

Vlad-Alexandru FLOREA (78767) - Motor Tuner

Autorul poate fi contactat la adresa: **Login pentru adresa**

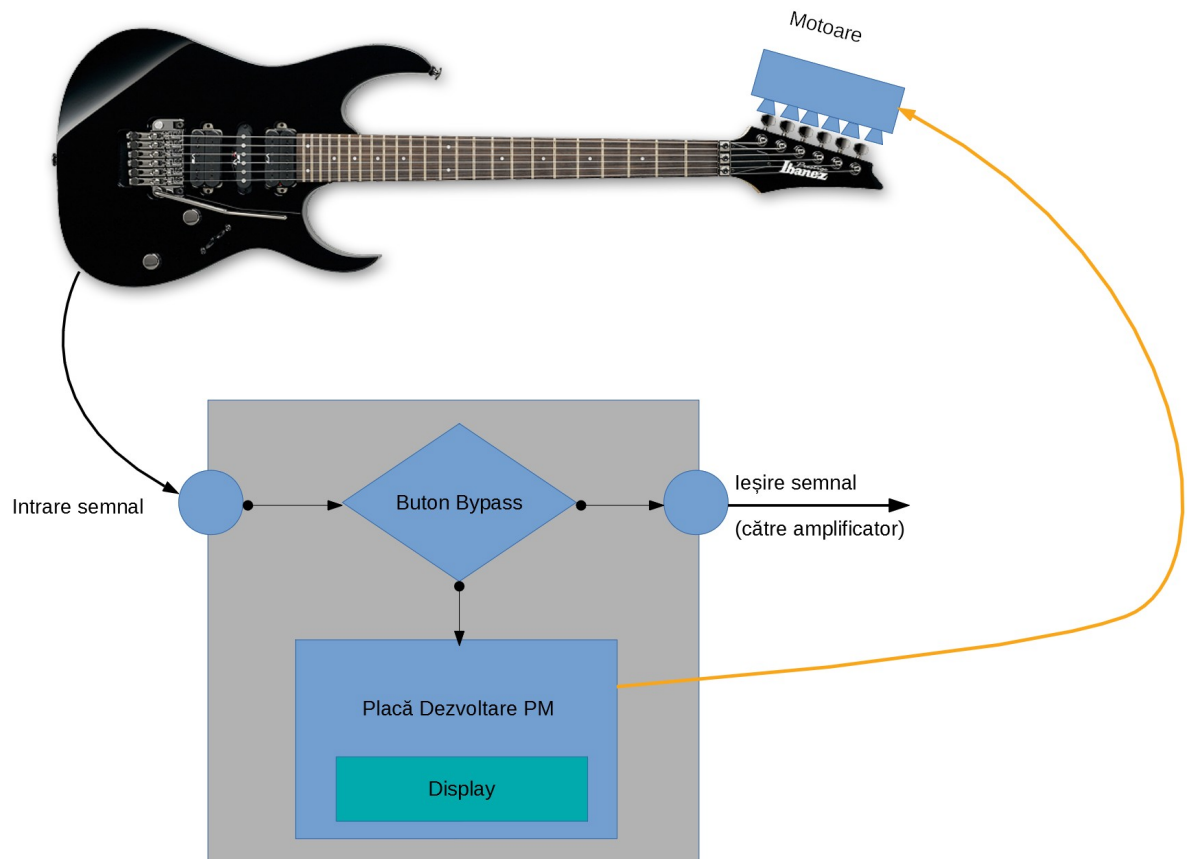
Introducere

Proiectul are ca scop construirea unui acordor automat pentru chitare electrice (sau electro-acustice). Acest acordor se va interpune în lanțul de semnal dintre chitară și amplificator, va prelua semnalul de la chitară (atunci când este activat), iar pe baza frecvențelor recepționate va ajusta cu ajutorul unui motor/unor motoare electrice poziția cheilor de acordare de pe chitară, până când acestea ajung în poziția corectă.

Am ales să construiesc un dispozitiv de acest tip după ce am văzut o prezentare pe YouTube pentru "Tronical Tune" ([dfQgqjFjc74](#)).

