

Autonomous Surveillance Camera

by [apple_fan](#) on May 21, 2011

Table of Contents

Autonomous Surveillance Camera	1
Intro: Autonomous Surveillance Camera	2
Step 1: Required Parts	2
Step 2: Assemble Breadboard	3
Step 3: Programming	4
Step 4: Circuit Assembly	4
Step 5: Body Design	4
Step 6: 3D Print	6
Step 7: Assemble	6
Step 8: Result	7

Intro: Autonomous Surveillance Camera

I would like to tell you about a DIY surveillance camera I have built. It is quite different from anything out there on the market. It works for up to a month on a single charge and emails surveillance updates over cellular network. All you need to do is send it an email and the camera would reply you with a picture of your home, country house or car.

Here is a video of how it all fits together:



The device checks pre-defined email mailbox once an hour. If a new email is received the camera takes a picture, saves it on a SD-card and emails the image to its owner.

Making of the camera was also an interesting journey I would like to share.



Step 1: Required Parts

I have used following basic components for the device:

MultiTech MTSMC-G2-IP

This GSM modem is great because of its built-in POP3 and SMTP support. Although these protocols are not very complicated to implement having them implemented saves some time.

LinkSpite JPEG Camera

This is a great little device. Built-in JPEG compression support saves a lot of development effort. The camera produces a 640x480 image. The image is around 50K; it is small enough to rapidly transmit over a cellular network.

Sparkfun's microSD Shield

Although there are a few SD card circuits available I decided to go with this one because it is very easy to work with both in software (Sparkfun provides great APIs) and in hardware (again, thanks Sparkfun for built-in voltage conversion).

ATmega 328P

This micro-controller was a natural choice. Arduino Uno uses this chip, so there are plenty of libraries and examples out there. Unfortunately, Arduino Uno itself consumes too much power when it is in a sleep mode, so I had to build a board with more efficient power supply system myself.

TPS2020 and LM2936

TPS2020 is a great power distribution IC. It allowed me turning circuit components on and off as I needed it. LM2936 is an awesome regulator with really low quiescent current. Combination of the two parts allowed my circuit to consume as little as 0.14mA in sleep mode. It stretches battery life for up to a month on a single charge; the camera is powered by 5V from two CR123A batteries.

<http://www.instructables.com/id/Autonomous-Surveillance-Camera/>

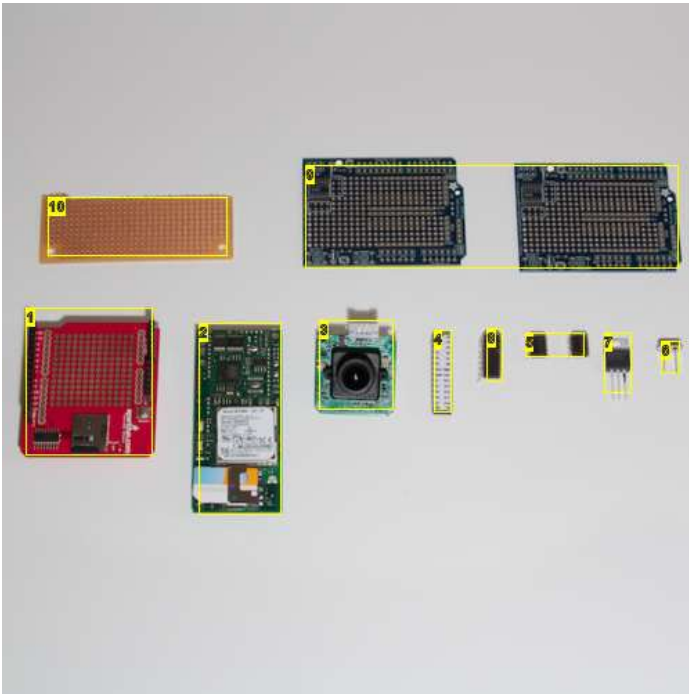


Image Notes

1. Sparkfun's microSD Shield
2. MultiTech MTSMC-G2-IP
3. LinkSpite JPEG Camera
4. ATmega 328P
5. TPS2020
6. 16Mhz Crystal
7. 3V Regulator
8. 74HC4052
9. Adafruit Proto Shield for Arduino PCB
10. A Piece to Host MultiTech MTSMC-G2-IP

Step 2: Assemble Breadboard

A proto-prototype breadboard layout would help to debug the circuit. It would also transform your table into a small cyberpunk jungle.

I would highly recommend laying out everything on the breadboard before assembling it.

