

كتاب تجارب معمل الدوائر الإلكترونية

أولاً: بيانات المعمل الأساسية

اسم المعمل: الدوائر الإلكترونية

القسم العلمي: هندسة الإلكترونيات والاتصالات

المشرف: أ.د./ شريف السيد كشك

مهندس المعمل: د./ مجدي محمد فاضل

أمين المعمل: أ./ محمد يوسف مصطفى

التليفون: 1301

الموقع بالنسبة للكلية: الدور الثالث – مبنى المعامل البحرية - بالكلية

مساحة المعمل: 124 متر²

ثانياً: قائمة بالأجهزة والمعدات الموجودة بالمعمل:

Serial Number	العدد	اسم الجهاز	م
	1	Motorola Codex 3266 single channel data modem	1
	1	Oscilloscope Channel Digital	2
	1	Sound System 2 power Amplifier 500W	3
	1	لوحة تجارب لتنشيت الدوائر الإلكترونية (TestBoard)	4
	1	شاشة عرض	5
	1	DC-to-AC Converter 600W	6
	1	Laser Printer HP 1320	7
	1	Color Printer HP 1280	8
	1	Scanner HP 2400	9
	1	كاميرا تليفزيون ناشيونال كاملة	10
	1	ميكروفون دايكو 2 خط أمبليفيير FM	11
	1	كاميرا زينيث	12
	1	كلوز (UP) 52	13
	1	CPU Intel 7500, 2GB Memory	14
	1	FPGA unit	15
	1	Digital Video Camera per DVD	16
	1	Micro-Program	17
	1	Micro Electronic	18
	1	Sanyo Data show 2000 Ansi XVGA	19
	1	Spartan-3A FPGA	20
	1	Microelectronika MMB for PIC 32	21
	1	Laptop Dell 1 GB RAM, 80 MB HDD, 15.4" Display	22

رابعاً: الخدمات المجتمعية التي يؤديها المعمل:

- عدد المستفيدين من المعمل: جميع طلاب مراحل البكالوريوس والماجستير والدكتوراه بالقسم والأقسام الأخرى.
 - الجهات التي تتعاون مع المعمل: جميع أقسام الكلية
 - الدخل السنوي للمعمل: لا يوجد دخل مادي
 - الجهات الممولة لأنشطة المعمل: لا يوجد جهات تمويل
 - المشاريع التنافسية التي يشارك فيها المعمل: مشاريع طلاب مرحلة البكالوريوس وكذلك المسابقات على مستوى المحافظة والجمهورية.
-

خامساً: الخدمات الطلابية التي يؤديها المعمل:

- عدد الطلاب المستفيدين من المعمل: جميع طلاب مراحل البكالوريوس والماجستير والدكتوراه بالقسم والأقسام الأخرى.
- الأقسام العلمية المستفيدة من المعمل: جميع أقسام الكلية
- الفرق الدراسية المستفيدة من المعمل: جميع أقسام الكلية
- المقررات الدراسية التي تستفيد من المعمل: جميع أقسام الكلية
- الأنشطة الطلابية داخل المعمل: مشاريع بحثية ومحاضرات
- عدد طلاب الدراسات العليا المستفيدين من المعمل: طلاب القسم والأقسام والأخرى.
- عدد الرسائل العلمية التي تمت في المعمل: العديد
- عدد الدورات التدريبية التي تمت في المعمل: العديد
- المسابقات العملية التي شارك فيها طلاب من المستفيدين من المعمل: العديد

ثالثاً: قائمة بالتجارب التي تؤدي داخل المعمل:

FPGA Projects

This page presents FPGA projects on fpga4student.com. The first FPGA project helps students understand the basics of FPGAs and how Verilog/ VHDL works on [FPGA](#).



FPGA Projects



on FPGA4student



Some of the FPGA projects can be FPGA tutorials such as [What is FPGA Programming](#), [image processing on FPGA](#), [matrix multiplication on FPGA](#) Xilinx using Core Generator, [Verilog vs VHDL: Explain by Examples](#) and [how to load text files or images into FPGA](#). Many others FPGA projects provide students with full Verilog/ VHDL source code to practice and run on FPGA boards. Some of them can be used for another bigger FPGA projects.

Following are the [FPGA](#) projects on fpga4student.com:

1. [What is an FPGA? How does FPGA work?](#)
2. [Basys 3 FPGA OV7670 Camera](#)
3. [How to load text file or image into FPGA](#)

4. [Image processing on FPGA using Verilog](#)
5. [License Plate Recognition on FPGA](#)
6. [Alarm Clock on FPGA using Verilog](#)
7. [Digital Clock on FPGA using VHDL](#)
8. [Simple Verilog code for debouncing buttons on FPGA](#)
9. [Traffic Light Controller on FPGA](#)
10. [Car Parking System on FPGA in Verilog](#)
11. [VHDL code for comparator on FPGA](#)
12. [Verilog code for Multiplier on FPGA](#)
13. [N-bit Ring Counter in VHDL on FPGA](#)
14. [Verilog implementation of Microcontroller on FPGA](#)
15. [Verilog Carry Look Ahead Multiplier on FPGA](#)
16. [VHDL Matrix Multiplication on FPGA Xilinx](#)
17. [Fixed Point Matrix Multiplication on FPGA using Verilog](#)
18. [Verilog Divider on FPGA](#)
19. [VHDL code for Microcontroller on FPGA](#)
20. [VHDL code for FIR Filter on FPGA](#)
21. [Verilog code for Digital logic components on FPGA](#)
22. [Delay Timer Implementation on FPGA using Verilog](#)
23. [Single-Cycle MIPS processor on FPGA using Verilog](#)
24. [FIFO Verilog Implementation on FPGA](#)
25. [FIFO VHDL Implementation on FPGA](#)
26. [Verilog D Flip Flop on FPGA](#)
27. [Comparator Design on FPGA using Verilog](#)
28. [D Flip Flop on FPGA using VHDL](#)
29. [Full Adder Design on FPGA using Verilog](#)
30. [Full Adder Design on FPGA using VHDL](#)
31. [Counters on FPGA with Verilog Testbench](#)
32. [RISC Processor Design on FPGA using Verilog](#)
33. [Verilog test bench for inout ports on FPGA](#)
34. [PWM Generator on FPGA using VHDL](#)
35. [Tic Tac Toe Game on FPGA using Verilog](#)
36. [VHDL code for ALU on FPGA](#)
37. [Verilog code for ALU on FPGA](#)
38. [Counter design on FPGA with VHDL test bench](#)
39. [Pipelined MIPS Processor on FPGA in Verilog \(Part-1\)](#)

