ Follow their instructions for downloading and installing the latest Arduino software, and installing the USB driver your system will need to communicate with the Arduino.
- ✓ Make sure to follow the instructions through the part where you connect your Arduino Uno module to your computer with a USB A to B cable, and successfully load a sample sketch to confirm your programming connection.
- ✓ Download the Shield-Bot Arduino code from this book's product page at www.parallax.com. It is product #122-32335.
- ✓ Save the file to your desktop, and un-zip it before trying to use the sketches in it.

Hands-On Labs: Coding Examples…



Simple Hello! sketch

Hands-On Labs: Coding Examples…



Modified Simple Hello! sketch

Hands-On Labs: Coding Examples…

Hello Messages on New Lines

How about having each "Hello!" message on a new line? That would make the messages scroll down the Serial Monitor, instead of across it. All you have to do is change `print` to `println`, which is short for 'print line.'

Using the Serial.println() instruction

- ✓ Change `Serial.print("Hello!")` to `Serial.println("Hello!")`.
- ✓ Run the modified sketch and watch it print each "Hello!" message on a new line.

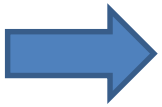
Question 5

Contrast Serial.print() and Serial.println().

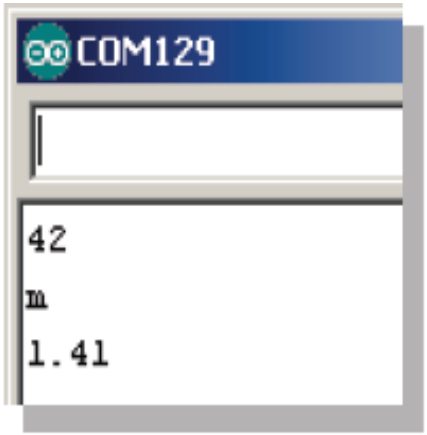
Hands-On Labs: Coding Examples…

Examples of Variable assignment statements

```
int a = 42;
char c = 'm';
float root2 = sqrt(2.0);
```



```
Serial.println(a);
Serial.println(c);
Serial.println(root2);
```



Hands-On Labs: Coding Examples…

Example Sketch – StoreRetrieveLocal

- ✓ Create a new sketch, and save it as StoreRetrieveLocal.
- ✓ Open or create and save the StoreRetrieveLocal sketch, and run it on your Arduino.
- ✓ Open the Serial Monitor and verify that the values display correctly.

```
// Robotics with the BOE Shield - StoreRetrieveLocal

void setup()
{
  Serial.begin(9600);

  int a = 42;
  char c = 'm';
  float root2 = sqrt(2.0);

  Serial.println(a);
  Serial.println(c);
  Serial.println(root2);
}

void loop()
{
  // Empty, no repeating code.
}
```

Code for Variable assignment statements lab activity.

Hands-On Labs: Coding Examples…

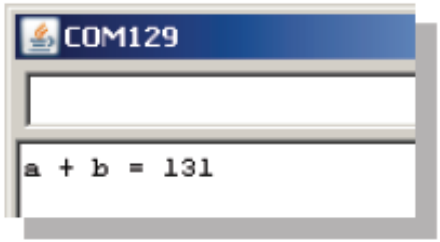
Activity 4: Solve Math Problems

Arithmetic operators are useful for doing calculations in your sketch. In this activity, we'll focus on the basics: assignment (=), addition (+), subtraction (–), multiplication (*), division (/), and modulus (%, the remainder of a division calculation).

- ✓ Open up the Arduino Language Reference, and take a look at the list of Arithmetic Operators.

The next example sketch, SimpleMath, adds the variables `a` and `b` together and stores the result in `c`. It also displays the result in the Serial Monitor.

Notice that `c` is now declared as an `int`, not a `char` variable type. Another point, `int c = a + b` uses the assignment operator (=) to copy the result of the addition operation that adds `a` to `b`. The next screen capture shows the expected result of 89 + 42 = 131 in the Serial Monitor.



