



STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

MULTIFUNKČNÍ ROTAČNÍ ZAŘÍZENÍ ZE STARÉ TISKÁRNY

AUTOR	Martina Hanusová
ŠKOLA	Gymnázium a Střední průmyslová škola elektrotechniky a informatiky, Frenštát pod Radhoštěm, příspěvková organizace
ROČNÍK	Prima, obor osmileté gymnázium
KRAJ	Moravskoslezský
OBOR	10. Elektrotechnika, elektronika a telekomunikace



STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

MULTIFUNKČNÍ ROTAČNÍ ZAŘÍZENÍ ZE STARÉ TISKÁRNY

MULTIFUNCTIONAL ROTATION MACHINE FROM AN OLD PRINTER

AUTOR	Martina Hanusová
ŠKOLA	Gymnázium a Střední průmyslová škola elektrotechniky a informatiky, Frenštát pod Radhoštěm, příspěvková organizace
ROČNÍK	Prima, obor osmileté gymnázium
KRAJ	Moravskoslezský
KONZULTANT	Ing. Marcel Hanus, Ph.D.
OBOR	10. Elektrotechnika, elektronika a telekomunikace

Prohlášení

Prohlašuji, že svou práci na téma *Multifunkční rotační zařízení ze staré tiskárny* jsem vypracovala samostatně pod vedením Ing. Marcela Hanuse, Ph.D. a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Dále prohlašuji, že tištěná i elektronická verze práce SOČ jsou shodné a nemám závažný důvod proti zpřístupňování této práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a změně některých zákonů (autorský zákon) v platném znění.

V Kunčicích pod Ondřejníkem dne 9.3.2017

Podpis:



Poděkování

Děkuji svému konzultantovi Ing. Marcelu Hanusovi, Ph.D. za obětavou pomoc, podnětné připomínky, povzbuzující slova a nekonečnou trpělivost, kterou mi během práce poskytoval a za přednastavení registrů použitých mikrokontrolérů. Dále děkuji svým učitelům: panu Mgr. Richardu Štěpánovi za skvělý kroužek robotiky a panu Ing. Liboru Otáhalíkovi za skvělý kroužek Arduino a doporučení přihlásit se do SOČ. V neposlední řadě pak děkuji svému sousedovi Ing. Davidu Strnadelovi za nápad na realizaci tohoto projektu. Nakonec moc děkuji autorům šablony SOČ Lucii Vaškeové, Jaroslavu Páralovi a Romanu Beránkovi, která mi hodně pomohla při psaní této práce.

Anotace

Náplní této práce je popis způsobu recyklace dílů ze starých tiskáren k sestavení multifunkčního rotačního zařízení, s primárním zaměřením na konstrukci natahovače automatických hodinek, ale majícím i další využití, např. k míchání vzorků v chemických a biologických laboratořích, homogenizaci sprejů s barvami, míchání potravin nebo otáčení vajec v inkubátorech.

Práce obsahuje přehled a rozbor konstrukce několika komerčních a amatérských řešení natahovačů automatických hodinek, popis konstrukce základní verze natahovače s jednoduchým ovládním pro jeden kus hodinek „Watch Winder Orbit Peppermint“ a popis konstrukce pokročilé verze „Watch Winder Multi Deluxe“ s větší kapacitou hodinek (až 6 ks), řídicím systémem s LCD displejem a možností širokého nastavení parametrů otáčení v několika nezávislých krocích.

Pro možnost využití této práce širší odbornou veřejností i pro jiné aplikace je zde navržena konstrukce elektronické desky (shield) pro běžně dostupné Arduino UNO, popsán postup demontáže staré tiskárny (na příkladu HP DeskJet 990cxi) pro získání potřebných součástek ke konstrukci míšícího zařízení, zpracován model krabičky pro řídicí jednotku k vytištění na 3D tiskárně, napsán program pro Arduino UNO a zpracován návod k montáži a použití celého zařízení.

Materiálové náklady na výrobu popsaných rotačních zařízení se pohybují řádově ve stovkách Kč a zařízení jsou tedy cenově dostupná i pro mladé začínající vědce shánějící potřebné laboratorní vybavení pro svou práci.

Klíčová slova

recyklace starých tiskáren; míšící zařízení; natahovač automatických hodinek; Arduino;

Annotation

This project describes a way of recycling old printers into multifunctional rotational machines, focused primarily on a design of watch winders for automatic wrist watches, but having wider use, such as mixing samples in chemical or biological laboratories, paint sprays homogenization, food mixing or turning eggs in incubators.

The study contains a review and analysis of applied designs and parameters of several commercial and home-made (DIY) types of watch winders. Design of a basic version of the winder with a simple operation for single wrist watch, called „Watch Winder Orbit Peppermint“ is described. The following chapter describes a more advanced version, called „Watch Winder Multi Deluxe“, having larger capacity (up to 6 watches), a control system with LCD display and a wide possibility of setting the rotational parameters in several independent steps.

In order to allow a wider public use of this study for other similar applications, a design of an electronic shield for commonly available Arduino UNO is described, together with a procedure for disassembly of an old inkjet printer (on example of HP Deskjet 990cxi) for salvaging the parts necessary for making the mixing devices, a 3D model of a case for the control unit to be printed on 3D printer, a software for Arduino UNO and a manual for assembly and use of the final device.

Material costs for creating the rotational devices range in low hundreds of CZK, making them accessible even for young scientists looking for a laboratory equipment for their work.

Obsah

1	ÚVOD	7
2	PŘEHLED KOMERČNÍCH A AMATÉRSKÝCH ŘEŠENÍ NATAHOVAČŮ HODINEK	8
2.1	KOMERČNÍ ŘEŠENÍ NATAHOVAČŮ HODINEK	8
2.2	AMATÉRSKÁ ŘEŠENÍ NATAHOVAČŮ HODINEK	11
2.3	TECHNICKÉ POŽADAVKY NA NATAHOVAČE HODINEK	14
3	ZÁKLADNÍ VERZE NATAHOVAČE HODINEK „WATCH WINDER ORBIT PEPPERMINT“	15
3.1	POPIS KONSTRUKCE	15
3.2	TABULKA SOUČÁSTEK ŘÍDÍCÍ JEDNOTKY	17
3.3	SCHÉMATA ZAPOJENÍ PLOŠNÉHO SPOJE	18
3.4	ALGORITMUS PROGRAMU	20
3.5	VÝPIS PROGRAMU	21
3.6	FOTODOKUMENTACE	22
3.7	ZHODNOCENÍ ZÁKLADNÍ VERZE	23
4	POKROČILÁ VERZE NATAHOVAČE HODINEK „WATCH WINDER MULTI DELUXE“	24
4.1	POPIS KONSTRUKCE	24
4.2	TABULKA SOUČÁSTEK	27
4.3	SCHÉMATA ZAPOJENÍ	29
4.4	ALGORITMUS PROGRAMU	31
4.5	VÝPIS PROGRAMU	32
4.6	FOTODOKUMENTACE	37
4.7	ZHODNOCENÍ POKROČILÉ VERZE	40
5	FINÁLNÍ MODEL ROTAČNÍHO ZAŘÍZENÍ	41
5.1	POPIS VÝROBKU	41
5.2	TABULKA SOUČÁSTEK OVLÁDACÍ DESKY (SHIELD) PRO ARDUINO UNO	41
5.3	SCHÉMATA PLOŠNÉHO SPOJE OVLÁDACÍ DESKY	43
5.4	MODEL KRABÍČKY ŘÍDÍCÍ JEDNOTKY	46
5.5	POSTUP DEMONTÁŽE TISKÁRNY	49
5.6	ALGORITMUS PROGRAMU	55
5.7	VÝPIS PROGRAMU PRO ARDUINO UNO	56
5.8	NÁVOD K MONTÁŽI ZAŘÍZENÍ	61
5.9	TECHNICKÉ PARAMETRY ZAŘÍZENÍ	65
5.10	NÁVOD K POUŽITÍ ZAŘÍZENÍ	66
6	ZÁVĚR	68
6.1	SEZNAM POUŽITÝCH OBRÁZKŮ	69
6.2	SEZNAM POUŽITÝCH TABULEK	70
6.3	SEZNAM POUŽITÝCH ODKAZŮ	71
6.4	SEZNAM POUŽITÝCH ZKRATEK	72
6.5	SEZNAM PŘÍLOH ELEKTRONICKÉ VERZE SOČ	72

1 Úvod

Staré vyřazené tiskárny jsou nečekaně bohatým zdrojem využitelných součástek, například stejnosměrných i krokových motorů, kompletních mechanismů rotačního i lineárního pohybu a přesných optických enkodérů pro tyto pohyby, optických závor, adaptérů nebo desek síťového napájení, desek s ovládacími tlačítky, plochých FFC kabelů, kovových a plastových dílů (podrobněji viz kapitolu 5.5), které mohou být použity amatéry i profesionály ke konstrukci řady elektronických i mechanických výrobků. Zejména inkoustové tiskárny vyhazované do elektronického odpadu, získatelné zdarma z firem či domácností nebo levně poříditelné z aukčních portálů (Aukro, eBay) jsou výborným zdrojem prakticky neporušených součástek, protože nejčastěji přestaly fungovat kvůli zaschnutí inkoustu v tiskové hlavě nebo prasknutí některé plastové zarážky.

Příkladem různých výrobků sestavených z dílů starých tiskáren jsou i výrobky našeho týmu HaHaHa Robotics, zhlédnutelné na stránkách www.hahaharobotics.tk.

V této práci jsem se rozhodla využít díly ze starých tiskáren pro sestavení levných, výkonných a uživatelsky nenáročných natahovačů automatických hodinek, anglicky nazývaných watch winder. Automatické hodinky jsou náramkové hodinky, které k chodu nepotřebují baterii, ale nabíjí se pomocí v hodinkách uloženého závažíčka, překlápěného pohybem ruky za chůze. Můj soused je sběratelem těchto hodinek a má jich tolik, že je nemůže všechny nosit najednou, takže si je potřebuje čas od času nabít. Cena komerčně dostupných natahovačů pro větší počet hodinek se ale pohybuje řádově v desítkách tisíc korun. Soused mne tedy požádal, abych mu vyrobila amatérskou verzi těchto natahovačů, s cenou součástek řádově ve stovkách korun.

V průběhu řešení jsem přišla na to, že použitá konstrukce rotačního mechanismu staré vyřazené tiskárny může mít i širší využití, např. pro levné mísiče vzorků v chemických a biologických laboratořích (kultivátory buněk), homogenizátory barev ve sprejích, míchání potravin či otáčení vajec v inkubátorech a rozhodla jsem se zpracovat obecněji zaměřený „open hardware + open software“ model levného mísiče s použitím běžně dostupné řídicí desky Arduino Uno, aby mohl být tento nápad případně využit dalšími začínajícími mladými vědci či techniky.

2 Přehled komerčních a amatérských řešení natahovačů hodinek

Tato kapitola obsahuje stručný přehled hlavních typů komerčních i amatérských typů natahovačů hodinek, náhodně vybraných na internetu, pro získání představy o konstrukčních řešeních použitelných pro návrh svých prototypů natahovačů hodinek.

2.1 Komerční řešení natahovačů hodinek

Název	Výrobce	Rozměry (mm)	Hmotnost (kg)	Způsob ovládání	Programy	LCD	Počet hodinek	Cena (Kč)
TIMELESS 102 031 KRAFF^[1]	KRAFF	320 x 240 x 210	4,8	potenciometry	4	ne	4	7 642
WOLF 70039/18 Viceroy 3+6^[2]	WOLF DESINGS	445 x 223 x 197	6	potenciometry	57	ano	3	29 160
MARS 201011^[3]	KRAFF	170 x 240 x 190	2	potenciometry	4	ne	2	5 669
KRAFF 196 012 Saturn^[4]	KRAFF	390 x 290 x 200	7	tlačítka	12	ano	6	13 410
WOLF 70039/23 Roadstar 1+3^[5]	WOLF DESINGS	191 x 223 x 197	3	potenciometry	57	ano	1	11 720
Senna Klarstein^[6]	Klarstein	110 x 150 x 130	0,5	tlačítkem	4	ne	1	1 999
Klagenfurt Klarstein^[7]	Klarstein	335 x 525 x 205	6,6	tlačítka	neomezeně	ano	12	22 590
WOLF Viceroy Watch Winder^[8]	WOLF DESINGS	19,05 x 20,32 x 22,86	neznámá	potenciometry	neomezeně	ano	1	-
CHIYODA Watch Winder^[9]	Rolex Tips and Tricks	neznámé	neznámá	potenciometr	4	ne	1	-
Wolf Designs Viceroy 6 Watch Winder^[10]	aBlog-toWatch.com	41,91 x 15,88 x 32,39	neznámá	potenciometry	neomezeně	ano	6	-

Tabulka 1: Tabulka komerčních natahovačů hodinek



Obrázek 1: TIMELESS 102 031 KRAFF^[1]



Obrázek 2: WOLF 70039/18 Viceroy 3+6^[2]



Obrázek 3: MARS 201011^[3]



Obrázek 4: KRAFF 196 012 Saturn^[4]



Obrázek 5: WOLF 70039/23 Roadstar 1+3^[5]



Obrázek 6: Senna Klarstein^[6]



Obrázek 7: Klagenfurt Klarstein^[7]



Obrázek 8: WOLF Viceroy Watch Winder^[8]



Obrázek 9: CHIYODA Watch Winder^[9]



Obrázek 10: Wolf Designs Viceroy 6 Watch Winder^[10]

2.2 Amatérská řešení natahovačů hodinek

Název	Autor	Způsob ovládnání	Napájení	Programy	LCD	Počet hodinek	Materiál
DIY watch winder^[11]	kha-nhquagioi	vypínač	ze zdroje	ne	ne	4	4 plechovky držené na dřevěné hřídeli na motoru, podpíraném železnou konstrukcí
DIY Homemade Watch Winder^[12]	The-MrJava-guy	vypínač	ze zdroje	ne	ne	1	plastový držák připevněný zátkou od vína k motoru, držéném plastovou konstrukcí
Cheapest homemade diy watch winder^[13]	Ted Valez	neovládáno	ze zdroje	ne	ne	1	plastový kelímek připevněný k motoru, držéném dřevěnou konstrukcí
Homemade Automatic Watch Winder^[14]	whaczup	vypínač	ze zdroje	ne	ne	2	válec s hodinkami připevněný na plastové hřídeli na motoru
Homemade Automatic Watch Winder^[15]	SoCal Skateboards	neovládáno	ze zdroje	ne	ne	4	dřevěný kříž připevněný k motoru, držéném železnou konstrukcí
DIY watch winder by wood^[16]	Pham KS	vypínač	ze zdroje	ne	ne	1	plastová krabička s hodinkami, připevněná k hřídeli motoru v dřevěné krabičce
DIY Watch Winder^[17]	Best Video From Cambodia	vypínač	ze zdroje	ne	ne	1	plechovka s hodinkami umístěná na hřídeli motoru v železné krabičce
My Kinetic watch charger^[18]	Mortnoiruk	tlačítko	ze dvou AA baterií	ne	ne	1	hodinky držené plasickou hmotou na hřídeli zřevodovaného motoru, držéném plastovou konstrukcí na dřevěném podkladu
Automatic Watch Winder DIY^[19]	Horology Zone	vypínač	ze zdroje	ne	ne	4	hodinky uchycené v plastové pění v plastovém kelímku na hřídeli motoru, umístěného v plastové krabičce
Lego Watch Winder (Mk 4)^[20]	Eddie Koh	potenciometr	ze zdroje	neomezeně	ano	1	válec s hodinkami, točený pneumatikami, připevněnými k motoru na lego podkladu

Tabulka 2: Tabulka amatérských řešení natahovačů hodinek



Obrázek 11: DIY watch winder od kha-nhquagioi^[11]



Obrázek 12: DIY Homemade Watch Winder od TheMrJavaguy^[12]



Obrázek 13: Cheapest homemade diy watch winder od Ted Valez^[13]



Obrázek 14: Homemade Automatic Watch Winder od whaczup^[14]



Obrázek 15: Homemade Automatic Watch Winder od SoCal Skateboards^[15]



Obrázek 16: DIY watch winder by wood od Pham KS^[16]



Obrázek 17: DIY Watch Winder od Best Video From Cambodia^[17]



Obrázek 18: My Kinetic watch charger od Mortnoiruk^[18]



Obrázek 19: Automatic Watch Winder od Horology Zone



DIY Obrázek 20: Lego Watch Winder (Mk 4) od Eddie Koh

2.3 Technické požadavky na natahovače hodinek

Rozborem konstrukce amatérských a komerčních řešení natahovačů automatických hodinek a konzultací se zadavatelem práce (sběratelem automatických hodinek) bylo zjištěno, že natahovač automatických hodinek by měl mít tyto hlavní parametry:

Základní (minimální) verze:

- minimální cena (výrobek vyroben z recyklovaných elektronických součástek);
- jednoduchý na obsluhu;
- nastavitelné otáčky v rozmezí řádově desítek až stovek otáček za minutu;
- možnost změny směru otáčení;
- napájení bezpečným napětím.

Pokročilá verze:

- kapacita pro co největší počet hodinek;
- nastavení několika různých kroků otáčení s volitelnými parametry;
- LCD displej;
- ovládání joystickem;
- počítadlo rychlosti a doby otáčení;
- ochrana proti přetížení motoru;

S tímto zadáním byly zkonstruovány dvě verze natahovačů automatických hodinek – základní a pokročilá, popisované v následujících kapitolách.

3 Základní verze natahovače hodinek „Watch Winder Orbit Peppermint“



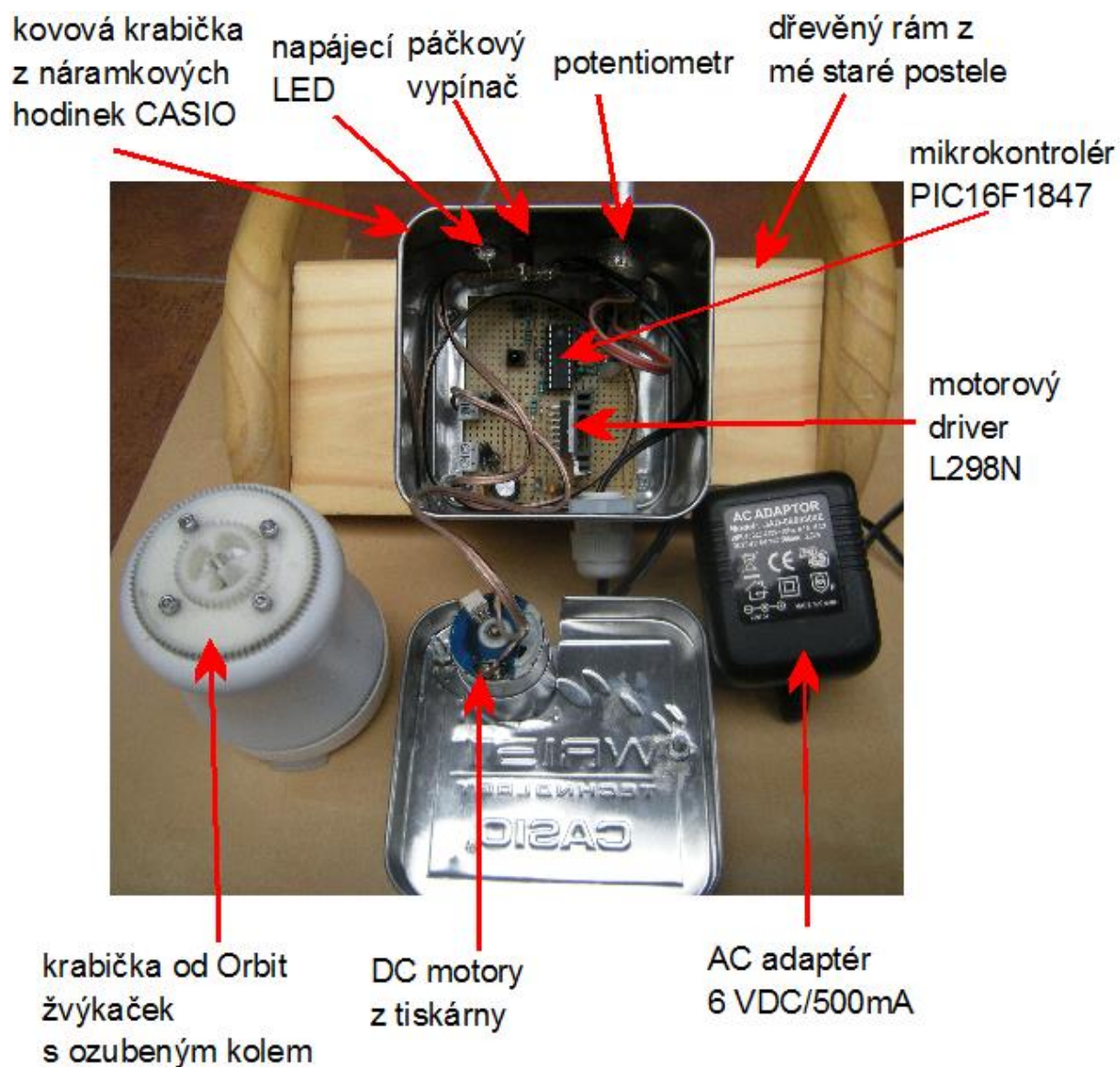
Obrázek 21: Celková sestava Watch Winder Orbit Peppermint

3.1 Popis konstrukce

Základní verze natahovače hodinek nazvaná podle použité krabičky „Watch Winder Orbit Peppermint“ obsahuje řídicí desku z mého starého robota, sledovače černé čáry [BÉDA4^{\[21\]}](#). Plošný spoj s mikrokontrolérem PIC16F1847 a obousměrným motorovým driverem L298N zajišťuje kontrolu rychlosti a směru otáčení zpřevodovaného motoru ze staré tiskárny neznámého typu. K řízení směru a rychlosti otáčení slouží 10kΩ potenciometr ze starého rádia. Deska je napájena z AC adaptéru poskytujícího 6 VDC / 500 mA. Současná spotřeba při provozu je cca 250 mA (1,5 W).

Plastové ozubené kolo z převodu tiskárny je přišroubováno ke spodní části plastové krabičky od žvýkaček Orbit Peppermint, do které jsou vloženy náramkové hodinky držené pěnovým plastem. Motor s převodovkou je přišroubován k víku kovové krabičky od hodinek značky Casio, její dno je pak přišroubováno k dřevěnému rámu z mé staré postele.

Návod na Watch Winder Orbit Peppermint jsem publikovala dne 14.7.2015 na fóru Let's Make Robots^[22], proto jsou některé popisy v angličtině. Na této stránce je dostupné i video chodu natahovače.