40. [Pipelined MIPS Processor on FPGA in Verilog \(Part-2\)](#)
41. [Pipelined MIPS Processor on FPGA in Verilog \(Part-3\)](#)
42. [Verilog Decoder on FPGA](#)
43. [Verilog Multiplexers on FPGA](#)
44. [N-bit Adder Design on FPGA in Verilog](#)
45. [VHDL ALU on FPGA using N-bit Verilog Adder](#)
46. [VHDL Shifter on FPGA](#)
47. [Lookup Table VHDL example code on FPGA](#)
48. [Coprocessor VHDL Implementation on FPGA](#)
49. [Affordable Xilinx FPGA boards for beginners](#)
50. [Affordable Altera FPGA boards for beginners](#)
51. [What is FPGA Programming?](#)
52. [Verilog vs VHDL: Explain by Examples](#)
53. [VHDL code for clock divider on FPGA](#)
54. [Verilog code for clock divider on FPGA](#)
55. [How to generate a clock enable signal instead of creating another clock domain](#)
56. [VHDL code for debouncing buttons on FPGA](#)
57. [VHDL code for Traffic light controller on FPGA](#)
58. [Verilog code for PWM Generator on FPGA](#)
59. [VHDL code for a simple 2-bit comparator on FPGA](#)
60. [VHDL code for a single-port RAM](#)
61. [FPGA car Parking System in VHDL using Finite State Machine \(FSM\)](#)
62. [VHDL code for MIPS Processor](#)
63. [Verilog code for Sequence Detector using Moore FSM](#)
64. [Full VHDL code for Sequence Detector using Moore FSM](#)
65. [\[FPGA Tutorial\] Seven-Segment LED Display Controller on Basys 3 FPGA](#)
66. [VHDL code for Seven-Segment Display on Basys 3 FPGA](#)
67. [\[FPGA tutorial\] How to interface a mouse with Basys 3 FPGA](#)
68. [Verilog Code for Ripple Carry Adder](#)
69. [How to Read Image into FPGA using VHDL](#)

FPGA4student will keep updating upcoming FPGA projects with full Verilog/VHDL source code. Hence, keep you up to date with FPGA projects on fpga4student.com. You can subscribe [here](#) to get FPGA projects directly to your inbox. You can also ask for help [here](#) if you encounter any difficulty in FPGA projects.

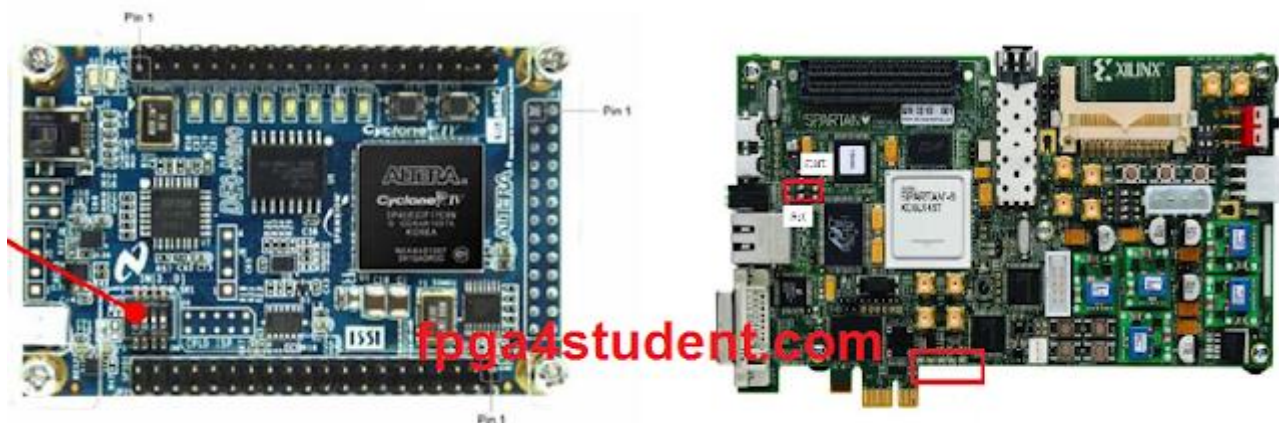
What is an FPGA?

What is [FPGA](#)? FPGA stands for Field Programmable Gate Array. Let's analyze the term:

1. **Field-Programmable**: An [FPGA](#) is manufactured to be easily reconfigured by developers, designers or customers. To program an FPGA as a specific configuration, Verilog HDL or VHDL (Hardware Description Language) is used as the standard language for FPGA programming.

2. **Gate-Array**: An FPGA consists of an array of programmable logic gates/ blocks such as AND, OR, XOR, NOT, memory elements, DSP components, etc., and reconfigurable interconnects which are to connect logic gates together for performing a specific function.

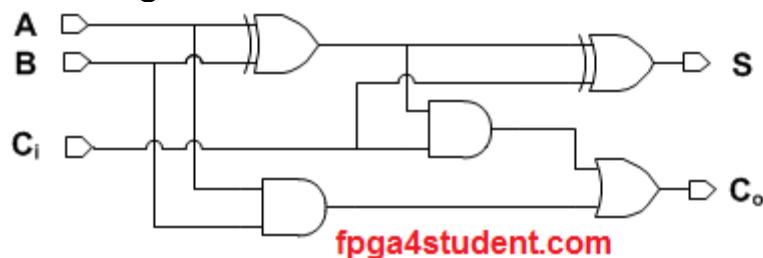
What is an FPGA?



Thus, FPGAs are nothing, but logic blocks and interconnects that can be programmable by [Hardware Description Languages](#) (Verilog HDL/ VHDL) to perform different complex functions.

In fact, [FPGAs](#) can be used to implement almost any DSP algorithm. Some FPGAs also obtain embedded soft-core processors such as Xilinx's MicroBlaze, Altera's Nios II, etc. so that we can use C, C++, etc. to program the processor like what we do with a microcontroller. Besides, the soft processors can communicate with hardware accelerators to speed up complex DSP operations so that we can obtain a better flexible embedded system for niche applications.

Let's take a very basic example on how to use an FPGA. Let's assume that you are designing a 1-bit full adder and you already obtained the logic diagram of the adder as shown in the figure below.



As mentioned above, there are necessary logic gates on FPGA such as XOR, AND and OR in order to implement the above adder on FPGA. To demonstrate the operation of the adder on FPGA, either Verilog or VHDL can be easily used to connect those gates together as shown in the logic diagram of the adder.