Hands-On Labs: Coding Examples…

Performing math on an Arduino.

```
// Robotics with the BOE Shield - SimpleMath

void setup()
{
  Serial.begin(9600);

  int a = 89;
  int b = 42;
  int c = a + b;

  Serial.print("a + b = ");
  Serial.println(c);
}

void loop()
{
  // Empty, no repeating code.
}
```

Hands-On Labs: Coding Examples…

```
float r = 0.75;
float c = 2.0 * PI * r;
```

Example Sketch - Circumference

- ✓ Create the Circumference sketch and save it.
- ✓ Make sure to use the values 0.75 and 2.0. Do not try to use 2 instead of 2.0.
- ✓ Run your sketch on the Arduino and check the results with the Serial Monitor.

```
// Robotics with the BOE Shield - Circumference

void setup()
{
  Serial.begin(9600);

  float r = 0.75;
  float c = 2.0 * PI * r;

  Serial.print("circumference = ");
  Serial.println(c);
}

void loop()
{
  // Empty, no repeating code.
}
```

Code demonstrating Floating Point Math

Question 6

The area of circle is $A = \pi r^2$. How can r^2 be expressed in Arduino code?

Hands-On Labs: Coding Examples…

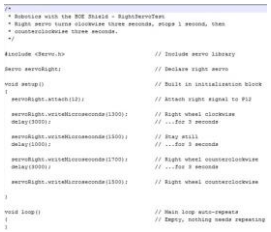
Activity 5: Make Decisions

Your BOE Shield-Bot will need to make a lot of navigation decisions based on sensor inputs. Here is a simple sketch that demonstrates decision-making. It compares the value of `a` to `b`, and sends a message to tell you whether or not `a` is greater than `b`, with an `if…else` statement.

If the condition `(a > b)` is true, it executes the `if` statement's code block: `Serial.print("a is greater than b")`. If `a` is *not* greater than `b`, it executes the `else` code block instead: `Serial.print("a is not greater than b")`.

- ✓ Create the SimpleDecisions sketch, save it, and run it on the Arduino.
- ✓ Open the Serial Monitor and test to make sure you got the right message.
- ✓ Try swapping the values for `a` and `b`.
- ✓ Re-load the sketch and verify that it printed the other message.

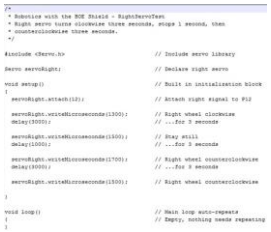
Hands-On Labs: Coding Examples…



```
// Robotics with the BOE Shield - SimpleDecisions

void setup()
{
  Serial.begin(9600);

  int a = 89;
  int b = 42;

  if(a > b)
  {
    Serial.print("a is greater than b");
  }
  else
  {
    Serial.print("a is not greater than b");
  }
}

void loop()
{
  // Empty, no repeating code.
}
```

Making Decisions Code

Hands-On Labs: Coding Examples…



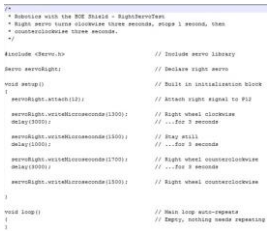
```
void setup()
{
  Serial.begin(9600);

  int a = 89;
  int b = 42;

  if(a > b)
  {
    Serial.print("a is greater than b");
  }
}
```

If, else if, else decision making

Maybe your sketch needs to monitor for three conditions: greater than, less than, or equal. Then, you could use an `if…else if…else` statement.

```
if(a > b)
  {
    Serial.print("a is greater than b");
  }
  else if(a < b)
  {
    Serial.print("a is not greater than b");
  }
  else
  {
    Serial.print("a is equal to b");
  }
```

Hands-On Labs: Coding Examples…



Boolean Operators: AND and OR

A sketch can also have multiple conditions with the Arduino's *Boolean operators*, such as `&&` and `||`. The `&&` operator means AND; the `||` operator means OR. For example, this statement's block will execute only if `a` is greater than 50 AND `b` is less than 50:

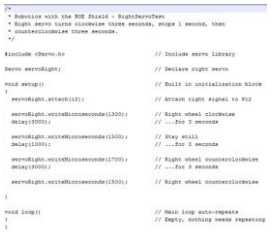
```
if((a > 50) && (b < 50))
  {
    Serial.print("Values in normal range");
  }
```

This example prints the warning message if `a` is greater than 100 OR `b` is less than zero.

```
if((a > 100) || (b < 0))
  {
    Serial.print("Danger Will Robinson!");
  }
```

One last example: if you want to make a comparison to find out if two values are equal, you have to use two equal signs next to each other: `==`.

Hands-On Labs: Coding Examples…



This example prints the warning message if a is greater than 100 OR b is less than zero.

```
if((a > 100) || (b < 0))
  {
    Serial.print("Danger Will Robinson!");
  }
```

One last example: if you want to make a comparison to find out if two values are equal, you have to use two equal signs next to each other: ==.

```
if(a == b)
  {
    Serial.print("a and b are equal");
  }
```