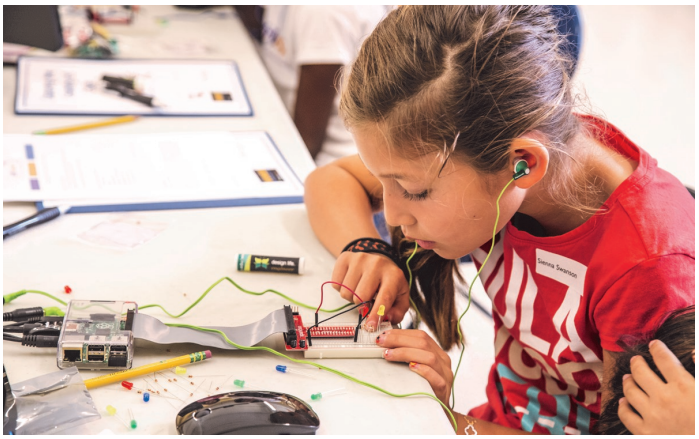
Best Practices for Using the Raspberry Pi Projects Kit in a Classroom or Out-of-School Setting



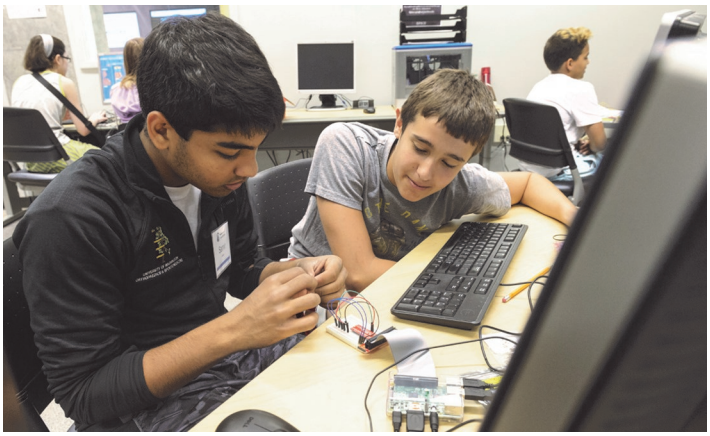
In the interest of maximizing student engagement and learning, Science Buddies has compiled a list of best practices for using the Raspberry Pi Projects Kit with groups of students in and out of the classroom.

Student Audience

Ages:	3rd-10th grade (ages 8-16)
Group size:	1-3 students
Experience Level:	Beginner
Minimum Time:	60 minutes
Maximum Time:	20 hours, cumulative

Skills Gained

- Programming
- Circuit building
- Problem solving
- Art & creativity
- Team communication (if working in teams)
- Self confidence



Logistics

Training: Adult mentors or educators who are helping out will feel most comfortable if they have had the chance to use the Raspberry Pi Projects Kit themselves first. At minimum, have mentors watch the videos for each project on the Science Buddies YouTube channel.

Mentors explore the kit ahead of time—at least watch the videos.

Adult help: For 5th grade and under, a ratio of 1 adult mentor for every 8 students is ideal. This can be increased for older students or for younger students who already have experience programming in Scratch and/or building circuits. If this ratio is not possible, good results can be achieved by carefully setting student expectations (*see next page*).

Have enough mentors. Set expectations for the event.

Session time: The more projects the students get to work on, the stronger their skills will become. However, the kit can be used as a one-time introduction for as little as an hour (expect to get through only the first project) or for several hours (more projects will be finished, scaling in number with student age, experience, and the amount of time spent on personalizing their projects). For maximum skill building, the kit can be used over multiple days for up to 20 cumulative hours of curriculum. Engagement over multiple days allows students to do all eight projects, explore variations on the projects, and ultimately use their sum learning to create an innovative project of their own.

For maximum skill building, do all eight activities.

Hooking up Raspberry Pi: Students can gain self-confidence and insight into the hardware by hooking the Raspberry Pi up themselves. If time allows, simply play the setup video and have students follow along. Students doing the setup works best when there is enough time to do the setup *and* several fun projects or when students will be using the kit over multiple sessions.

Emphasize coding and circuit building first. Incorporate setup skills if time allows.

If the session is:

<2 hours: pre-set up the Raspberry Pi and have students jump directly into the projects to maximize their time working on programming, circuit building, and problem solving skills.

>2 hours: have students hook up the Raspberry Pi.

Multiple days (75 minutes+ per day): have students hook up the Raspberry Pi.



Audio: Consider headphones or earbuds for each student so that they can go through the teaching videos at their own pace without adding a lot of noise to the room. If students are working as teams, you may need headphone splitters for the audio.

Earbuds or headphones for everyone.



Implementation

Setting student expectations: Students should know that they *will* make mistakes and that this is normal. Set up an expectation that they will work to problem solve on their own *before* they come to an adult. This promotes problem solving skills, communication, attention to detail, independence, and self-confidence. If a project is not working for them, ask the students to take these three steps:

- Troubleshoot your own project. Look at the troubleshooting section of the onscreen instructions for your project. Carefully go back over the circuit and the Scratch program looking for errors.
- Ask another student/team for help. Think through the problem and try solutions together.
- If all else fails, ask an adult for help. Be patient. Problem solving takes time and effort.

Adult role: Adults can facilitate student problem solving. Ask students to seek adult help only after they have tried to solve the problem themselves. If they have tried and still need assistance, adults should lead the students to the answer by asking them questions. Sometimes an adult mentor may not know the answer – this is fine and an excellent learning opportunity for students. Adults can simply state that they do not know the answer and invite the student(s) to problem solve with them. It is helpful for all adults to have a printed copy of the final breadboard diagram and program for each project. (These are provided at the end of this guide.)

Help students troubleshoot by asking questions.



Teamwork: Students learn the most if they are hands-on with the kit. To make sure *every* student is getting a chance to be hands-on, it is best to make teams of no more than three students. Within a team, students should be fairly well matched in age and starting skill level so that no one student dominates. To further facilitate shared responsibility, have students rotate roles for each project. Possible roles include:

Teams of 3 or less. Assign team roles.