- **An example Verilog code for the adder:**

```
-- FPGA Project: What is an FPGA?
-- Verilog example code for Adder on FPGA
module fpga4student_adder(input A,B,Ci, output S,Co);
  wire tmp1,tmp2,tmp3; //FPGA projects
  xor u1(tmp1,A,B); // Verilog projects
  and u2(tmp2,A,B);
  and u3(tmp3,tmp1,Ci);
  or u4(Co,tmp2,tmp3);
  xor u5(S,tmp1,Ci);
endmodule
```

- **An example VHDL code for the adder:**

```
-- FPGA projects: What is an FPGA?
```

```

-- VHDL example code for adder on FPGA
library ieee;
use ieee.std_logic_1164.all;
entity fpga4student_Adder is
  port( A, B, Ci : in std_logic;
        S, Co : out std_logic);
end fpga4student_Adder;
architecture structural of fpga4student_Adder is
  signal tmp1, tmp2, tmp3: std_logic;
begin
  tmp1 <= A xor B;
  tmp2 <= A and B;
  tmp3 <= tmp1 and Ci;
  Co <= tmp2 or tmp3;
  S <= tmp1 xor Ci;
end structural;

```

The [Verilog/ VHDL](#) code for the adder simply does the "wiring" job by connecting the logic gates on FPGA together to be functional as a full adder. After verifying the code by simulation, we can synthesize, run the adder on FPGA and see how it works. For simulation, there are various available simulators that we can use, but for students, I would recommend you to use ModelSim of Mentor Graphics. You can see more simulators [here](#). For FPGA synthesis and FPGA programming, you can use Quartus II for Altera's FPGA Boards and Xilinx ISE or Vivado for Xilinx's FPGA Boards. As an FPGA designer for several years, here are five reasons why I love FPGA design:

1. Very fast on-chip (FPGA) demonstration

The top reason why I love [FPGA design](#) is that it is very fast to verify a design on FPGA. While ASICs could take several months only for tape-out and another latency for PCB design, everything is settled on FPGA and we just need to download the program file to FPGA using a specific FPGA programming tool and see how it works on FPGA. FPGA boards of Xilinx and Altera provide necessary IOs and additional support circuits such as

LCD, single LEDs, 7-segment LEDs, communication ports (USB, UART, VGA, HDMI, PS2, FMC, etc.), ADCs, DACs, CODEC, etc. so that FPGA can easily communicate with other chips for the verification process.

2. Simple and fast design process on FPGA

Another great thing to say about FPGA is that the design process is pretty simple and really easy to learn. The design flow for ASICs is very complicated and time-consuming since it needs a lot of complex steps for designing, verification, and implementation. On the other hand, FPGA design process mostly avoids sophisticated and time-consuming steps like Floor-planing, Timing Analysis, Physical Implementation, etc. because FPGA is already a characterized and verified chip. Of course, when needed, FPGA vendors also provide necessary tools for floorplanning and timing analysis to enable users optimizing performance for niche very-demanding designs. In fact, FPGA design flow only takes several steps such as HDL design and coding, functional simulation, synthesis, timing or post-synthesis simulation if needed, and Place And Route. Furthermore, many FPGA design tools are free and very easy for users to learn and design. FPGA vendors provide free user guides and tutorials to facilitate user's learning process. It could take very short time for students to be familiar with FPGA design if they have a good background in digital logic design.

3. FPGA's programmability

The highlight feature of FPGA we obviously could not omit is its programmability. While ASICs or microcontrollers are fixed in term of hardware (it can be programmable at the software level), FPGAs can be programmable at the hardware level. We can program FPGAs to perform

almost any digital complex functionality and reconfigure it to whatever we want in the future if needed. FPGAs can be programmed as a microprocessor, a microcontroller, DSPs, VGA controllers, digital filters, etc.

4. FPGA's high performance

Another superb feature in [FPGA design](#) is high performance. While processor-based ASICs or DSPs are sequential executed, FPGAs exploits the hardware parallelism to obtain a breakthrough performance for demanding designs. Thus, FPGAs provides faster implementations that processor-based ASICs could not match. The FPGA's parallelism can be effectively exploited to implement digital signal processing algorithms in order to speed up the processing time.

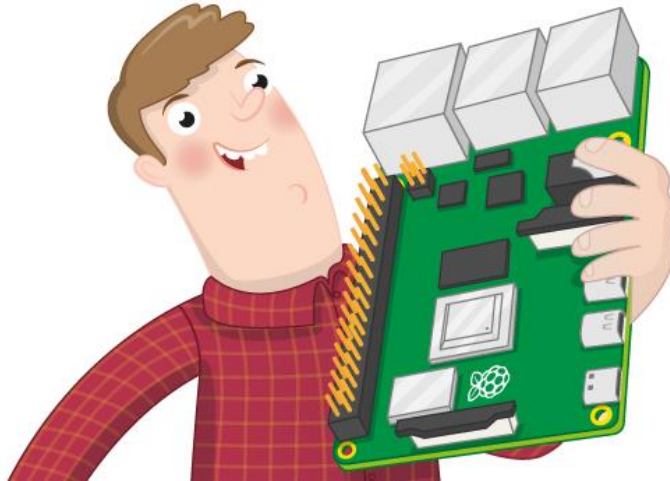
5. FPGA's flexibility

FPGAs are more and more flexible for designers to make their own applications. As mentioned above, FPGA vendors provide their own soft processors such as Xilinx's MicroBlaze, Altera's Nios II, etc. so that designers can be more flexible in design and programming process. Indeed, on FPGA, you can use the soft processors with the software-level-reconfigurable capability(C, C++, etc.) for low and average speed applications, and FPGA hardware accelerators with the hardware-level-reconfigurable capability(Verilog/ VHDL) for high-speed operations. Thus, designers can obtain a suitable embedded system which meets their design requirements. For example, when designing an embedded real-time tracking system, designers can use the soft processor for camera interface and FPGA hardware accelerators for tracking processing.