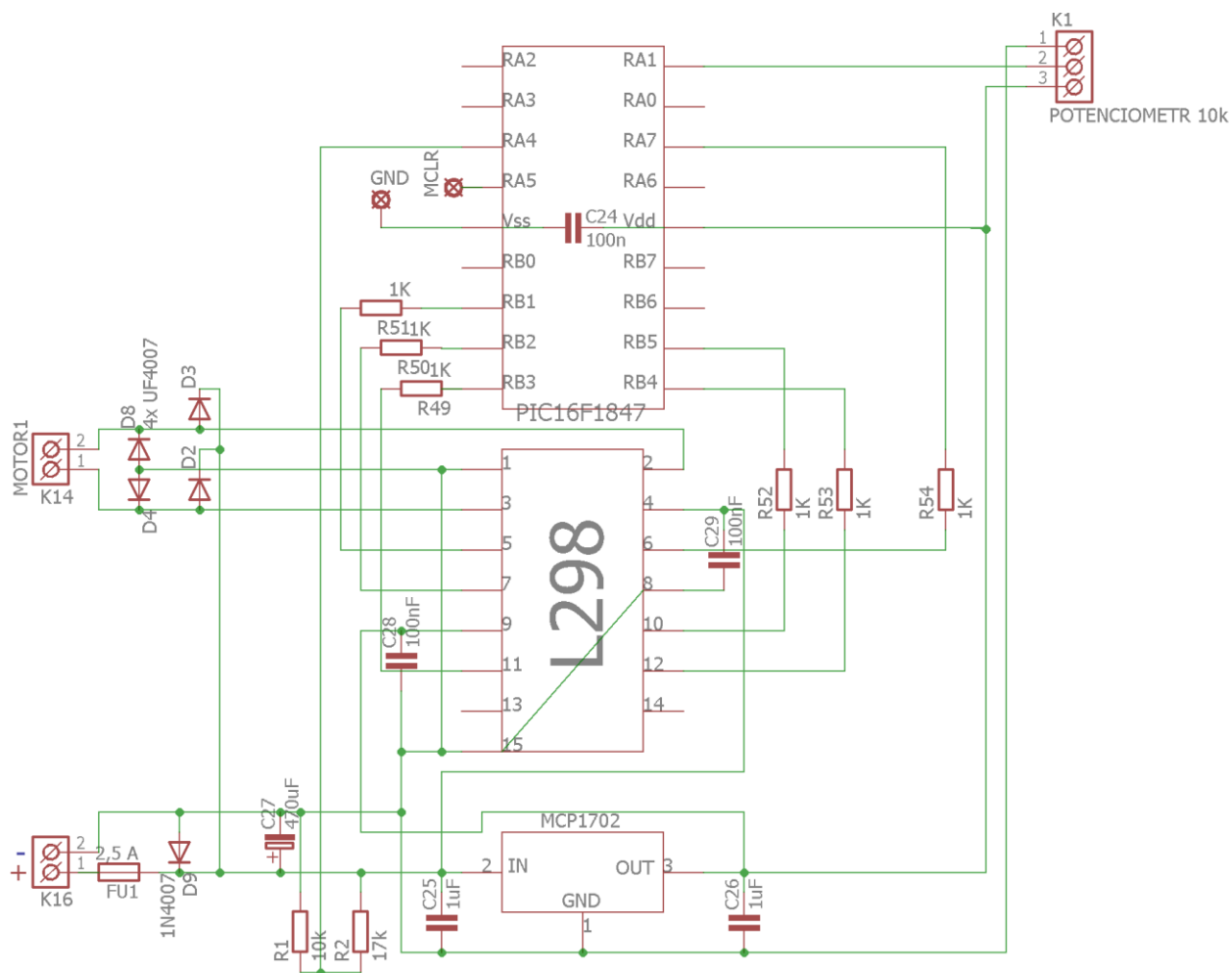
Obrázek 22: Obrázek součástí natahovače Watch Winder Orbit Peppermint

3.2 Tabulka součástek řídicí jednotky

Kód	Typ	Parametry	Funkce	Kód (TME)	Cena (Kč)
C24	kondenzátor	keramický 100 nF SMD 0805	filtrace napájení mikrokontroléru	02016D104KAT2A	0,13
C25 - C26	kondenzátor	keramický 1 µF THT	stabilizace MCP1702	0201X105K6R3CT	0,54
C27	kondenzátor	elektrolytický 470 µF / 50V THT	filtrace a ochrana napájení	UVR1H471MHD	6,34
C28 - C29	kondenzátor	keramický 100 nF THT	filtrace výstupu driveru	CM-100N	0,76
D1- D4, D8	dioda	UF4007 THT	ochrana výstupu driveru	UF4007	1,30
D9	dioda	1N4007 THT	polaritní ochrana	1N4007-DC	0,66
FU1	polymerová pojistka	2,5 A THT	protiproudová a polaritní ochrana	30R250UU	7,56
IC1	MCU	PIC16F1847 THT	mikrokontrolér	PIC16F1847-I/P	55,77
IC2	stabilizátor	MCP1702-5000, 5 V, 250 mA THT	5 V napájecí stabilizátor	MCP1702-5002ET	16,30
IC3	driver	L298N THT	obousměrný motorový driver	L298N	67,49
K1	svorkovnice	ARK500/3	připojení potenciometru	TB-5.0-P-3P/GY	5,31
K15 - K16	svorkovnice	ARK500/2	připojení motorů a napájení	TB-5.0-P-2P/GY	4,09
PAD1 - PAD5	piny		programovací piny pro mikrokontrolér	ZL201-20G	4,39
	plošný spoj	univerzální plošný spoj řádkový	plošný spoj	H25SR160	51,34
P1	potenciometr	10 kΩ , vymontovaný ze starého rádia	regulace rychlosti motoru	ze starého rádia	-
R1	rezistor	10 kΩ 0,6 W	napěťový dělič pro kontrolu napájení	1/2W-10K	0,41
R2	rezistor	18 kΩ 0,6 W	napěťový dělič pro kontrolu napájení	1/4W18K	0,20
R49 - R54	rezistor	1 kΩ, 0,6 W	ochrana výstupu mikrokontroléru na driver (motor 1 s rezistory nezapojen)	OX102KE	0,41
S1	vypínač	páčkový vypínač	hlavní napájecí vypínač	TS-14	13,48
Celkem					237

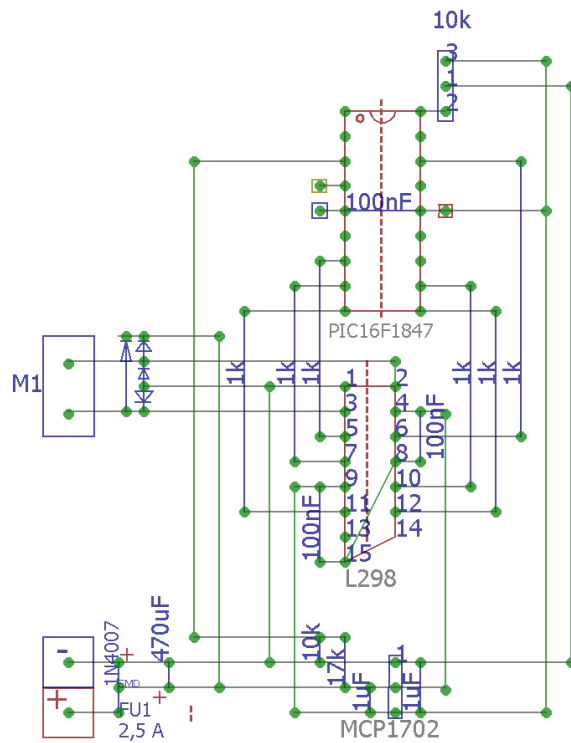
Tabulka 3: Watch Winder Orbit Peppermint, tabulka součástek

3.3 Schémata zapojení plošného spoje

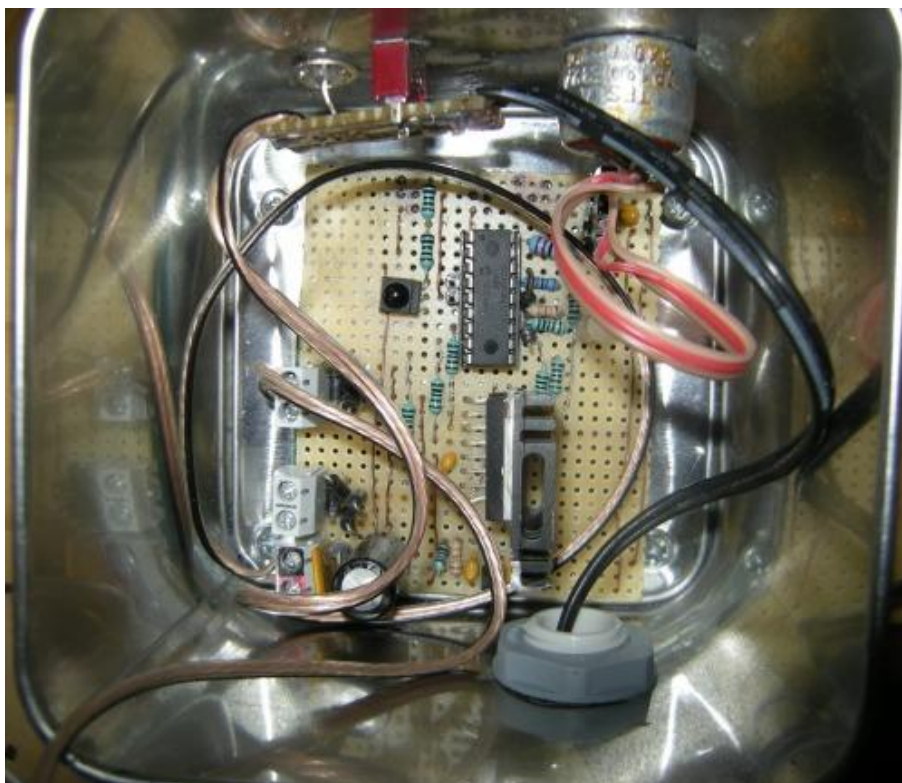


Obrázek 23: Watch Winder Orbit Peppermint, funkční schéma

Pozn.: Ze schématu jsou odstraněny díly z původní desky robota Bědy 4, které nejsou ve Watch Winderu Orbit Peppermint využity.

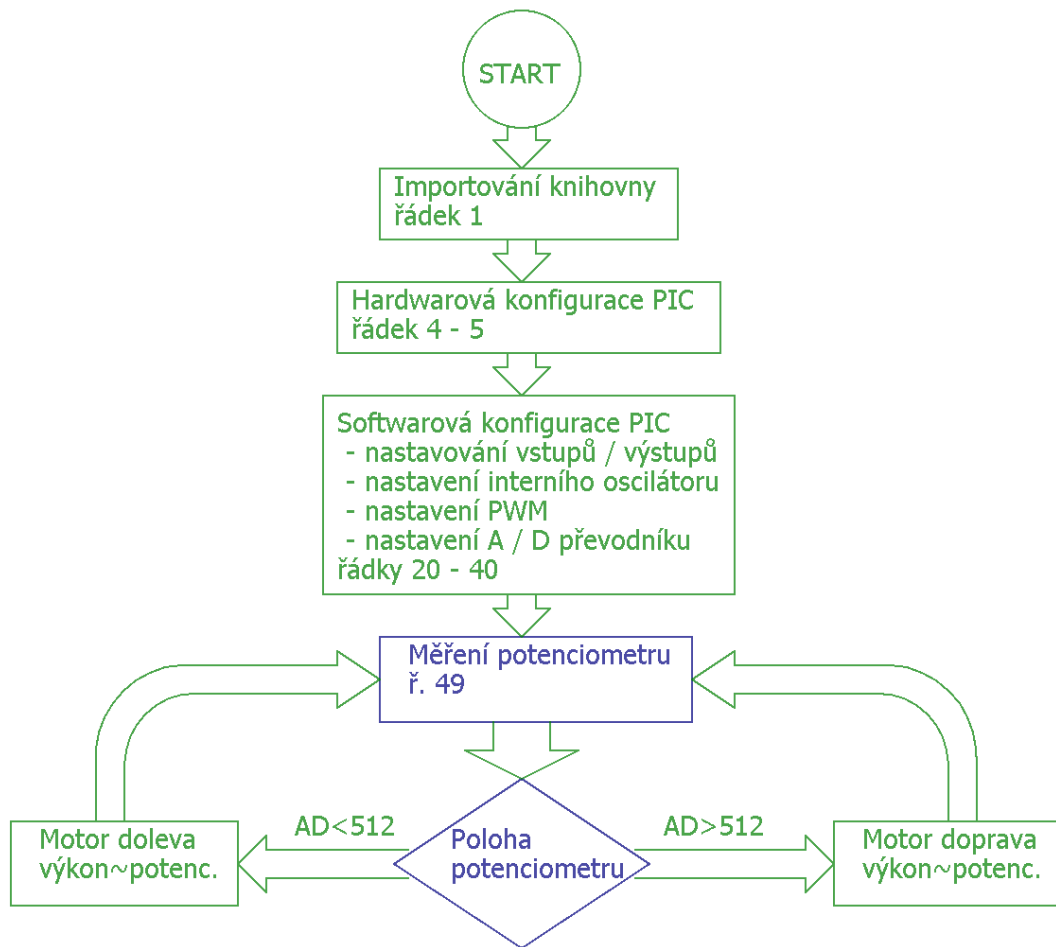


Obrázek 24: Watch Winder Orbit Peppermint, pájecí schéma na univerzálním řádkovém plošném spoji (rozteč = 2,54 mm)



Obrázek 25: Řídící deska Watch Winder Orbit Peppermint s krabičkou

3.4 Algoritmus programu



Obrázek 26: Watch Winder Orbit Peppermint algoritmus programu

3.5 Výpis programu

```
1 #include <xc.h>
2
3 //zakladni konfigurace PIC16F1847
4 #pragma config FOSC=INTOSC,WDTE=OFF,PWRITE=OFF,MCLRE=OFF,CP=OFF,CPD=OFF,BOREN=ON,CLKOUTEN=OFF,IE
5 #pragma config WRTI=OFF,PLLEN=OFF,STVREN=ON,BORV=LO,LVP=OFF
6
7 #define IN1          LATBbits.LATB1
8 #define IN2          LATBbits.LATB2
9 #define _XTAL_FREQ  16000000
10
11 long Potent=0,power=0;
12
13 void Wait(long ms);
14 void PIC_Init();
15 void POTENCIOMETR();
16 void MOTOR_L(int powerL);
17 void MOTOR_R(int powerR);
18
19
20 void PIC_Init()
21 {
22     TRISA=0b00111111; TRISB=0b00000001;// nastavování vstupu=1/vystupu=0
23     ANSELA=0b00010000; ANSELB=0b00000000;// nastavování analogových=1/digitalnich=0 vstupu
24     LATA=0; LATB=0; // vynulování vstupu
25
26     OSCCON=0b01111010; // 16 MHz interni oscilator
27     APFCON0bits.CCP2SEL=1; // PWM CCP2 na RA7
28
29     //nastavování PWM na RA7, PWM frekvence = 244 Hz
30     TRISAbits.TRISA7=1;
31     PR4=255;
32     CCP2CON=0b00001100;
33     CCP2L=0;
34     PIR3bits.TMR4IF=0;
35     T2CON=0b00000111; // preddelicka 64x pro PWM
36     TRISAbits.TRISA7=0;
37
38     // nastavování A/D převodníku
39     ADCON1=0b10100000; // zarovnaní doprava, konverzní frekv. FOSC/32, VDD reference
40     ADCON0=0b00010001; // ADC na kanál RA4=AN4, ADC modul je zapnutý
41 }
42
43 void main()
44 {
45     PIC_Init();
46
47     while(1)
48     {
49         POTENCIOMETR();
50         if(Potent<512)
51             {power=255-(Potent/2);
52             MOTOR_L(power);}
53         else {
54             power=(Potent-512)/2;
55             MOTOR_R(power);}
56     }
57
58
59 void MOTOR_L(int powerL)
60 {
61     IN1=1;IN2=0;
62     CCP2L=powerL;
63 }
64
65 void MOTOR_R(int powerR)
66 {
67     IN1=0;IN2=1;
68     CCP2L=powerR;
69 }
70
71 void Wait(long ms)
72 {
73     for(long bb=0;bb<ms;bb++) __delay_ms(1);
74 }
75 void POTENCIOMETR()
76 {
77     Potent=0;
78     for(int i=0;i<100;i++)
79     {
80         ADCON0=0b00000111; // Spustení ADC konverze
81         while(ADCON0==0b00000111); // čekání na konverzi
82         Potent+=ADRES;
83     }
84     Potent=Potent/100;
85 }
```

Obrázek 27: Watch Winder Orbit Peppermint, výpis programu pro PIC16F1847

3.6 Fotodokumentace



Obrázek 28: Hlavní díly ke stavbě Watch Winder Orbit Peppermint



Obrázek 29: Watch Winder Orbit Peppermint, plastová krabička s automatickými hodinami



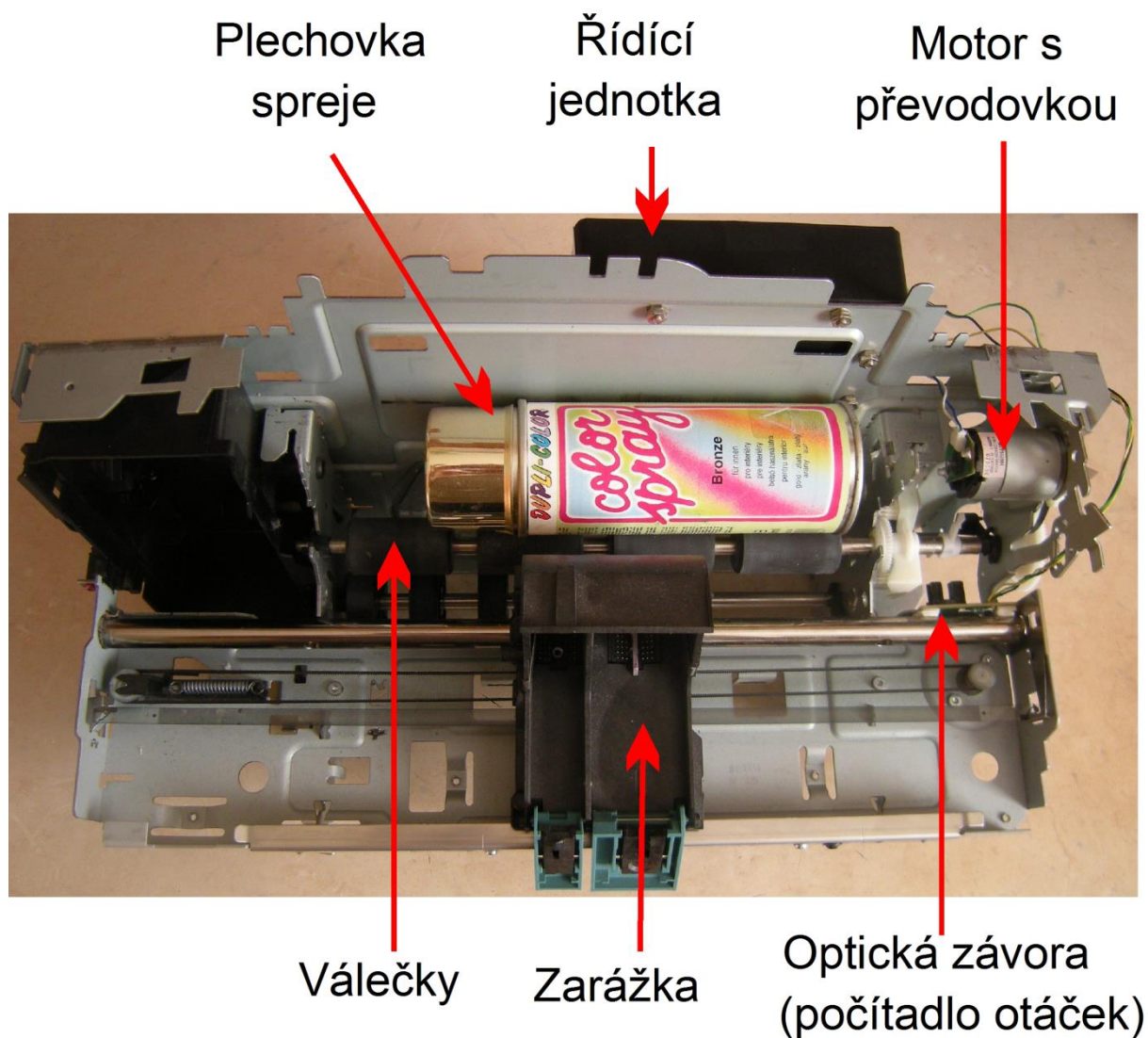
Obrázek 30: Zkompletovaný Watch Winder Orbit Peppermint

3.7 Zhodnocení základní verze

První prototyp základní verze natahovače hodinek splnil požadavky specifikované pro tuto verzi v zadání (kapitola 2.3):

- minimální cena (výrobek vyroben z recyklovaných elektronických součástek)
 - materiálové náklady na výrobu jsou 238 Kč
 - k výrobě byly použity staré domácí elektrospotřebiče a díly starého nábytku
- jednoduchý na obsluhu
 - ovládal se pouze spouštěcím vypínačem a potenciometrem
- nastavitelné otáčky v řádově desítek až stovek otáček za minutu
 - výkon motoru je plynule volitelný potenciometrem.
- možnost změny směru otáčení
 - směr otáčení se rovněž volí potenciometrem přes středovou nulovou polohu
- napájení bezpečným napětím
 - napájen ze síťového zdroje napětí 6 VDC / 500 mA

4 Pokročilá verze natahovače hodinek „Watch Winder Multi Deluxe“



Obrázek 31: Watch winder multi deluxe, profilový obrázek

4.1 Popis konstrukce

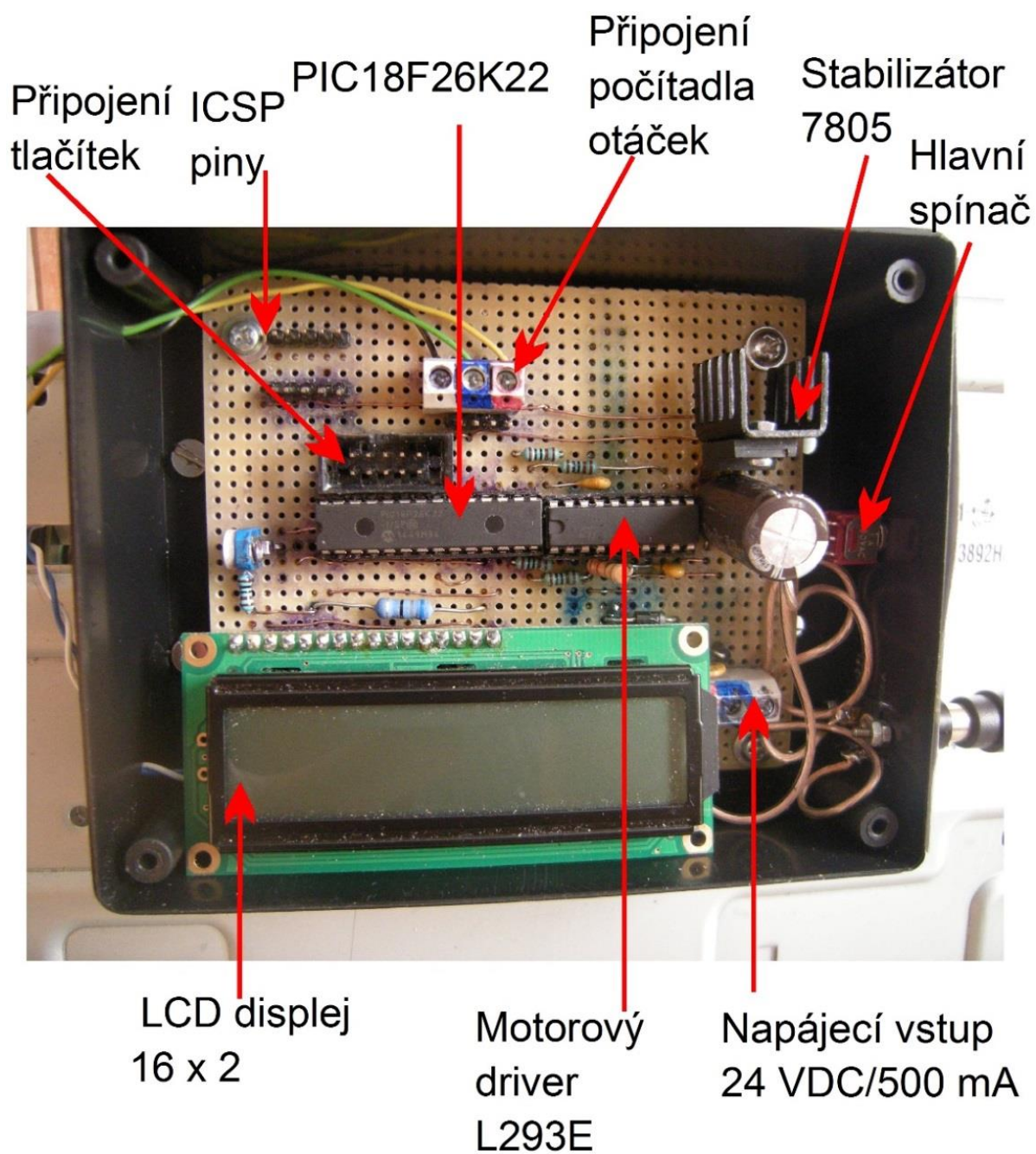
Ke konstrukci pokročilé verze automatického natahovače hodinek jsem zvolila starou inkoustovou tiskárnu HP Deskjet PhotoSmart 1215, z níž byl použit celý ocelový rám s zpřevodovaným stejnosměrným DC motorem pohánějícím válečky pro podávání papíru. Původní tisková hlava byla použita jako zábránka proti vypadnutí tuby s hodinkami. K počítání otáček bylo použito počítadlo otáček z optické závory (fotopřerušovače), použité jinde v této tiskárně, v kombinaci s plastovou ručkou nalepenou na rotační hřídeli.

Řídící deska je umístěna v krabici na zadní straně rámu tiskárny a je napájena síťovým zdrojem (max. 24 VDC/500 mA). Uživatelské rozhraní je tvořeno alfanumerickým LCD displejem a pěti tlačítky. V prvním kroku uživatel zadá počet kroků otáčení. Potom uživatel nastaví dobu otáčení, směr a rychlost otáčení v každém kroku. Po potvrzení startu program měří čas každé otáčky a reguluje PWM výstup na motor pro udržení otáček na požadovanou hodnotu. Program rovněž měří proud na motoru a vypne výstup, pokud je zjištěna hodnota nad 500 mA. Motor se také zastaví v případě, že rychlost otáčení klesne pod 0,5 otáčky za sekundu (zablokování válců). Efektivní rozsah otáčení je 1,0 – 10,0 otáček za sekundu. Hodinky se vloží do plastové tuby od malé hobby vrtačky a jsou od sebe vzájemně odděleny pěnovým plastem.

Návod na Watch Winder Multi Deluxe jsem publikovala dne 17.11.2015 na fóru Let's Make Robots ^[23], proto jsou některé popisky programu v angličtině. Na této stránce bylo publikováno i video z chodu zařízení.

Hlavními součástkami zařízení jsou:

- Procesor: PIC18F26K22
- Vstupy: 5 tlačítek pro řízení displeje, optická závora pro počítání otáček. měření proudu na motoru přes driver L293E.
- Výstupy: obousměrný motorový driver pro řízení motoru, LCD alfanumerický displej 16x2.
- Napájení: 24 VDC/500 mA síťový zdroj, 5 VDC přes 7805 lineární stabilizátor pro procesor a zbytek.
- Plošný spoj: univerzální řádkový plošný spoj (e=2,54 mm).