- Software Engineer: responsible for writing the Scratch code for the project.
- Electrical Engineer: responsible for building the circuit for the project.
- Project Manager & QA Engineer: responsible for keeping everyone on task, settling disagreements about how

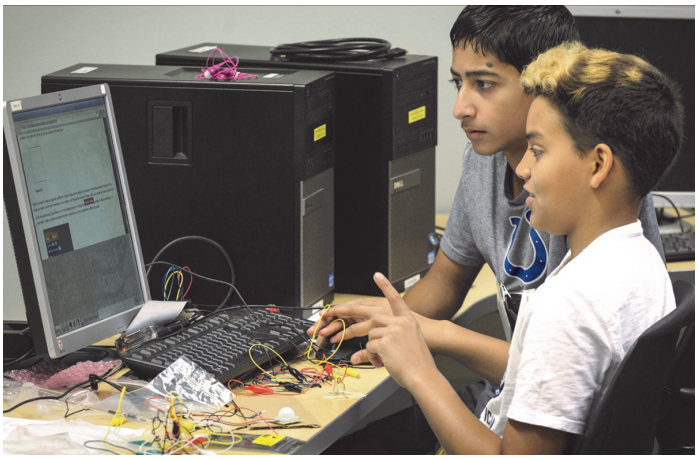
to proceed, and error checking the code and circuit. In projects with an art component, they are also the main creator of the art.

Career connections: If you are using roles within student teams, this is an opportunity to discuss what those careers are like in the real world. More information on each of the careers is available in Science Buddies' online career profiles.

Connect activities to real-world career paths.

Workflow: The on-screen resources mean that each team of students can go through the projects at their own pace. If you want everyone to work on the same project at the same time, encourage those that finish earlier to go back and try some of the variations suggested in the project and put their own creative twist on the project.

Help students troubleshoot by asking questions.



Further Help Resources

The following Science Buddies pages and resources may be helpful:

- Raspberry Pi Projects Kit videos at YouTube (playlist): www.sciencebuddies.org/rpivideos
- Creative Raspberry Pi Projects for Beginners (kit overview): www.sciencebuddies.org/raspberry-pi-projects-kit
- Raspberry Pi Projects Kit FAQ: www.sciencebuddies.org/raspberry-pi-projects-kit#help
- Kit Purchasing: store.sciencebuddies.org/SPF-6000-KIT/raspberry-pi-projects-kit.aspx

Development of Best Practices



The best practices described in this document are the culmination of a year-long collaboration with our partners: Purdue University's Women in Engineering Program and girlSPARC. Funding for the collaboration and publishing of the resulting conclusions was generously provided by Google RISE Awards and Sparkfactor.org.

Purdue University's Women in Engineering Program (WIEP)

Purdue University's Women in Engineering Program (WIEP) helps women and girls discover their inner engineer. From mentoring to career development, WIEP continually encourages current and future women engineering students by providing interesting and engaging programming relevant to their lives. WIEP programs are a place to learn, discover and explore aspects of engineering and connect with others who are also interested. They are a place of encouragement, support, and positive perspectives for those who are interested in following their dreams by pursuing an engineering degree.

girlSPARC

girlSPARC™ (A Girls Science Plus Arithmetic Club) is dedicated to increasing the number of girls exposed to science, technology, engineering and math (STEM) in grades 2-6. Our goal is to inspire girls to embrace STEM in a real world, hands-on setting. The program helps girls build problem solving skills, increases their confidence and exposes them to these subjects in a fun and collaborative environment.

Google RISE Awards

The Google RISE Awards is an annual grant program for informal education organizations around the world that promote computer science for K-12/pre-university age youth.

Sparkfactor.org

Sparkfactor.org is the nonprofit division of Sparkfactor Design. In keeping with our Maker roots, our programs center around technological and philanthropic innovation and STEM Education for underprivileged communities.

Appendix A

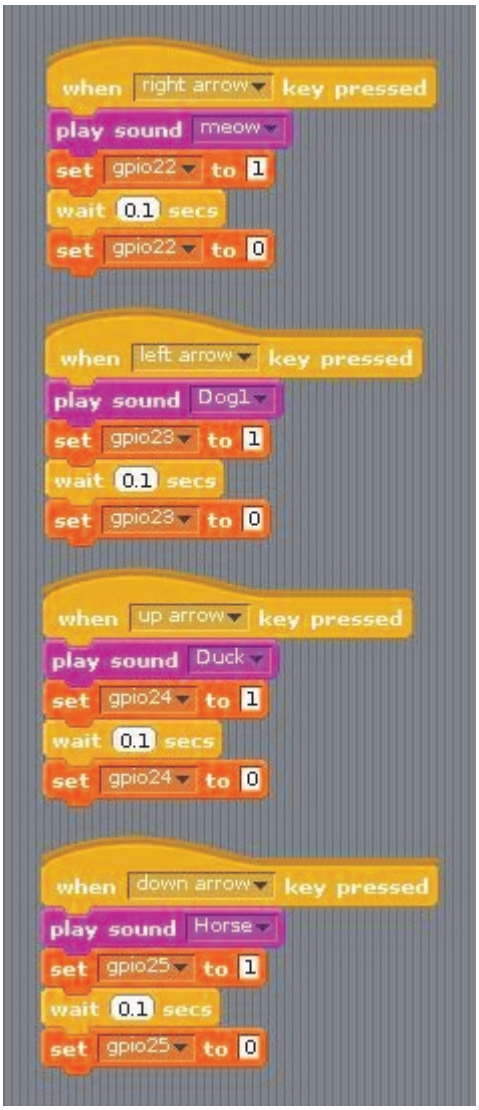
Science Buddies Raspberry Pi Kit Instructor Reference Sheet

