

Lesson 5 (PC – Arduino Integration)

The Best of Both Worlds

A POSTERIORI
Play · Experience · Learn

Advantages

PC

- Large display
- Keyboard and mouse
- Sound, graphics, video
- Great for games and animation

Arduino

- Read wide variety of sensors
- Control electronics (eg. lights, sound, motors)
- Great for real-world interaction

Nintendo Wii

Integrated real-world sensors with on-screen games



Vastly more successfully than its competitors (PS3 and Xbox 360)

A POSTERIORI

Play · Experience · Learn

Slides available at: <http://a9i.sg/huayi>

Real-World Applications



Rehabilitation training for stroke patients
(Games can have a strong motivating effect on patients)



Flight simulator
(...this one was built using an Arduino)

Exercise

Create a Scratch game that works with the
Arduino

A POSTERIORI
Play · Experience · Learn

Game Idea

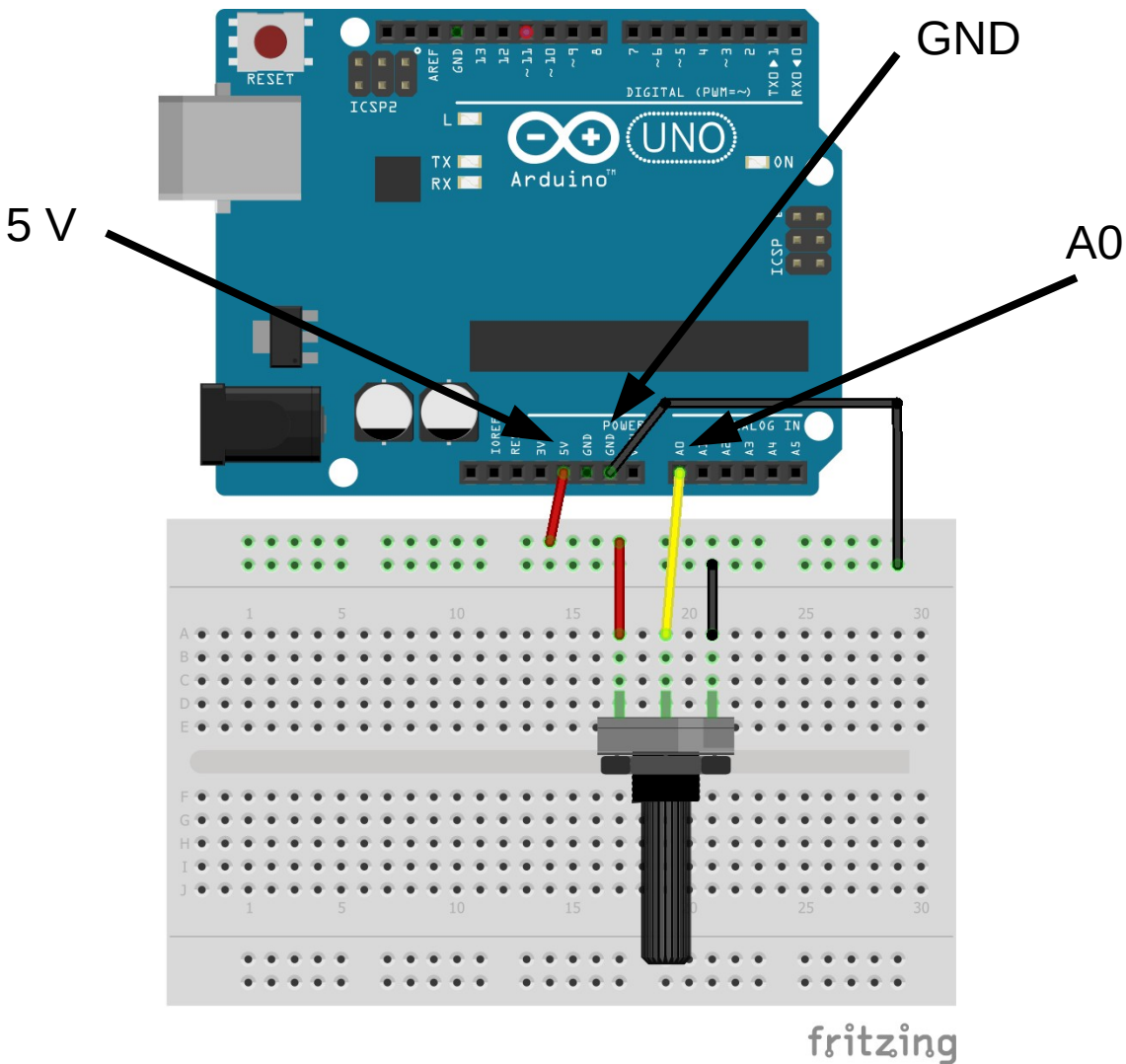
- Side Scroller
- Control using potentiometer
- Flash Red LED if balloon is missed
- Flash Green LED when balloon is caught



Important Tips!