Obrázek 32: Watch winder multi deluxe, řídicí deska s popisky

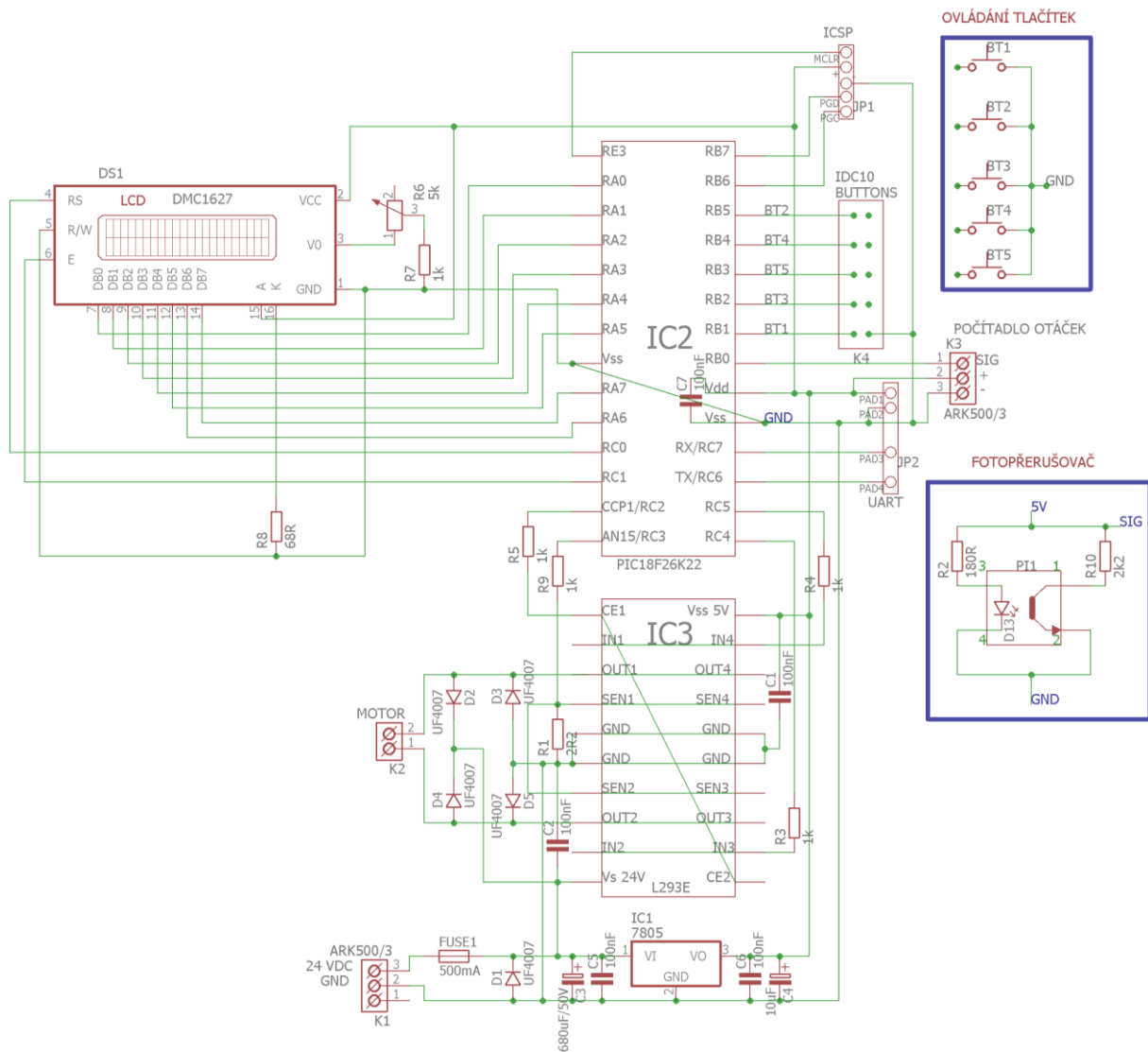
4.2 Tabulka součástek

Kód	Typ	Parametry	Funkce	Kód (TME)	Cena (Kč)
C1	kondenzátor	keramický 100 nF, THT	filtrace napájení drive- ru	CM-100N	0,76
C2	kondenzátor	keramický 100 nF, THT	filtrace napájení drive- ru	CM-100N	0,76
C3	kondenzátor	elektrolytický 680 µF, THT	filtrace napájení	WL1H687M1 2025BB	5,07
C4	kondenzátor	SMD 0805 keramický 10 µF	výstupní kondenzátor stabilizátoru 7805	0805X106K10 OCT	0,37
C5	kondenzátor	SMD 0805 keramický 100 nF	vstupní kondenzátor stabilizátoru	CL05A104KP 5NNNC	0,11
C6	kondenzátor	SMD 0805 keramický 100 nF	výstupní kondenzátor stabilizátoru	CL05A104KP 5NNNC	0,11
C7	kondenzátor	keramický 100 nF, THT	filtrace napájení proces- oru	CM-100N	0,76
D1	dioda	UF4007, THT	polaritní ochrana napá- jení	UF4007	1,3
D2-D5	dioda	UF4007, THT	ochrana výstupu drive- ru	UF4007	1,3
DS1	LCD displej 16x2	RC1602B2- GHW-CSX	nastavování a zobra- zování programových parametrů	RC1602B2- GHW-CSX	158,58
FU1	polymerová pojistka	500 mA, THT	proudová a polaritní ochrana	EX050-V1	8,7
IC1	stabilizátor	7805	5V stabilizátor	L7805ABV	5,7
IC2	MCU	PIC18F26K22	mikrokontrolér	PIC18F26K22 -I/SP	72,43
IC3	motorový driver	L293E, THT	obousměrný motorový driver	L293E	91,49
JP1	piny		ISCP programovací piny	ZL201-20G	4,39
JP2	piny		UART komunikace (nepoužita)	ZL201-20G	
K1	svorkovnice	ARK500/3, THT	připojení hlavního napájení	TB-5.0-P- 3P/GY	5,31
K2	svorkovnice	ARK500/2, THT	připojení motoru	TB-5.0-P- 2P/GY	4,09
K3	svorkovnice	ARK500/3, THT	připojení počítačla otáček	TB-5.0-P- 3P/GY	5,31
K4	IDC konek- tor	IDC10, THT	připojení nastavova- cích tlačítek	ZL231-10PG	2,25

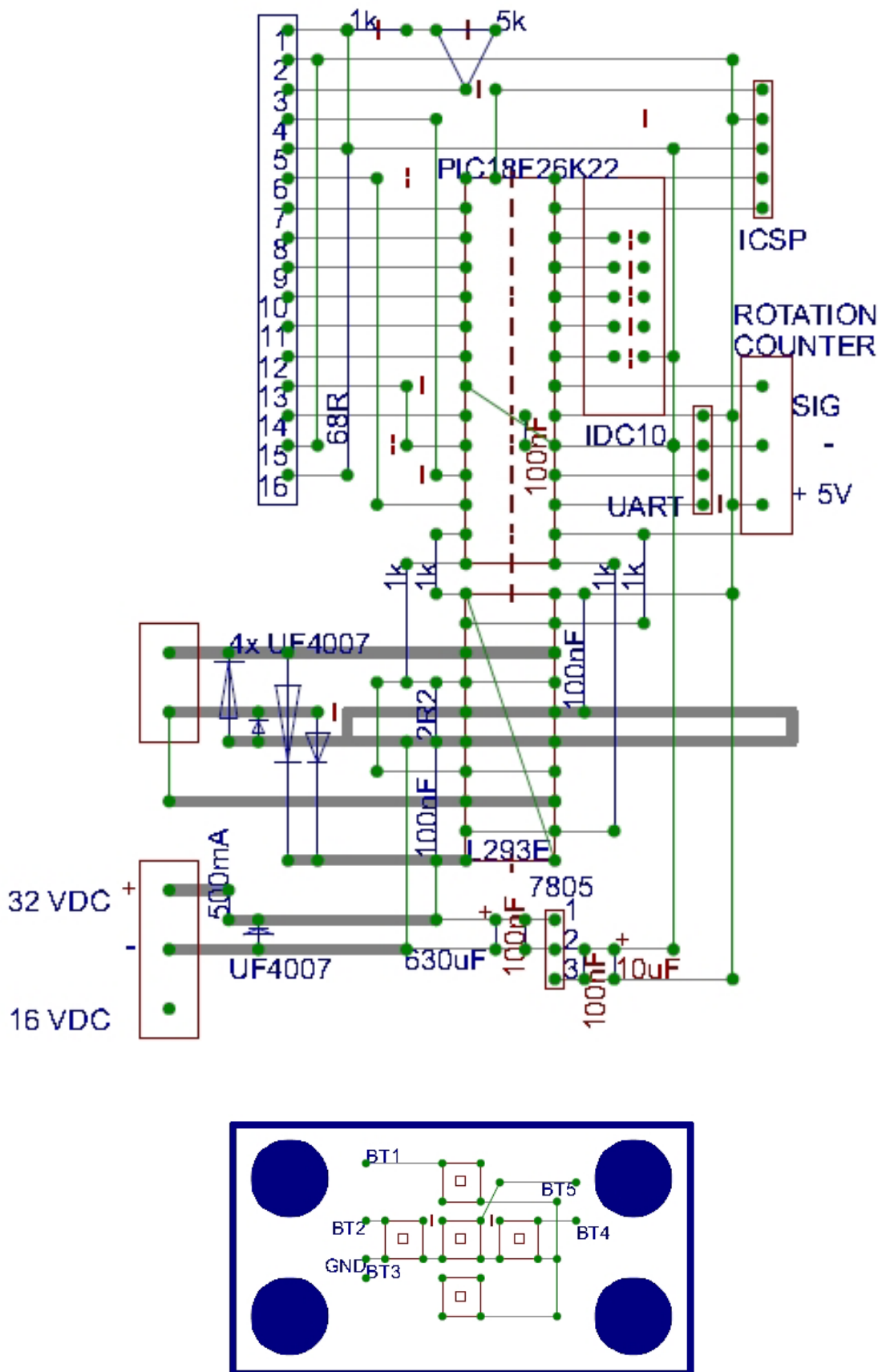
M1	motor	motor s převodovkou (C6429-60004, MITSUMI) z HP DJ PhotoSmart 1215	hlavní motor		
PI1	optická závora	z HP tiskárny	počítadlo otáček		
	plošný spoj	univerzální plošný spoj řádkový, rozteč 2,54 mm	plošný spoj	H25SR160	51,34
R1	rezistor	2,2 Ω, THT, 0,6 W	bočníkový rezistor pro měření proudu	OX222KE	0,41
R2	rezistor	180 Ω, THT, 0,6 W	vstupní rezistor optické závory	KNPA1W-180R	0,41
R3	rezistor	1 kΩ THT, 0,6 W	vstupní ochrana motorového driveru	OX102KE	0,41
R4	rezistor	1 kΩ THT, 0,6 W	vstupní ochrana motorového driveru	OX102KE	0,41
R5	rezistor	1 kΩ THT, 0,6 W	vstupní ochrana motorového driveru	OX102KE	0,41
R6	trimr	5 kΩ THT, 0,6 W	nastavení kontrastu LCD displeje	1028G-5K	9,04
R7	rezistor	1 kΩ THT, 0,6 W	nastavení kontrastu LCD displeje	OX102KE	0,41
R8	rezistor	68 Ω	nastavení podsvícení LCD displeje	KNPA1W-68R	0,2
R9	rezistor	1 kΩ THT, 0,6 W	analogová vstupní ochrana mikroprocesoru	OX102KE	0,41
R10	rezistor	2,2 kΩ THT, 0,6 W	výstupní rezistor optické závory	OX222KE	0,41
S1	spínač	páčkový spínač	hlavní napájecí vypínač	TS-14	13,48
Celkem					447

Tabulka 4: Watch Winder Multi Deluxe, tabulka součástek

4.3 Schémata zapojení

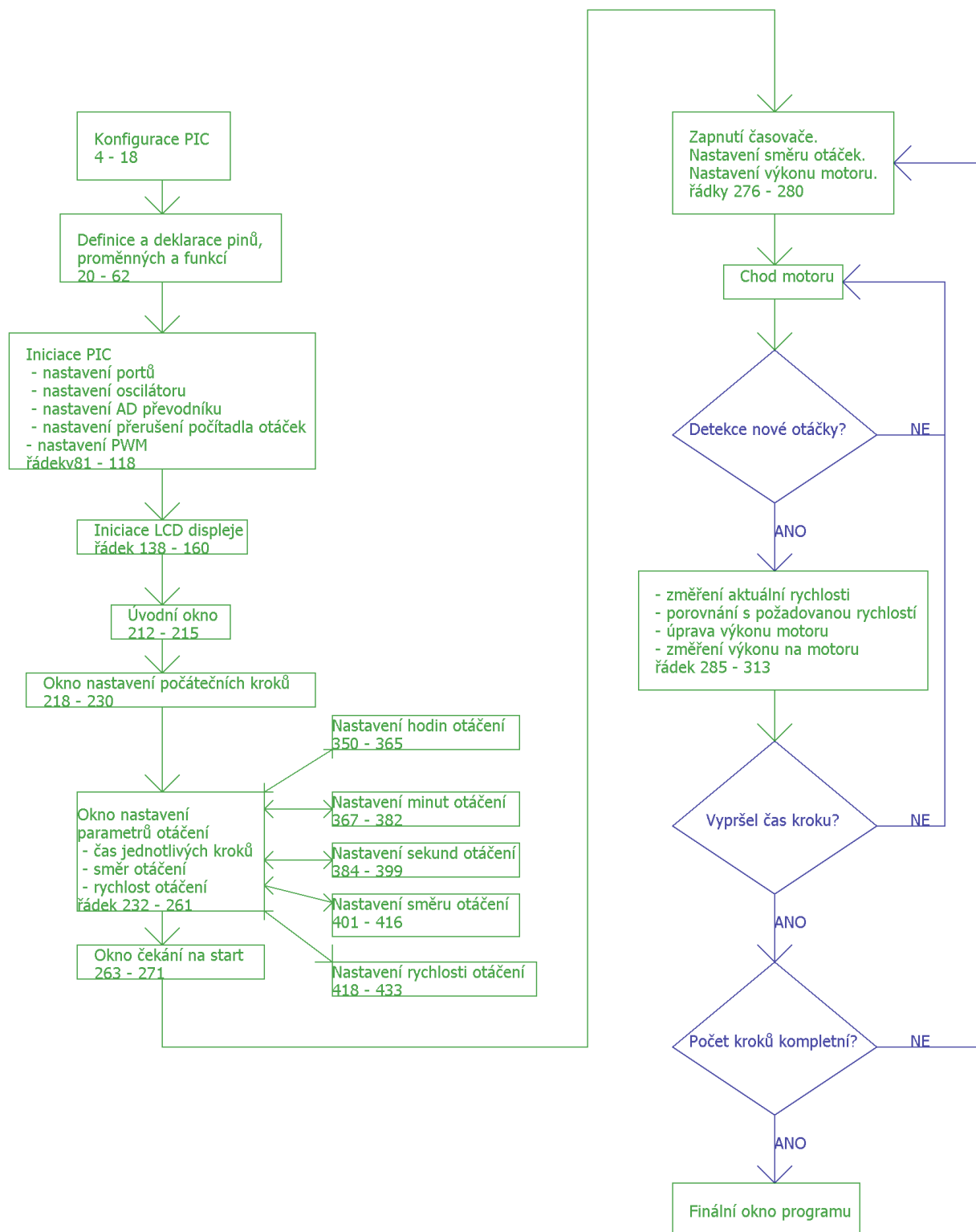


Obrázek 33: Watch winder multi deluxe, funkční schéma



Obrázek 34: Watch winder multi deluxe, pájecí schéma (univerzální řádkový plošný spoj; $e=2,54\text{ mm}$)

4.4 Algoritmus programu



Obrázek 35: Watch Winder Multi Deluxe, algoritmus programu

4.5 Výpis programu

```
1 // Multifunkční rotací zařízení
2
3 // Importované knihovny
4 #include <xc.h>
5 #include <stdio.h>
6 #include <math.h>
7 #include <plib/timers.h>
8 #include <plib/usart.h>
9 #include <plib/pwm.h>
10 #include <stdbool.h>
11
12 // konfigurace PIC18F26K22
13 #pragma config FOSC=INTIO67, PLLCFG=ON, PRICLKEN=OFF, FCMEN=OFF, IESO=OFF, PWRTEN=ON
14 #pragma config BOREN=OFF, BORV=190, WDTE=OFF, WDTPS=512, PBADEN=OFF, MCLRE=INTMCLR
15 #pragma config STVREN=OFF, LVP=OFF, DEBUG=OFF, CP0=OFF, CP1=OFF, CP2=OFF, CP3=OFF
16 #pragma config CPB=OFF, CPD=OFF, WRT0=OFF, WRT1=OFF, WRT2=OFF, WRT3=OFF, WRT4=OFF
17 #pragma config WRB=OFF, WRTD=OFF, EBTR0=OFF, EBTR1=OFF, EBTR2=OFF, EBTR3=OFF, EBTR4=OFF
18 #pragma config CCP3MX=PORTBS
19
20 // I/O definice pinů
21 // RA0-RA7 = LCD
22 #define ENK PORTBbits.RB0 // Vstup počítadla otacek
23 #define UP PORTBbits.RB1 == 0 // Zmáknutí horního tlačítka
24 #define DOWN PORTBbits.RB2 == 0 // Zmáknutí dolního tlačítka
25 #define START PORTBbits.RB3 == 0 // Zmáknutí startovního tlačítka
26 #define RIGHT PORTBbits.RB4 == 0 // Zmáknutí pravého tlačítka
27 #define LEFT PORTBbits.RB5 == 0 // Zmáknutí levého tlačítka
28
29 // RB6-RB7 programovací piny
30 #define LCDRS LATCbits.LATC0 // LCD
31 #define LCDE LATCbits.LATC1 // LCD
32
33 // RC2 L293E PWM regulace
34 // RC3 L293E měření proudu
35 #define IN2 LATCbits.LATC4 // vstup L293E
36 #define IN1 LATCbits.LATC5 // vstup L293E
37
38 // RC6-RC7 UART
39
40 #define TimeOver INTCONbits.TMR0IF // Timer0 praporek přetečení
41 #define XTAL_FREQ 8000000 // PIC frekvence oscilátoru
42 #define USE_AND_MASKS
43
44 // Deklarace proměnných
45 unsigned char command, direction[15];
46 int pwm=0, hours[15], minutes[15], seconds[15], step=1, steps=1, position=1, cycle, NewEnc=0;;
47 float TotalTime=0, speed[15], TotalStepTime[15], RotationTime=0, current=0;
48
49 // Deklarace funkcí
50 void PICinit();
51 void LCDinit();
52 void LCDsend(unsigned int command);
53 void LCDwrite(unsigned int command);
54 void LCDposition(unsigned int position, unsigned int row);
55 void LCDclear();
56
57 void Delay(unsigned long b);
58 void Wait(long ms);
59 void SetHours();
60 void SetMinutes();
61 void SetSeconds();
62 void SetDirection();
63 void SetSpeed();
64 void interrupt Encoder();
65
66 //*****
67 // Měření proudu na motoru
68 int Current(int samples) // RC3 = AN15
69 {
70     long j2, result=0;
71     for(j2=0; j2<samples; j2++)
72     {
73         ADCON0=0b00111101; // nastavení kanálu RC3(AN15), zapnutí ADC
74         ADCON0=0b00111111; // zapnutí konverze
75         while(ADCON0==0b00111111); // čekání na dokončení konverze
76         result+=ADRES;
77     }
78     return (result/samples);
79 }
80
81 //*****
82 // Nastavení PIC
83 void PICinit()
84 {
85     BSR=15;
86
87     // I/O nastavení pinů
88     TRISA=0; ANSELA=0; LATA=0;
89     TRISB=0b00111111; ANSELB=0; LATB=0;
90     TRISC=0b00001000; ANSELC=0b00001000; LATC=0;
91
92     INTCON2=0; WPUB=0b00111110; // nastavení Pull-up na tlačítkách
93
94     OSCCON=0b01100000; // 8MHz interní oscilátor
95     OSCCON2=0;
96
97     // Nastavení ADC převodníku
98     ADCON2=0b10111001; // zarovnaní doprava, preruseni 20TAD, konverzní frekvence FOSC/8
99     ADCON1=0; // napětová reference = VDD
100 }
```



```

100 // Nastaveni enkoderoveho preruseni
101 INTCONbits.GIE=1; //zapnuti globalnich preruseni
102 INTCON2bits.INTEDG0=0; // enkoder nastaveny na pin INTO, spinany z 1 na 0
103 INTCONbits.INT0IE=1; // zapnuti preruseni na INTO
104 INTCONbits.INT0IF=0; // praporek preruseni na INTO
105
106
107 // Nastaveni PWM na RC2
108 TRISCbits.TRISC2=1;
109 CCP1TMR1=0; // CCP1 pouziva Timer 2 pro PWM
110 CCP1CON=0b00001100; // zapnuti PWM
111 PR2=255; // casova zakladna
112 CCPR1L=0; // vykon PWM
113 PIR1bits.TMR2IF=0;
114 T2CON=0b00000110; // zapnuti Timer 2, preddelicka 1:16
115 TRISCbits.TRISC2=0;
116
117 TOCON=0b10000101; // TMR0 16bit preddelicka 64
118
119 }
120
121 //*****
122 // casovaci funkce
123
124 void Delay(unsigned long b)
125 {
126     for(;b>0;b--);
127 }
128
129 void Wait(long ms)
130 {
131     for(long bb=0;bb<ms;bb++) __delay_ms(1);
132 }
133
134

```

```

135 //*****
136 // INICIACNI KOD PRO LCD DISPLAY
137
138 void LCDinit()
139 {
140     // Z datasheetu
141     LCDRS=0;
142
143     Wait(50);
144     command=0b00111000;
145     LCDsend(command);
146
147     Wait(1);
148     command=0b00111000;
149     LCDsend(command);
150
151     Wait(1);
152     command=0b00001100;
153     LCDsend(command);
154
155     Wait(1);
156     command=0b00000001;
157     LCDsend(command);
158
159     Wait(3);
160     command=0b00000110;
161     LCDsend(command);
162
163 }
164
165 //*****
166 // Rizeni LCD displeje
167
168 void LCDposition(unsigned int position, unsigned int row)
169 {
170     LCDRS=0;
171     if(position>16 || row>2 || position==0 || row==0) LCDsend(128);
172     else LCDsend((position+((row-1)*64))+127);
173 }
174
175 void LCDwrite(unsigned int command)
176 {
177     LCDRS=1;
178     LCDsend(command);
179 }
180
181 void LCDsend(unsigned int command)
182 {
183     LATA=command;
184     LCDE=1;
185     Delay(5);
186     LCDE=0;
187     Delay(5);
188 }

```

```

190 void LCDclear()
191 {
192     LCDposition(1,1);
193     printf("          ");
194     LCDposition(1,2);
195     printf("          ");
196     LCDposition(1,1);
197 }
198
199 extern void putchar(char c) // zavedeni funkce printf
200 {
201     LCDwrite(c);
202     return;
203 }
204
205
206 //-----
207 //-----
208 //-----
209 void main()
210 {
211     PICinit();LCDinit();
212
213     // UVODNI OKNO
214     LCDposition(3,1); printf("WatchWinder");
215     LCDposition(2,2); printf("Multi Deluxe");
216     Wait(1000);LCDclear();
217
218     //-----
219     // NASTAVENÍ POČTU KROKŮ
220
221     LCDposition(5,1);printf("SETTINGS");
222     LCDposition(1,2);printf("Steps: ");
223
224     while(!START)
225     {
226         if(UP) steps++; if(DOWN) steps--;
227         if(steps<1) steps=9; if(steps>9) steps=1;
228
229         LCDposition(8,2);printf("%i",steps);Wait(150);
230         LCDposition(8,2);printf(" ",steps);Wait(100);
231     }
232     //-----
233     // NASTAVOVACÍ OKNO PARAMETRU OTACENÍ (čas kroku, smer, rychlost otacek za vterinu)
234
235     LCDclear();
236     LCDposition(1,1); printf("# Time : ");
237     LCDposition(1,2); printf("Dir: Rps:");
238     for(int a=0;a<=9;a++) {hours[a]=0; minutes[a]=0;seconds[a]=0;direction[a]='<';speed[a]=3.0;}
239
240     for(step=1;step<=steps;step++)
241     {
242         LCDposition(2,1); printf("%i",step);
243         LCDposition(9,1); printf("%02i",hours[step]);
244         LCDposition(12,1); printf("%02i",minutes[step]);
245         LCDposition(15,1); printf("%02i",seconds[step]);
246         LCDposition(5,2); printf("%c",direction[step]);
247         LCDposition(12,2); printf("%.1f",speed[step]);
248         Wait(500);
249
250         while(!START)
251         {
252             if(LEFT) {position--;if(position<1) position=5;Wait(200);}
253             if(RIGHT){position++;if(position>5) position=1;Wait(200);}
254             if(position==1) SetHours();
255             if(position==2) SetMinutes();
256             if(position==3) SetSeconds();
257             if(position==4) SetDirection();
258             if(position==5) SetSpeed();
259         }
260     }
261
262     for(int a=1;a<=9;a++) TotalStepTime[a]=(hours[a]*3600.0)+(minutes[a]*60.0)+seconds[a];
263
264     //-----
265     // CEKANI NA START
266
267     LCDclear();
268     Wait(500);
269     LCDposition(3,1);printf("Press any key");
270     LCDposition(4,2);printf("for START!");
271     while(!START && !DOWN && !UP && !RIGHT && !LEFT);
272     LCDclear();
273     //-----
274
275     for(cycle=1;cycle<=steps;cycle++)
276     {
277         TotalTime=0; TimeOver=0; WRITETIMERO(0); // zapnuti casovace
278         if(direction[cycle]=='>') {IN1=1;IN2=0;} // nastaveni otacek motoru
279         else {IN1=0;IN2=1;}
280         pwm=50;
281         CCP1L=pwm; // nastaveni vykonu motoru
282     }