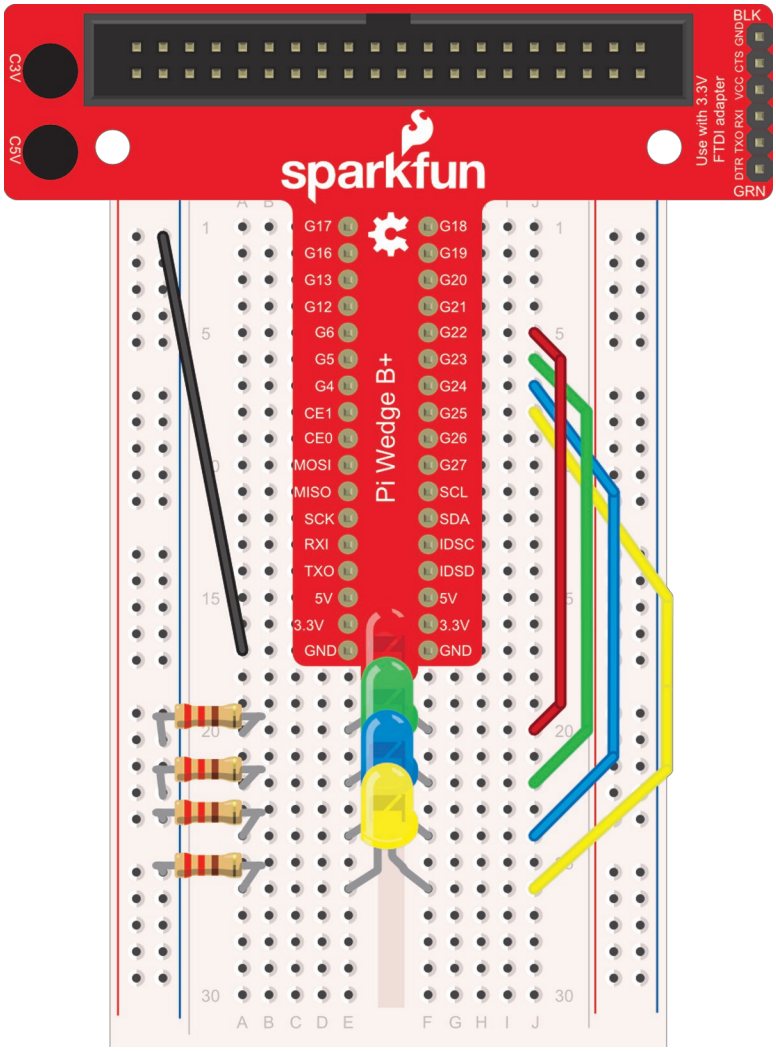
Version 1.0



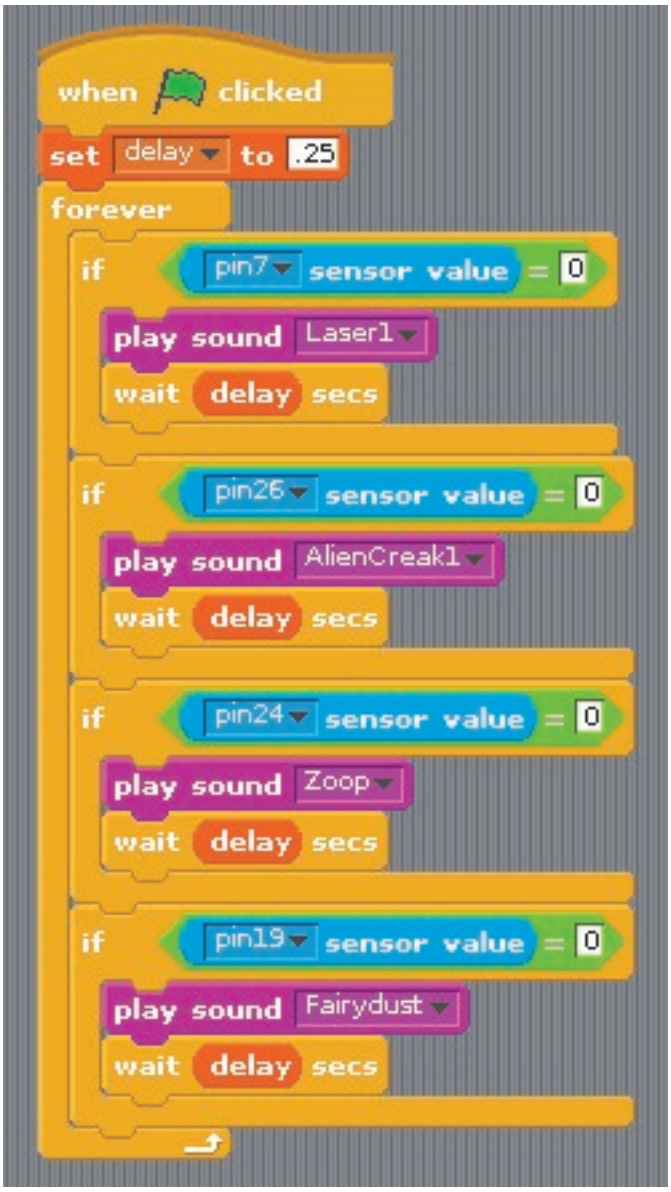
The following pages show the *final* Scratch program and circuit diagram for each of the eight Raspberry Pi Projects Kit projects. These reference screens do not show any intermediate steps.