- In mBlock, the sprites cannot directly read or control an Arduino; you need to use **Broadcast**
- Use **variables** to share the result of an analog read with other sprites
- It's possible for an Arduino to communicate with a PC in “**Upload**” mode, but it's much easier in “**Live**” mode (...we will only be using Live mode for this exercise)

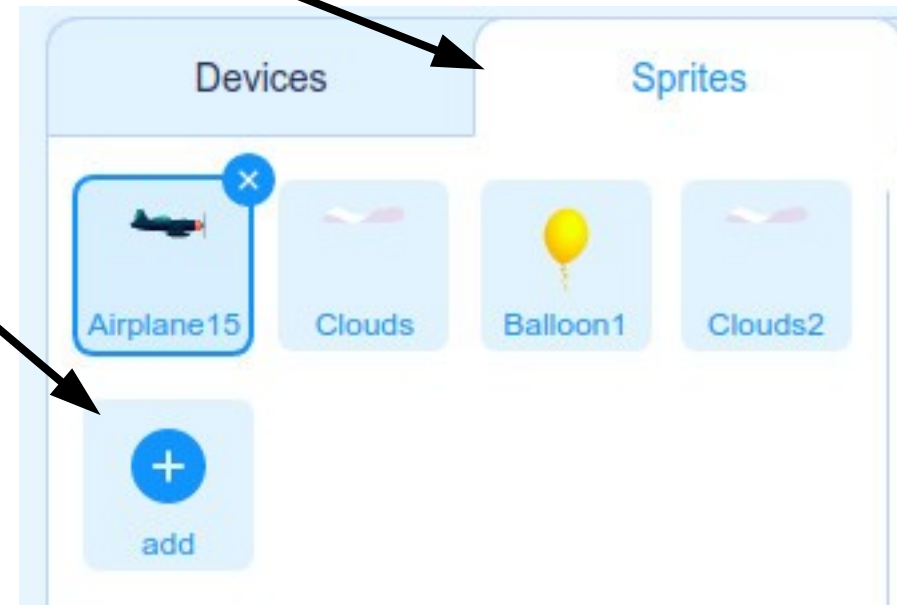
1) Basic Control



- Connect the potentiometer to one of the analog input
- Analog inputs are pin A0 to A5

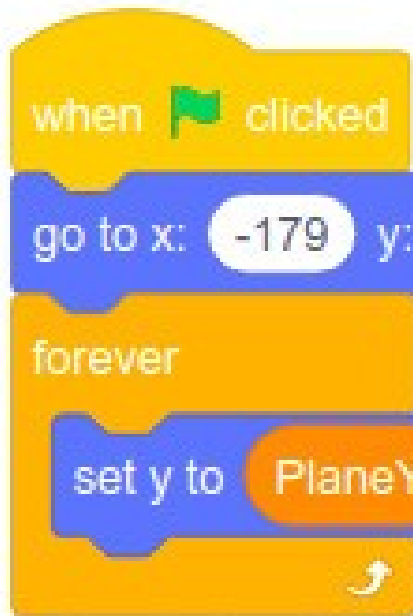
1) Basic Control

- Go to the “Sprites” tab...
- ...and add a new sprite
- I’m using a plane here, but you can go wild with this



1) Basic Control

- In the plane sprite, create this script...



Starting position of plane, experiment to get a suitable x and y

You'll need to create the "PlaneY" variable first

1) Basic Control

- In the Arduino device, create this script...