```

```

283 while(TotalTime<TotalStepTime[cycle])
284 {
285
286     if(NewEnc==1) // detekovana nova otacka
287     {
288         NewEnc=0;
289         TotalTime+=RotationTime/1000000.0;
290         current=(float)Current(100)*2.222;
291
292         TotalStepTime[0]=TotalStepTime[cycle]-TotalTime;
293
294         if(current>500) // vypni motor, pokud proud presahuje 500 mA
295         {
296             IN1=0;IN2=0;
297             LCDposition(1,2);printf(" Overcurrent !!!");
298             while(1);
299         }
300
301         hours[0]=(int)(TotalStepTime[0]/3600.0);
302         minutes[0]=(int)((unsigned long)TotalStepTime[0]*3600/60.0);
303         seconds[0]=(unsigned long)TotalStepTime[0]-(hours[0]*3600+minutes[0]*60);
304         speed[0]=1000000.0/RotationTime;
305         if(speed[cycle]>speed[0]) {pwm++;if(pwm>255)pwm=255;}
306         if(speed[cycle]<speed[0]) {pwm--;if(pwm<0) pwm=0;}
307         CCP1L=pwm;
308
309         LCDclear();
310         LCDposition(1,1);printf("#%i" ,cycle);
311         LCDposition(4,1);printf("%02i:%02i:%02i" ,hours[0],minutes[0],seconds[0]);
312         LCDposition(13,1);printf("%0.f%%" ,(float)(pwm*0.3922));
313         LCDposition(1,2);printf("%c" ,direction[cycle]);
314         LCDposition(3,2);printf("%.1frps" ,speed[0]);
315         LCDposition(12,2);printf("%0.fmA" ,current);
316     }
317
318     if(TimeOver==1) // vypni motor, pokud otacka trva dele nez 2 sekundy
319     {
320         IN1=0;IN2=0;
321         LCDposition(1,2);printf(" Roll blocked!!!");
322         while(1);
323     }
324 }
325
326
327 }
328
329 //*****
330 // FINALNI OKNO PROGRAMU
331
332 IN1=0;IN2=0;
333 LCDclear();
334 LCDposition(1,1);printf(" The program is");
335 LCDposition(3,2);printf("FINISHED!!!");
336
337 while(!START && !LEFT && !RIGHT && !UP && !DOWN);
338
339 LCDclear();
340 LCDposition(1,1);printf("Thanks for using");
341 LCDposition(3,2);printf("WatchWinder!");
342
343 while(1);
344
345 //*****
346 //*****
347 //*****
348
349 // NASTAVOVANI PARAMETRU OTACENI
350
351 void SetHours()
352 {
353     while(!RIGHT && !LEFT && !START)
354     {
355         if(UP) hours[step]++;
356         if(DOWN) hours[step]--;
357         if(hours[step]<0) hours[step]=24; if(hours[step]>24) hours[step]=0;
358
359         LCDposition(9,1);
360         printf(" ");
361         Wait(100);
362         LCDposition(9,1);
363         printf("%02i" ,hours[step]);
364         Wait(150);
365     }
366 }
367
368 void SetMinutes ()
369 {
370     while(!RIGHT && !LEFT && !START)
371     {
372         if(UP) minutes[step]++;
373         if(DOWN) minutes[step]--;

```

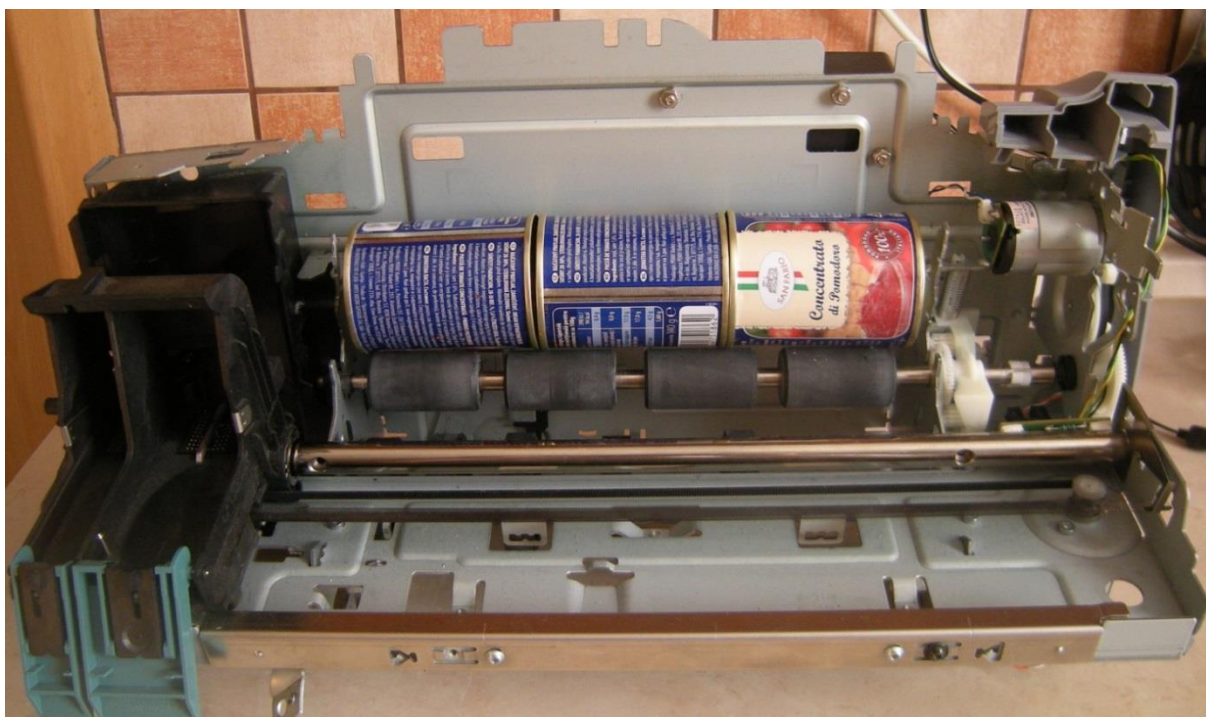
```

373         if (DOWN) minutes[step]--;
374         if (minutes[step]<0) minutes[step]=59; if (minutes[step]>59) minutes[step]=0;
375
376         LCDposition(12,1);
377         printf(" ");
378         Wait(100);
379         LCDposition(12,1);
380         printf("%02i" ,minutes[step]);
381         Wait(150);
382     }
383 }
384
385 void SetSeconds ()
386 {
387     while(!RIGHT && !LEFT && !START)
388     {
389         if (UP) seconds[step]++;
390         if (DOWN) seconds[step]--;
391         if (seconds[step]<0) seconds[step]=59; if (seconds[step]>59) seconds[step]=0;
392
393         LCDposition(15,1);
394         printf(" ");
395         Wait(100);
396         LCDposition(15,1);
397         printf("%02i" ,seconds[step]);
398         Wait(150);
399     }
400 }
401
402 void SetDirection ()
403 {
404     while(!RIGHT && !LEFT && !START)
405     {
406         if (DOWN) direction[step]='<';
407         if (UP) direction[step]='>';
408
409         LCDposition(5,2);
410         printf(" ");
411         Wait(100);
412
413         LCDposition(5,2);
414         printf("%c" ,direction[step]);
415         Wait(150);
416     }
417 }
418
419 void SetSpeed ()
420 {
421     while(!RIGHT && !LEFT && !START)
422     {
423         if (UP) speed[step]=speed[step]+0.1;
424         if (DOWN) speed[step]=speed[step]-0.1;
425         if (speed[step]<1) speed[step]=10; if (speed[step]>10) speed[step]=1;
426         LCDposition(12,2);
427         printf(" " ,speed[step]);
428         Wait(100);
429
430         LCDposition(12,2);
431         printf("%.1f" ,speed[step]);
432         Wait(150);
433     }
434 }
435
436 //*****
437 //*****
438 //*****
439 // DETEKCE POCITADLA OTACEK
440
441 void interrupt Encoder(void)
442 {
443     if (INICONbits.INT0IF==1 && INICONbits.INT0IE==1)
444     {
445         INICONbits.INT0IE=0; INICONbits.INT0IF=0;
446         NewEnc=1;
447         RotationTime=(READTIMER0()*32.0);
448         TimeOver=0; WRITETIMER0(0);
449
450         INICONbits.INT0IE=1;INICONbits.INT0IF=0;
451
452         return;
453     }
454 }

```

Obrázek 36: Watch Winder Multi Deluxe, výpis programu pro PIC18F26K22

4.6 Fotodokumentace



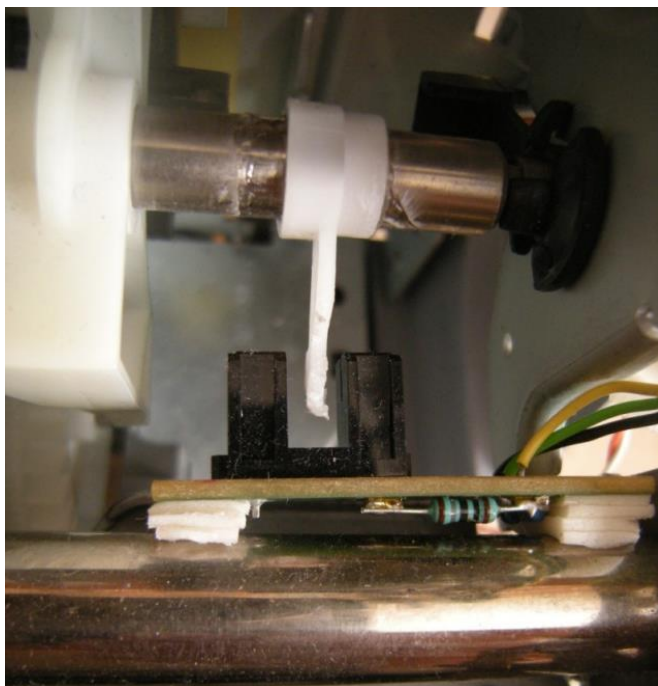
Obrázek 37: Watch winder multi deluxe, pohled zepředu



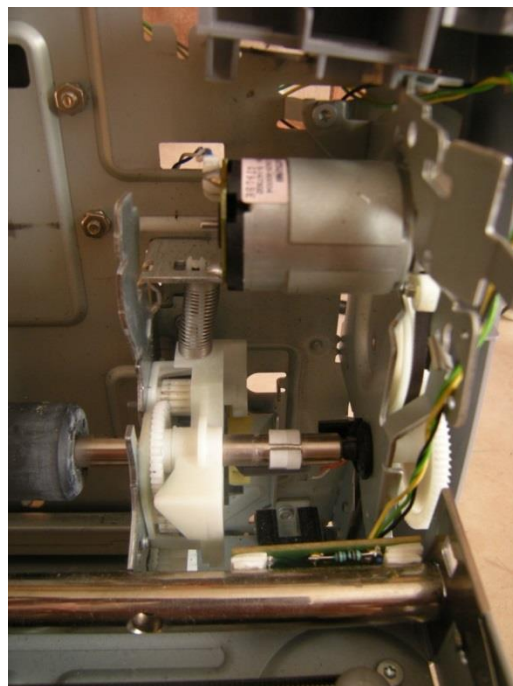
Obrázek 38: Watch winder multi deluxe, pohled zezadu



Obrázek 39: Watch winder multi deluxe, řídicí jednotka



Obrázek 40: Watch winder multi deluxe, počítadlo otáček s optickou závorou (vlevo)



Obrázek 41: Watch winder multi deluxe, motor s převodovkou (vpravo)



Obrázek 42: Watch winder multi deluxe, hodinky, vyplněné plastovou pěnou



Obrázek 43: Watch winder multi deluxe, programová okna

4.7 Zhodnocení pokročilé verze

I pokročilá verze natahovače hodinek popsaná v této kapitole splnila požadavky specifikované pro v zadání úkolu (kapitola 2.3):

- minimální cena
 - náklady na součástky s použitím recyklovaných dílů z tiskáren nepřekračují 500 Kč bez DPH.
- kapacita pro co největší počet hodinek
 - tuba pro otáčení hodinek měří 195 mm, což znamená, že má kapacitu až 6 hodinek
- nastavení několika různých programů s volitelnými parametry
 - nastavení až devíti samostatných kroků s volbou rychlosti, směru a doby otáčení nezávisle na sobě
- LCD displej
 - alfanumerický LCD display 16x2 s podsvícením
- ovládání joystickem
 - 5tlačítkové ovládání
- počítadlo otáček
 - optická závora vymontovaná ze staré tiskárny
- ochrana proti přetížení motoru
 - ano, softwarově, s limitem 500 mA
- širší využitelnost pro mísení jiných materiálů
 - ano, například plechovky s barvami, od kečupu nebo i exotické nápoje :-).

5 Finální model rotačního zařízení

5.1 Popis výrobku

Pro možnost širšího využití vyvinutého řešení rotačního mísiče ze staré tiskárny i pro jiné účely než jsou natahovače hodinek (např. pro mísení vzorků v chemických a biologických laboratořích) jsem pro finální model zvolila použít celý rám tiskárny, podobně jako u druhého modelu popsáno v předchozí kapitole, ale pozměnit jeho elektroniku a software, aby byl univerzálnějším a jednodušším pro výrobu. Hlavní změny oproti předchozí verzi:

1. Jednoúčelová deska s řídicím systémem na bázi mikrokontrolérů PIC bude nahrazena široce dostupným modulem Arduino UNO, na který se pouze připojí ovládací deska plošného spoje, ve formě tzv. shieldu.
2. Ovládání 5 tlačítka bude nahrazeno levným, komerčně dostupným joystickem, dodávaným v rámci různých Arduino sad.
3. K počítání otáček bude použit původní rotační enkodér zabudovaný v tiskárně, který umožňuje mnohem přesnější počítání otáček a jejich regulaci než dříve použitý fotopřerušovač (optická závora). Připojení optické závory bude ale na desce zachováno jako rezerva.
4. Pro zjednodušení konstrukce ovládací desky bude motorový driver L293E nahrazen motorovým driverem L293D, který má dostatečnou proudovou kapacitu (1A v paralelním zapojení výstupů), nepotřebuje ale dodatečné ochranné diody (má je již zabudované).
5. Nebude prováděno přímé měření proudu, nadproudová ochrana bude místo toho řešena polymerovou pojistkou v obvodu a detekce zaseknutí otáčení řešena softwarově pomocí signálů z enkodéru.
6. Krabíčka pro řídicí jednotku bude nakreslena jako jednoúčelová a vytištěna na 3D tiskárně.

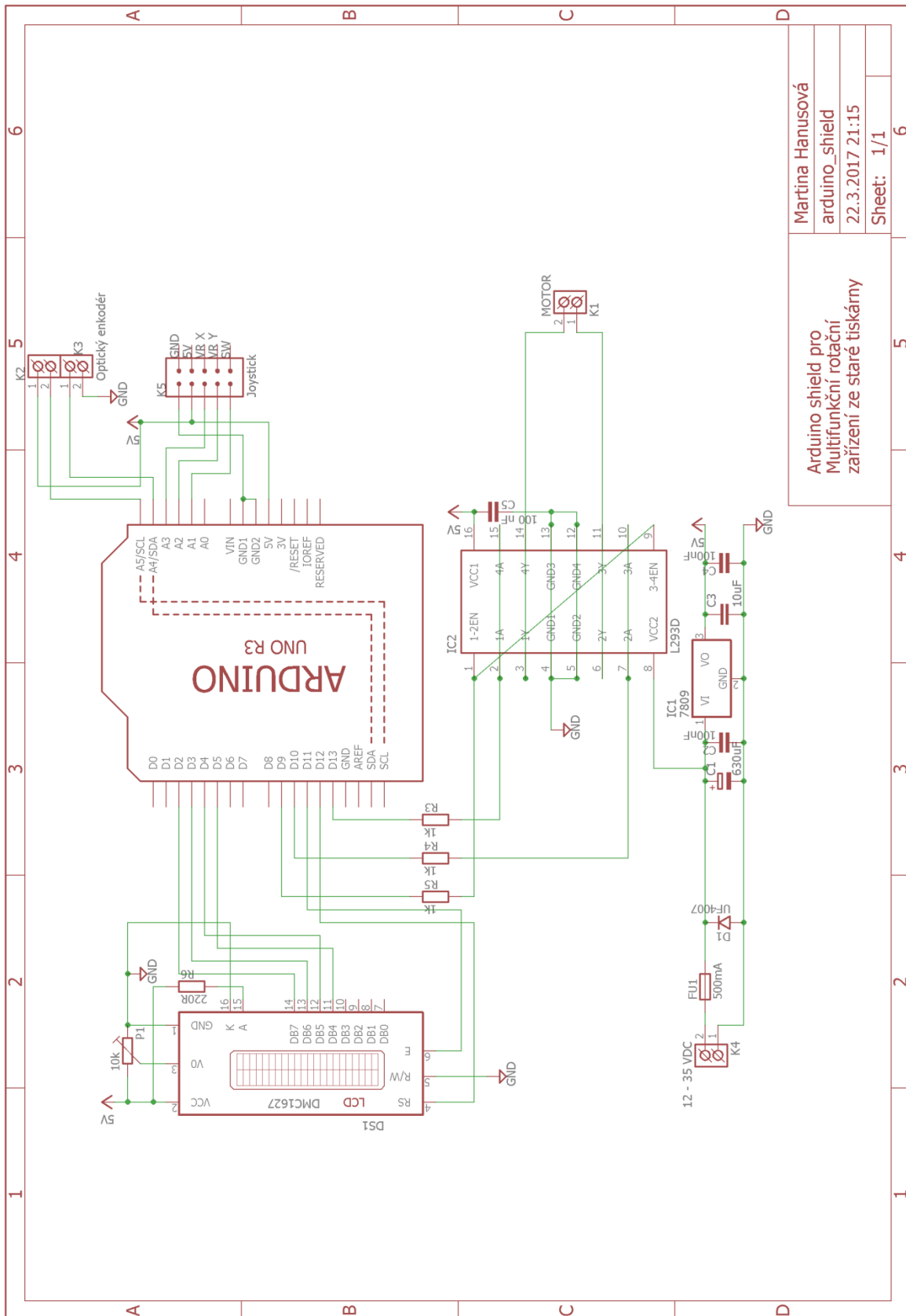
5.2 Tabulka součástek ovládací desky (shield) pro Arduino Uno

Kód	Typ	Parametry	Funkce	Kód (TME)	Cena (Kč)
C1	kondenzátor	elektrolytický, 630 uF, THT	filtrace vstupu stabilizátoru	ED1I681MNN1240	13,77
C2	kondenzátor	keramický, 100 nF, SMD 0805	filtrace vstupu stabilizátoru	CL05A104KP5NNNC	0,11
C3	kondenzátor	keramický, 10 uF SMD 1206	filtrace výstupu stabilizátoru	CL31F106ZPENNNE	0,57
C4	kondenzátor	keramický, 100 nF, SMD	filtrace výstupu	CL05A104KP5NNNC	0,11

Kód	Typ	Parametry	Funkce	Kód (TME)	Cena (Kč)
		0805	stabilizátoru		
C5	kondenzátor	keramický, 100 nF, THT	filtrace napájení driveru	CM-100N	0,76
D1	dioda	UF4007, THT	polaritní ochrana napájení	UF4007	1,30
DS1	displej	alfanumerický 16 x 2	nastavování a zobrazování programových parametrů	RC1602B2-GHW- CSX	158,58
FU1	pojistka	polymerová, 500 mA	proudová a polaritní ochra- na	EX050-V1	8,70
IC1	stabilizátor	7809, THT	9V napájecí zdroj	L7809CV	3,96
IC2	motorový driver	L293D, THT, DIP16	obousměrný mo- torový driver	L293D	63,77
K1	svorkovnice	ARK500/2	připojení motoru	TB-5.0-P-2P/GY	4,10
K2 – K3	svorkovnice	ARK500/2	připojení optického enkodéru	TB-5.0-P-2P/GY	4,10
K4	svorkovnice	ARK500/2	připojení hlavního napájení	TB-5.0-P-2P/GY	4,10
K5	IDC konek- tor	IDC10, THT	připojení joystick- ku	ZL231-10PG	2,25
K6	svorkovnice	ARK500/3	připojení optické závory (rezerva)	TB-5.0-P-3P/GY	5,31
	plošný spoj	plošný spoj s fotocitlivou vrstvou 16 x 10 cm		FR4100X160/3500	45,31
P1	trimr	10 kΩ, THT	nastavování kon- trastu LCD dis- pleje	1028F-10K	10,36
R1	rezistor	220 Ω, THT, 0,6 W	snížení proudu LED	1/2W-220R	0,41
R2	rezistor	2k2 Ω, THT, 0,6 W	napěťový výstup z kolektoru	1/4W2.2K	0,20
R3	rezistor	1 kΩ, THT, 0,6 W	ochrana výstupu MCU	OX102KE	0,41
R4	rezistor	1 kΩ, THT, 0,6 W	ochrana výstupu MCU	OX102KE	0,41
R5	rezistor	1 kΩ, THT, 0,6 W	ochrana výstupu MCU	OX102KE	0,41
R6	rezistor	220 Ω, THT, 0,6 W	nastavení podsví- cení LCD displeje	1/2W-220R	0,41
Celkem					304

Tabulka 5: Finální verze, tabulka součástek

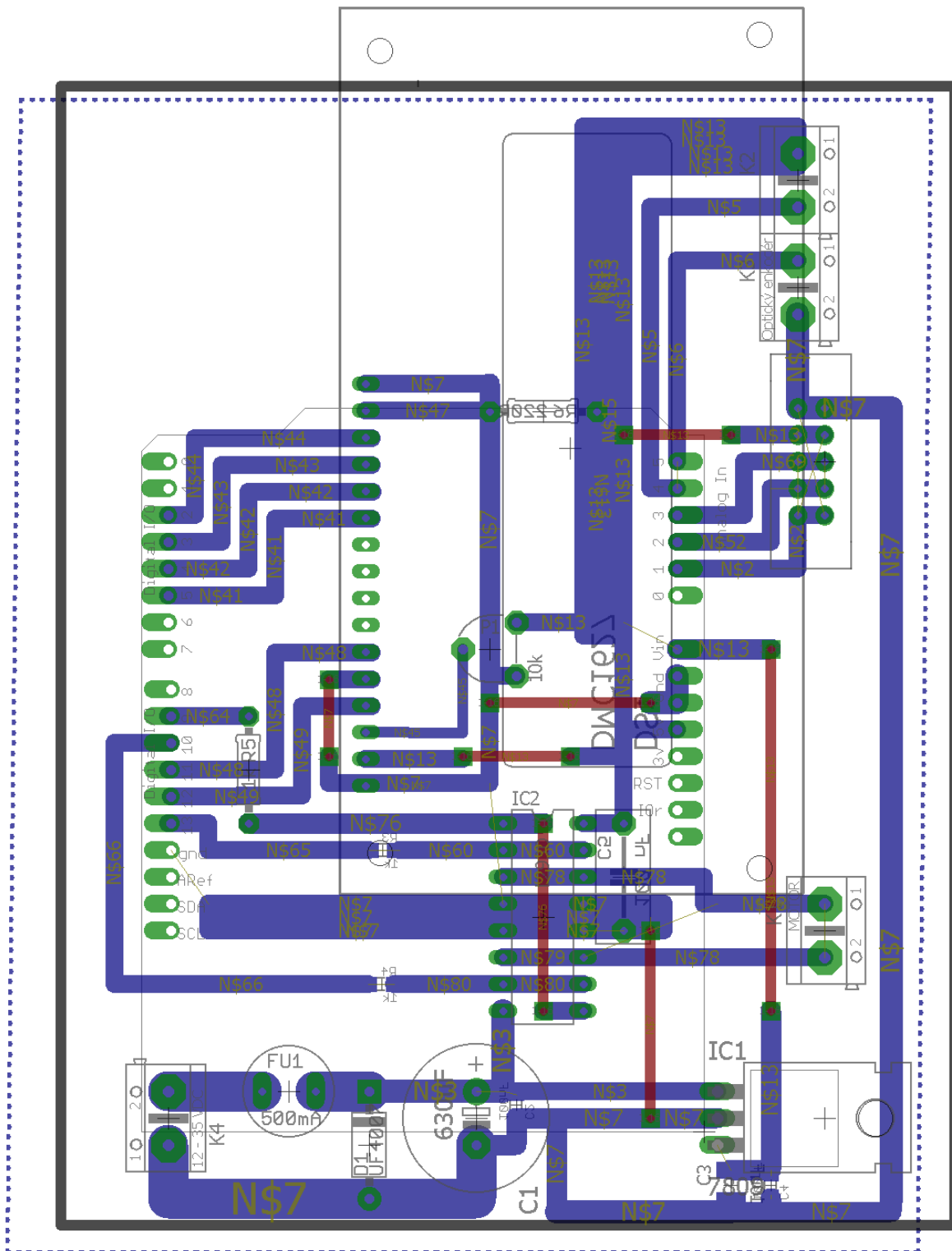
5.3 Schémata plošného spoje ovládací desky



Arduino shield pro
Multifunkční rotační
zařízení ze staré tiskárny

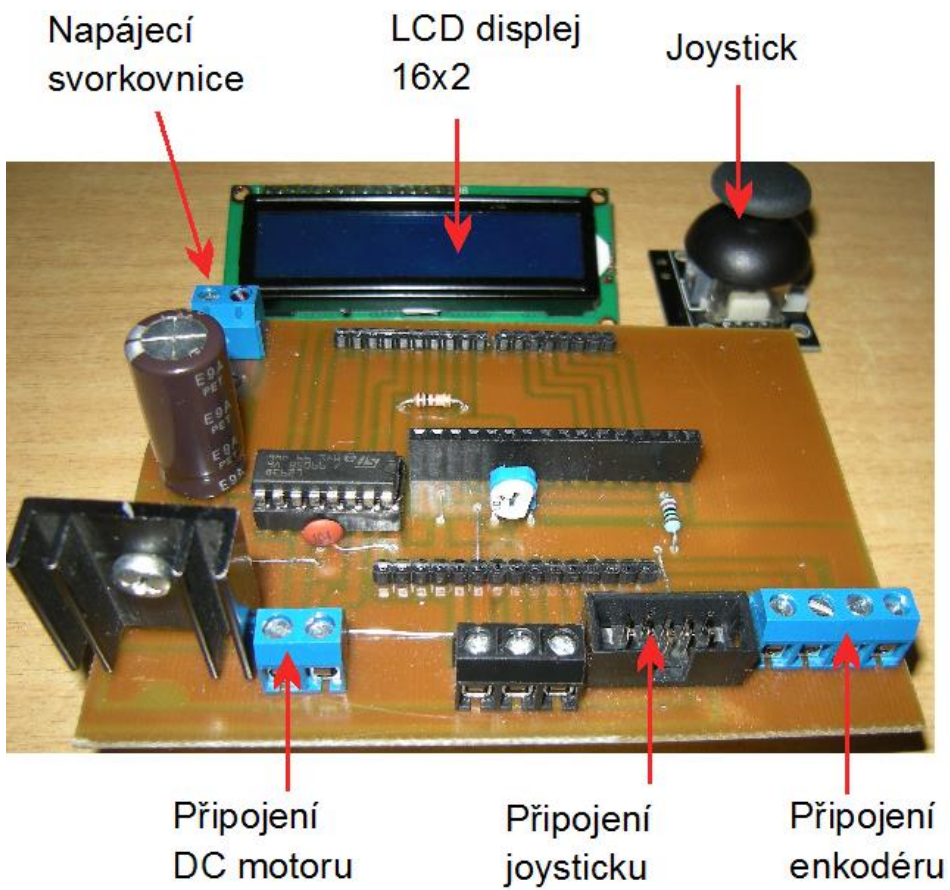
Martina Hanusová
arduino_shield
22.3.2017 21:15
Sheet: 1/1

Obrázek 44: Funkční schéma zapojení ovládací desky (shield) pro Arduino UNO

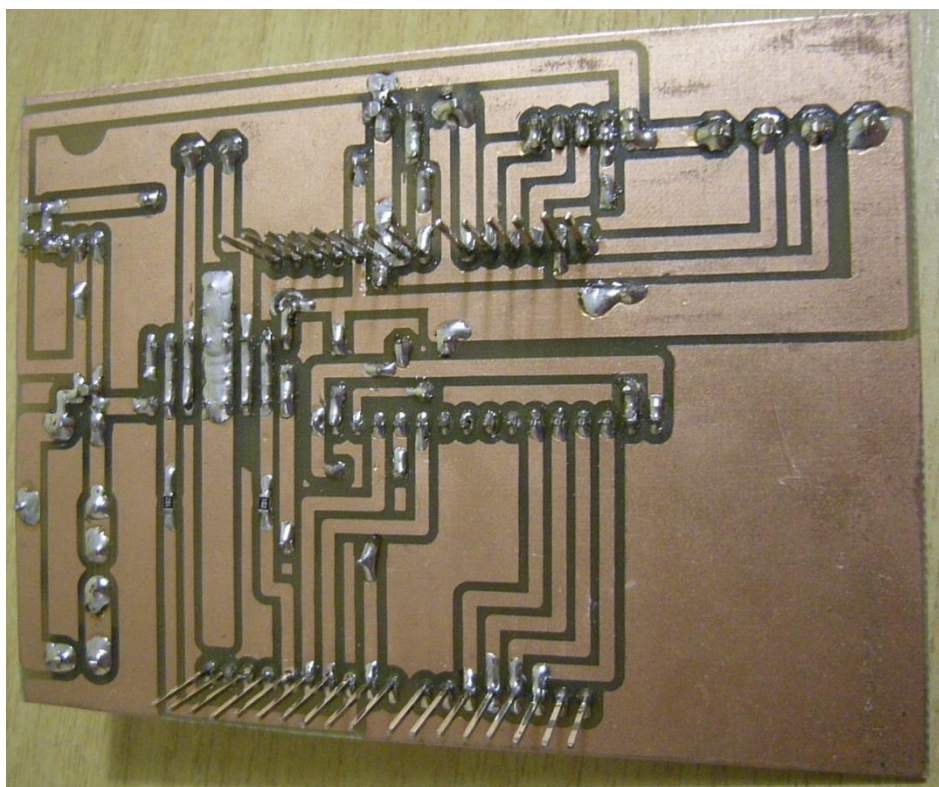


Obrázek 45: Schéma plošného spoje ovládací desky (shield) pro Arduino Uno

Poznámka: Schémata plošného spoje ve formátu Eagle CAD (*.sch, *.brd) jsou přiloženy v elektronických přílohách této práce a mohou být použity k výrobě ovládací desky.