Get started with Raspberry Pi

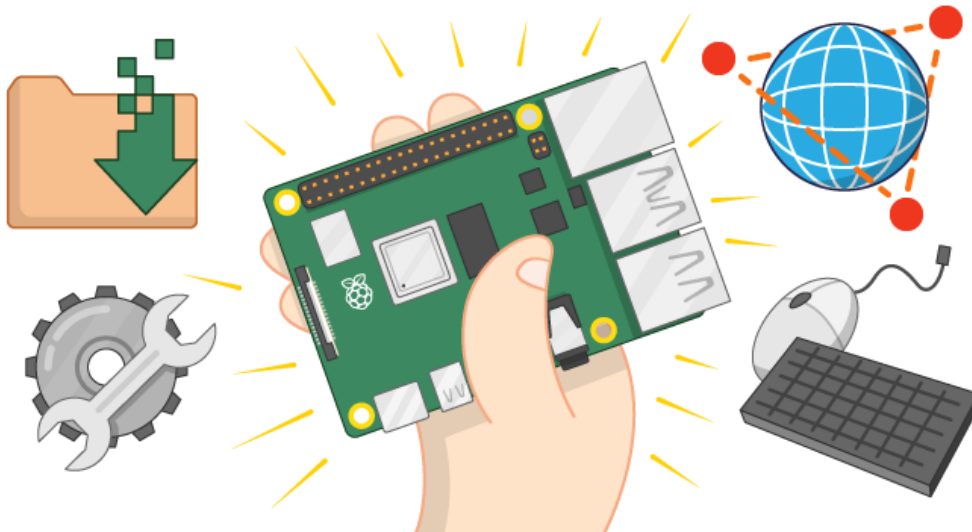
Set up your Raspberry Pi and see what it can do!

Setting up your Raspberry Pi



How to set up and start your Raspberry Pi for the first time
Raspberry Pi

Using your Raspberry Pi



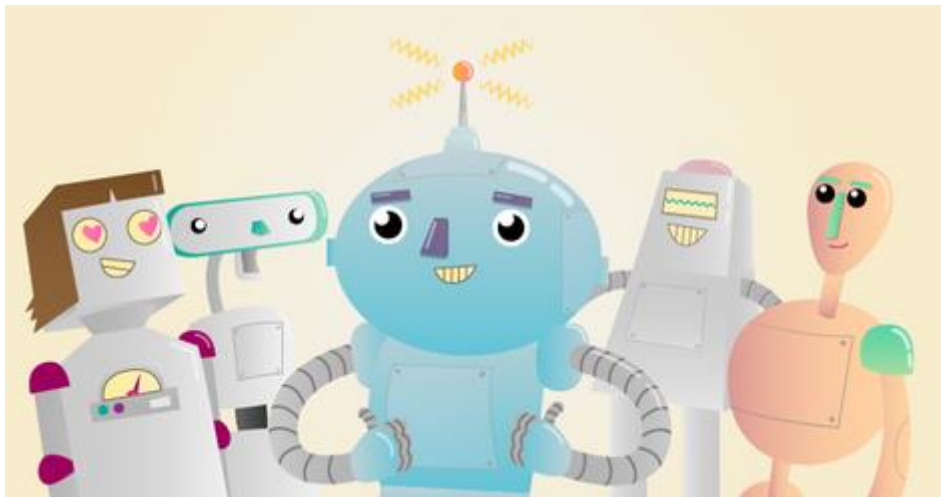
How to configure, update, and navigate your Pi once it's
set up Raspberry Pi

Getting started with the Camera Module



Take pictures and video with the Raspberry Pi Camera Module and Python Pi Camera Module, Electronic components, Python

Robot antenna



Control a robot's antenna light with a Raspberry Pi and code blocks Electronic components, Raspberry Pi, Scratch

Push Button Stop Motion



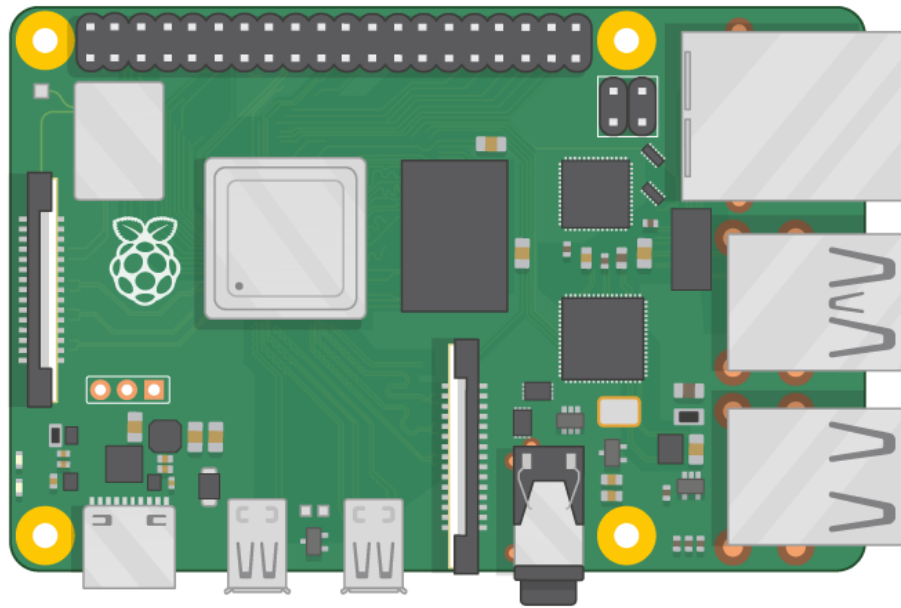
Make your own stop motion animation rig with a push button, using Python Pi-camera and GPIO Electronic components, Python

What you will need

Which Raspberry Pi?

There are several [models of Raspberry Pi](#), and for most people Raspberry Pi 4 Model B is the one to choose. Raspberry Pi 4 Model B is the newest, fastest, and easiest to use.

Raspberry Pi 4 comes with either 1GB, 2GB, or 4GB of RAM. For most educational purposes and many hobbyist projects, 1GB is enough; for use as a desktop computer, we recommend 2GB.

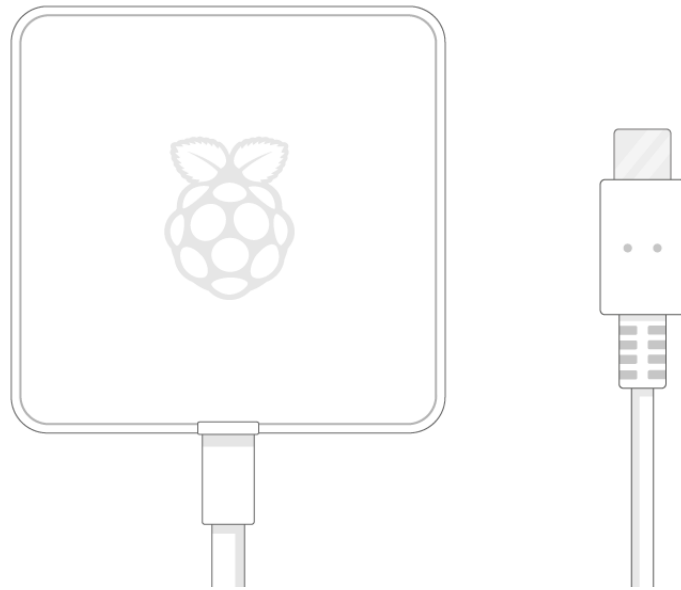


Raspberry Pi Zero and Zero W are smaller and require less power, so they're useful for portable projects such as robots. It's generally easier to start a project with Raspberry Pi 4, and to move to Pi Zero when you have a working prototype that a smaller Pi would be useful for.