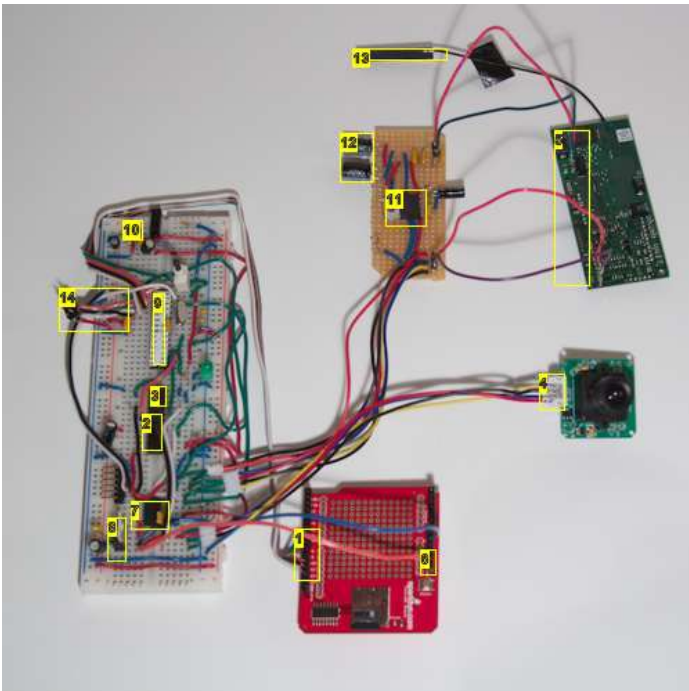


Image Notes

1. Talk to MicroSD via pins 13,12,11 and 8

2. Pins 0,1 and 2 to 74HC4052: 0,1 would be connected either to GSM Modem or to the Camera. Pin 2 controls where to connect.
3. Pins 12,1 out to GSM Modem, Pins 14,5 out to the Camera
4. Camera uses four lines: 5V power, ground, TX, RX
5. Modem uses 4 lines: 5V power, ground, TX, RX
6. 3V regulated power
7. TPS2020 that turns SD Card on and off. Final board has one for SD Card, Modem and Camera
8. 3V Regulator for SD Card. Final board has 3V regulator for SD Card, and 2 5V regulators for Modem and Camera. Plant those regulators _after_ TPS2020.
9. Microcontroller
10. LM2936
11. 5V Regulator for the Modem
12. For some reason, MultiTech development board suggests using 4 470uF capacitors. I've used two.
13. Modem antenna
14. Programming interface on pins 13,12,11 and Reset. I used AVRISP mkII to program the chip.

Step 3: Programming

I would recommend using Eclipse with AVR Eclipse Plugin . You would also need a programmer to flush your program on to the chip. I've used a standard AVRISP mkII , which worked great with my OS X).

Source code for the project could be found at [google code](#) .

Step 4: Circuit Assembly

I assembled the circuit with the simplest way I could think of: just solder everything on the prototype boards. It ended up looking like a sandwich with Adafruit boards on the outside and Sparkfun MicoSD card in the middle.

It takes about a day of soldering, cursing, un-soldering and soldering again to put all components together.



Image Notes

1. Standard Arduino Connectors
2. Adafruit Proto Shield
3. Adafruit Proto Shield
4. Sparkfun MicroSD Shield

Step 5: Body Design

Enclosure Motivation

Unfortunately, not all DIY projects have nice looking enclosures. I believe that a project is fully finished when it fully finished and the circuit, the program and the enclosure are in harmony. So, I decided to use 3D printing service to achieve a look I want.

CAD Tool

After struggling quite a bit with 3D CAD tools, I have discovered a great program OpenSCAD. It is a great script-based CAD editing software. Instead of using mouse to draw 3D shapes one just types in commands. For example, a sphere with radius of 5 would be `sphere(r=5)`. Awesome, right?

Body Design

I decided to go with a clean sphere design. Body consists of three parts: top part that hosts the camera, bottom part that hosts primary circuit and batteries and a middle "holder" piece that binds top and bottom. The design was inspired by an 8-ball from any toy store.

Caliper and OpenSCAD allowed to make precise models of both circuit components and body parts. Source for the CAD files could be found under `enclosure` folder in the project source.