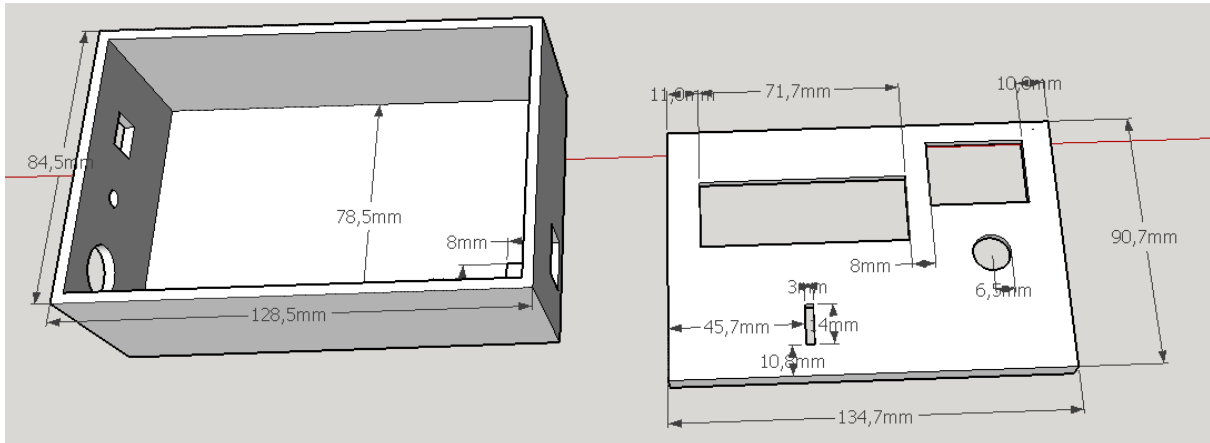
Obrázek 46: Finální model, ovládací deska s popiskami



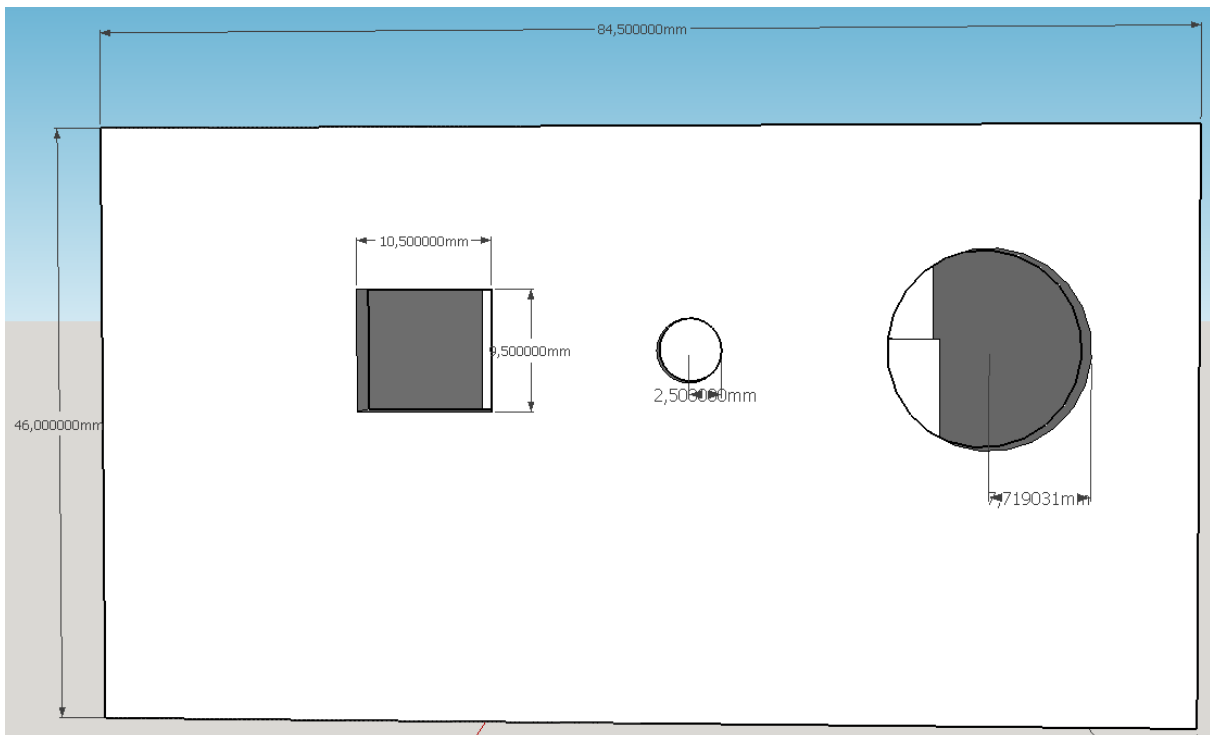
Obrázek 47: Finální model, pohled na ovládací desku zespodu

5.4 Model krabičky řídicí jednotky

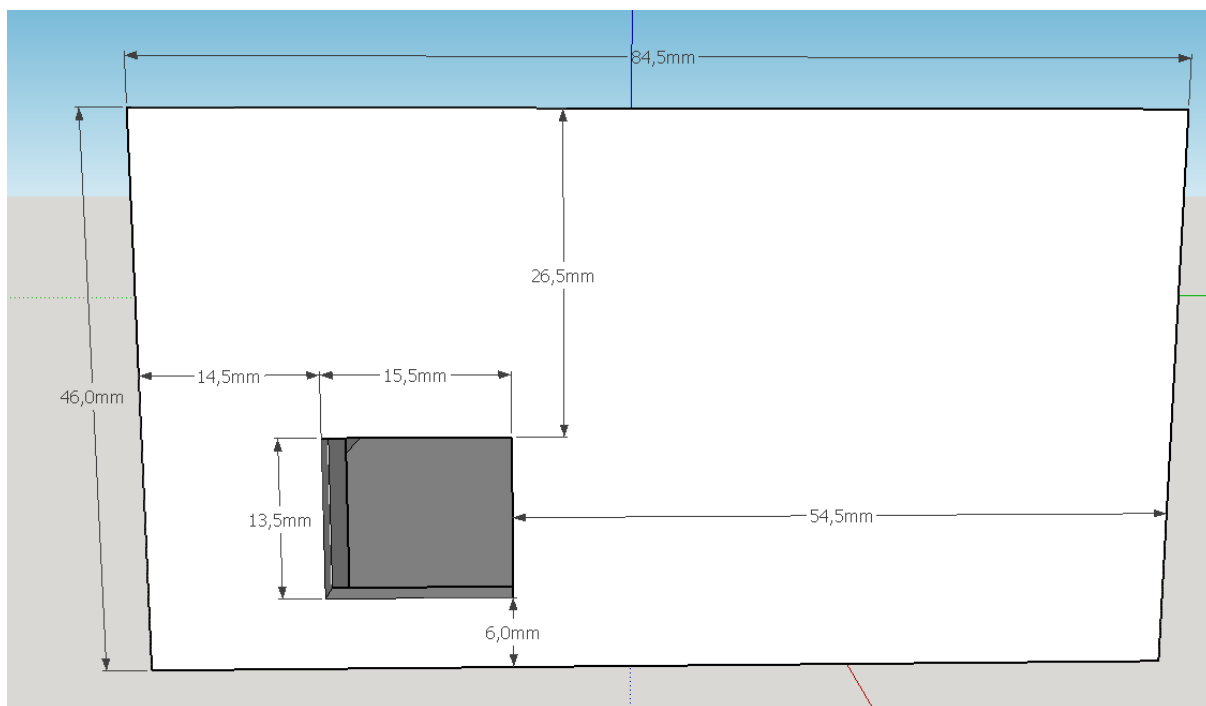
Model krabičky s víkem pro řídicí jednotku jsem nakreslila v programu SketchUp (soubor *.skp je uveden v přílohách elektronické verze této SOČ) a obě součásti jsem vyexportovala ve formátu STL pro možnost jejich vytištění na 3D tiskárně. Krabička vymezuje plošný spoj s displejem proti pohybu a obsahuje otvory pro zavedení kabelů, chlazení stabilizátoru i zasunutí vyššího kondenzátoru. Krabičku jsem vytiskla na 3D tiskárně Průša i3 Mk2 z materiálu PETG transparentní, s 20% výplně. Tloušťka stěn krabičky jsou 3 mm.



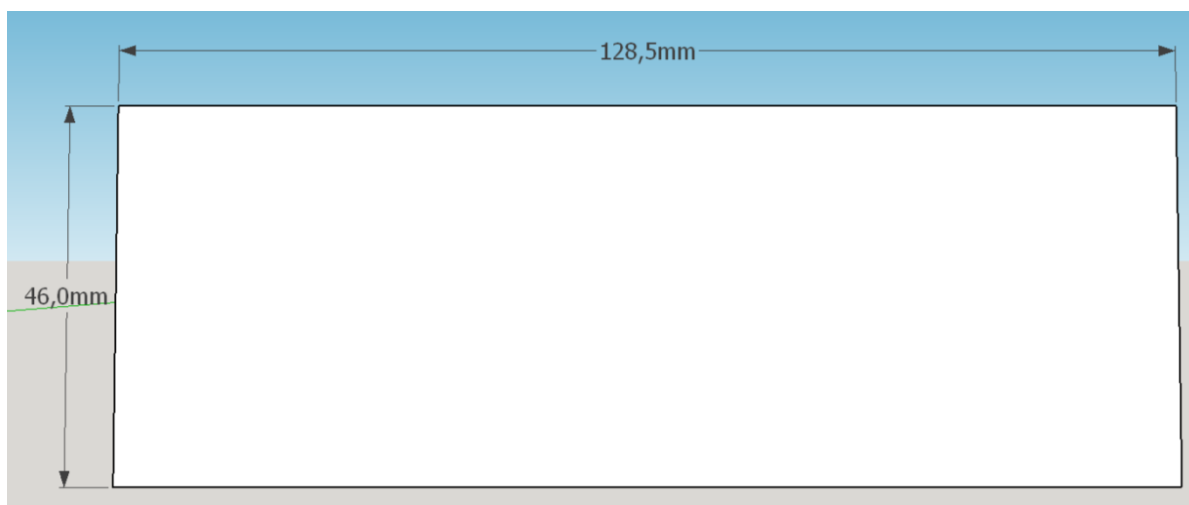
Obrázek 48: Celkový pohled na krabičku a víko



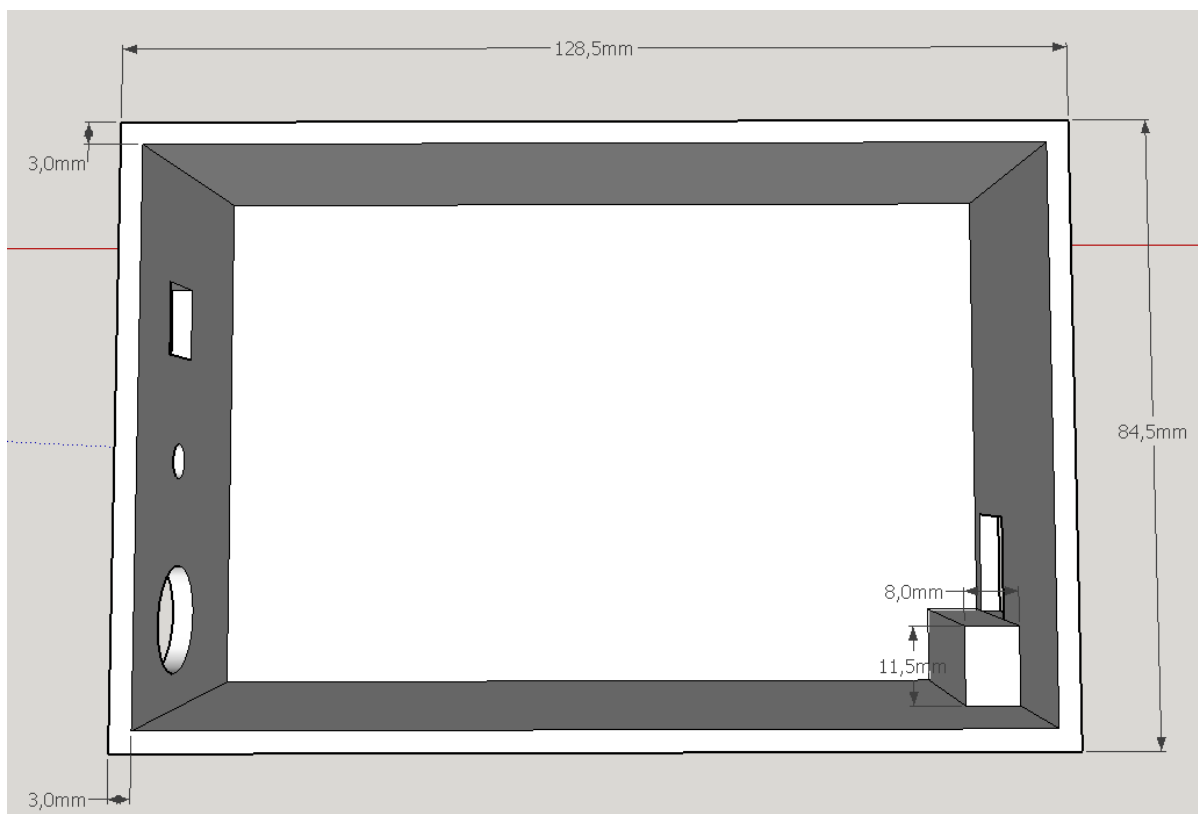
Obrázek 49: Pravý bokorys krabičky s otvory pro napájení, vypínač a průchodku kabelů



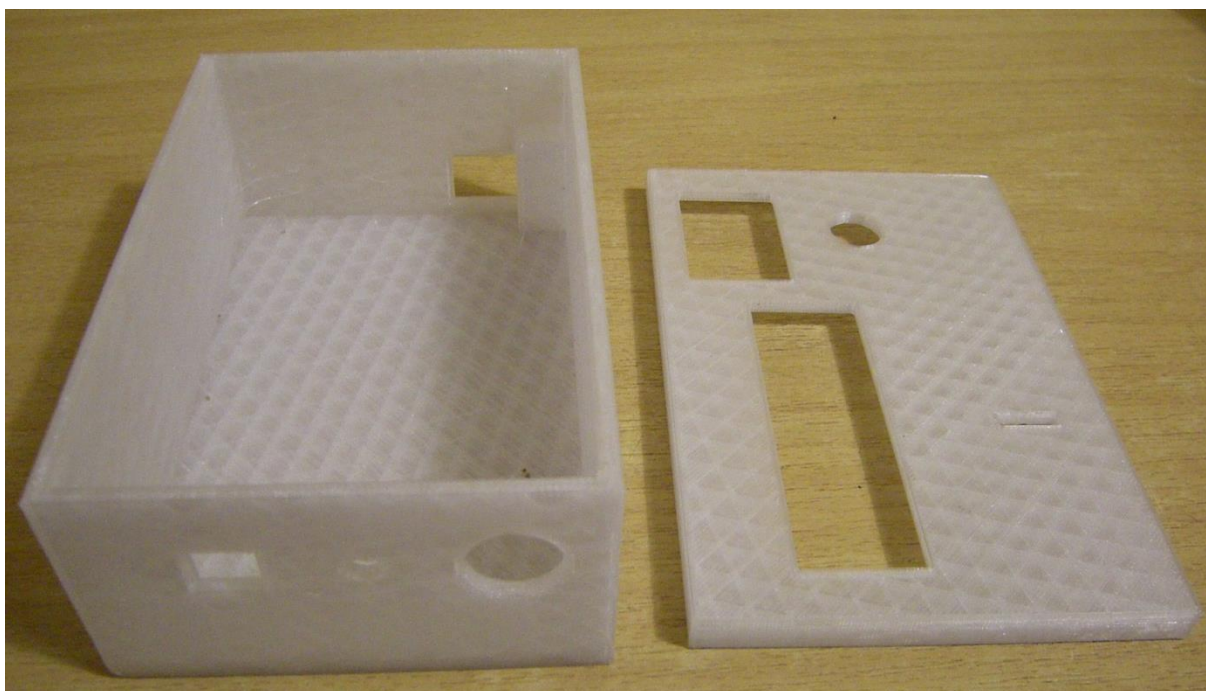
Obrázek 50: Levý bokorys krabičky s otvorem pro programovací kabel Arduino



Obrázek 51: Narys krabičky



Obrázek 52: Půdorys krabičky



Obrázek 53: Finální model, foto vytisknuté krabičky

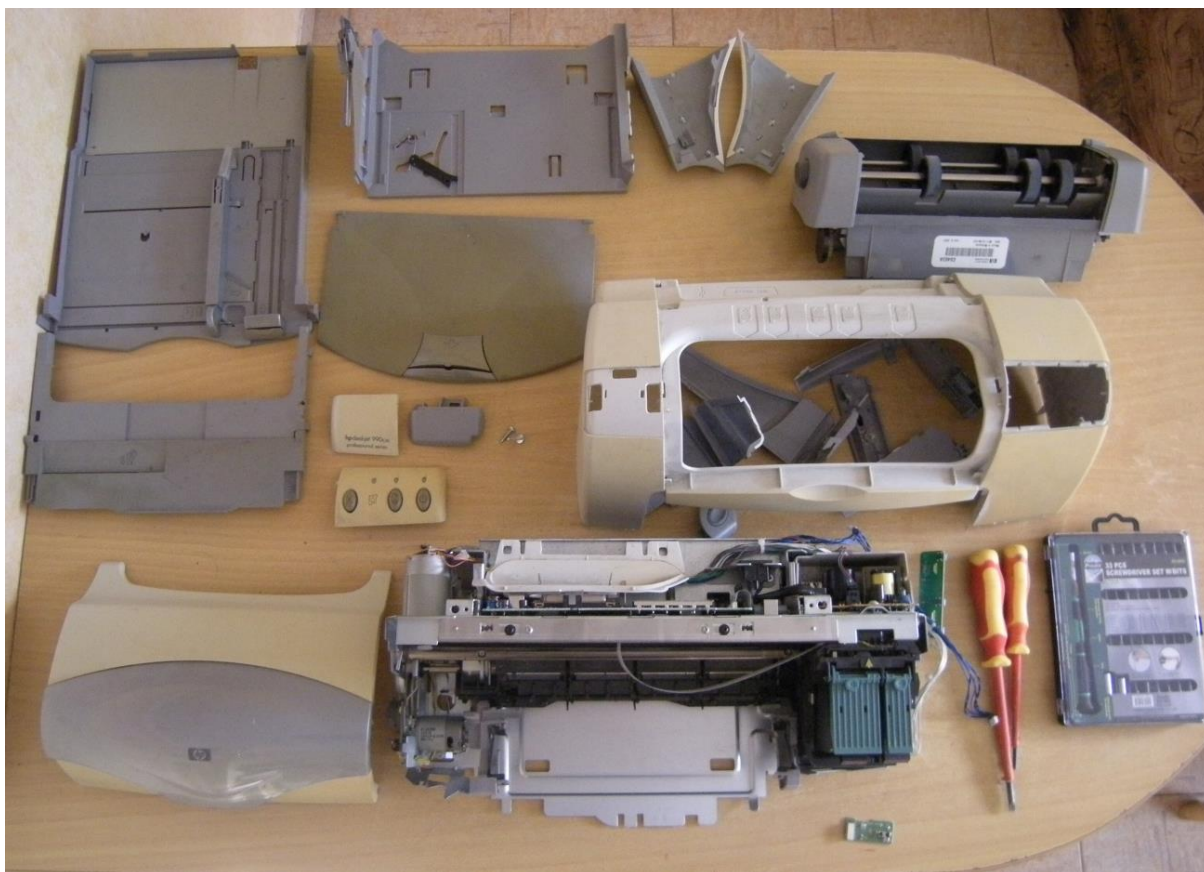
5.5 Postup demontáže tiskárny

V této kapitole je popsán a vyobrazen postup demontáže tiskárny HP DeskJet 990cxi pro získání součástí potřebných ke konstrukci rotačního zařízení. Postup je uveden pouze jako modelový příklad, využitelnou rámovou konstrukci mají i další modely tiskáren firmy Hewlett Packard (např. z řad DeskJet 9XX a 6XX nebo Photosmart 12XX), které se ale konstrukčními detaily liší, takže i konkrétní postup montáže bude trochu odlišný.

- 1) Z tiskárny odstraníme všechny plastové díly, ponecháme pouze ocelovou rámovou konstrukci s motory a enkodérem. Většina plastů je buď zaklapnutá do sebe nebo přišroubovaná k ocelové konstrukci a jde odejmout bez poškození. Rovné plastové díly jde použít jako konstrukční materiál pro jiné výrobky. Demontáž pochopitelně provádíme bez připojeného napájení tiskárny.