Consider că acest proiect va fi util atât pentru mine cât și pentru ceilalți, deoarece elimină singura parte plictisitoare din cântatul la chitară și anume, acordatul instrumentului. De asemenea, un acordor automat va putea comuta într-un timp foarte scurt între mai multe variante de acordaj (E Standard, Drop D, Drop C, D Standard, etc.) pentru a face rapid trecerea între diverse melodii, un proces anevoios și de durată atunci când este executat manual.

Descriere generală



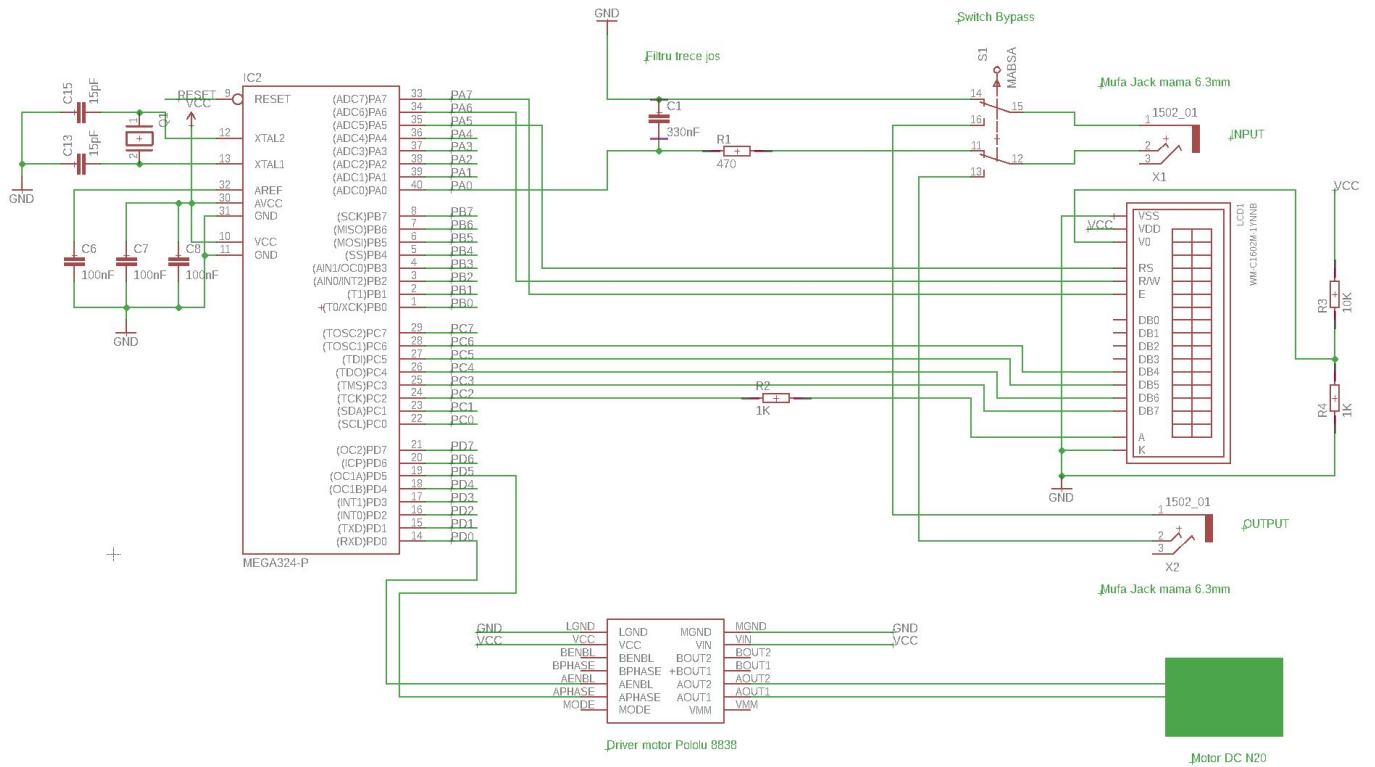
Așa cum am menționat mai sus, acordul se va interpune în lanțul de semnal dintre chitară și amplificator, semnalul va circula prin el, iar la acționarea butonului de bypass de către utilizator acesta se va activa și va prelua semnalul de la chitară, având ca efect și întreruperea semnalului către amplificator. Utilizatorul va acționa fiecare coardă de pe chitară în succesiune rapidă, iar acordul va arăta pe ecran starea fiecărei corzi și va ajusta cheile de acordaj corespunzător cu ajutorul motorului/motoarelor. Atunci când utilizatorul este mulțumit de rezultat acesta va apăsa din nou butonul de bypass pentru a comuta semnalul înapoi către amplificator.

