

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor SOČ: 10 Elektrotechnika, elektronika a telekomunikace

Robotická ruka

Autor: Lukáš Pizúr

Škola: Vyšší odborná škola s Střední průmyslová škola
elektrotechnická, Božetěchova 3, Olomouc

Kraj: Olomoucký kraj

Konzultant: Ing. Jaroslav Pospíšil

Olomouc 2014

Prohlášení

Prohlašuji, že jsem svou práci SOČ vypracoval(a) samostatně a použil(a) jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v seznamu vloženém v práci SOČ.

Prohlašuji, že tištěná verze a elektronická verze soutěžní práce SOČ jsou shodné.

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 o změně některých zákonů (autorský zákon) v platném znění.

V Olomouci dne 28. 3. 2014

podpis:

Poděkování

Tímto bych chtěl poděkovat Ing. Jaroslavovi Pospíšilovi za obětavou pomoc a podnětné připomínky, které mi během práce poskytoval. Dále bych chtěl poděkovat svému otci Karlovi Pizúrovi za značnou pomoc při výrobě mechanické konstrukce.

Anotace

Tato práce pojednává o návrhu a vytvoření robotického ramene. Robotické rameno je možné ovládat pomocí programu pro PC nebo programu pro systém Android. Zároveň je tu možnost automatického provádění pohybů uložených v externí paměti EEPROM, a to bez přítomnosti jiného externího zařízení. Programy pro PC a Android byly naprogramovány v jazyce Java. Programy pro řídicí jednotku a jednotku generování signálů pro serva byly naprogramovány v jazyce Assembler.

Klíčová slova: robotické rameno; robotická ruka; programování; assembler; java

Obsah

Úvod	8
1. Mechanika a použité serva	9
1.1 Mechanická konstrukce	9
1.2 Použité serva a princip ovládání.....	10
1.2.1 Princip ovládání serv.....	10
2. Hardware	11
2.1 Použité mikrokontoléry	12
2.1.1 Zapojení mikrokontroléru AT89CX051	12
2.2 Komunikace.....	13
2.2.1 Komunikace mezi jednotlivými bloky	13
2.2.2 Sériové rozhraní UART.....	14
2.2.3 Sériová sběrnice I2C.....	14
2.2.4 Bluetooth	15
2.2.5 Komunikační rozhraní RS232	16
2.3 Řídicí jednotka pro serva	17
2.3.1 Generování signálu pro serva	18
2.4 Hlavní řídicí jednotka	20
2.4.1 Display s řadičem S6A0069 (kompatibilní s HD44780).....	21
2.4.2 Paměť EEPROM	22
2.4.3 Automatické provádění programu	23
2.5 Napájecí zdroj.....	24
3. Software pro operační systém	25
3.1 Ovládací program pro PC	25
3.1.1 Ovládání ramene.....	26
3.1.2 Programování paměti EEPROM	27
3.2 Ovládací program pro operační systém android.....	28
4. Programování pohybů.....	29
4.1 Popis programování	29
4.2 Ukázky naprogramovaných pohybů.....	30
Závěr	31
Seznam použitych zdrojů	32
Příloha.....	33

Seznam zkratek

LCD	- Liquid Crystal Display
TxD	- Transmit Data
RxD	- Receive Data
EEPROM	- Electrically Erasable Programmable Read-Only Memory
UART	- Universal Asynchronous Receiver-Transmitter
SCON	- Serial Control
I2C	- Inter-Integrated Circuit
SDA	- Synchronous Data
SCL	- Synchronous Clock

Seznam tabulek

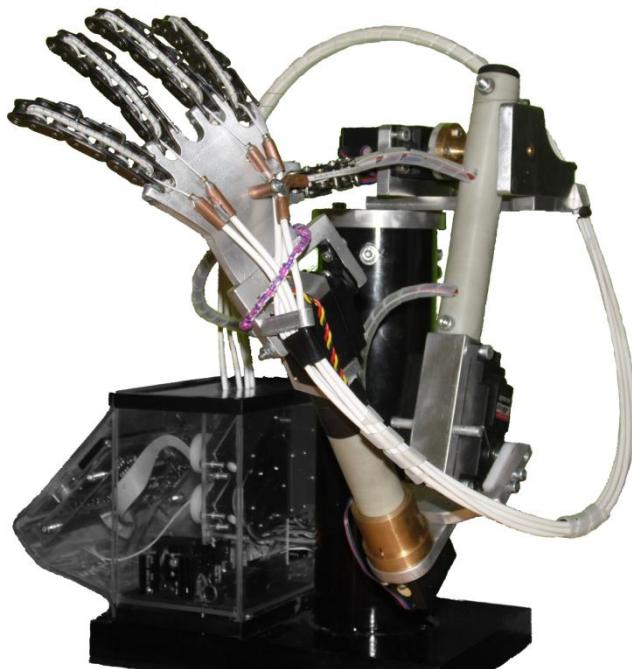
Tabulka 1: Použitá serva	10
Tabulka 2: Seznam příkazů pro ovládání řídicí jednotky serv.....	17
Tabulka 3: Seznam adres příslušných serv.....	18
Tabulka 4: Seznam příkazů pro ovládání hlavní jednotky	21
Tabulka 5: Klávesnicové zkratek v PC programu.....	26