If you want to buy a Raspberry Pi, head to rpf.io/products.

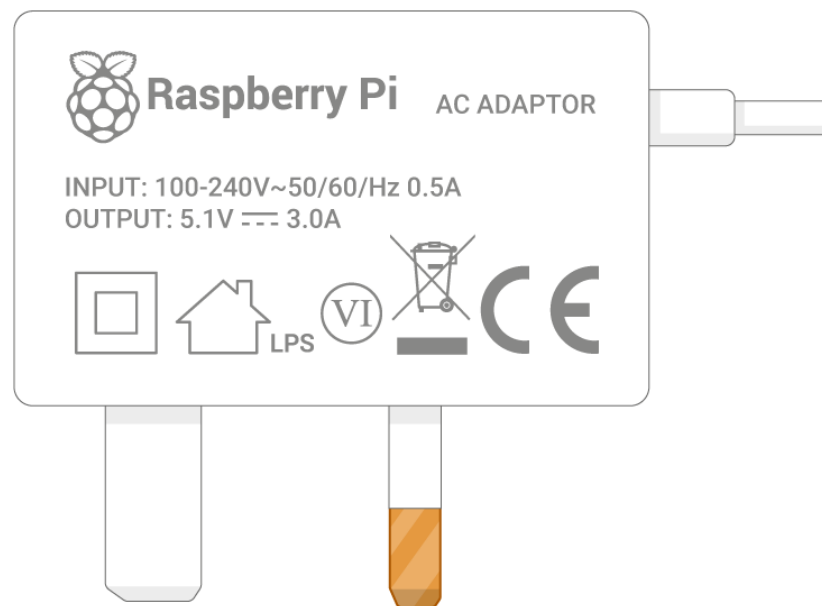
A power supply

To connect to a power socket, all Raspberry Pi models have a USB port (the same found on many mobile phones): either USB-C for Raspberry Pi 4, or micro USB for Raspberry Pi 3, 2 and 1.

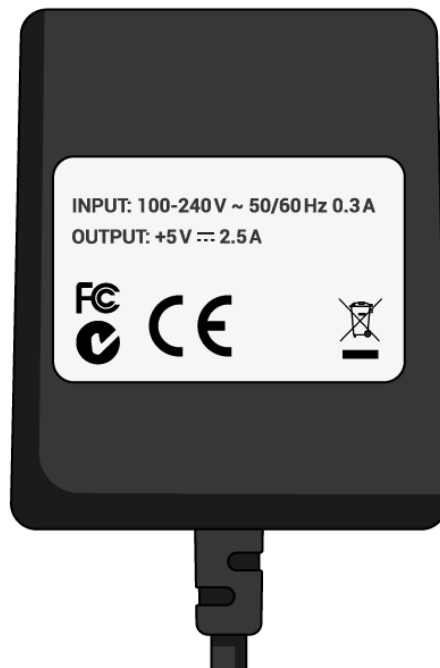


You need a power supply that provides:

- At least 3.0 amps for Raspberry Pi 4



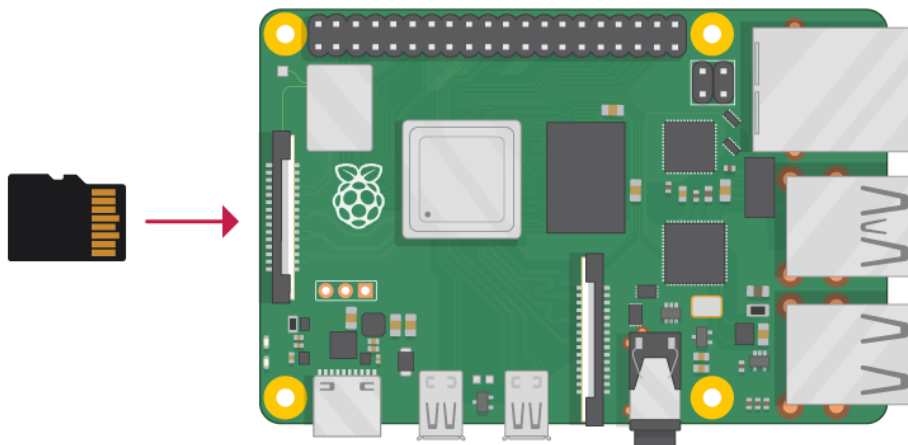
- At least 2.5 amps for Raspberry Pi 3



We recommend using [our official Universal Power Supply](#).

A microSD card

Your Raspberry Pi needs an SD card to store all its files and the Raspbian operating system.



You need a microSD card with a capacity of at least 8 GB.

Many sellers supply SD cards for Raspberry Pi that are already set up with Raspbian and ready to go.

A keyboard and a mouse

To start using your Raspberry Pi, you need a USB keyboard and a USB mouse.

Once you've set your Pi up, you can use a Bluetooth keyboard and mouse, but you'll need a USB keyboard and mouse for the first setup.

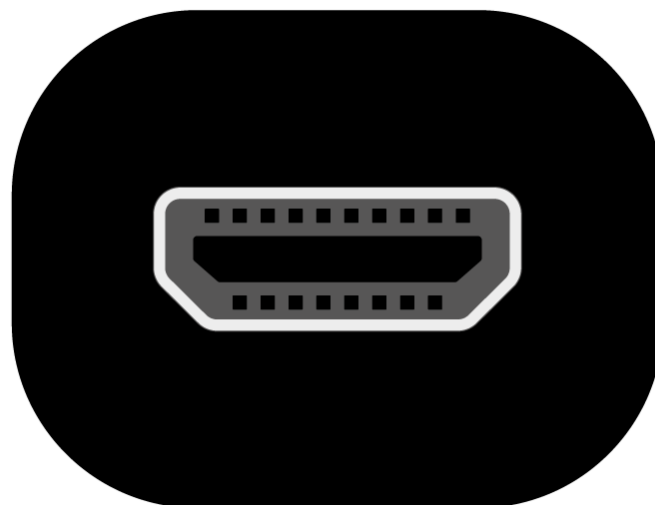
A TV or computer screen

To view the Raspbian desktop environment, you need a screen, and a cable to link the screen and the Pi. The screen can be a TV or a computer monitor. If the screen has built-in speakers, the Pi is able to use these to play sound.