Hardware Design

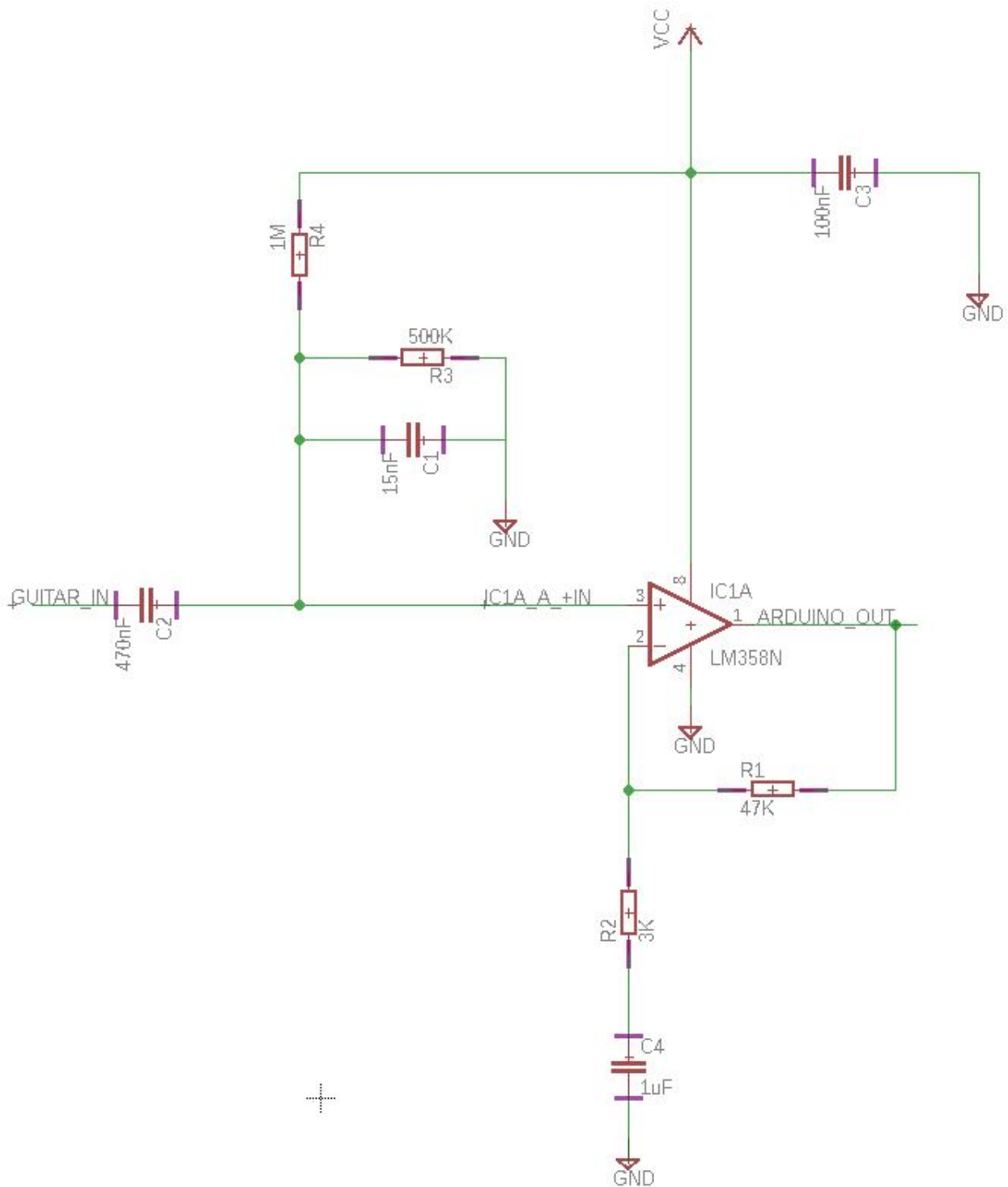
Piese necesare:

- 1x motor DC N20 Pololu cu reductor 1:380
- Driver motor Pololu 8835 (<https://www.pololu.com/product/2135>)
- display simplist 16x02 - similar cu cel de la laborator
- 2 x mufe mama jack mono 6.3mm
- Latching Stomp Foot Switch - pentru activare
- Componente electronice necesare pentru stagiul de preamplificare folosit: amplificator LM385N, condensatori 470nF, 100nF, 15nF, 1uF, rezistori 47k, 500k (in cazul meu comușă din 200k și 300k în serie), 1M, 3k

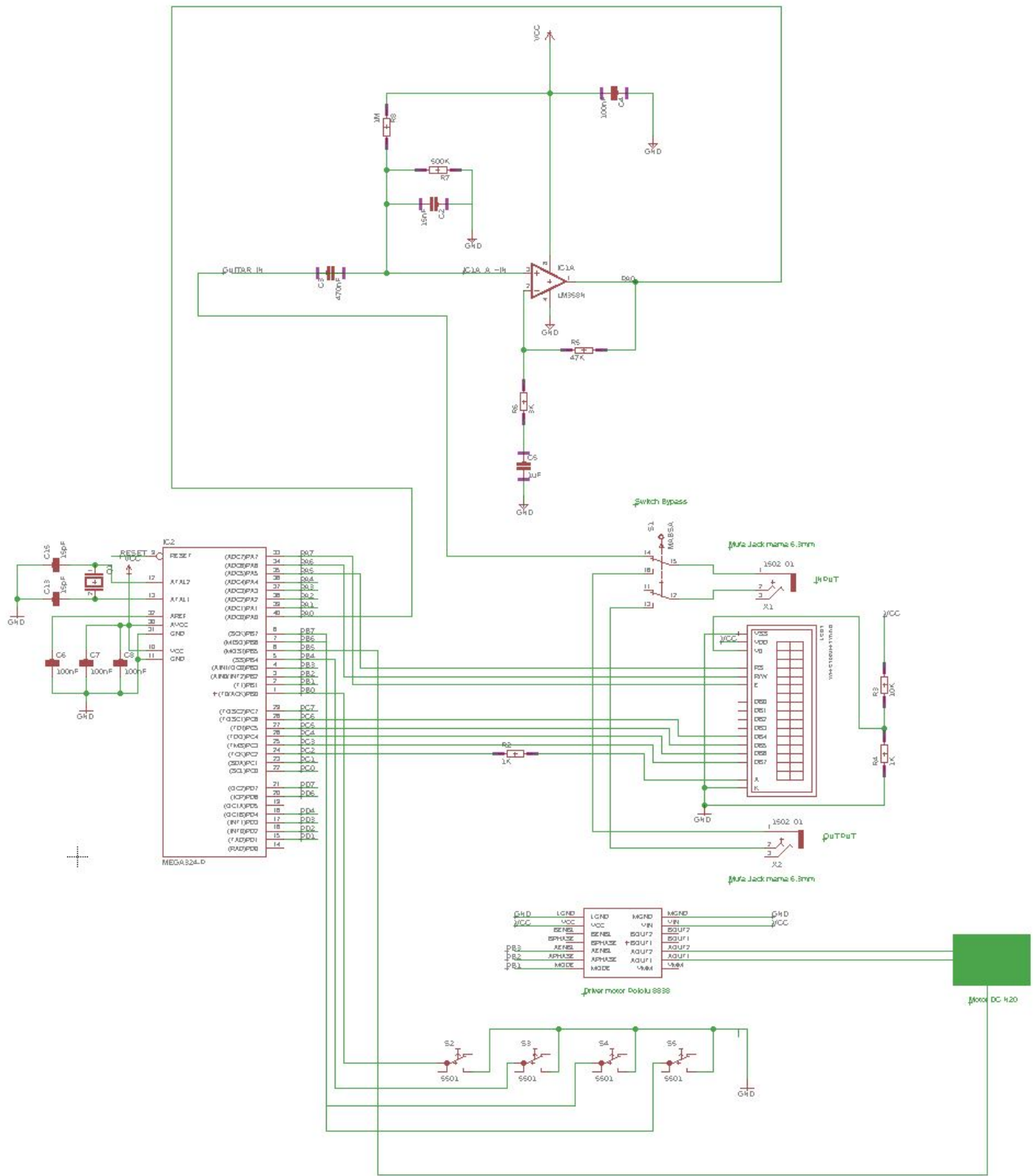
Schema electrică:



Datorită zgomotului nedorit foarte mare prezent la intrare, am ales să construiesc un circuit de preamplificare care să facă legătura între semnalul de la chitară și intrarea ADC0 (am folosit prin urmare ADC-ul cu gain 1x și nu 10x așa cum am scris în schema inițială; în combinație cu referința internă de tensiune de 2.56 V). Am înlocuit filtrul trece jos din schema inițială cu următorul montaj:



În concluzie schema electrică finală arată astfel:



Am adăugat 2 butoane pentru a selecta coarda ce urmează a fi acordată. Dacă se apasă aceste butoane simultan se va putea trece într-un acordaj diferit (D Standard, Drop D, etc.). Am adăugat alte 2 butoane pentru a putea controla manual motorul în cazul în care acesta a rămas într-o poziție neconfortabilă sau atunci când se dorește schimbarea corzilor. Dacă se apasă simultan aceste două butoane se va inversa sensul de rotație al motorului în procesul de acordare automată.

Software Design

Pentru determinarea frecvenței am utilizat algoritmul scris de Amanda Ghassaei (prezent în majoritatea tutorialurilor de gitar tuner cu Arduino)

<http://www.instructables.com/id/Arduino-Guitar-Tuner/>.

Am utilizat ADC-ul cu referința de tensiune internă de 2.56V, în modul auto-trigger, prescaler 32, și rezoluție 8 biți, folosind rezultatul din ADCH pentru rularea algoritmului din tutorialul de mai sus.

Am folosit instrucțiuni pentru printare pe LCD din biblioteca lcd.h pusă la dispoziție în cadrul laboratorului.

Pentru debugging am folosit un cablu UART și biblioteca usart.h din cadrul laboratorului.

Pentru controlul motorului am generat un semnal PWM pe ieșirea OC0A, care va comanda driverul. Motorul este controlat proporțional cu eroarea măsurată între frecvența dorită și cea actuală. Am utilizat în acest scop encoderul motorului și am măsurat cu o întrerupere GPIO numărul de tranziții din semnalul generat de acesta. În cazul motorului meu, encoderul generează 16 tranziții ale semnalului pentru o rotație completă a axului. Știind că pe motor este montat un reductor 1:380, înseamnă că pentru 16 * 380 tranziții ale semnalului de pe encoder axul exterior va face o rotație completă, astfel putem ști exact unghiul la care vom roti motorul. Totuși, puteam folosi și un motor fără encoder pe care să îl controlez similar, de data aceasta proporțional în domeniul timp, cu ajutorul unui timer.

Descrierea codului aplicației (firmware):

- mediu de dezvoltare: CLion + WinAVR + BootloadHID
- biblioteci și surse 3rd-party: bibliotecile lcd.h și usart.h din codul laboratoarelor (usart.h folosit mai mult pentru debugging cu ajutorul unui cablu UART), algoritmul de măsurare a frecvenței din acest tutorial de pe Instructables, scris de Amanda Ghassaei (prezent în majoritatea tutorialurilor de gitar tuner cu Arduino) <http://www.instructables.com/id/Arduino-Guitar-Tuner/>

Rezultate Obținute

Am reușit să obțin o precizie de +/- 1 sau 2 Hz, cu ajutorul configurației de preamplificare din schema de mai sus și a algoritmului din tutorialul respectiv.

Rezultatul final: <https://www.youtube.com/watch?v=fRMH0ej9VBY>

Concluzii

Deși am plecat de la "Tronical Tune" ([dfQgqjFJc74](https://www.youtube.com/watch?v=dfQgqjFJc74)) m-am inspirat de la un alt tuner, cu un singur motor, "ROADIE" www.roadietuner.com, pentru simplitatea părții mecanice.