Seznam obrázků

Obrázek 1: Sestavená robotická ruka	8
Obrázek 2: Závislost natočení serva na šířce impulzu	10
Obrázek 3: Hlavní řídicí deska	11
Obrázek 4: Deska spínaného zdroje	11
Obrázek 5: Pouzdro mikrokontroléru AT89C2051	12
Obrázek 6: Základní zapojení mikrokontroléru AT89C2051	12
Obrázek 7: Komunikace mezi jednotlivými bloky	13
Obrázek 8: Propojení EEPROM paměti a hlavní jednotky	14
Obrázek 9: Bluetooth modul	15
Obrázek 10: Převod 3,3 V na TTL	15
Obrázek 11: Převodník RS232 na TTL	16
Obrázek 12: Algoritmus jednotky generování signálu pro serva	19
Obrázek 13: Znázornění uživatelského menu v hlavní jednotce	20
Obrázek 14: Ovládací tlačítka hlavní jednotky	20
Obrázek 15: Zapojení displeje ve 4 bitovém módu	21
Obrázek 17: Pouzdro paměti 24LC16B	22
Obrázek 16: Připojení paměti EEPROM k mikrokontroléru	22
Obrázek 18: Algoritmus provádění programu	23
Obrázek 19: Schéma spínaného zdroje pro modelářská serva	24
Obrázek 20: Spínáný napájecí zdroj pro řídicí obvody	24
Obrázek 21: Připojovací a ovládací okno PC aplikace	25
Obrázek 22: Prostředí pro programování paměti	27
Obrázek 23: Připojovací okno android aplikace	28
Obrázek 24: Ovládací okno android aplikace	28
Obrázek 25: Prostředí pro vytváření pohybů	29
Obrázek 26: Schéma kompletního spínaného zdroje	33
Obrázek 27: Kompletní schéma řídicí jednotky	34
Obrázek 28: Osazovací plán desky zdrojů	35
Obrázek 29: Osazovací plán řídicí desky	35
Obrázek 30: Výkres podstavce pro robotické rameno (měřítko 1:4)	36
Obrázek 31: Výkres robotického ramene (měřítko 1:2)	37

Úvod

Cílem této práce je návrh robotického ramene (robotické ruky) ovládaného pomocí programu pro PC. Dále i možnost robotické rameno naprogramovat, aby bylo schopno automaticky provádět úkon i bez připojeného PC. Komunikaci mezi zařízením a uživatelem zajišťuje LCD display a čtyři tlačítka.



Obrázek 1: Sestavená robotická ruka

Veškeré mechanické díly jsou vlastní (domácí) výroby. Elektronické části byly navrženy přímo pro tento projekt, nebyly použity žádné koupené ovladače serv. Ovládací software byl navržen tak, aby byl kompatibilní téměř se všemi operačními systémy. Návrh a vývoj všech součástí je detailně popsán v příslušných kapitolách.

1. Mechanika a použité serva

V této kapitole je popsána mechanická část robotického ramene. Základním materiélem při výrobě byl hliník (přesněji dural), některé díly jsou i z mosazi. Veškeré díly této konstrukce byly vyrobeny v domácí dílně. Dále jsou zde popsány použité pohony a princip ovládání.

1.1 Mechanická konstrukce

Konstrukci lze rozdělit na tři základní části:

- Podstavec – Je vyroben ze železa. Díky své velké hmotnosti zajišťuje stabilitu celé konstrukce.
- Pohyblivá část – Jedná se o soustavu serv, pospojovanou pomocí dalších dílů do tvaru, který se podobá lidské ruce.
- Krabice – Zde jsou umístěny spolu s elektronikou serva, které ovládají jednotlivé prsty. Je vytvořena z plexiskla.

Ruka se stejně jako lidská skládá z pěti prstů, tyto prsty jsou vytvořeny úpravou řetězu na kolo. Každý prst je opatřen gumičkou, která působí proti gravitaci a zajišťuje, že jsou v klidném stavu natažené. Všechny prsty jsou ovládány pomocí provázků uložených v bowdenech, konce provázků jsou přidělány k jednotlivým servům. Zavírání jednotlivých prstů ovládají příslušná serva, které namotávají provázek a působí tak proti gumičkám. S celou rukou lze pomocí serva hýbat i ve vertikálním směru. Všechny serva použitá zde jsou standardní velikosti.