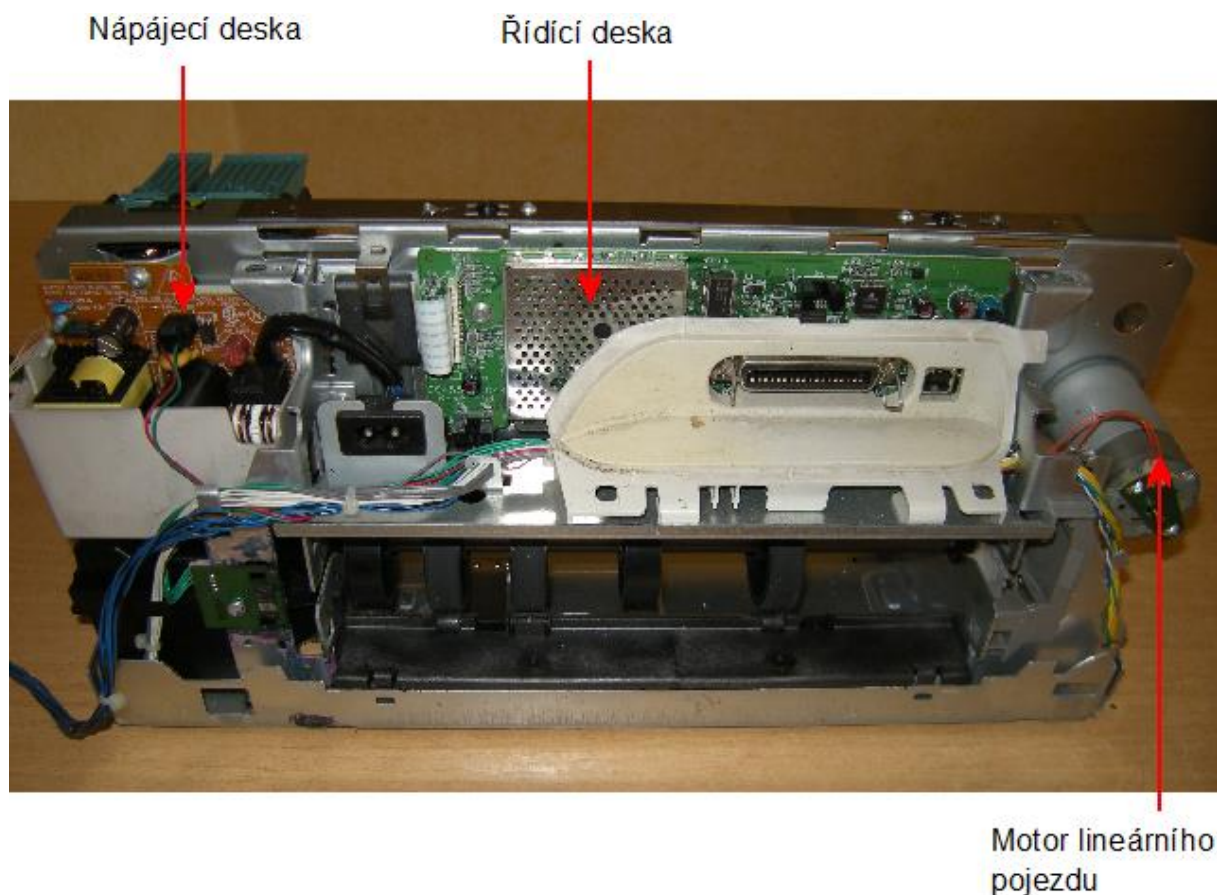
Obrázek 54: Tiskárna HP DeskJet 990cxi v původním stavu



Obrázek 55: Tiskárna po odstranění všech plastových krytů

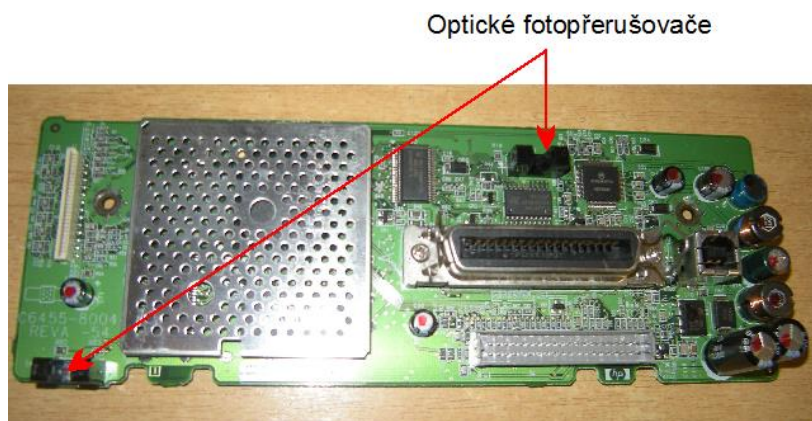


Obrázek 56: Rám tiskárny zepředu po odstranění plastových krytů



Obrázek 57: Rám tiskárny zezadu po odstranění plastových krytů

- 2) V dalším kroku demontujeme všechny desky plošných spojů s výjimkou desky rotačního enkodéru (viz dále). U některých tiskáren jsou síťové napájecí adaptéry (dávající 12 – 24 VDC) externí a dají se proto použít i pro napájení upravené tiskárny, v tomto případě je napájecí deska součástí tiskárny a nejde proto dále použít bez rizika kontaktu s nebezpečným napětím. Původní řídicí deska tiskárny obsahuje některé zajímavé součástky, které jde po odpájení použít v jiných výrobcích, např. optické fotopřerušovače.



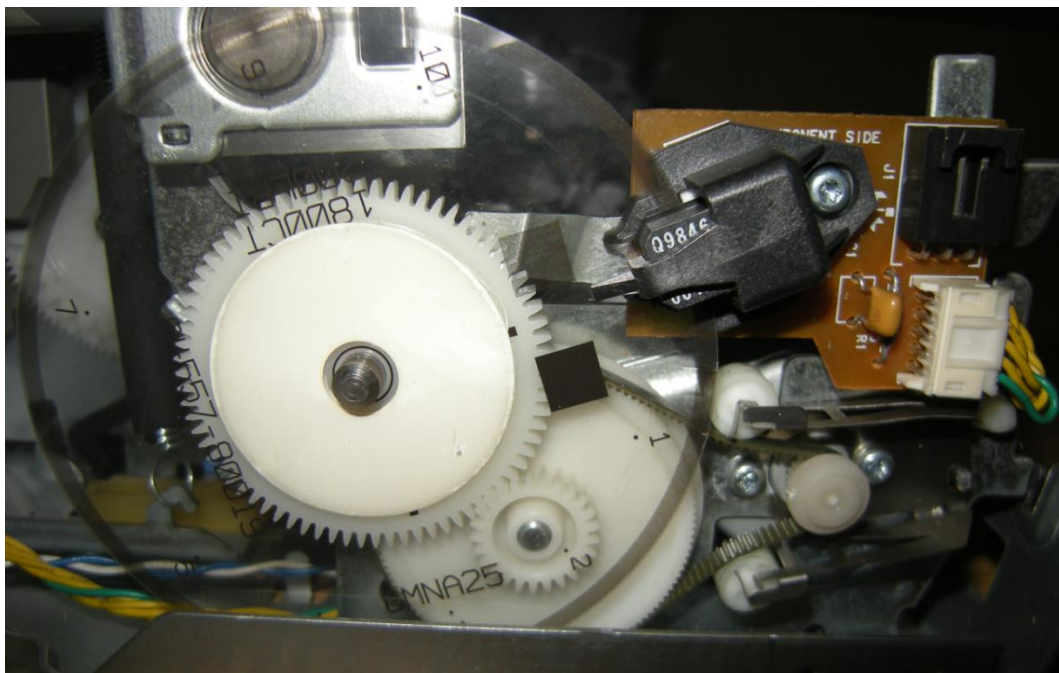
Obrázek 58: Detail původní řídicí desky tiskárny

- 3) Opatrně vyjmeme nádoby s tonerem a demontujeme mechanismus čištění tiskových hlav umístěný obvykle na boku tiskárny a poháněný vlastním motorem. Tiskové patrony a všechny materiály znečištěné tiskovým inkoustem předáme do sběrného dvora jako nebezpečný odpad. Držák nádobek ponecháme, bude sloužit jako zarážka pro otáčení předmětů.



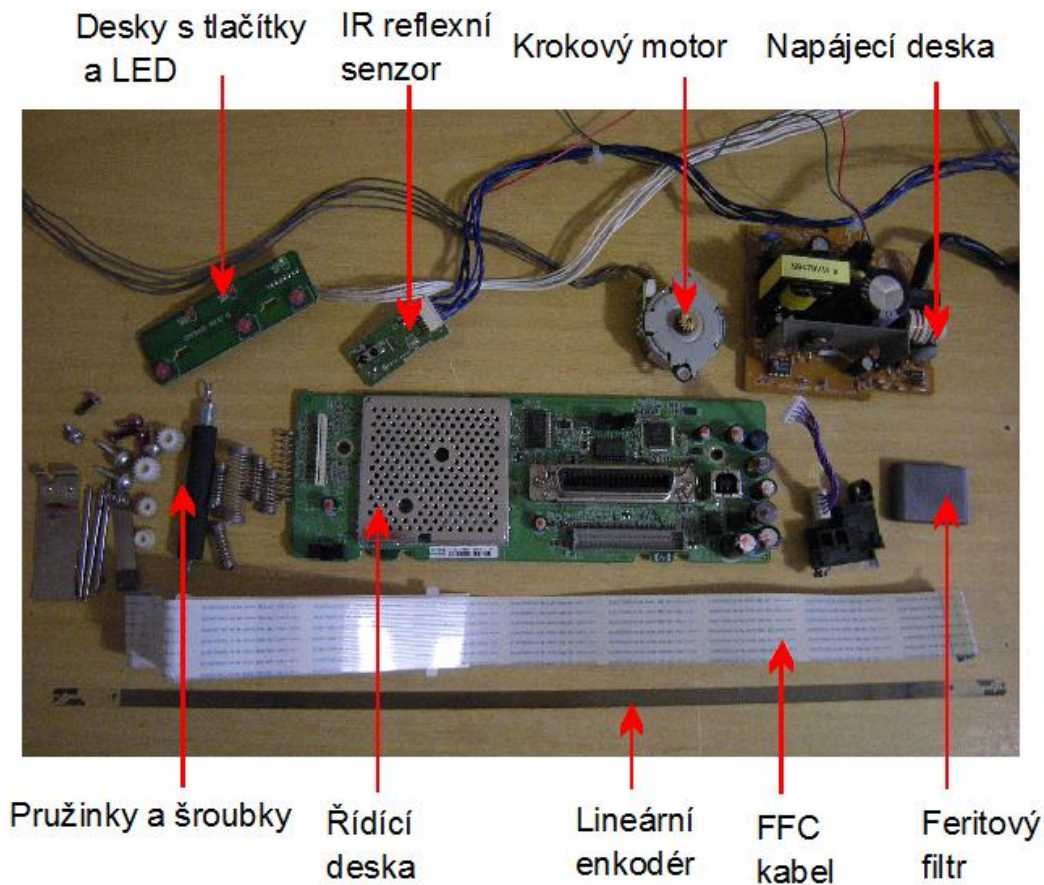
Obrázek 59: Tiskové patrony na lineárních pojezdech tiskárny.

- 4) Z rotačního mechanismu podávání papíru odstraníme všechny plastové díly a zarážky, které brání volnému pohybu válců oběma směry. Dáváme pozor, abychom při demontáži nepoškodili optický enkodér, jeho elektroniku a kabeláž. Použitý optický enkodér dává 1800 pulzů na otáčku a je pro naše účely dostatečně citlivý.

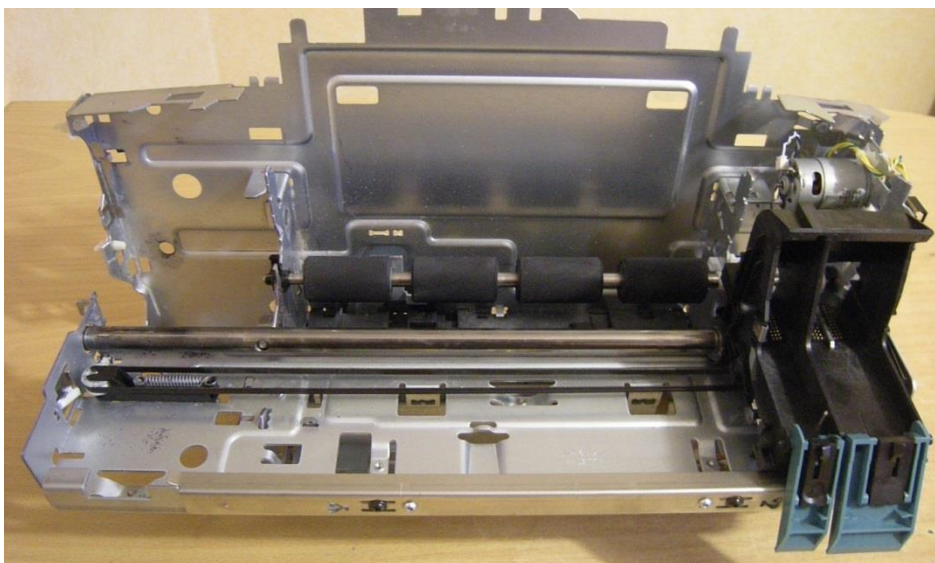


Obrázek 60: Detail optického enkodéru

- 5) Po demontáži získáme nejen vlastní rám tiskárny se zabudovaným rotačním mechanismem, jeho DC motorem, optickým enkodérem a dalším DC motorem pohybu lineární osy (v našem míšiči nepoužívaném), ale i celou řadu rozmanitých elektronických součástek, vyobrazených na následujícím obrázku a využitelných v jiných elektronických či robotických výrobcích.



Obrázek 61: Souhrn elektronických součástek vyjmutých z tiskárny



Obrázek 62: Rám tiskárny s válcem, motory a enkodérem po dokončené demontáži

6) Posledním potřebným krokem je identifikace zapojení optického enkodéru, k němuž obvykle vede 4 – 5 vodičů (v případě HP DeskJet 990cxi 4 žluté a 1 zelený), z nichž dva jsou signály enkodéru, dva jsou napájecí (VDD, GND) a jeden nevyužitý. K identifikaci vodičů enkodéru slouží následující postup navržený konzultantem této práce:

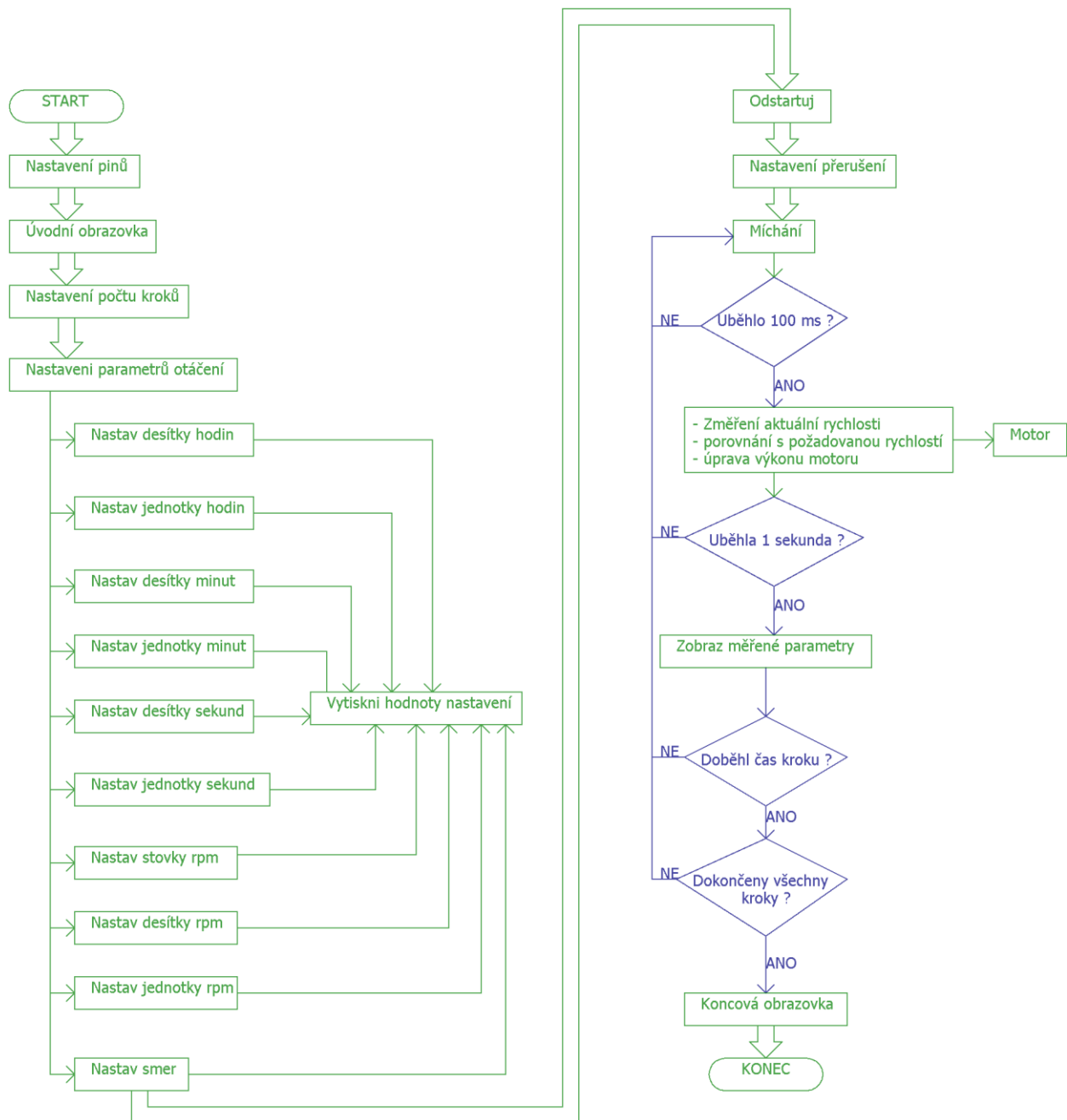
- Žluté vodiče označíme fixem 1 – 4 čárkami.
- Postupně měříme multimetrem v nastavení pro měření diod všechny vodiče navzájem (červený konektor vs. černý konektor multimetru) a výsledky zaznamenáme do tabulky:

	1	2	3	4	Z
1	X	X	X	X	X
2	X	X	925	1853	660
3	X	872	X	1854	660
4	X	769	768	X	644
Z	X	1627	1672	1414	X
	X	SIG1	SIG2	VDD	GND

Tabulka 6: Tabulka dekodování vodičů enkodéru

- Vodič, na kterém neměříme žádnou hodnotu, se nepoužívá (v našem případě č. 1).
- Vodič, na kterém naměříme největší hodnoty, je kladný pól napájení VDD (č. 4).
- Vodič, na kterém naměříme nejmenší hodnoty, je záporný pól napájení GND (Z).
- Zbylé vodiče, dávající přibližně shodné výsledky, jsou signály enkodéru SIG1 a SIG2 (č. 2 a č. 3).

5.6 Algoritmus programu



Obrázek 63: Finální model, algoritmus programu

5.7 Výpis programu pro Arduino Uno

```
1 #include <math.h>
2 #include <LiquidCrystal.h>
3
4 LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
5
6 int JOY_TLA_pin=A1, JOY_Y_pin=A2, JOY_X_pin=A3, ENK1_pin = A4,
7 ENK2_pin=A5, MOTOR_EN_pin=9, MOTOR_IN2_pin=10, MOTOR_IN1_pin=13;
8 int pocet_kroku=1, krok=1, pozice=1, casovac_sekund=0,
9 casovac_desetin=0, pwm=0, zobraz_data=0, nastav_motor=0, blik=100;
10 int hodiny[10], minuty[10], sekundy[10], smer[10],
11 rychlost[10], desitkyhodin[10], jednotkyhodin[10],
12 desitkyminut[10], jednotkyminut[10];
13 int desitkysekund[10], jednotkysekund[10],
14 stovkyrpm[10], desitkyrpm[10], jednotkyrpm[10];
15 long CelkovaDobaKroku[10], enkoder=0;
16
17 void NastaveniPinu();
18 void UvodniObrazovka();
19 void NastaveniPoctuKroku();
20 void NastaveniParametruOtaceni();
21 void NastavDesitkyHodin();
22 void NastavJednotkyHodin();
23 void NastavDesitkyMinut();
24 void NastavJednotkyMinut();
25 void NastavDesitkySekund();
26 void NastavJednotkySekund();
27 void NastavStovkyRpm();
28 void NastavDesitkyRpm();
29 void NastavJednotkyRpm();
30 void NastavSmer();
31 void VytiskniHodnotyNastaveni();
32 void Odstartuj();
33 void NastaveniPreruseni();
34 void Michani();
35 void ZobrazMereneParametry();
36 void KoncovaObrazovka();
37 void Motor(bool Smer, int PWM);
38
39 #define JOY_Y      analogRead(JOY_Y_pin)
40 #define JOY_X      analogRead(JOY_X_pin)
41 #define JOY_TLA    digitalRead(JOY_TLA_pin)
42 #define ENK1       digitalRead(ENK1_pin)
43 #define ENK2       digitalRead(ENK2_pin)
44 #define JOY_VLEVO  JOY_X<100
45 #define JOY_VPRAVO JOY_X>900
46 #define JOY_NAHORU JOY_Y<100
47 #define JOY_DOLU   JOY_Y>900
48 #define JOY_TLA_ON JOY_TLA==0
49 #define JOY_TLA_OFF JOY_TLA==1
50 #define SMER_VLEVO 0
51 #define SMER_VPRAVO 1
52
53 void setup()
54 {
55     NastaveniPinu();
56     UvodniObrazovka();
57     NastaveniPoctuKroku();
58     NastaveniParametruOtaceni();
59     Odstartuj();
60     NastaveniPreruseni();
61     Michani();
62     KoncovaObrazovka();
63 }
64
65
66 /*****
67 ISR(TIMER2_OVF_vect) // funkce obsluhy preruseni
68 {
69     TCNT2 = 0; // nastavi hodnotu casovac_sekunde
70     casovac_desetin++; casovac_sekund++;
71
72     if(casovac_desetin == 6) // ubehlo 100 ms
73     {
74         casovac_desetin=0;
75         nastav_motor=1;
76     }
77
78     if(casovac_sekund == 61) // ubehla 1 sekunda
79     {
80         casovac_sekund=0;
81         zobraz_data=1;
82     }
83 }
84
```

```

87 void NastaveniPinu()
88 {
89     pinMode(JOY_TLA_pin, INPUT_PULLUP);
90     pinMode(ENK1_pin, INPUT);
91     pinMode(ENK2_pin, INPUT);
92     pinMode(MOTOR_IN1_pin, OUTPUT);
93     pinMode(MOTOR_IN2_pin, OUTPUT);
94     pinMode(MOTOR_EN_pin, OUTPUT);
95
96     lcd.begin(16, 2);
97 }
98
99 void UvodniObrazovka()
100 {
101     lcd.setCursor(2,0); lcd.print("Multifunkcni");
102     lcd.setCursor(0,1); lcd.print("rotacni zarizeni");
103     delay(1000); lcd.clear();
104 }
105
106 void NastaveniPocetukroku()
107 {
108     lcd.setCursor(3,0); lcd.print("NASTAVENI");
109     lcd.setCursor(0,1); lcd.print("Kroku: ");
110
111     while(JOY_TLA_OFF)
112     {
113         if(JOY_NAHORU) {pocet_kroku++;}
114         else if(JOY_DOLU) {pocet_kroku--;}
115
116         if(pocet_kroku>9) pocet_kroku=9;
117         if(pocet_kroku<1) pocet_kroku=1;
118
119         lcd.setCursor(7,1); lcd.print(" "); delay(200);
120         lcd.setCursor(7,1); lcd.print(pocet_kroku); delay(200);
121     }
122 }
123
124 void NastaveniParametruOtaceni()
125 {
126     lcd.clear();
127     lcd.setCursor(0,0); lcd.print("# Cas : :");
128     lcd.setCursor(0,1); lcd.print("Smer: Rpm:");
129
130     for(int a=0;a<=9;a++)
131     {hodiny[a]=0; minuty[a]=0; sekundy[a]=0;
132       smer[a]=SMER_VLEVO; rychlost[a]=300;
133
134       desitkyhodin[a]=0; jednotkyhodin[a]=0;
135       desitkyminut[a]=0; jednotkyminut[a]=0;
136       desitkysekund[a]=0; jednotkysekund[a]=0;
137       stovkyrpm[a]=3; desitkyrpm[a]=0; jednotkyrpm[a]=0;}
138
139     for(krok=1;krok<=pocet_kroku;krok++)
140     {
141         VytiskniHodnotyNastaveni();
142         delay(500);
143
144         while(JOY_TLA_OFF)
145         {
146             if(JOY_VLEVO) {pozice--; if(pozice<1) pozice=10; delay(200);}
147             if(JOY_VPRAVO) {pozice++; if(pozice>10) pozice=1; delay(200);}
148
149             switch(pozice)
150             {
151                 case 1: NastavDesitkyHodin(); break;
152                 case 2: NastavJednotkyHodin(); break;
153                 case 3: NastavDesitkyMinut(); break;
154                 case 4: NastavJednotkyMinut(); break;
155                 case 5: NastavDesitkySekund(); break;
156                 case 6: NastavJednotkySekund(); break;
157                 case 7: NastavSmer(); break;
158                 case 8: NastavStovkyRpm(); break;
159                 case 9: NastavDesitkyRpm(); break;
160                 case 10: NastavJednotkyRpm(); break;
161             }
162         }
163
164         for(int a=1;a<=9;a++)
165         {
166             hodiny[a] = (desitkyhodin[a]*10.0) + jednotkyhodin[a];
167             minuty[a] = (desitkyminut[a]*10.0) + jednotkyminut[a];
168             sekundy[a] = (desitkysekund[a]*10.0) + jednotkysekund[a];
169             rychlost[a] = (stovkyrpm[a]*100.0) + (desitkyrpm[a]*10.0) + jednotkyrpm[a];
170             CelkovaDobaKroku[a]=(hodiny[a]*3600.0)+(minuty[a]*60.0)+sekundy[a];
171         }
172     }
173 }
174