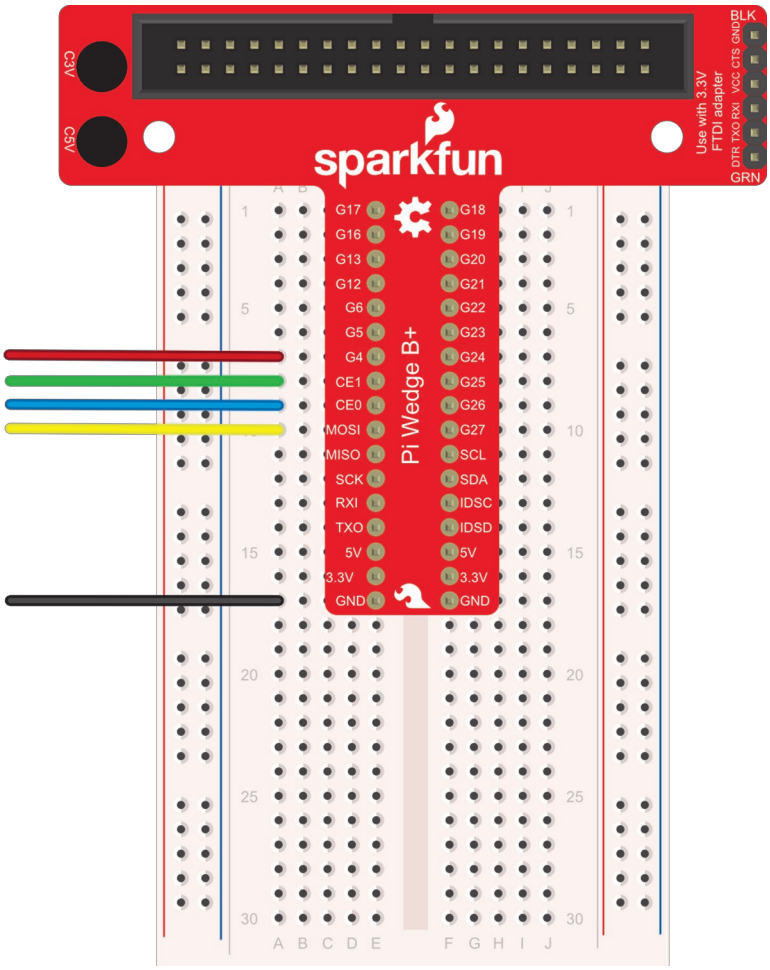
Project 1 – Light-up Piano

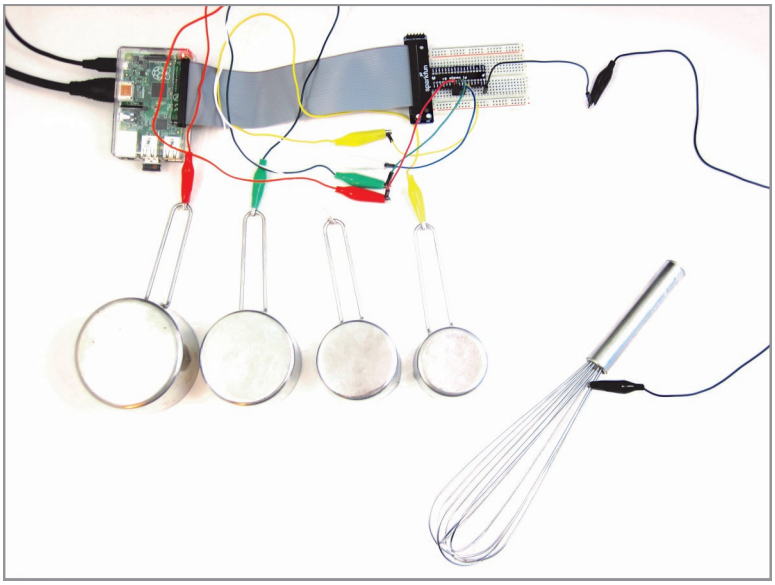




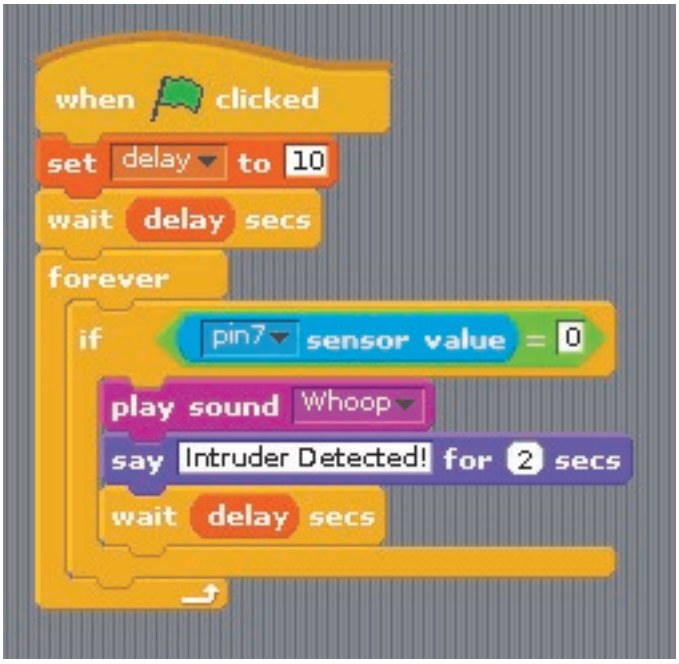
Project 2 – Electronic Drumset

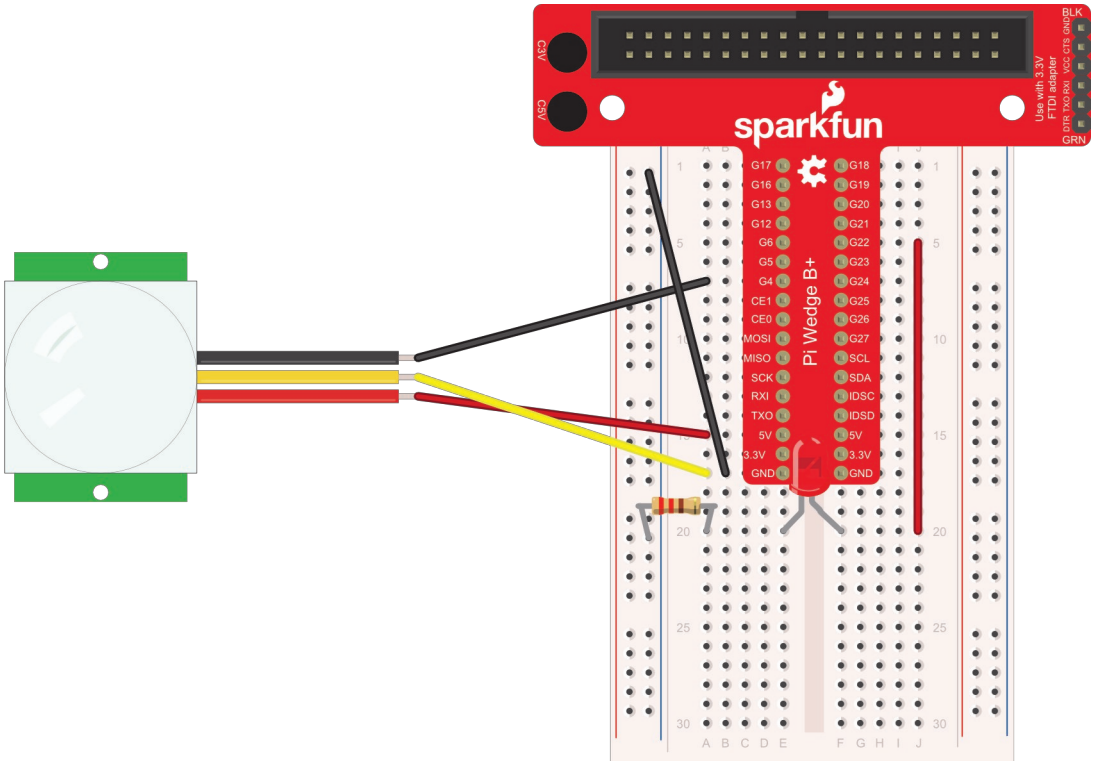




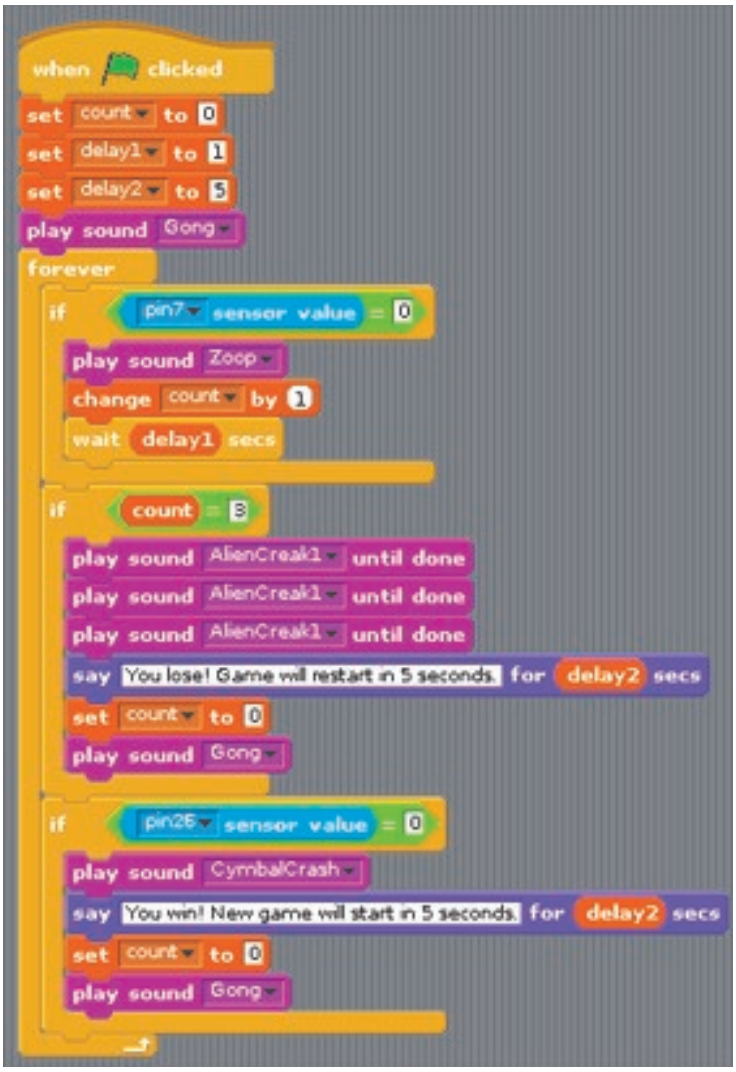