Loketní kloub se skládá ze dvou serv. První (menší) servo zajišťuje otáčení ruky s prsty, zde není kladen důraz na velkou sílu serva. Druhé servo patří do kategorie „Maxi serva“, tohle servo má podstatně větší sílu, neboť hmotnost ruky je přes 0,4kg a délka páky okolo 20cm. Výrobce udává, že tah servo je 24,7kg/cm. K servu byly přidány po obou stranách pružiny, které působí proti gravitaci a tím bylo dosaženo menšího náporu na servo.

Ramenní kloub, tento kloub je podobný loketnímu kloubu. I zde jsou dvě serva jedno větší a druhé menší - otáčecí. Otáčecí mechanizmus je uvnitř podstavce. Osa otáčení je uložená mezi axálními ložisky, které zabraňují zlomení osy uvnitř serva. Servo je namáhané pouze ve směru otáčení. K velkému servu je předělána oska, na které je přiděláno rameno. Na tohle servo působí největší síla, i zde bylo zapotřebí přidělat pružiny.

1.2 Použité serva a princip ovládání

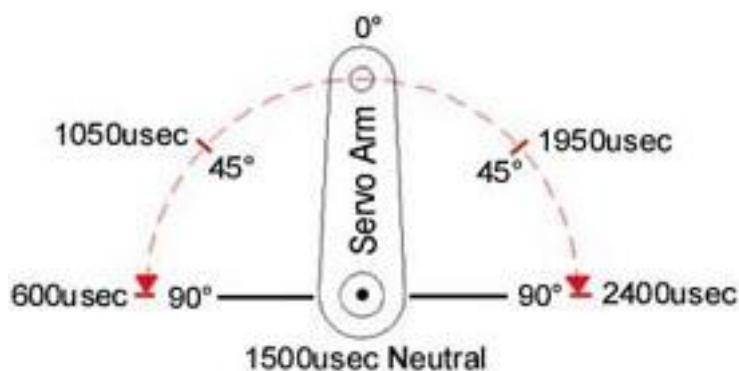
V celé konstrukci bylo použito celkem deset serv. Použité typy jednotlivých serv jsou napsány v tabulce č. 1.

Tabulka 1: Použitá serva

Umístění serva	Typ serva	Velikost	Poznámka
Ovládání prstů	HS-311	Standardní	Tah: 3,5kg/cm analogové servo
Pohyb ruky s prsty ve vertikálním směru	HS-645	Standardní	Tah: 9,6kg/cm digitální servo
Loketní kloub otáčení	HS-311	Standardní	Tah: 3,5kg/cm analogové servo
Loketní kloub zvedání	HS-805	Maxi	Tah: 24,7kg/cm analogové servo
Ramenní kloub otáčení	HS-311	Standardní	Tah: 3,5kg/cm analogové servo
Ramenní kloub zvedání	HS-805	Maxi	Tah: 24,7kg/cm analogové servo

1.2.1 Princip ovládání serv

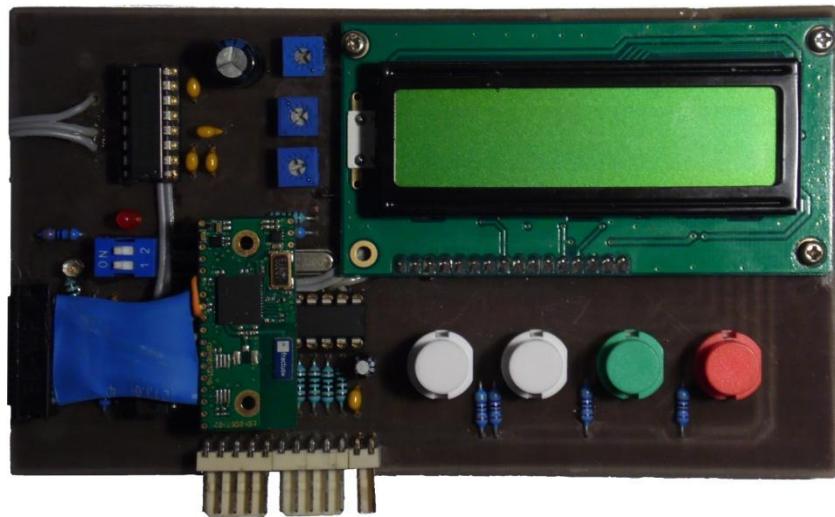
Natočení modelářských serv určuje šířka impulzu posílaného na datový vodič serva. Impulzy by měly být posílány v opakovací frekvenci 50Hz. Při zkracování impulzů se servo natáčí vlevo a při rozširování vpravo. Každé servo má jiné krajní hodnoty. Délka impulzů je v dnešní době v rozmezí 600μs až 2400μs. [1]