“Read analog” will give you a value between 0 to 1023

“map” is used to convert the analog read range (0 to 1023) to the y coordinate range (-150 to 150)

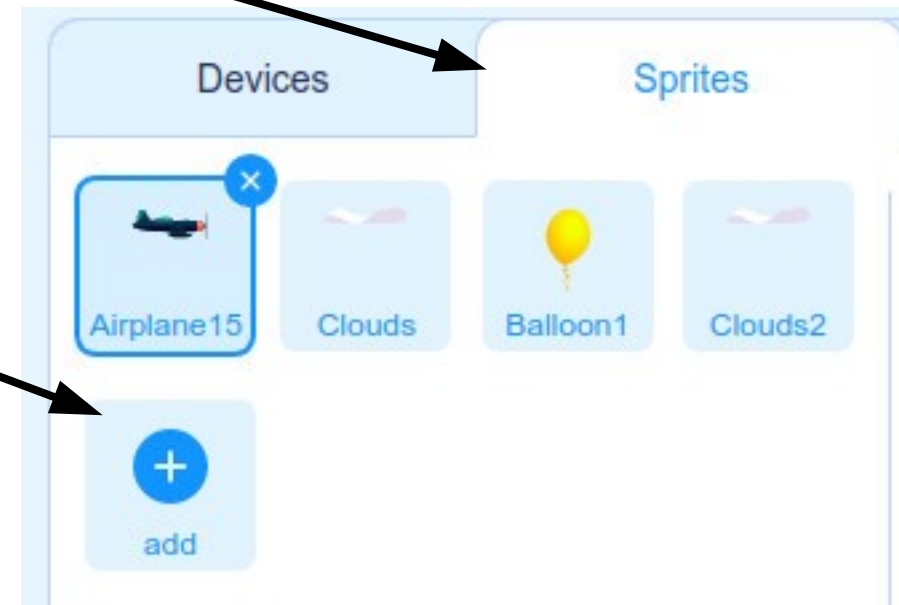


Challenge

1) In the example program, the potentiometer controls the **position** of the plane. That makes the game too easy. Can you make it control the **Y-velocity** instead?

2) Adding Balloons

- Go to the “Sprites” tab again...
- ...and add a balloon sprite



2) Adding Balloons

- Add the following script to the balloon...



Starting position of balloon. The Y-position is randomized

“Changing” the X-position by a negative number makes the balloon move to the left

If the balloon is too far to the left, we'll move it back to the right and randomize the Y position.

2) Adding Balloons

- Add the following code to the previous script...



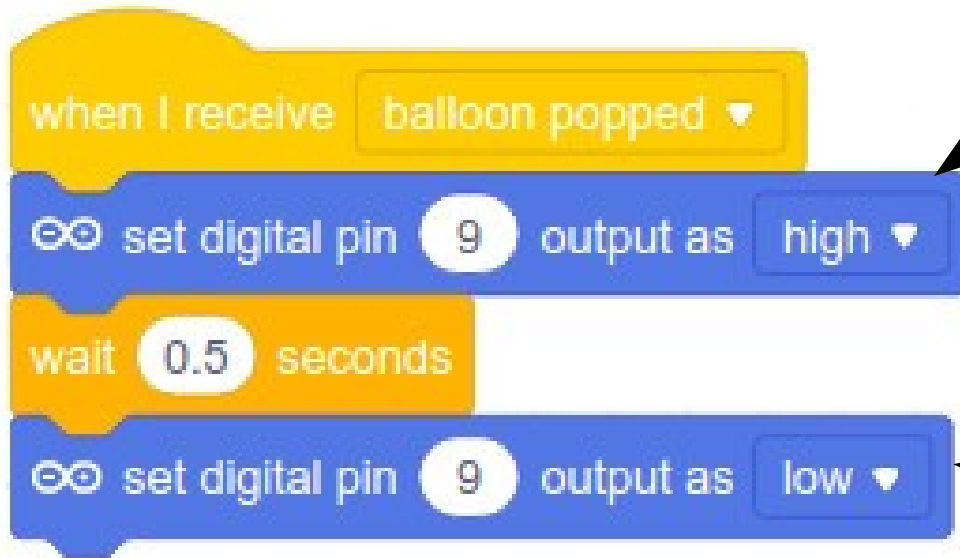
You'll need to create a "Score" variable first

We'll use this broadcast to inform the Arduino to flash the LED

Move the balloon back to the right and randomize the Y position.