```

```

175 void NastavDesitkyHodin()
176 {
177     while(1)
178     {
179         if(JOY_NAHORU) desitkyhodin[krok]++;
180         if(JOY_DOLU) desitkyhodin[krok]--;
181         if(desitkyhodin[krok]<0) desitkyhodin[krok]=9; if(desitkyhodin[krok]>9) desitkyhodin[krok]=0;
182
183         lcd.setCursor(7,0);lcd.print(" "); delay(blik);
184         VytiskniHodnotyNastaveni(); delay(blik);
185         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
186     }
187 }
188
189 void NastavJednotkyHodin()
190 {
191     while(1)
192     {
193         if(JOY_NAHORU) jednotkyhodin[krok]++;
194         if(JOY_DOLU) jednotkyhodin[krok]--;
195         if(jednotkyhodin[krok]<0) jednotkyhodin[krok]=9; if(jednotkyhodin[krok]>9) jednotkyhodin[krok]=0;
196
197         lcd.setCursor(8,0);lcd.print(" "); delay(blik);
198         VytiskniHodnotyNastaveni(); delay(blik);
199         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
200     }
201 }
202
203
204 void NastavDesitkyMinut()
205 {
206     while(1)
207     {
208         if(JOY_NAHORU) desitkyminut[krok]++;
209         if(JOY_DOLU) desitkyminut[krok]--;
210         if(desitkyminut[krok]<0) desitkyminut[krok]=5; if(desitkyminut[krok]>5) desitkyminut[krok]=0;
211
212         lcd.setCursor(10,0);lcd.print(" "); delay(blik);
213         VytiskniHodnotyNastaveni(); delay(blik);
214         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
215     }
216 }
217
218 void NastavJednotkyMinut()
219 {
220     while(1)
221     {
222         if(JOY_NAHORU) jednotkyminut[krok]++;
223         if(JOY_DOLU) jednotkyminut[krok]--;
224         if(jednotkyminut[krok]<0) jednotkyminut[krok]=9; if(jednotkyminut[krok]>9) jednotkyminut[krok]=0;
225
226         lcd.setCursor(11,0);lcd.print(" "); delay(blik);
227         VytiskniHodnotyNastaveni(); delay(blik);
228         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
229     }
230 }
231
232
233 void NastavDesitkySekund()
234 {
235     while(1)
236     {
237         if(JOY_NAHORU) desitkysekund[krok]++;
238         if(JOY_DOLU) desitkysekund[krok]--;
239         if(desitkysekund[krok]<0) desitkysekund[krok]=5; if(desitkysekund[krok]>5) desitkysekund[krok]=0;
240
241         lcd.setCursor(13,0);lcd.print(" "); delay(blik);
242         VytiskniHodnotyNastaveni(); delay(blik);
243         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
244     }
245 }
246
247 void NastavJednotkySekund()
248 {
249     while(1)
250     {
251         if(JOY_NAHORU) jednotkysekund[krok]++;
252         if(JOY_DOLU) jednotkysekund[krok]--;
253         if(jednotkysekund[krok]<0) jednotkysekund[krok]=9; if(jednotkysekund[krok]>9) jednotkysekund[krok]=0;
254
255         lcd.setCursor(14,0);lcd.print(" "); delay(blik);
256         VytiskniHodnotyNastaveni(); delay(blik);
257         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
258     }
259 }

```



```

261 void NastavSmer()
262 {
263     while(1)
264     {
265         if(JOY_DOLU) smer[krok]=SMER_VLEVO;
266         if(JOY_NAHORU) smer[krok]=SMER_VPRAVO;
267
268         lcd.setCursor(6,1);lcd.print(" ");delay(blik);
269         VytiskniHodnotyNastaveni();delay(blik);
270         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
271     }
272 }
273
274
275 void NastavStovkyRpm()
276 {
277     while(1)
278     {
279         if(JOY_NAHORU) stovkyrpm[krok]++;
280         if(JOY_DOLU) stovkyrpm[krok]--;
281         if(stovkyrpm[krok]<0) stovkyrpm[krok]=6; if(stovkyrpm[krok]>6) stovkyrpm[krok]=0;
282
283         lcd.setCursor(12,1);lcd.print(" "); delay(blik);
284         VytiskniHodnotyNastaveni(); delay(blik);
285         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
286     }
287 }
288
289 void NastavDesitkyRpm()
290 {
291     while(1)
292     {
293         if(JOY_NAHORU) desitkyrpm[krok]++;
294         if(JOY_DOLU) desitkyrpm[krok]--;
295         if(desitkyrpm[krok]<0) desitkyrpm[krok]=9; if(desitkyrpm[krok]>9) desitkyrpm[krok]=0;
296         if(stovkyrpm[krok]==0 && desitkyrpm[krok]<6) desitkyrpm[krok]=6;
297
298         lcd.setCursor(13,1);lcd.print(" "); delay(blik);
299         VytiskniHodnotyNastaveni(); delay(blik);
300         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
301     }
302 }
303
304
305 void NastavJednotkyRpm()
306 {
307     while(1)
308     {
309         if(JOY_NAHORU) jednotkyrpm[krok]++;
310         if(JOY_DOLU) jednotkyrpm[krok]--;
311         if(jednotkyrpm[krok]<0) jednotkyrpm[krok]=9; if(jednotkyrpm[krok]>9) jednotkyrpm[krok]=0;
312
313         lcd.setCursor(14,1);lcd.print(" "); delay(blik);
314         VytiskniHodnotyNastaveni(); delay(blik);
315         if(JOY_TLA_ON || JOY_VPRAVO || JOY_VLEVO) break;
316     }
317 }
318
319 void VytiskniHodnotyNastaveni()
320 {
321     lcd.setCursor(1,0); lcd.print(krok);
322     lcd.setCursor(7,0); lcd.print(desitkyhodin[krok]); lcd.setCursor(8,0); lcd.print(jednotkyhodin[krok]);
323     lcd.setCursor(10,0); lcd.print(desitkyminut[krok]); lcd.setCursor(11,0); lcd.print(jednotkyminut[krok]);
324     lcd.setCursor(13,0); lcd.print(desitkysekund[krok]); lcd.setCursor(14,0); lcd.print(jednotkysekund[krok]);
325
326     if(smer[krok]== SMER_VLEVO) {lcd.setCursor(6,1); lcd.print("<");}
327     else {lcd.setCursor(6,1); lcd.print(">");}
328
329     lcd.setCursor(12,1); lcd.print(stovkyrpm[krok]); lcd.setCursor(13,1); lcd.print(desitkyrpm[krok]);lcd.setCursor(14,1); lcd.print(jednotkyrpm[krok]);
330 }
331
332 void Odstartuj()
333 {
334     lcd.clear();
335     lcd.setCursor(0,0); lcd.print("Stiskni joystick");
336     lcd.setCursor(3,1); lcd.print("pro start!");
337     delay(500);
338     while(JOY_TLA_OFF);
339 }
340
341
342 void NastaveniPreruseni()
343 {
344     // Nastaveni preruseni TIMER2 prevzato z https://arduino-info.wikispaces.com/Timers-Arduino
345
346     noInterrupts(); // vypne vsechna preruseni
347     TCCR2A = 0; TCCR2B = 0;
348     TCNT2 = 0; // vynuluje casovac_sekund
349     TCCR2B |= (1 << CS20) | (1 << CS21) | (1 << CS22); // preddelicka casovace 1:1024
350     TIMSK2 |= (1 << TOIE2); // nastaveni preruseni pretocenim casovace
351     interrupts(); // zapne vsechna preruseni
352 }

```

```

354 void ZobrazMereneParametry()
355 {
356     CelkovaDobaKroku[0]--;
357     sekundy[0]--;
358     if (sekundy[0]<0) {minuty[0]--; sekundy[0]=59;}
359     if (minuty[0]<0) {hodiny[0]--; minuty[0]=59;}
360     if (hodiny[0]<0) {hodiny[0]=0; minuty[0]=0; sekundy[0]=0;}
361
362     lcd.clear();lcd.setCursor(0,0); lcd.print("#      : ");
363     lcd.setCursor(0,1);          lcd.print("      rpm pwm:");
364
365     lcd.setCursor(1,0);lcd.print(krok);
366
367     if (hodiny[0]<10) {lcd.setCursor(6,0); lcd.print("0"); lcd.setCursor(7,0); lcd.print(hodiny[0]); }
368     else {lcd.setCursor(6,0); lcd.print(hodiny[0]);}
369
370     if (minuty[0]<10) {lcd.setCursor(9,0); lcd.print("0"); lcd.setCursor(10,0); lcd.print(minuty[0]); }
371     else {lcd.setCursor(9,0); lcd.print(minuty[0]);}
372
373     if (sekundy[0]<10) {lcd.setCursor(12,0); lcd.print("0"); lcd.setCursor(13,0); lcd.print(sekundy[0]); }
374     else {lcd.setCursor(12,0); lcd.print(sekundy[0]);}
375
376     if (rychlost[0]<100) {lcd.setCursor(2,1); lcd.print(" "); lcd.setCursor(3,1); lcd.print(rychlost[0]); }
377     else {lcd.setCursor(2,1); lcd.print(rychlost[0]);}
378
379     if (smer[krok] == SMER_VLEVO) {lcd.setCursor(0,2); lcd.print("<");}
380     else {lcd.setCursor(0,2); lcd.print(">");}
381
382     if (pwm<10) {lcd.setCursor(13,1); lcd.print(" "); lcd.setCursor(15,1); lcd.print(pwm); }
383     else if (pwm<100) {lcd.setCursor(13,1); lcd.print(" "); lcd.setCursor(14,1); lcd.print(pwm); }
384     else {lcd.setCursor(13,1); lcd.print(pwm);}
385 }
386
387 /*****
388 /*****
389
390 void Michani()
391 {
392     for (krok=1;krok<=pocet_kroku;krok++)
393     {
394         hodiny[0]=hodiny[krok]; minuty[0]=minuty[krok]; sekundy[0]=sekundy[krok]; CelkovaDobaKroku[0]=CelkovaDobaKroku[krok];
395         pwm = 128;
396         Motor(smer[krok],pwm);
397
398         while(CelkovaDobaKroku[0] > 0)
399         {
400             while (ENK1); while (!ENK1);
401             if (ENK2) enkoder++;
402             else enkoder--;
403
404             if (nastav_motor == 1)
405             {
406                 nastav_motor = 0;
407                 rychlost[0] = abs(enkoder) / 3; // enkoder 1800 tiků/ot; 1 rpm = 30 tiků/ot/min = 3 tiků/ot/100 ms
408                 if (rychlost[0] < rychlost[krok]) pwm++;
409                 if (rychlost[0] > rychlost[krok]) pwm--;
410                 if (abs(enkoder) < 3) pwm = 0;
411                 if (pwm < 0) pwm = 0;
412                 if (pwm > 255) pwm = 255;
413                 Motor(smer[krok], pwm);
414                 enkoder = 0;
415             }
416
417             if (zobraz_data == 1) {zobraz_data = 0; ZobrazMereneParametry();}
418         }
419     }
420 }
421
422 void KoncovaObrazovka()
423 {
424     Motor(0,0);
425     noInterrupts(); // vypne vsechna preruseni
426     lcd.clear();
427     lcd.setCursor(0,0); lcd.print("Program skoncil");
428     lcd.setCursor(0,1); lcd.print("Prijdte zas :->");
429     while(1);
430 }
431
432 /*****
433
434 void Motor(bool Smer, int PWM)
435 {
436     if (Smer == SMER_VLEVO) {digitalWrite(MOTOR_IN1_pin, 1); digitalWrite(MOTOR_IN2_pin, 0);}
437     else {digitalWrite(MOTOR_IN1_pin, 0); digitalWrite(MOTOR_IN2_pin, 1);}
438     analogWrite(MOTOR_EN_pin, PWM);
439 }
440
441
442 void loop() {}
443
444

```

Obrázek 64: Finální model, výpis programu pro Arduino Uno

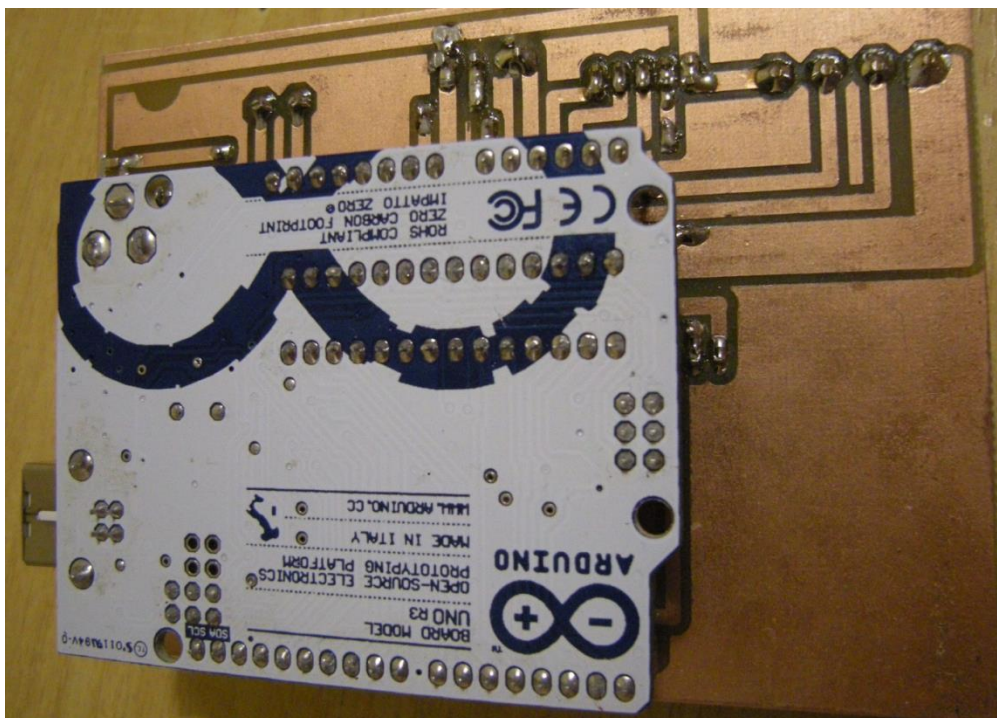
5.8 Návod k montáži zařízení

K finální montáži celého mísičního zařízení jsou potřeba tyto součástky:

- Arduino Uno
- LCD displej, alfanumerický 16 x 2 (z Arduino sady), s napájenou řadou dlouhých pinů k zasunutí do dutinek v ovládací desce.
- joystick (z Arduino sady)
- ovládací deska (shield), vyrobená a napájená dle schémat uvedených v kapitole 5.3
- IDC konektor pro 10žilový plochý kabel (cca 20 cm dlouhý), zakončený 5 dutinkami se signály VDD, GND, VR_X, VR_Y a SW dle schématu v kapitole 5.3
- krabičku s víkem vytištěnou na 3D tiskárně dle kapitoly 5.4
- rám tiskárny, demontovaný postupem dle kapitoly 5.5
- napájecí síťový adaptér (např. z jiné tiskárny 12 – 24 VDC) zakončený konektorem 5,5 mm (plus v jádře, mínus na plášti konektoru)
- napájecí zásuvka pro konektor 5,5 mm
- páčkový vypínač
- kabelová průchodka PG9

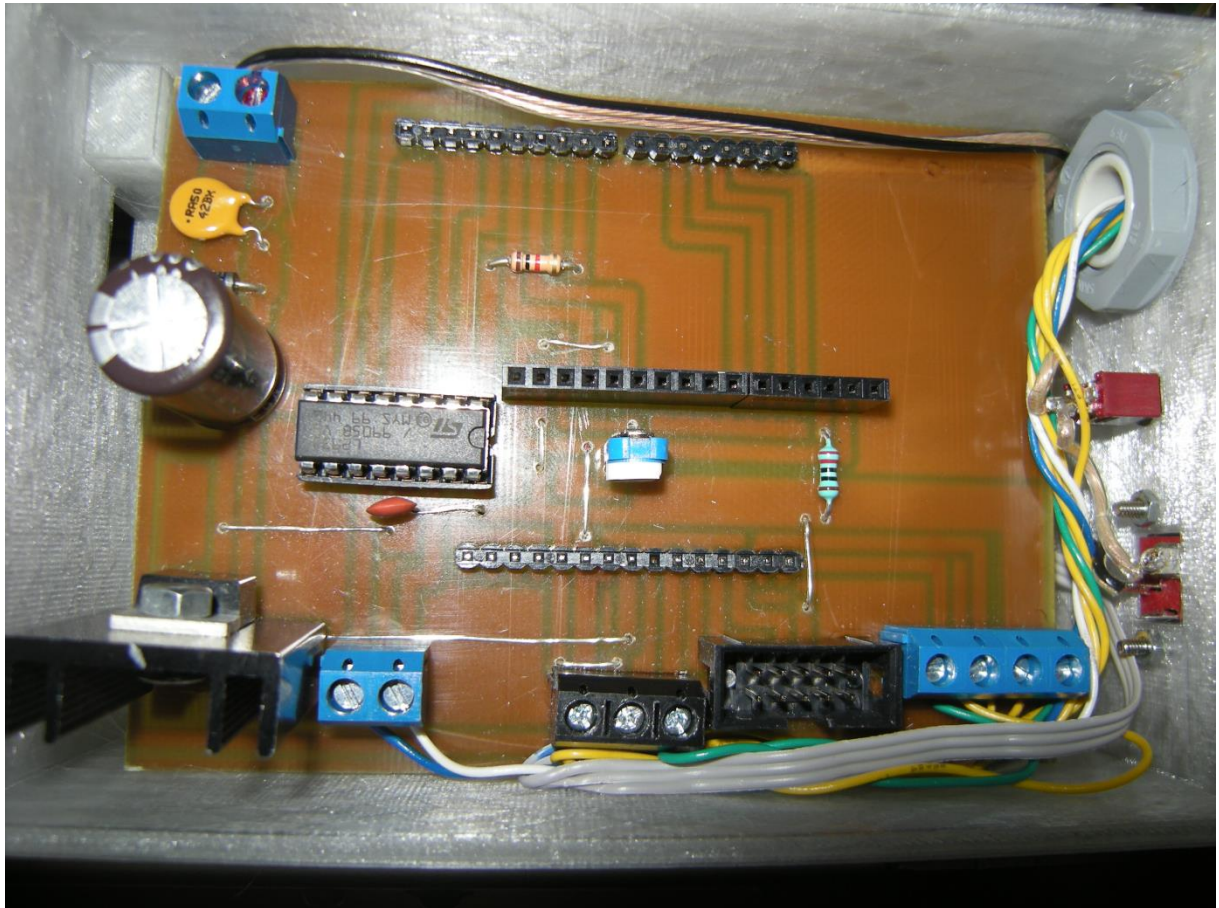
Součástky zkompletujeme následujícím postupem:

- 1) Piny ovládací desky zasuneme do příslušných dutinek desky Arduino Uno



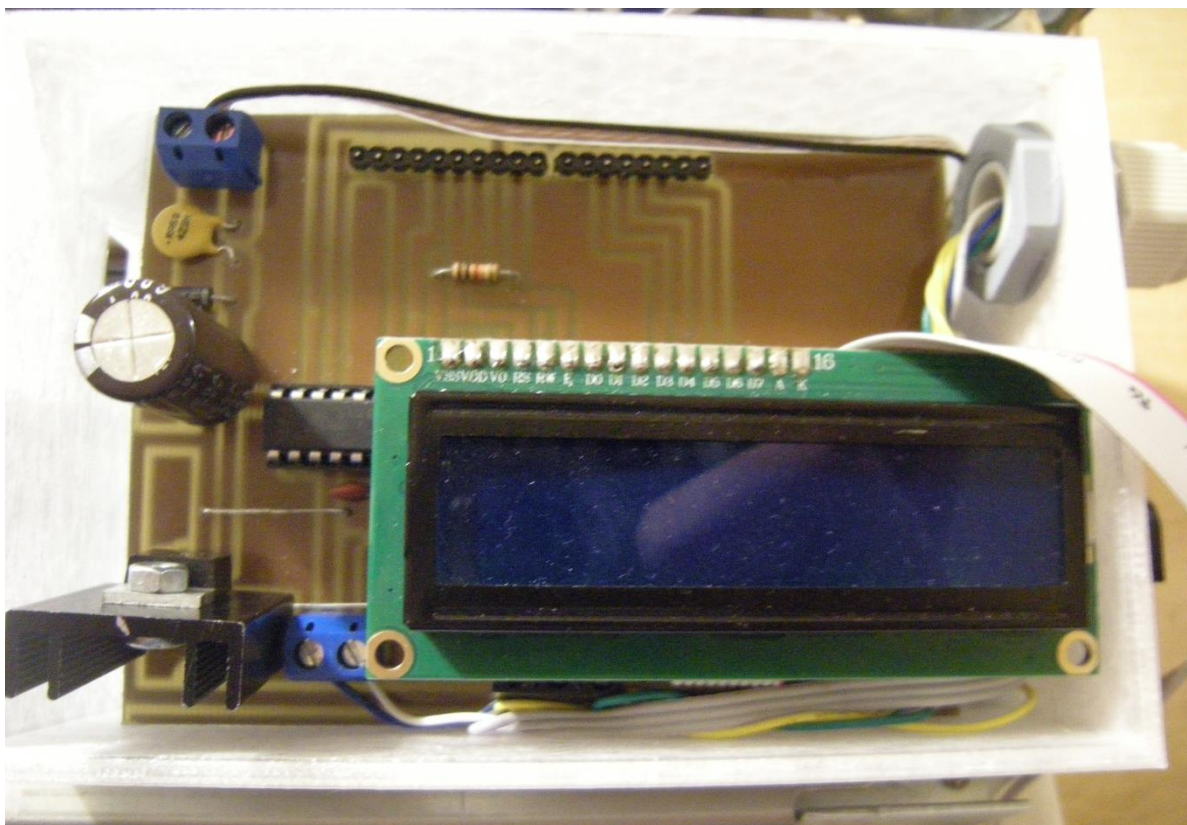
Obrázek 65: Finální model, ovládací deska zapojená do Arduina

- 2) Protáhneme kabely od motoru a enkodéru průchodkou PG9 a zapojíme dle schématu, uvedeného v kapitole 5.3.
- 3) Připájíme vodiče zásuvky napájení a páčkového vypínače.
- 4) Napájecí kabely z vypínače připojíme do svorkovnice napájení dle schématu uvedeného v kapitole 5.3.



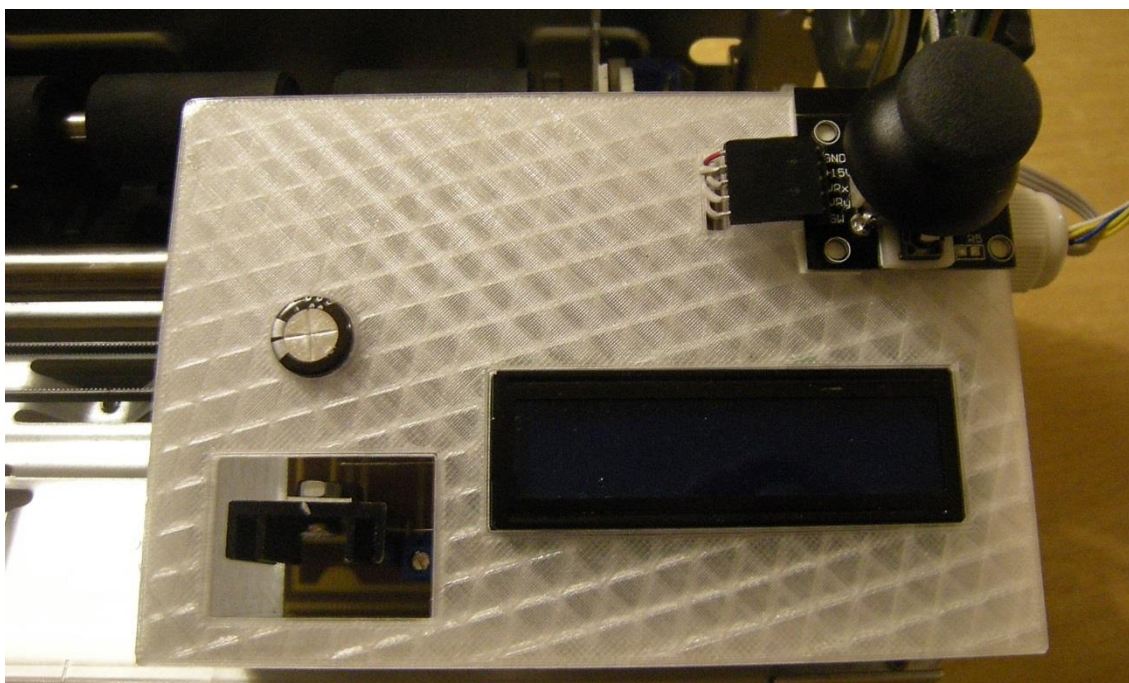
Obrázek 66: Finální model, ovládací deska s připojenými kabely usazená v krabičce

- 5) IDC konektor s plochým kabelem zapojíme do konektoru na desce a víkem krabičky protáhneme dutinky pro připojení joysticku.
- 6) Vložíme sestavu ovládací desku s Arduinem a připojenými kabely do krabičky.
- 7) Zasuneme LCD displej do dutinek dle schématu v kapitole 5.3



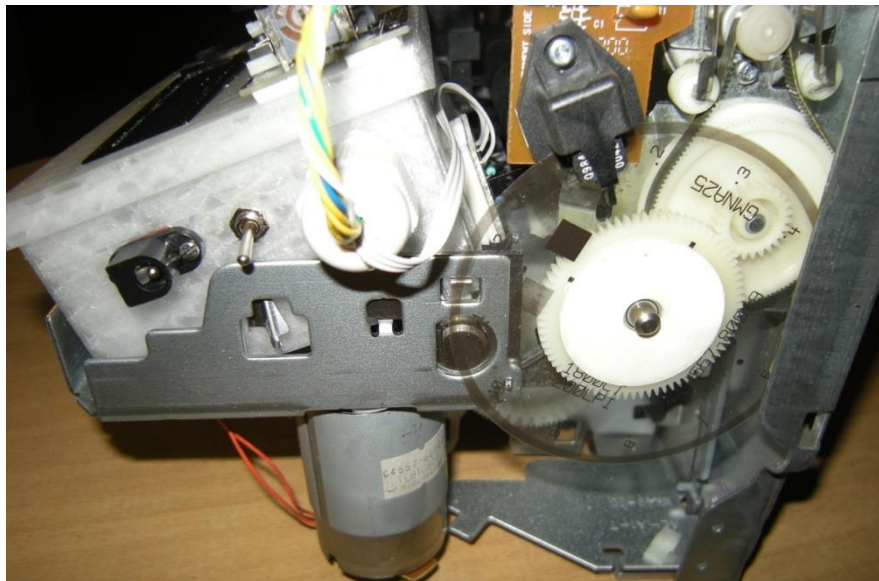
Obrázek 67: Finální model, ovládací deska s připojenými LCD displejem a IDC konektorem

8) Joystick přilepíme oboustrannou lepicí páskou na víko krabičky

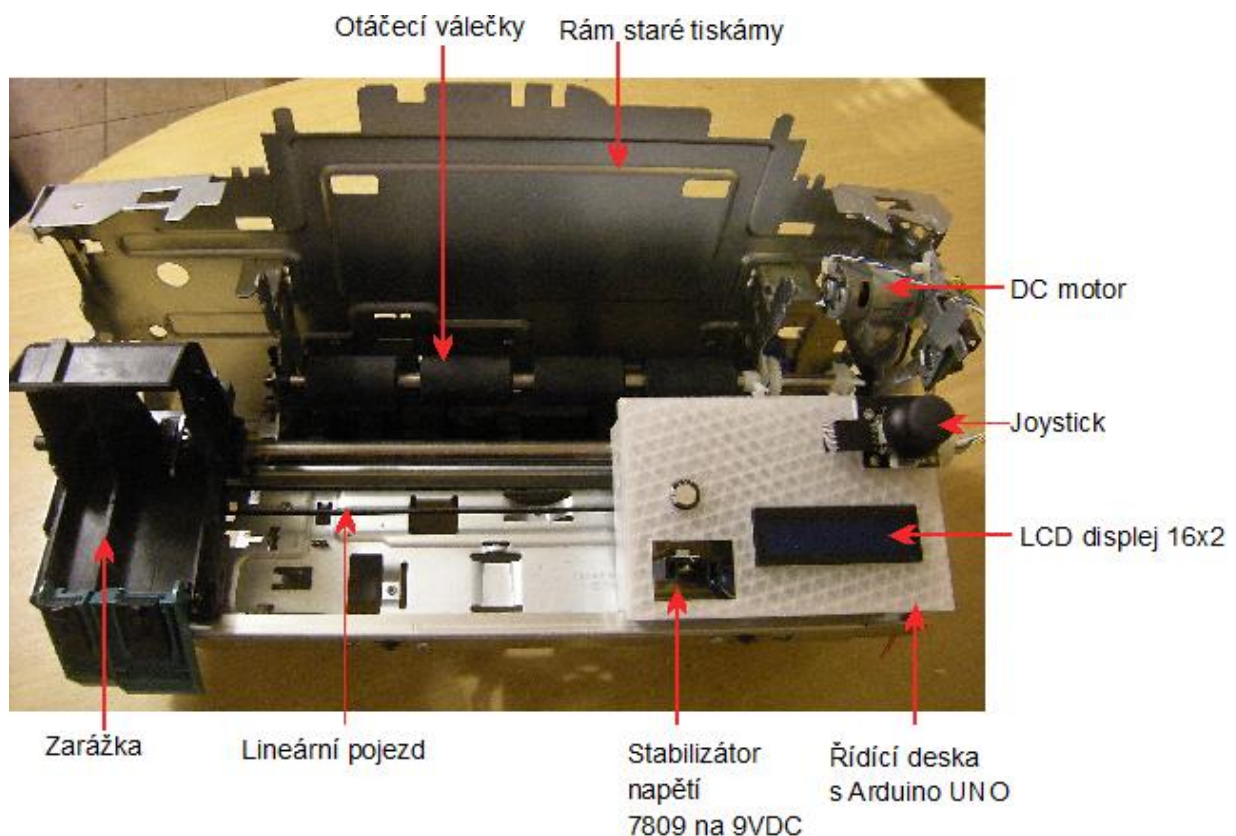


Obrázek 68: Finální model, obrázek řídicí jednotky v krabičce

- 9) Dno krabičky přilepíme oboustrannou lepící páskou k rámu tiskárny a dotáhneme kabelovou průchodku PG9. Kabely od motoru a enkodéru vedeme tak, aby se při pohybu tiskárny nemohly poškodit.