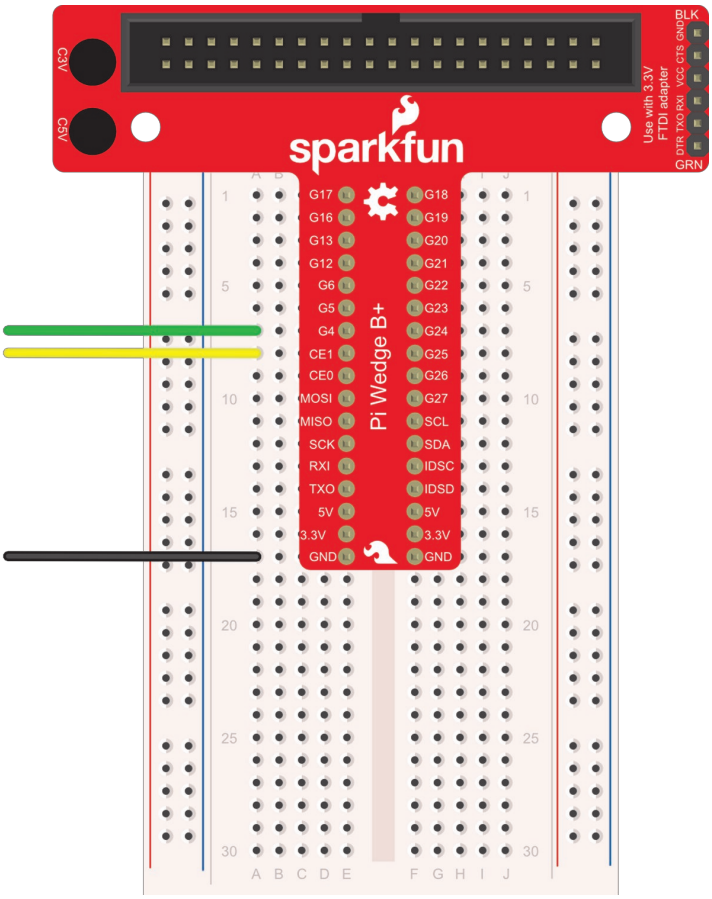
Project 3 – Invader Alert

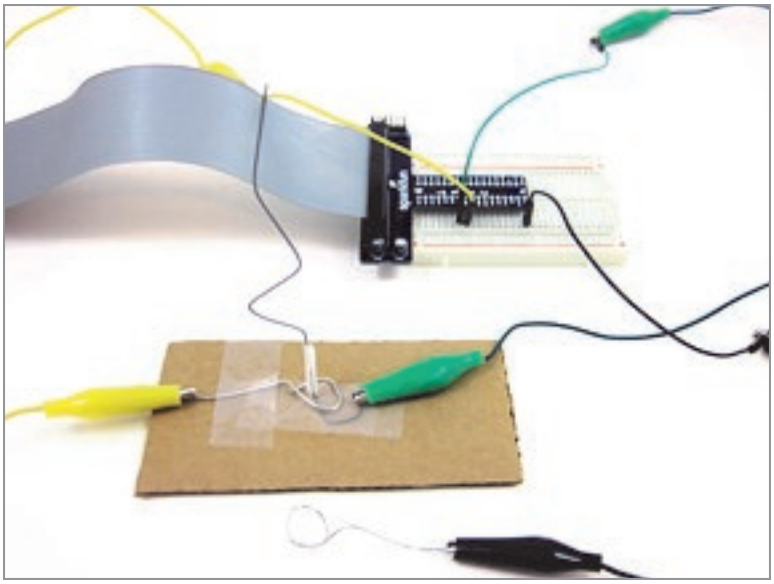




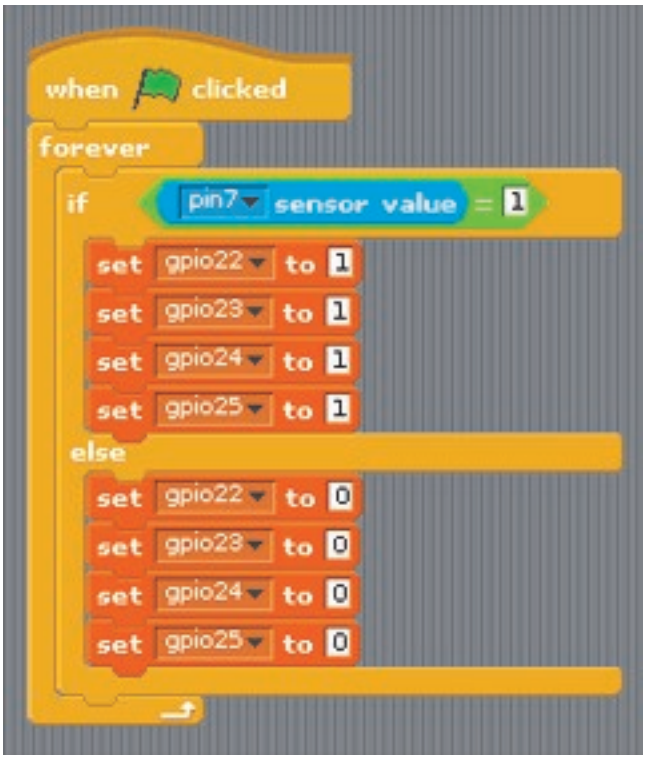
Project 4 – Buzzer Game

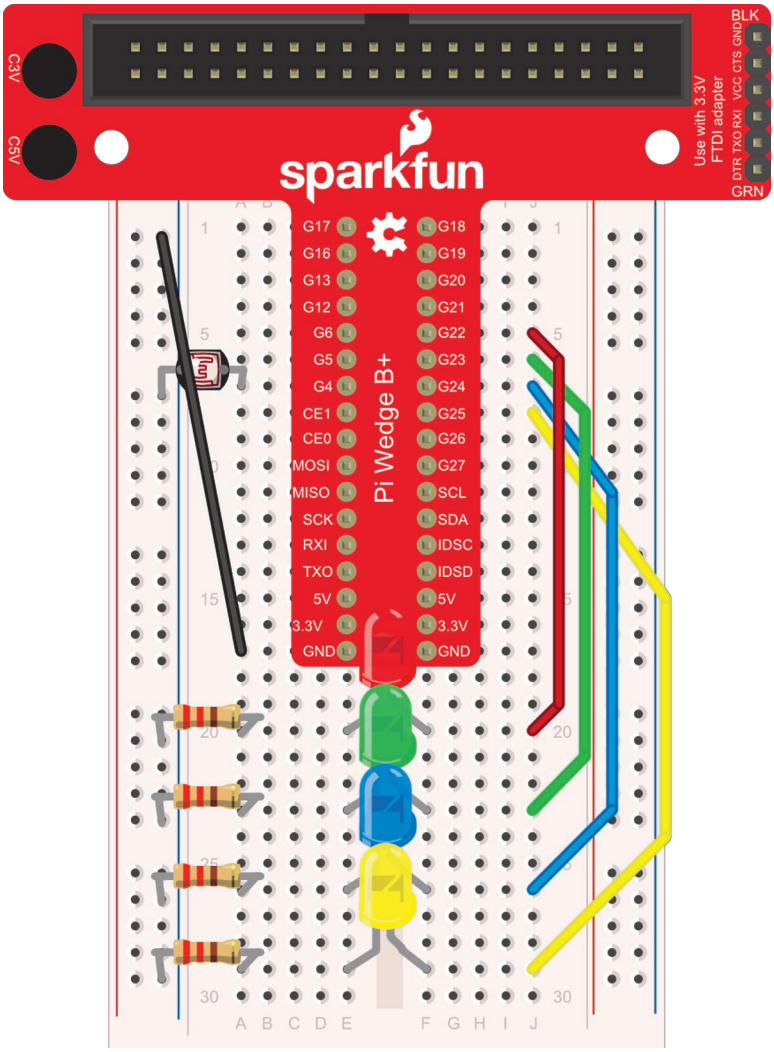




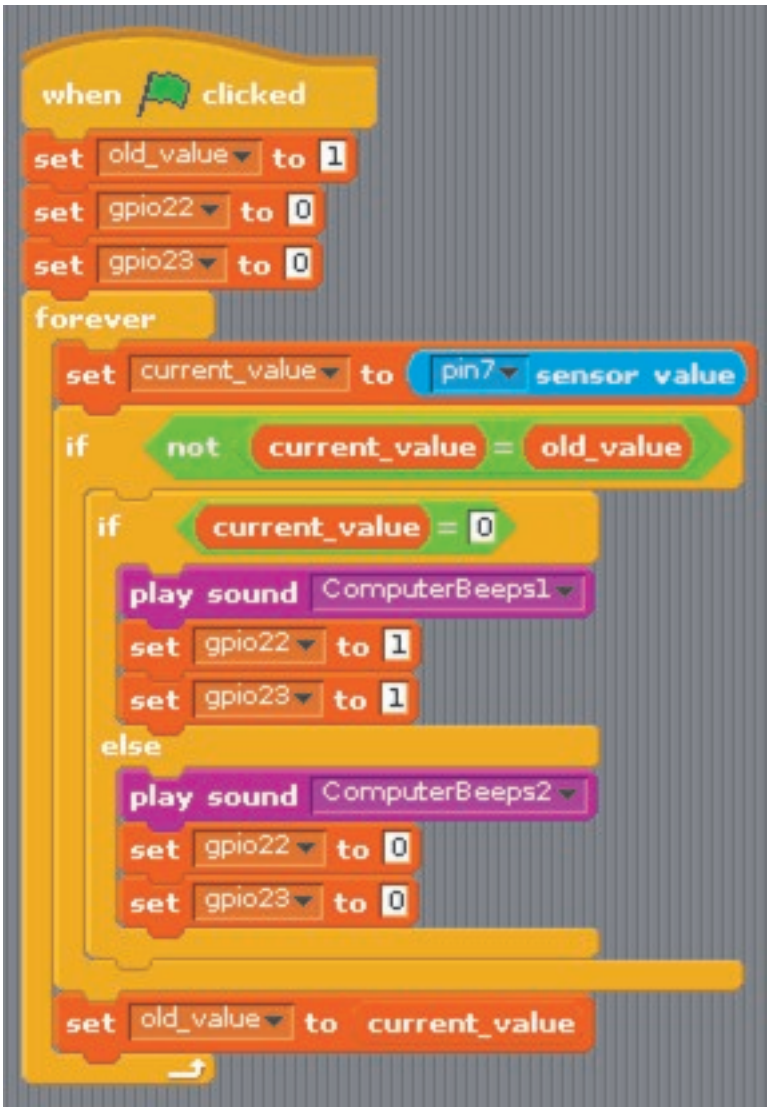


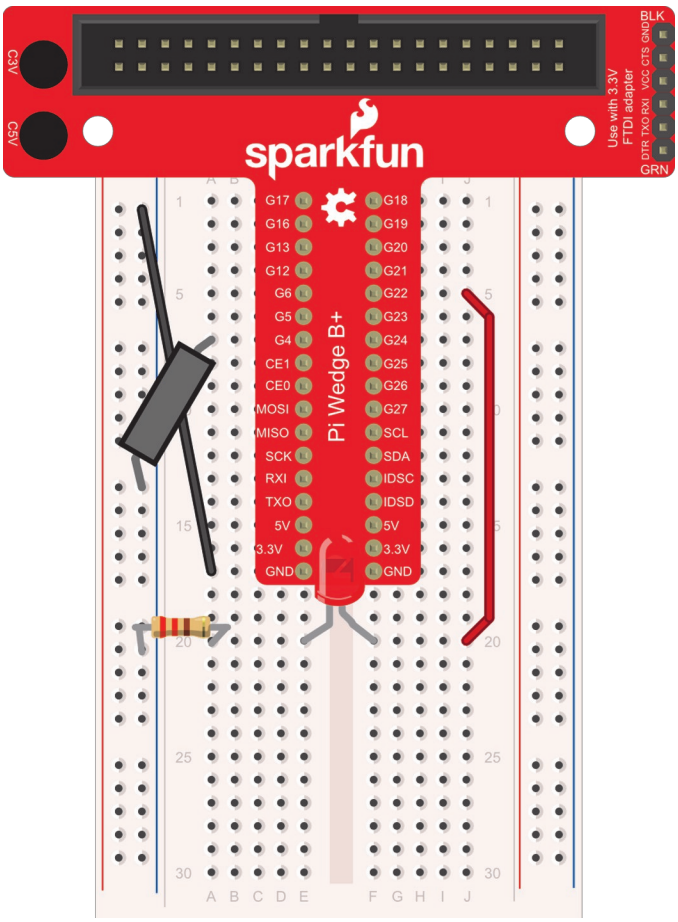