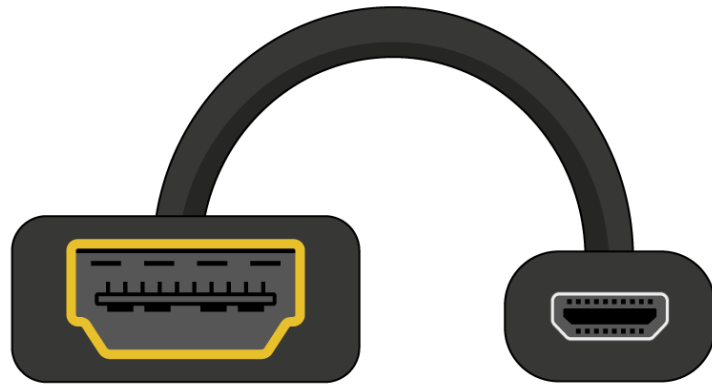
HDMI

The Raspberry Pi has a HDMI output port that is compatible with the HDMI port of most modern TVs and computer monitors. Many computer monitors may also have DVI or VGA ports.

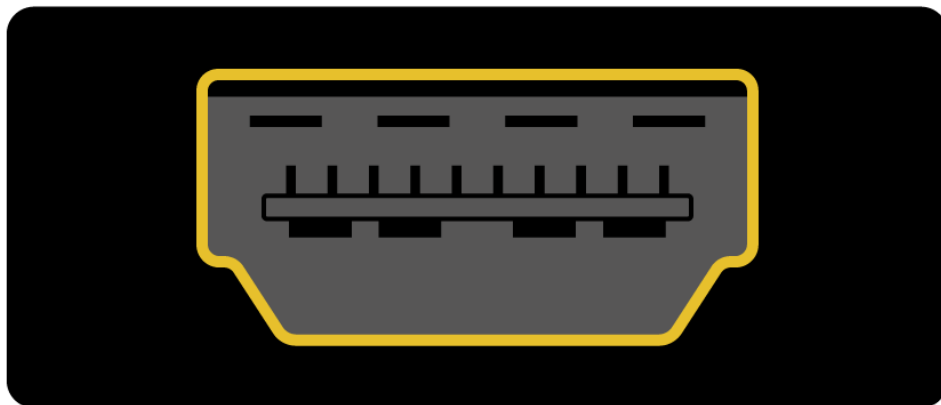
Raspberry Pi 4 has two micro HDMI ports, allowing you to connect two separate monitors.



You need either a micro HDMI-to-HDMI cable, or a standard HDMI-to-HDMI cable plus a micro HDMI-to-HDMI adapter, to connect Raspberry Pi 4 to a screen.

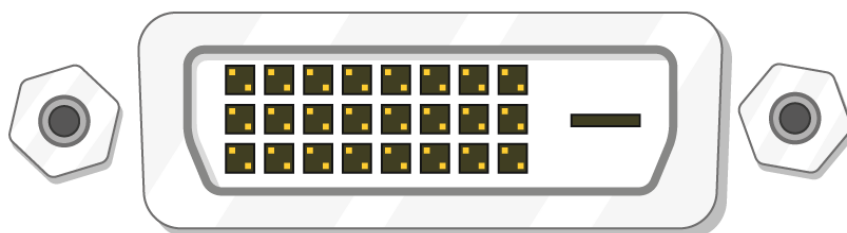


Raspberry Pi 1, 2 and 3 have a single full-size HDMI port, so you can connect them to a screen using a standard HDMI-to-HDMI cable.



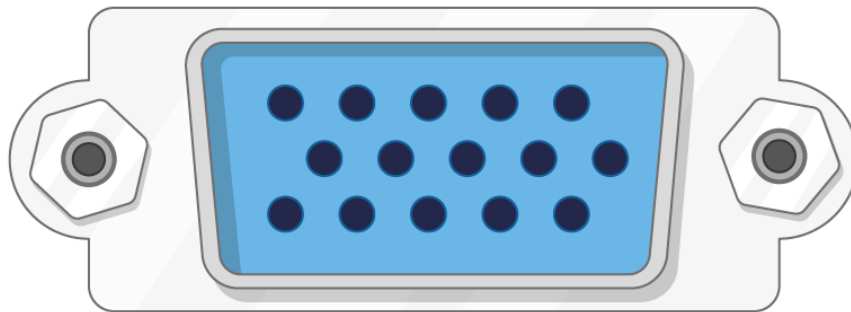
DVI

If your screen has a DVI port, you can connect the Pi to it using a HDMI-to-DVI cable.

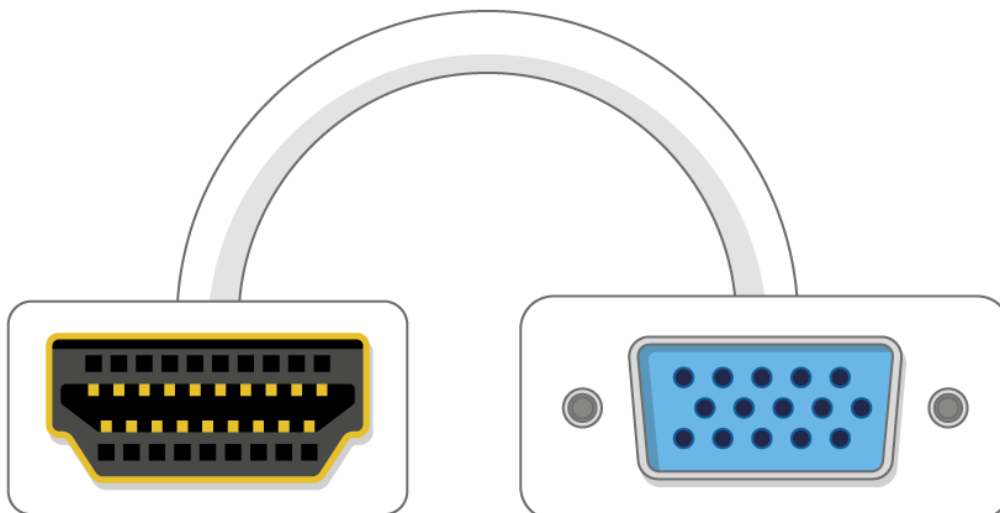


VGA

Some screens only have a VGA port.