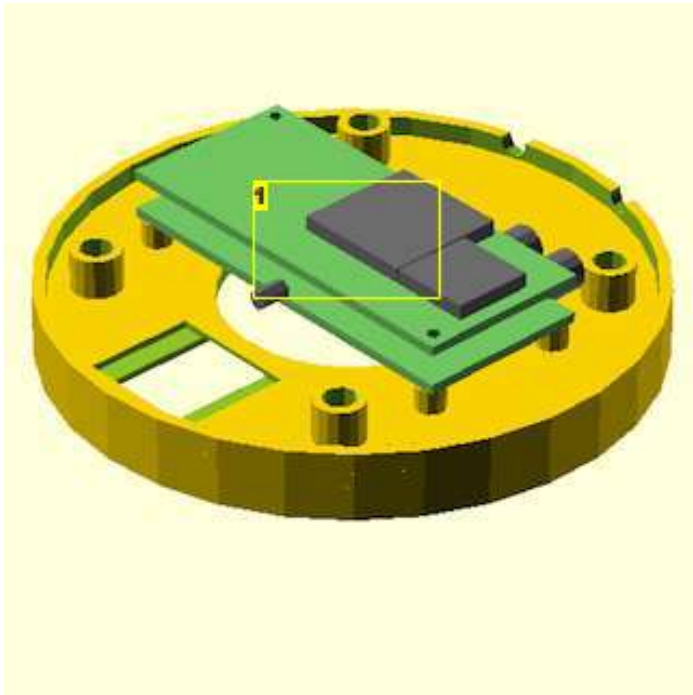


Image Notes

1. GSM Modem on its holder



Image Notes

1. Main circuit board
2. Battery (CR123A) Holder
3. Battery (CR123A) Holder

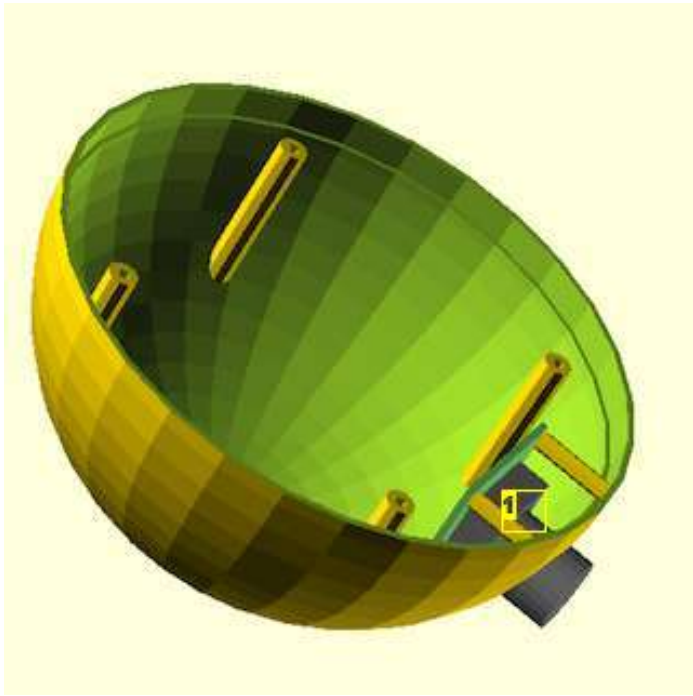


Image Notes

1. Camera

Step 6: 3D Print

Once CAD model is exported to STL, it could be sent to any 3D Printing service to be printed. I have chosen Ponoko because of their clear interface and reasonable prices.



Step 7: Assemble

Once 3D printed case arrives components fit nicely in their places. I have used a combination of screws and double-sided adhesive tape to put everything together.

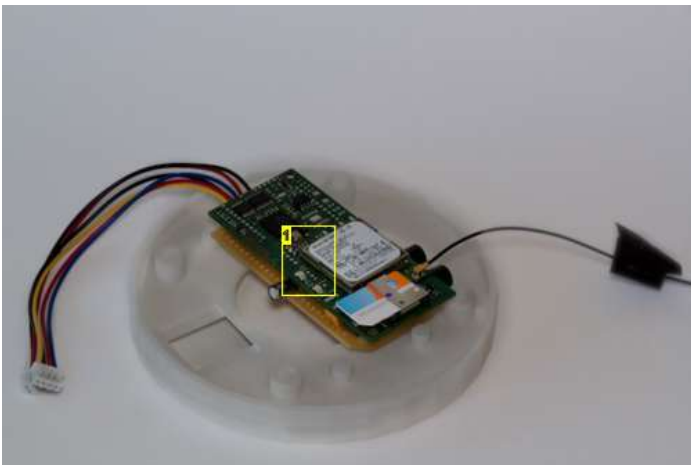


Image Notes
1. Modem in its holder

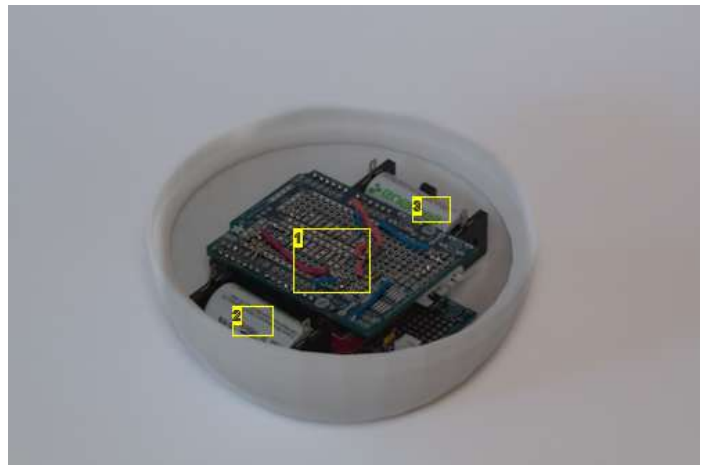


Image Notes
1. Main Circuit
2. CR123A Battery in its holder
3. CR123A Battery in its holder



Image Notes

1. Camera

Step 8: Result

If everything fit together nicely the camera should be working and carefully surveilling your house. Even though emailed image resolution is not HD, it is quite enough to tell if everything is fine at home.

I am using AT&T GoPhone as a network plan. It costs about 8 cents to check an email and about 70 cents to send an email with a picture.