Sunt sincer surprins că am reușit până la urmă să obțin un tuner funcțional, cu o precizie foarte bună, care chiar îmi va fi util de acum înainte pentru a trece rapid între acordaje, a re-acorda chitara atunci când se dezacordează, și chiar pentru a schimba mai rapid corzile (folosind motorul electric în modul manual pentru înfășurarea acestora pe cheițe).

Desigur montajul meu nu se compără cu variantele comerciale ca precizie sau rapiditate, însa pentru un montaj construit în timpul liber, la o fracțiune din cost și fără scule specializate a ieșit chiar foarte bine. Pot spune că sunt mulțumit de rezultatul final.

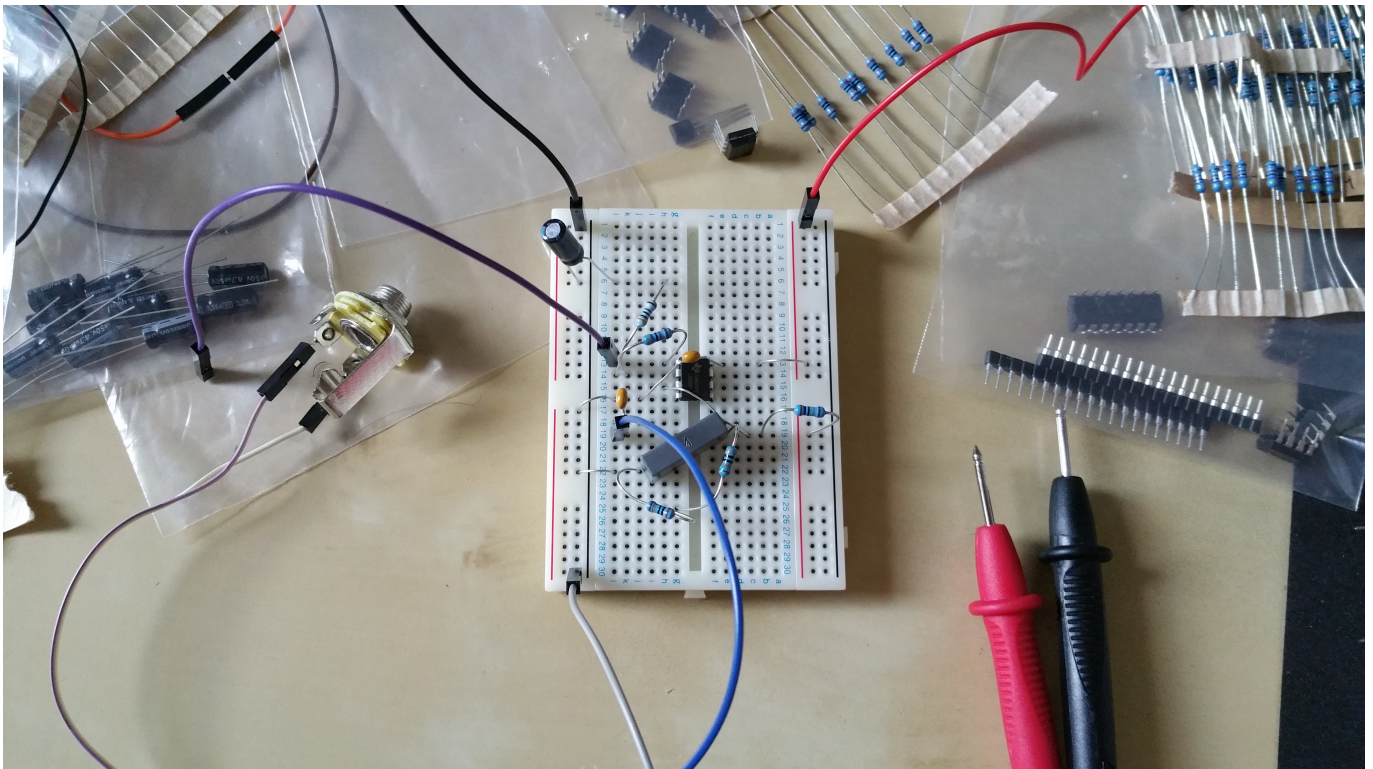
Download

Arhiva cu surse si Makefile:

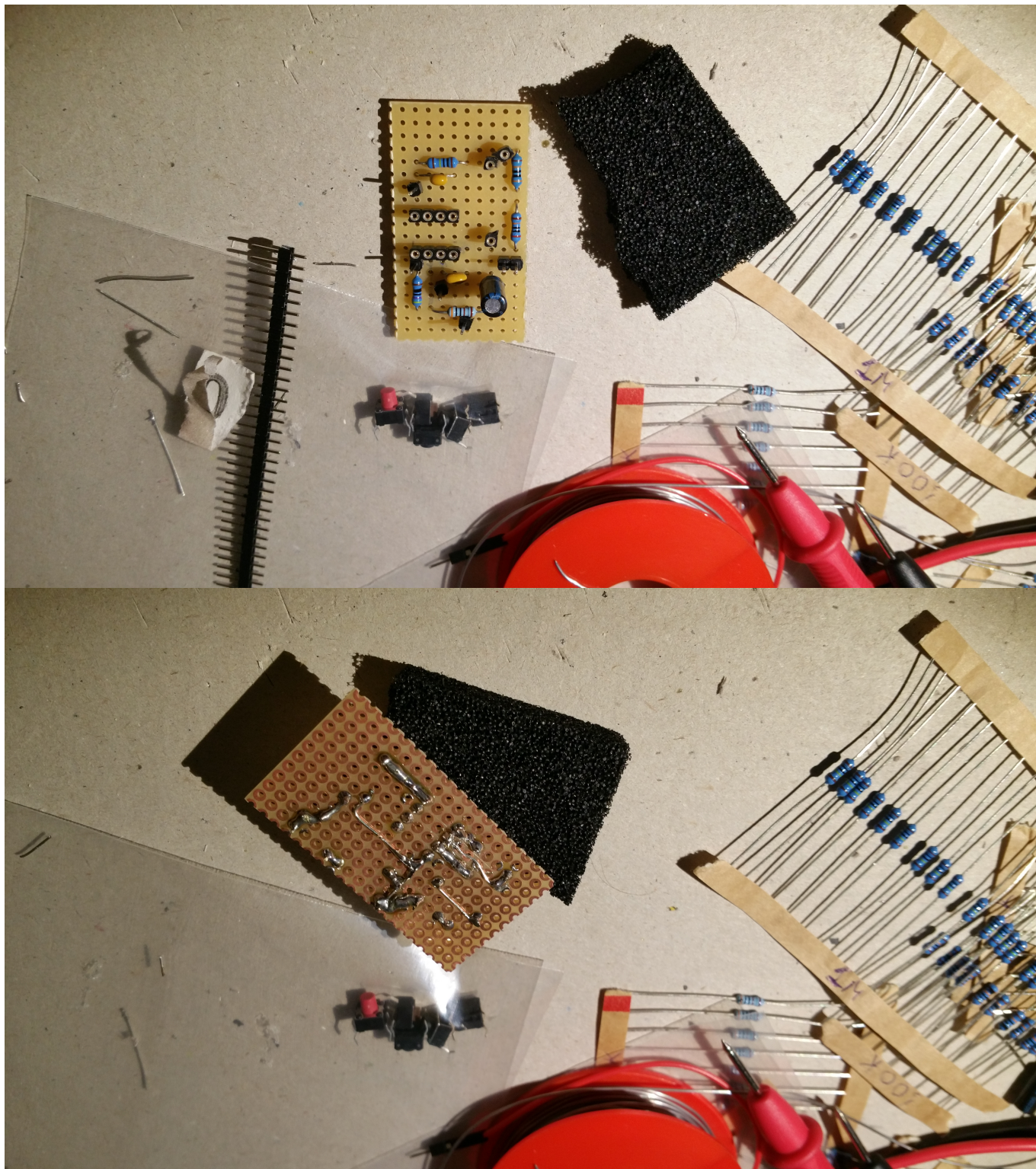
[final_code_mototuner.zip](#)