To connect your Pi to such a screen, you can use a HDMI-to-VGA adapter.



Optional extras

A case

You may want to put your Raspberry Pi in a case. This is not essential, but it will provide protection for your Raspberry Pi. If you'd like, you can use the official case for [Raspberry Pi 4](#) or [Pi Zero or Zero W](#).

Headphones or speakers

The large Raspberry Pi models (but not Pi Zero/Zero W) have a standard audio port like the one on your smart phone or MP3 player. If you want to, you can connect your

headphones or speakers so that your Raspberry Pi can play sound. If the screen you're connecting your Raspberry Pi to has built-in speakers, Raspberry Pi can play sound through these.

An Ethernet cable

The large Raspberry Pi models (but not Pi Zero/Zero W) have a standard Ethernet port to connect them to the internet; to connect Pi Zero to the internet, you need a USB-to-Ethernet adaptor.

Raspberry Pi 4, 3, and Pi Zero W can also be wirelessly connected to the internet.

Arduino Projects | Arduino Project Ideas

Arduino has been a revolution ever since it struck the market. Arduino or Genuino (as known in some places), has been a very lucrative option for students designing their first projects. This increase in interest is causing a lot of people to tinker with technology themselves. We compiled a list of Arduino projects that would be cool to design. If you have an Arduino lying around or are planning to buy one, these should come in handy. All you need is an Arduino board and some basic electronics supplies.

Temperature Based Fan Speed Control And Monitoring Using Arduino

This project is an automatic fan speed control and monitoring that controls the speed of an electric fan according to the requirement using Arduino.

This project is available at: [Temperature Based Fan Speed Control And Monitoring Using Arduino](#)

ESP8266 Wireless Web Server

This ESP8266 based wireless web server project is built around an Arduino.

Currently, [ESP8266](#) is gaining popularity in the field of electronics because it is low-cost, reliable and easily available in the market. Most documents related to this module are in Mandarin (Chinese language) and information provided in the data sheet is not sufficient enough for using ESP8266 for an application.

This project is available at: [ESP8266 wireless web server](#)

Digital IC Tester With Embedded Truth Table

This article represents an Arduino based digital IC tester that is highly capable, highly reliable as well as cost-effective. Here, we develop a program with different functions for checking different ICs. We systematically analyse and test the prototype for several ICs, accessing each individual pin with all possible inputs. We also investigate truth tables associated with different ICs over a display channel.

This project is available at: [Digital IC tester with the embedded truth table](#)

Arduino Based Audio Meter

Presented here is an Arduino based audio meter that uses a liquid crystal display (LCD). A [VU meter](#) or standard volume indicator (SVI) is a device that displays a representation of signal level in audio equipment. In this project, the intensity of left-channel and right-channel audio signals provided as input to the Arduino UNO board is displayed as bars on the 16×2 LCD. In this circuit, analogue input pins of Arduino UNO board are used for measurement of audio-signal levels.

This project is available at: [Arduino based audiometer](#)

Arduino Based RF Controlled Robot

Here we present a simple Arduino based RF controlled robot that can be driven remotely. This robot can be built very quickly on a small budget. The RF remote control provides the advantage of a

good controlling range (up to 100 metres with proper antennae) besides being omnidirectional.

This project is available at: [Arduino based RF controlled robot](#)

Home Automation System

Presented here is a home automation system using a simple Android app, which you can use to control electrical appliances with clicks or voice commands. Commands are sent via Bluetooth to Arduino Uno. So you need not get up to switch on or switch off the device while watching a movie or doing some work.

This project is available at: [Home automation system using android](#)

Making Arduino Shields

Fritzing is an open source software initiative that supports designers and artists who are ready to move from physical prototyping to the actual product. It was developed at University of Applied Sciences of Potsdam, Germany. The software allows designers, artists, researchers or hobbyists to document their Arduino based prototypes and create PCB layouts for manufacturing. The associated website helps users share and discuss drafts and experiences as well as reduce manufacturing costs.

The project is available at: [Making Arduino shield](#)

Fancy Lights Controller

The fancy lights controller described here is built around the Arduino (an Open Source single-board

microcontroller) platform that can be purchased in pre-assembled hardware form. The circuit is nothing but a portable four-channel, multi-mode digital light controller, realised using very few external components. Four LEDs are made to glow in different sequences and patterns, controlled from the Arduino board.

This project is available at: [Fancy lights controller](#)

Android controlled robot

Presented here is a phone controlled robot that can be controlled via an app on your mobile. The control commands are sent via Bluetooth and the robot has such features as:

1. It can be controlled from Android smart-phones by touch or voice commands.
2. The speed of the robot can also be controlled
3. The robot will sense and inform to the phone its distance from the nearest obstacle
4. It will also send information about the direction in which it is moving

This project is available at: [Android controlled robot](#)

PC based oscilloscope

Here we describe how you can make your own oscilloscope at a very low cost using your PC and an Arduino board as the hardware for signal acquisition. You can use this oscilloscope to capture frequency signals up to 5kHz. The Arduino board, the heart of the oscilloscope, reads the values from its inbuilt analogue-to-digital

converter (ADC) and pushes these to the PC via USB port.

The project is available at: [PC based oscilloscope](#)

Arduino based digital IC tester using MATLAB

Testing of digital electronic systems generally involves applying a set of test stimuli to inputs of the device-under-test (DUT) and analyzing responses of the system using a response analyzer. If the DUT generates correct output responses (also called the golden response) for all the input stimuli, the DUT is regarded as fault-free. Those DUTs that fail to meet the golden response are regarded as faulty or defective. This project describes a digital IC tester for testing 74xx series digital ICs using a MATLAB graphical user interface (GUI) drop-down menu based approach.

This project is available at: [Arduino based digital IC tester using MATLAB](#)

Humidity & temperature monitoring

The IoT using Arduino microcontroller (MCU) is easy and fun for those who are new to the field. Presented here is a humidity and temperature monitoring using Arduino. In this article, humidity and temperature information from DHT-11 Humidity and Temperature sensor is analysed graphically on ThingSpeak platform using Arduino MCU and ESP8266 Wi-Fi module.