Obrázek 69: Finální model, řídicí jednotka z boku, s detailem enkodéru

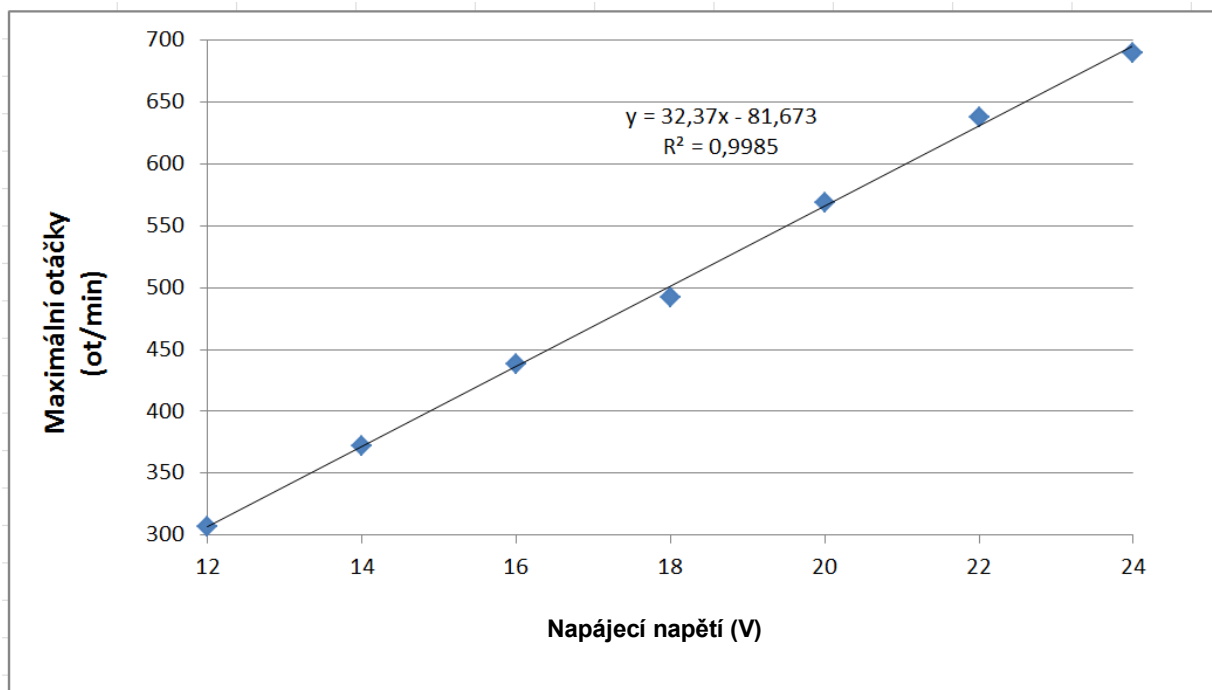


Obrázek 70: Finální model, celková sestava

5.9 Technické parametry zařízení

Vyrobené mísicí zařízení využívající konstrukci tiskárny HP DeskJet 990cxi má následující technické parametry:

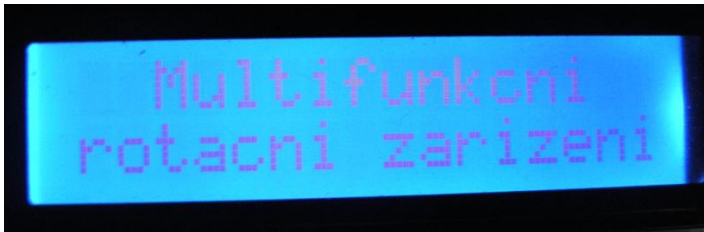
- Napájení: 12 – 24 VDC ze síťového adaptéru, například z jiné staré tiskárny
- Proudová spotřeba: méně než 500 mA
- Celkové rozměry zařízení (d x š x v): 425 mm x 180 mm x 200 mm
- Celková váha: 3,8 kg
- Maximální délka míchaného předmětu: 235 mm
- Minimální průměr míchaného předmětu: 35 mm
- Maximální průměr míchaného předmětu: 60 mm
- Minimální otáčky: 60 ot/min
- Maximální otáčky: viz následující graf závislosti maximálních otáček na napájecím napětí, v rozsahu 12 – 24 VDC.



Obrázek 71: Závislost maximálních otáček zařízení na maximálním napětí

5.10 Návod k použití zařízení

- 1) Zařízení připojíme k síťovému adaptéru dávajícímu bezpečné napětí 12 – 24 VDC a dimenzovanému na proud min. 500 mA.
- 2) Míchaný předmět vložíme na válečky rotačního zařízení a zajistíme posunutím držáku tiskových hlav proti jeho vypadnutí při otáčení.
- 3) Páčkovým vypínačem zapneme řídicí jednotku, na displeji se na 1 sekundu objeví úvodní obrazovka s názvem zařízení.



Obrázek 72: Úvodní obrazovka

- 4) V dalším kroku pohybem joysticku nahoru a dolů zvolíme počet kroků rotačního programu (max. 9).



Obrázek 73: Obrazovka nastavení počtu kroků

- 5) Následně nastavíme parametry otáčení v každém kroku – dobu otáčení ve formátu hodiny : minuty : sekundy (max. 99 hodin, 59 minut a 59 sekund), směr otáčení motoru a rychlost otáčení Rpm (v ot/min). Jednotlivé číslice se zvětšují pohybem joysticku nahoru a dolů, mezi pozicemi se přechází pohybem joysticku vlevo a vpravo. Volba nastavení parametrů daného kroku se potvrdí stisknutím joysticku.



Obrázek 74: Obrazovka nastavení parametrů otáčení

- 6) Po nastavení parametrů všech kroků přejde program na startovací okno a otáčení se spustí stisknutím joysticku.

A blue LCD display with red text that reads "Stiskni joystick Pro start!".

Obrázek 75: Startovací obrazovka

- 7) Při běhu otáčení se na obrazovce zobrazuje čas zbývající do ukončení daného kroku, aktuální otáčky (rpm = ot/min) a regulační výkon vkládaný na motor (pwm). Otáčky se regulují každých 100 ms podle naměřeného počtu pulzů z enkodéru (při 1800 pulzech na otáčku odpovídá 1 ot/min 3 pulzům za 100 ms). Pokud jsou aktuální otáčky jiné než požadované, zvýší nebo sníží se každých 100 ms regulační výkon motoru o jednotku pwm. Maximální hodnota 8bitového pwm modulu Arduina Uno je 255. Pokud nelze dosáhnout požadovaných otáček při maximální hodnotě pwm, je zapotřebí zvýšit napájecí napětí síťového adaptéru (viz graf na obr. 71). Program si v 100ms intervalech hlídá i případné zaseknutí chodu motoru (otáčky menší než 1 ot/min), v tomto případě odpojí napájení motoru.

A blue LCD display with red text showing "#1 01:03:04" on the top line and "< 299rpm PWM:254" on the bottom line.

Obrázek 76: Obrazovka běhu programu

- 8) Po uběhnutí nastavené doby všech kroků otáčení se zařízení automaticky vypne a objeví se koncová obrazovka. Zařízení se vypne páčkovým vypínačem a vytažením síťového zdroje z konektoru napájení.

A blue LCD display with red text that reads "Program skoncil" on the top line and "Prijďte zas !->" on the bottom line.

Obrázek 77: Koncová obrazovka.

6 Závěr

Zadání práce – sestrojil levné, výkonné a uživatelsky nenáročné natahovače automatických hodinek, v ceně řádově stovek korun, s použitím dílů ze starých tiskáren, se podařilo splnit. U všech zkonstruovaných modelů nepřekročila cena součástí pro výrobu mísičů 500 Kč bez DPH (bez započítání ceny vyřazené tiskárny, která se dá obvykle získat zdarma nebo za cenu poštovného).

Základní jednoduchá verze natahovače pro jedny hodinky, tvořená převodovkou s motorem z tiskárny a krabičkou od žvýkaček, se ovládá pomocí potenciometru, umožňujícího nastavit jak směr otáčení, tak jeho rychlost. Použit je možno prakticky jakoukoliv sestavu motory s převodovkou vymontovanou z různých typů tiskáren, popsané zapojení řídicí desky bude stejné.

Pokročilá verze natahovače, využívající celý rám inkoustové tiskárny HP DeskJet 990cxi (podobnou konstrukci mají i další tiskárny řady 9XX), umožňuje současně natahovat až 6 kusů hodinek, má ovládání pomocí 5 tlačítek a alfanumerického displeje a dovoluje nastavit parametry otáčení (rychlost, směr, čas) v až 9 nezávislých krocích.

Pro možnost většího rozšíření této varianty jsem sestrojila i nástavbu (shield) pro populární řídicí desku Arduino Uno, nakreslila model krabičky řídicí jednotky pro vytištění na 3D tiskárně, napsala řídicí software pro Arduino Uno a popsala postup demontáže typické inkoustové tiskárny HP DeskJet 990cxi. Technické podklady pro výrobu této varianty (model krabičky a víka ve SketchUp a formátu STL pro 3D tisk, program pro Arduino Uno, schémata plošných spojů v CAD Eagle) jsou jako open-software a open-hardware přiloženy k elektronické verzi této práce a budou také zveřejněny ke stažení na webu našeho týmu www.hahaharobotics.tk.

Vyvinuté zařízení využívající rotační mechanismy starých tiskáren ve spojení s novou řídicí deskou může najít i širší využití pro amatérskou vědu a techniku než jsou natahovače automatických hodinek. Příkladem může být míchání zkumavek v biologických a chemických laboratořích, míchání sprejů s barvami, mísení práškových komponent, míchání potravin nebo otáčení vajec v inkubátorech a výroba tohoto zařízení může začínajícím vědcům či technikům ušetřit tisíce korun za nákup komerčních mísičů.

Z časových důvodů jsem bohužel nestihla některá uživatelská vylepšení řídicího softwaru (např. možnost předvolby nejčastějších programů míchání a jejich zápis do EEPROM mikrokontroléru) a ověřit použitelnost tiskárny i pro další uvažované aplikace, tuto část proto již nechávám na potenciálních uživateli tohoto nápadu :)

6.1 Seznam použitých obrázků

Obrázek 1: TIMELESS 102 031 KRAFF ^[1]	9
Obrázek 2: WOLF 70039/18 Viceroy 3+6 ^[2]	9
Obrázek 3: MARS 201011 ^[3]	9
Obrázek 4: KRAFF 196 012 Saturn ^[4]	9
Obrázek 5: WOLF 70039/23	9
Obrázek 6: Senna Klarstein ^[6]	9
Obrázek 7: Klagenfurt Klarstein ^[7]	10
Obrázek 8: WOLF Viceroy Watch Winder ^[8]	10
Obrázek 9: CHIYODA Watch Winder ^[9]	10
Obrázek 10: Wolf Designs Viceroy 6 Watch Winder ^[10]	10
Obrázek 11: DIY watch winder od khanhquagioi ^[11]	12
Obrázek 12: DIY Homemade Watch Winder od TheMrJavaguy ^[12]	12
Obrázek 13: Cheapest homemade diy watch winder od Ted Valez ^[13]	12
Obrázek 14: Homemade Automatic Watch Winder od whaczup ^[14]	12
Obrázek 15: Homemade Automatic Watch Winder od SoCal Skateboards ^[15]	12
Obrázek 16: DIY watch winder by wood od Pham KS ^[16]	12
Obrázek 17: DIY Watch Winder od Best	13
Obrázek 18: My Kinetic watch charger od Mortnoiruk ^[18]	13
Obrázek 19: Automatic Watch Winder od Horology Zone	13
Obrázek 20: Lego Watch Winder	13
Obrázek 21: Celková sestava Watch Winder Orbit Peppermint	15
Obrázek 22: Obrázek součástí natahovače Watch Winder Orbit Peppermint	16
Obrázek 23: Watch Winder Orbit Peppermint, funkční schéma	18
Obrázek 24: Watch Winder Orbit Peppermint, pájecí schéma na univerzálním řádkovém plošném spoji (rozteč = 2,54 mm)	19
Obrázek 25: Řídící deska Watch Winder Orbit Peppermint s krabičkou	19
Obrázek 26: Watch Winder Orbit Peppermint algoritmus programu	20
Obrázek 27: Watch Winder Orbit Peppermint, výpis programu pro PIC16F1847	21
Obrázek 28: Hlavní díly ke stavbě Watch Winder Orbit Peppermint	22
Obrázek 29: Watch Winder Orbit Peppermint, plastová krabička s automatickými hodinkami ..	22
Obrázek 30: Zkompletovaný Watch Winder Orbit Peppermint	23
Obrázek 31: Watch winder multi deluxe, profilový obrázek	24
Obrázek 32: Watch winder multi deluxe, řídicí deska s popisky	26
Obrázek 33: Watch winder multi deluxe, funkční schéma	29
Obrázek 34: Watch winder multi deluxe, pájecí schéma (univerzální řádkový plošný spoj; e=2,54 mm)	30
Obrázek 35: Watch Winder Multi Deluxe, algoritmus programu	31
Obrázek 36: Watch Winder Multi Deluxe, výpis programu pro PIC18F26K22	36
Obrázek 37: Watch winder multi deluxe, pohled zepředu	37
Obrázek 38: Watch winder multi deluxe, pohled zezadu	37
Obrázek 39: Watch winder multi deluxe, řídicí jednotka	38
Obrázek 40: Watch winder multi deluxe, počítadlo otáček s optickou závorou (vlevo)	38
Obrázek 41: Watch winder multi deluxe, motor s převodovkou (vpravo)	38
Obrázek 42: Watch winder multi deluxe, hodinky, vyplněné plastovou pěnou	39

Obrázek 43: Watch winder multi deluxe, programová okna.....	39
Obrázek 44: Funkční schéma zapojení ovládací desky (shield) pro Arduino UNO.....	43
Obrázek 45: Schéma plošného spoje ovládací desky (shield) pro Arduino Uno.....	44
Obrázek 46: Finální model, ovládací deska s popiskami.....	45
Obrázek 47: Finální model, pohled na ovládací desku zesponu.....	45
Obrázek 48: Celkový pohled na krabičku a víko.....	46
Obrázek 49: Pravý bokorys krabičky s otvory pro napájení, vypínač a průchodku kabelů.....	46
Obrázek 50: Levý bokorys krabičky s otvorem pro programovací kabel Arduino.....	47
Obrázek 51: Nárýs krabičky.....	47
Obrázek 52: Půdorys krabičky.....	48
Obrázek 53: Finální model, foto vytisknuté krabičky.....	48
Obrázek 54: Tiskárna HP DeskJet 990cxi v původním stavu.....	49
Obrázek 55: Tiskárna po odstranění všech plastových krytů.....	50
Obrázek 56: Rám tiskárny zepředu po odstranění plastových krytů.....	50
Obrázek 57: Rám tiskárny zezadu po odstranění plastových krytů.....	51
Obrázek 58: Detail původní řídicí desky tiskárny.....	51
Obrázek 59: Tiskové patrony na lineárních pojezdech tiskárny.....	52
Obrázek 60: Detail optického enkodéru.....	52
Obrázek 61: Souhrn elektronických součástí vyjmutých z tiskárny.....	53
Obrázek 62: Rám tiskárny s válcem, motory a enkodérem po dokončené demontáži.....	53
Obrázek 63: Finální model, algoritmus programu.....	55
Obrázek 64: Finální model, výpis programu pro Arduino Uno.....	60
Obrázek 65: Finální model, ovládací deska zapojená do Arduina.....	61
Obrázek 66: Finální model, ovládací deska s připojenými kabely usazená v krabičce.....	62
Obrázek 67: Finální model, ovládací deska s připojenými LCD displejem a IDC konektorem.....	63
Obrázek 68: Finální model, obrázek řídicí jednotky v krabičce.....	63
Obrázek 69: Finální model, řídicí jednotka z boku, s detailem enkodéru.....	64
Obrázek 70: Finální model, celková sestava.....	64
Obrázek 71: Závislost maximálních otáček zařízení na maximálním napětí.....	65
Obrázek 72: Úvodní obrazovka.....	66
Obrázek 73: Obrazovka nastavení počtu kroků.....	66
Obrázek 74: Obrazovka nastavení parametrů otáčení.....	66
Obrázek 75: Startovací obrazovka.....	67
Obrázek 76: Obrazovka běhu programu.....	67
Obrázek 77: Koncová obrazovka.....	67

6.2 Seznam použitých tabulek

Tabulka 1: Tabulka komerčních natahovačů hodinek.....	8
Tabulka 2: Tabulka amatérských řešení natahovačů hodinek.....	11
Tabulka 3: Watch Winder Orbit Peppermint, tabulka součástí.....	17
Tabulka 4: Watch Winder Multi Deluxe, tabulka součástí.....	28
Tabulka 5: Finální verze, tabulka součástí.....	42
Tabulka 6: Tabulka dekodování vodičů enkodéru.....	54

6.3 Seznam použitých odkazů

TIMELESS 102 031 KRAFF^[1]: <http://www.hodinky-doplky.cz/natahovac-hodinek-timeless-102-031-kraff-watch-winder/d-70326/>

WOLF 70039/18: <http://www.hodinky.inshop.cz/natahovac-hodinek-wolf-70039-18-viceroy-3-6.html>

Viceroy3+6^[2]: <http://www.hodinky.inshop.cz/natahovac-hodinek-wolf-70039-18-viceroy-3-.html>

MARS 201011^[3]: <http://www.hodinky-time.cz/natahovac-hodinek-kraff-201-011-mars#tb1=2>

KRAFF 196 012 Saturn^[4]: <http://www.hodinky.inshop.cz/natahovac-hodinek-kraff-196-012-saturn.html>

WOLF 70039/23 Roadstar 1+3^[5]: <http://www.hodinky.inshop.cz/natahovac-hodinek-wolf-70039-23-roadstar-1-3.html>

Senna Klarstein^[6]: <http://www.klarstein.cz/cs/Moderni-bydleni/Natahovace-hodinek/Senna-cerveny-pohyblivy-stojan-na-hodinky-natahovac-vlevo-vpravo.html>

Klagenfurt Klarstein^[7]: <http://www.klarstein.cz/cs/Moderni-bydleni/Natahovace-hodinek/Klagenfurt-pohyblivy-stojan-na-hodinky-pohyb-vpravo-vlevo-12-hodinek-LED-dotykovy-displej.html>

WOLF Viceroy Watch Winder^[8]: <https://www.youtube.com/watch?v=ifsXITOIGD4>

CHIYODA Watch Winder^[9]: <https://www.youtube.com/watch?v=DlCd1cdQ2fE>

Wolf Designs Viceroy 6 Watch Winder^[10]: <http://www.ablogtowatch.com/wolf-designs-viceroy-6-piece-watch-winder-review/>

DIY watch winder^[11]: <https://www.youtube.com/watch?v=CMIWJ37WkSM>

DIY Homemade Watch Winder^[12]: <http://www.how-to-diy.org/8gtuqfix0adu0b/diy-homemade-watch-winder-in-hd.html>

Cheapest homemade diy watch winder^[13]: <http://www.how-to-diy.org/wF9-lFiXo04SKW/Cheapest-homemade-diy-watch-winder.html>

Homemade Automatic Watch Winder^[14]: <http://www.how-to-diy.org/ARpJlFiXrSmKit/Homemade-Automatic-Watch-Winder.html>

Homemade Automatic Watch Winder^[15]: https://www.youtube.com/watch?v=Rans_fj6l2M

DIY watch winder by wood^[16]: <https://www.youtube.com/watch?v=ChuRCA58AqQ>

DIY Watch Winder ^[17]: <https://www.youtube.com/watch?v=tvdHAtoZOH4>

My Kinetic watch charger^[18]: https://www.youtube.com/watch?v=qcpb0bbBi_Y

Automatic Watch Winder DIY ^[19]: <https://www.youtube.com/watch?v=SwVqwgROJKs>

Lego Watch Winder (Mk 4) ^[20]: <https://www.youtube.com/watch?v=caPrK7R9OI4>

Popis robota BÉDA4 ^[21]: <http://www.robotshop.com/letsmakerobots/beda-4-%E2%80%93-my-first-line-follower>

Fórum Let's Make Robots ^[22]: <http://www.robotshop.com/letsmakerobots/watchwinder-%E2%80%93-corbit-peppermint%E2%80%93>

Fórum Let's Make Robots ^[23]: <http://www.robotshop.com/letsmakerobots/mixer-a-printer-watch-winder-multi-deluxe>

Wiki Arduino ^[24]: <https://arduino-info.wikispaces.com/Timers-Arduino>

6.4 Seznam použitých zkratk

AC	Střídavé napětí
AD	analogově/digitální převodník
ADC	analogově/digitální převodník
DC	stejnoseměrné napětí
DIY	podomácku vyrobené (do it yourself)
DPH	Daň z přidané hodnoty
GND	záporný pól stejnosměrného napětí (ground)
HP	Hewlett Packard
LCD	displej s tekutými krystaly (liquid crystal display)
LED	světlo emitující dioda (light emitting diode)
mA	miliampéry
MCU	mikrořadič (microcontroller unit)
PIC	mikrořadič od firmy Microchip
PWM	pulzně šířková modulace (Pulse Width Modulation)
Rpm	otáčky za minutu (rotations per minute)
SMD	povrchově montované součástky (surface mounted device)
SOČ	Středoškolská odborná činnost
THT	součástky s drátovými vývody (through hole technology)
VDC	volty stejnosměrného napětí
VDD	kladný pól stejnosměrného napájení

6.5 Seznam příloh elektronické verze SOČ

Příloha 1: Model krabičky pro finální model řídicí desky v programu SketchUp (*.skp)

Příloha 2: Vyexportovaný STL model krabičky pro tisk na 3D tiskárně (*.stl)

Příloha 3: Vyexportovaný STL model víka krabičky pro tisk na 3D tiskárně (*.stl)

Příloha 4: Program finálního modelu pro Arduino UNO (*.ino)

Příloha 5: Funkční schéma plošného spoje pro Arduino UNO v Eagle CAD (*.sch)

Příloha 6: Schéma desky plošného spoje pro Arduino UNO v Eagle CAD (*.brd)