Jurnal

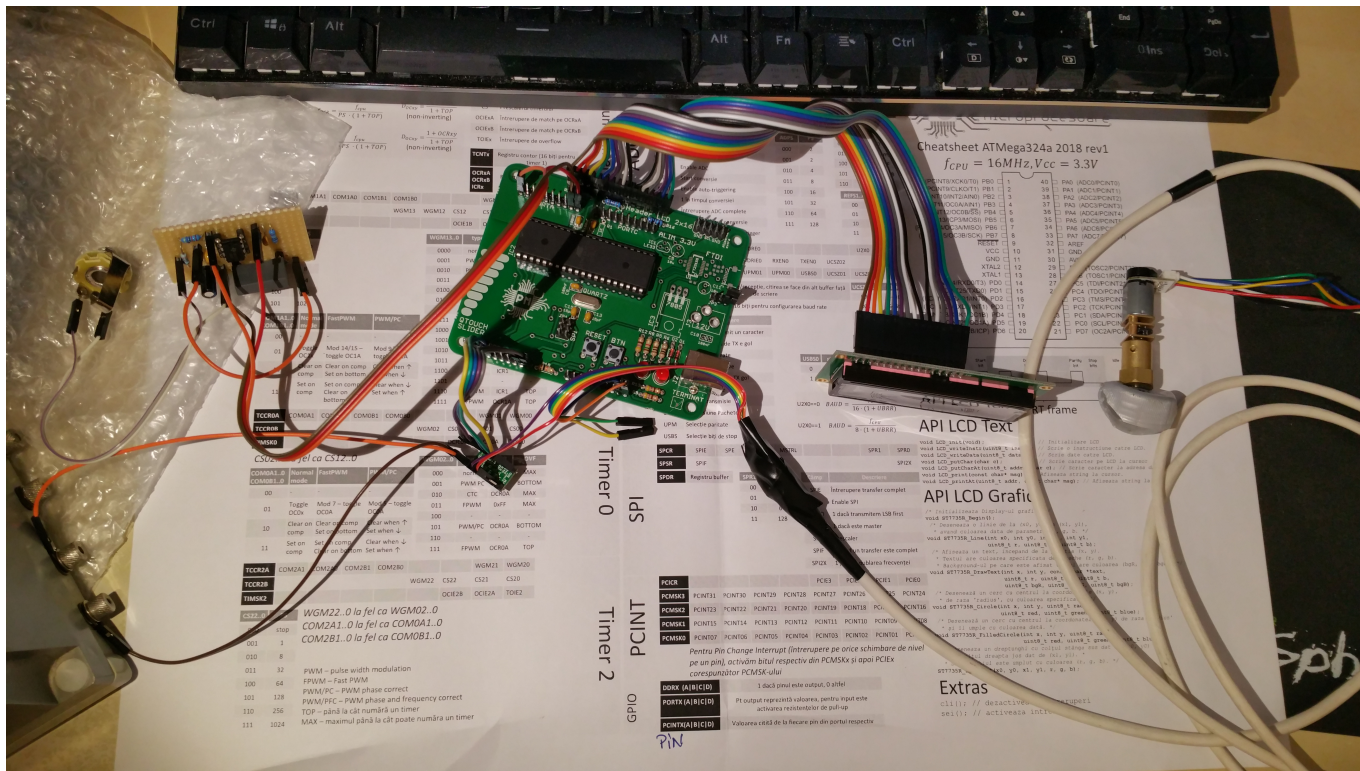
- Testarea circuitului pe breadboard:



- Mutarea circuitului pe placa de textolit:



- Implementarea hardware este completă:



• Construirea carcasei:





- Rezultatul final: <https://www.youtube.com/watch?v=fRMH0ej9VBY>

Bibliografie/Resurse

- <http://www.instructables.com/id/Arduino-Guitar-Tuner/>
- <http://www.akellyirl.com/arduino-guitar-tuner/>
- diagramele de conectare de pe pagina driverului <https://www.pololu.com/product/2135>
- tutorialele de lipit de aici sunt foarte bine explicate și mi-au fost de mare folos <https://www.youtube.com/watch?v=sTv3gK9tAKA>
- Documentația în format PDF

From:
<http://ocw.cs.pub.ro/courses/> - CS Open CourseWare

Permanent link:
http://ocw.cs.pub.ro/courses/pm/prj2018/avoinescu/vladflorean_motortuner



Last update: **2021/04/14 15:07**