Project 5 – Starry Night



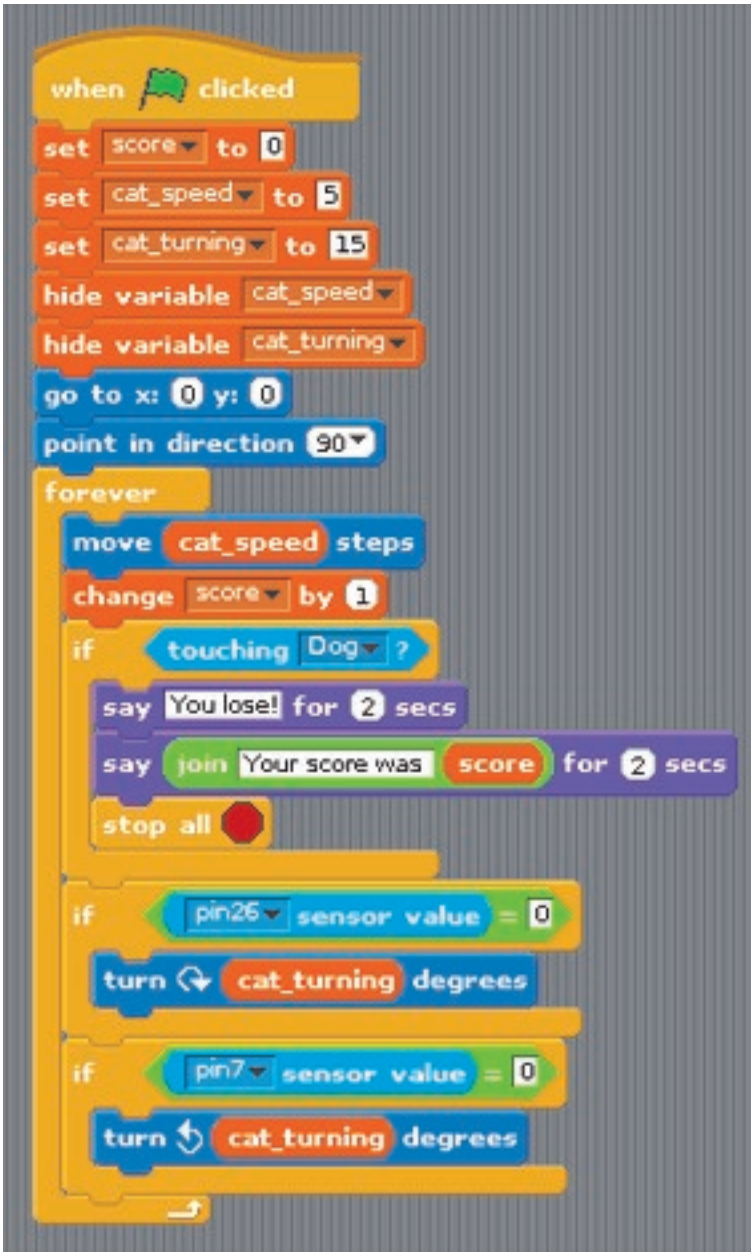


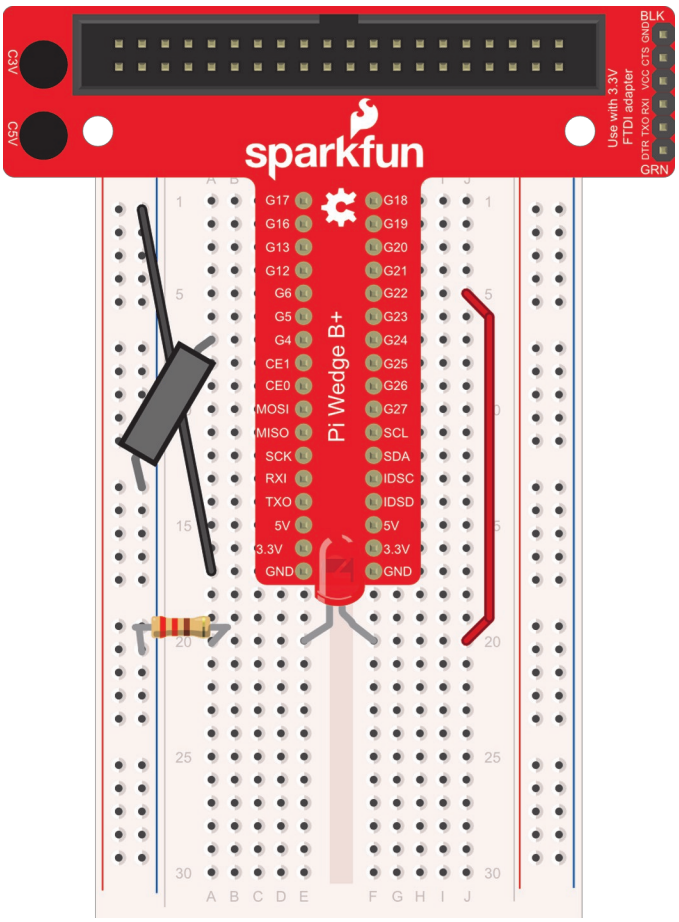
Project 6 – Digital Puppet





Project 7 – Make Your Own Video Game





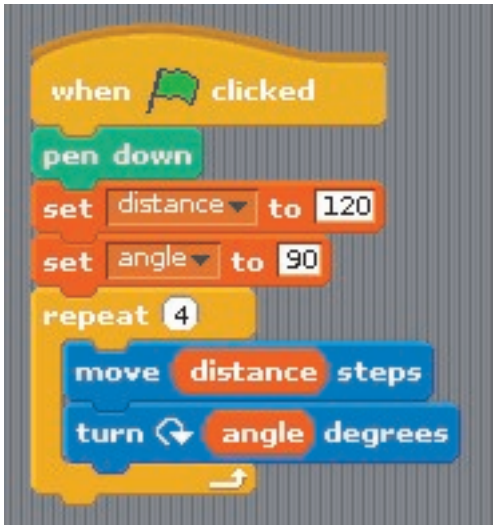
This project has a separate script for the dog sprite:



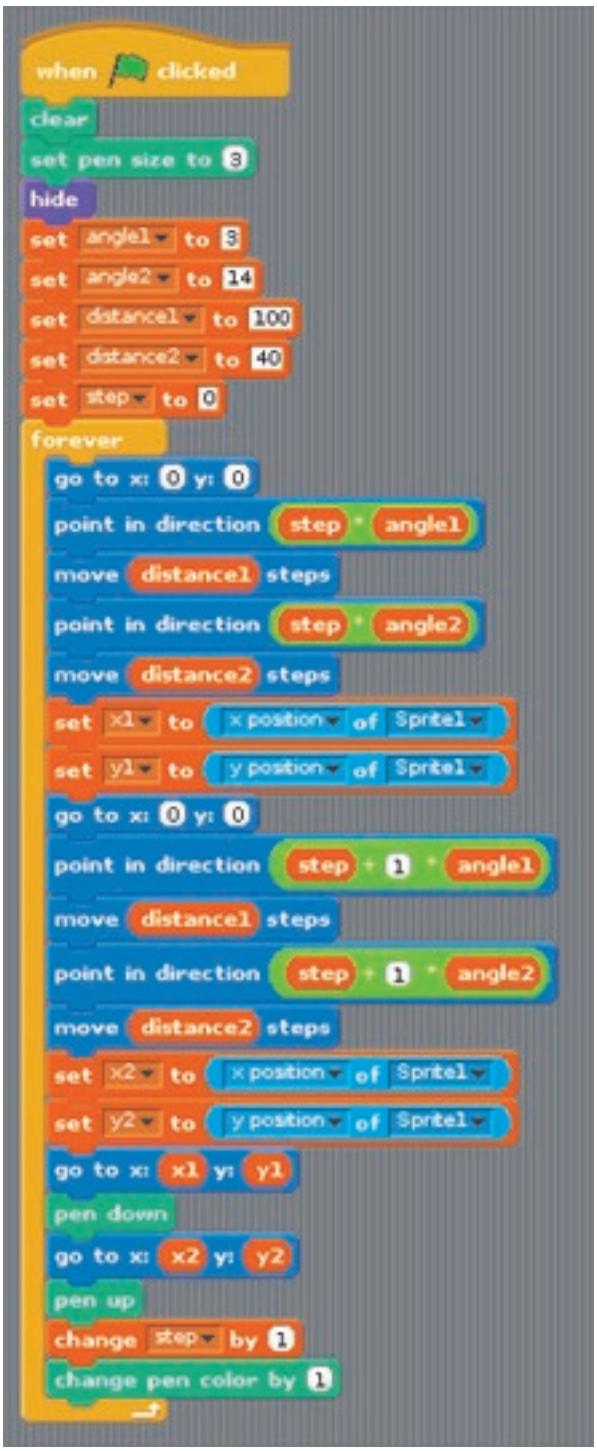
Project 8 – Spirograph

Note: unlike most of the other projects, this one has students step through progressively more complicated Scratch programs, to go from drawing a simple square all the way up to a Spirograph. The circuit is the same as in Project 7.

Code for square:



Code for Spirograph:



Note: there is a line of code misplaced in Figure 8.15 in this project. The line "set angle to 360/number_of_sides" should appear *inside* the first "repeat" loop, not before it. This allows the number of polygon sides to update each time you press a button.