This project is available at: [Humidity & temperature monitoring](#)

Wi-Fi controlled robot

This Wi-Fi controlled robot can be controlled from anywhere using the ARMA IoT shield for UNO and the Blynk App which provides the user interface. This project uses an Arduino Uno connected to the WiFi using an ARMA IoT shield and controlled using Blynk app which is available for both iOS as well as Android. The chassis is a simple one made from pressed sheet metal with high gauge thickness.

This project is available at: [Wi-Fi controlled robot](#)

Fingerprint sensor

Once you have all this electronic stuff around, you can't let just about anybody into your workshop. This Fingerprint door lock helps you in keeping unwanted people away from your workshop. You want some engineer to access the workshop, you can add his credentials to the database and he would have access as well.

This project is available at: [Fingerprint Lock](#)

Greenhouse

Another idea would be to build a greenhouse for yourself. You can have your own kitchen garden build inside a wrapped enclosure. Mark Watney (Matt Damon in Martian) will be proud. Let's look at the project before your shuttle malfunctions. In this slideshow, the designer takes you through the entire details about building an Arduino project. The project is easy to build and the software and hardware details are provided within the presentation. So, go ahead with building a greenhouse for yourself.

This project is available at: [Greenhouse](#)

Automated plant watering system

Once you are all set with your greenhouse, how does an alarm system for watering your plants sound? This next project could be an addition to your greenhouse as well. You can use both Arduino projects together to have your own automated kitchen garden. This project is a simple and exciting plant watering system that you can build yourself in just a few hours. With efficient use of moisture sensor and downloadable component layout and source code, this could fit in perfectly for potted plants, be it on your terrace or the balcony or the front lawn. It works everywhere. This [video](#) takes you through a very similar system that would be very helpful in understanding the project.

This project is available at: [Automated Plant Watering System](#)

Air quality controller

Some reports stated Bengaluru to be unliveable in the next 5 years. Top reason listed in the report was high air pollution. How about you build something for yourself that helps you in keeping the air clean at least in your home. The outside can be taken care by planting more trees, which is a longer process. You go ahead with building an air quality control system for yourself.

This project is available at: [Air Quality Controller](#)

Earthquake sensor

This list of Arduino projects starts off with an earthquake indicator. A highly-sensitive ADXL335 accelerometer is presented that can indicate vibrations. If motion is violent enough during an earthquake and crosses a certain threshold, a local alarm light (LED) glows, a buzzer sounds and a relay energises. With certain modifications, this can be turned into a knock-and-shake detector for ATMs, vehicles or door-break alarms as well.

This project is available at: [Earthquake sensor](#)

Light Animations Using Arduino and MATLAB

Light animations are visually appealing and hence widely used for advertising purposes. In this project given below, we present a MATLAB-based graphical user interface (GUI) approach to control the glowing pattern of a number of light-emitting diodes (LEDs). This project creates five different lighting patterns including ring counter and Johnson counter by clicking appropriate pushbuttons in the GUI. The blinking speed of LEDs can also be controlled using fast, normal and slow pushbuttons in the GUI.

This project is available at: [Light Animations Using Arduino and Matlab](#)

GPS Clock

Global positioning system (GPS) synchronised clocks give accurate time. These clocks are universal and commonly used at railway stations, bus stands and airports. These are widely used for military purposes too. Here we describe a GPS clock based on Arduino Uno R3—an AVR ATmega328-based microcontroller board with six

analogue inputs pins and 14 digital input/output (I/O) pins.

This project is available at: [GPS Clock Using Arduino](#)

Home Automation Using an Android Device

This project is based on Interfacing an android application to Arduino Uno board using Bluetooth. The result is a home automation system with minimal electronic components without complex soldering and simple and flexible design.

This project is available at: [Home Automation Using Arduino Through Android Device](#)

Inverter

How much does an inverter cost? A basic with a single battery costs about 20-25K INR. How about you just go about buying the battery and use this next Arduino project for your inverter. Money aside, you get to have fun while building it. Who knows, this very practical solution could generate interest in electronics for you.

This project is available at: [Inverter](#)

Arduino Projects from around the World

Turn signaling jacket

Taking your zeal to make the air clean inside your apartment, we would also suggest you build this next jacket which signals left or right turning for two wheelers. Left or right turning indicators are

often missing in bicycles, and most of the other people don't use them. This jacket could come in handy for such people. The blinking LED's indicate left or right turning movements.

This project is available at: [Turn Signalling Jacket](#)

Lawn mowing might be a bothersome task at times, but this next project is just straight up lazy. However, if you are that lazy, this next project might be the one for you. Statistics report a majority of people are uncomfortable with tying shoelaces, with about half of these don't know how to. The effort taken to put your feet in tighten the laces and tie the knot might be too much. This next project built around Arduino does that for you. After Nike's self-tying shoes, yours could be the one to hit the market. What do you think?

This project is available at: [Automatic Tying Shoelaces](#)

Walkie Talkie

Walkie talkies have always fascinated me. I used to wonder why don't people just use their phones. Turns out it makes much sense to use a walkie talkie for them. These are bi directional modules dedicated for conversation between just them. How about you make one for yourself? This next Arduino project uses a Bluetooth for communication. All you'll need are basic electronics supplies and an Arduino board. Hopefully, if your frequency crosses over with mine, we might talk?

This project is available at: [Walkie Talkie](#)

Splash Photography

This next one is for all the photographers out there. Splash photography is a very interesting form of art to look at. The camera timing might not be able to get just right with a manual photographer at the helm. However, if you have an automated system operating that high-resolution camera for you, it truly works wonders. With this next Arduino project, you can design your very own rig and have a high-speed flash photography setup in no time.

This project is available at: [Splashing Photography](#)

Rat Trap

The move to smart has now affected the rat trap too. This next project builds a rat trap using an Arduino. Seems like a suitable project to build and set up for those little thieves. Click through to prevent any more of these buggers from making any more holes in your celery boxes or rice bags.

This project is available at: [Find the Rat](#)

Spycam

This next James Bond device could be a double-edged sword. With the help of an Arduino, this surveillance camera is fitted into a coffee cup. Every time you pretend to drink coffee, the tilt sensors in the module takes an image for you. Make sure the base of the cup is pointed in the right direction.

This project is available at: [It's Bond, James Bond](#)

Tree climbing robot

This robotic project goes up a tree with ease. Unlike Naruto, this Sasuke (Ref. Naruto anime) keeps climbing a tree and this goes all the way. Trees with a long stalk and no branches could be very well scaled with this little bugger. Build it and see it go all the way.