2) Adding Balloons

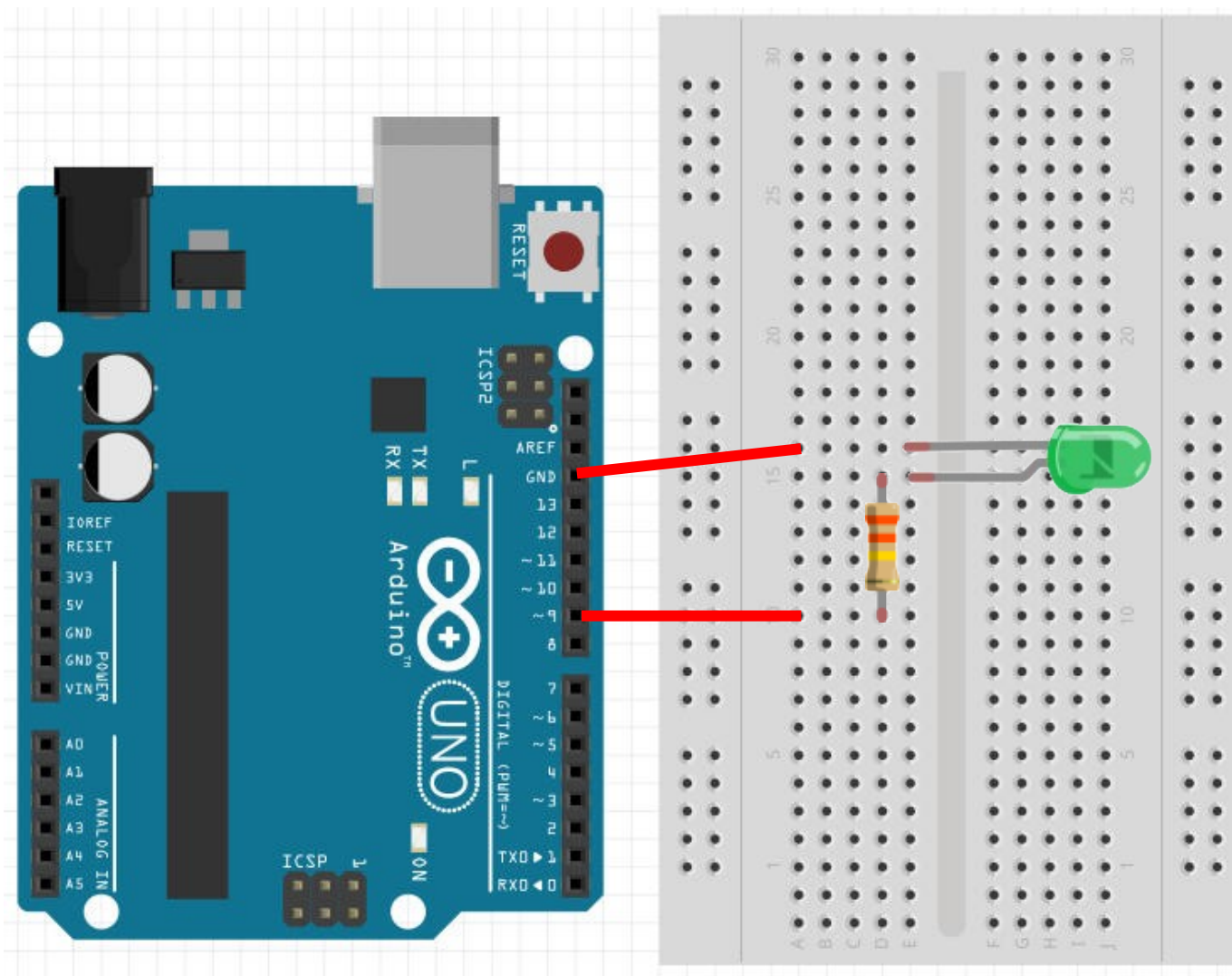
- Add this to the Arduino device



Turn on LED connected to pin 9, change this if you are using a different pin

Wait for a while then turn the LED off

Connecting an LED



Modify accordingly if you are not using pin 9

IMPORTANT!

- Short leg to ground
- Long leg to resistor

Challenge

2) Add in a Red LED and make it flash if the plane misses the balloon.

Tips

- Use a second resistor for the Red LED.
- Two LEDs cannot share a resistor, unless the LEDs are exactly the same voltage

3) Adding Clouds

- The clouds script is the same as the balloon, but without detecting contact with plane



Challenge

3) A parallax effect can simulate depth and improve the animation quality of your game. Read up on “parallax effect in games” and try to implement it on your clouds

Discuss

- How can you use games and the Arduino to encourage people to exercise more?

Copyright

- Created by A Posteriori LLP
- Visit <http://aposteriori.com.sg/> for more tips and tutorials
- This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

