Dzień dobry państwu, nazywam sie Paweł Marciniec, jestem studentem 3 roku informatyki stosowanej na IMIPie, należę do KN Energetyków Caloria i mam dzisiaj przyjemność zaprezentować państwu urządzenie jakie ostatnimi czasy udało mi się złożyć, a jest nim ten oto straszak na gołębie, działający na platformie arduino, dokładnie arduino Leonardo. Może zacznijmy od tego czym tak w ogóle jest arduino?

Jeżeli komuś się nie chce mnie słuchać to tutaj przygotowałem schemat urządzenia na którym ja operowałem. Arduino to open hardware’owa platforma programowalnych mikrokontrolerów z możliwościa obsługi urządzeń wejścia-wyjścia. Jest na tyle prosta w obsłudze iż nawet informatyk może pobawić się w elektronika. Programowanie tych mikrokontrolerów odbywa się zasadniczo w języku C/C++ z kilkoma przekształceniami. Wykorzystują one układy z serii Atmega8, 168, 328, 1280 i 2560, przeważnie posiadają 5v regulator napięcia i działają zasadniczo w częstotliwości 16MHz, zdarzają się wyjątki.

Arduino wybrałem z tego względu, iż początkiem tego semestru zaczęły mi się zajęcia z systemów wbudowanych, gdzie miałem przyjemność zapoznania się z uzytkowaniem owego urządzenia, zafascynowało mnie ono i stwierdziłem iż połączę swój uporczywy problem halasujących gołębi z tym oto mikrokontrolerem.

Dlaczego arduino? W zasadzie czemu nie, skoro ma pewien potencjał, chciałem sprawdzić w jakim stopniu mogę go wykorzystać w swoim zamyśle.

Krakowskie gołębie, są zmorą czystych balkonów, parapetów, źle uszczelnionych okien i tak dalej. Mnie na przykład strasznie denerwują kiedy o coraz to wcześniejszej porze typu 5-6 rano przylatują i gruchają. Potrafię wstawać i je przeganiać a one potrafia być na tyle bezczelne, że z upartością osła przylatują i robią swoje, koło się zamyka.

Przejdę teraz do konkretów, co przy składaniu takiego urządzenia może pójść nie tak? W zasadzie wszystko. Otóż, na slajdzie widzimy super teoretycznie działający prototyp, ale…

Niby jakoś to już wygląda, tutaj w innej perspektywie, ale nie działało.

Czemu to nie działało? Z kilku powodów, oto one.

Błędy amatora, to na przykład złe oporniki, wprawne oko zauważyło że to nie było 1.8 omów a znacznie więcej, zamiast 10 omów miałem tam 10 tysięcy…

Pospiech twój wróg, łatwo wtedy przeoczyć to „k” obok wartości opornika.

Podlączanie czytnika kart SD do „zwykłych” pinów, bazując na uno, jest totalnie inne na Leonardo, w Leonardo musiałem podpiąć go do interfejsu ICSP, chociaż dodam iż czasem zdarzyło mi się rozpoznać cokolwiek podpięte schematem z uno, niestety to tylko było czasem oraz totalnie mijało się z rzeczywistością.

Kolejną rzeczą jest odpowiednia biblioteka do odtwarzania dźwięku. Na Leonardo udało mi się jedynie uruchomić zewnętrzna bibliotekę Tmrpcm, nie działa SimplesdAudio i pokrewne, zapisujące dźwięki w formacie afm. Tmrpcm wymaga dźwięków w formacie wav sformatowanych do 8 bitrate na częstotliwości od 8 do 32 kHz.

Hardware robiony ręcznie przez kompletnego laika również jest nieciekawym rozwiązaniem pomijając już fakt że pomyliłem się w opornikach.

Czas potrzebny na eksperymenty jest w zasadzie kluczowy. Jeżeli usiadłbym z tym projektem wczoraj mając w głowie tyle samo ile wtedy gdy go zaczynałem, nie wyrobiłbym się na 100%. Cały projekt kosztował mnie dobrych 2 tygodni eksperymentowania i przeszukiwania róznych forów internetowych.

Może zaprezentuję teraz metode działania urządzenia. Gołąb sobie podlatuje i ku jego zdziwieniu słyszy to. Naszym gołebiem może być również droid republiki z gwiezdnych wojen. Bądź też kot. Zaletą jest to że mamy w zasadzie nieograniczone możliwości co do dźwięków jakie urządzenie ma wydawać, ogranicza tylko pamięc flash. Całe serce urządzenia zamknięte jest w tej skrzyneczce, wraz z konsolą do zmiany dźwięku oraz głośności.

Kilka zdjęć z mojej pracy nad projektem, schemat prototypu zrobiony w programie fritzing oraz kod jaki zawarłem w urzadzeniu.