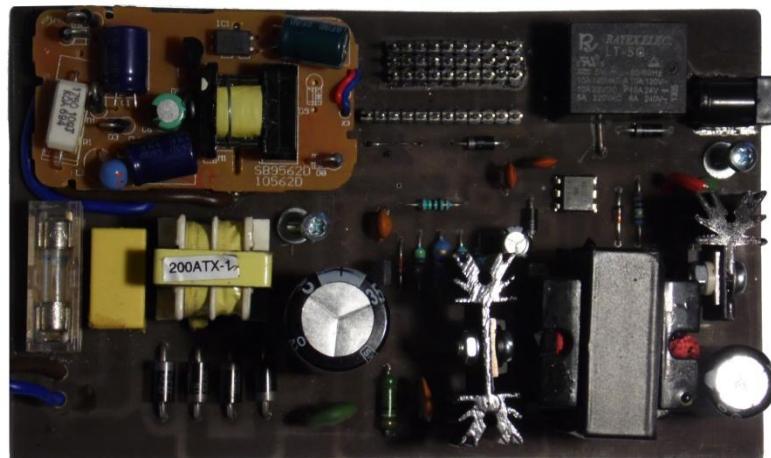
Obrázek 2: Závislost natočení serva na šířce impulzu

2. Hardware

V této kapitole jsou popsány použité mikrokontoléry, komunikace s PC, komunikace mezi jednotlivými bloky, řídicí jednotka pro serva, hlavní řídicí jednotka a nakonec spínaný napájecí zdroj. Příslušné bloky jsou popsány v jednotlivých kapitolách. Kompletní schémata, desky plošných spojů a osazovací výkresy jsou obsaženy v příloze. Poznámka: v těchto kapitolách budou udávány schémata bez konkrétních hodnot součástek, kompletní schémata jsou obsažena v příloze.



Obrázek 3: Hlavní řídicí deska

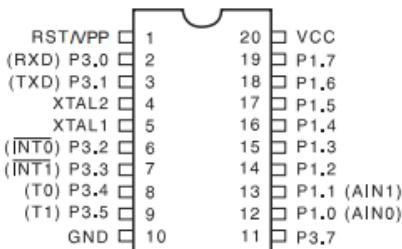


Obrázek 4: Deska spínaného zdroje

2.1 Použité mikrokontoléry

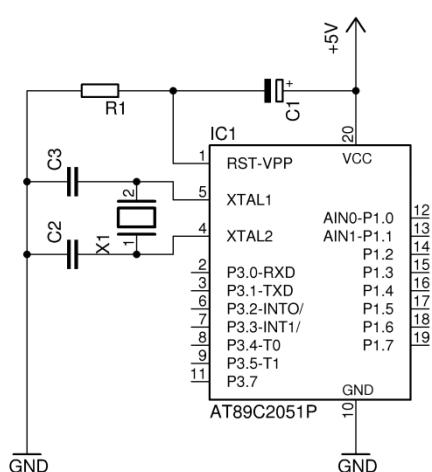
Jak pro hlavní řídicí jednotku, tak i pro obvod generování signálu byly použity mikrokontroléry s jádrem 8051 od firmy ATMEL. V obvodu hlavní řídicí jednotky je použit mikrokontrolér AT89C4051 a v obvodu generování signálu je použit AT89C2051. Oba tyto kontroléry mají stejné funkce, avšak mikrokontrolér AT89C4051 má větší paměť programu a to 4kB [2]. Taktovací frekvence pro oba mikrokontroléry je zajištěna externím krystalem o taktu 12MHz.

2.1.1 Zapojení mikrokontroléru AT89CX051



Obrázek 5: Pouzdro mikrokontroléru AT89C2051

AT89C2051 je 20 pinová verze mikrokontroléru s jádrem 8051 [3]. Jedná se o 8 bitový mikrokontrolér s interní pamětí RAM o velikosti 128bajtů a programovou pamětí typu Flash v základní verzi o velikosti 2kB. Napájecí napětí mikrokontroléru může být v rozmezí 2,7V – 6V. Obsahuje dva 16 bitové časovače/čítače a 15 programovatelných linek. Má šest zdrojů přerušení z toho dvě externí, které nebyly využity, jeden analogový komparátor. Dále obsahuje plně duplexní sériovou linku, což znamená, že může data zároveň přijímat i vysílat.



Aby mikrokontrolér začal provádět program od začátku, je nutné ho po přivedení napájecího napětí resetovat tzn. vynulovat programový čítač. K tomu slouží zapojení s rezistorem R1 a kondenzátorem C1, které musí udržet logickou úroveň H alespoň po dobu 2SC.

Kondenzátory C2 a C3 spolu s krystalem X1 jsou připojeny na vnitřní generátor hodin (XTAL1 a XTAL2). Tímto je zajištěna taktovací frekvence mikrokontroléru.

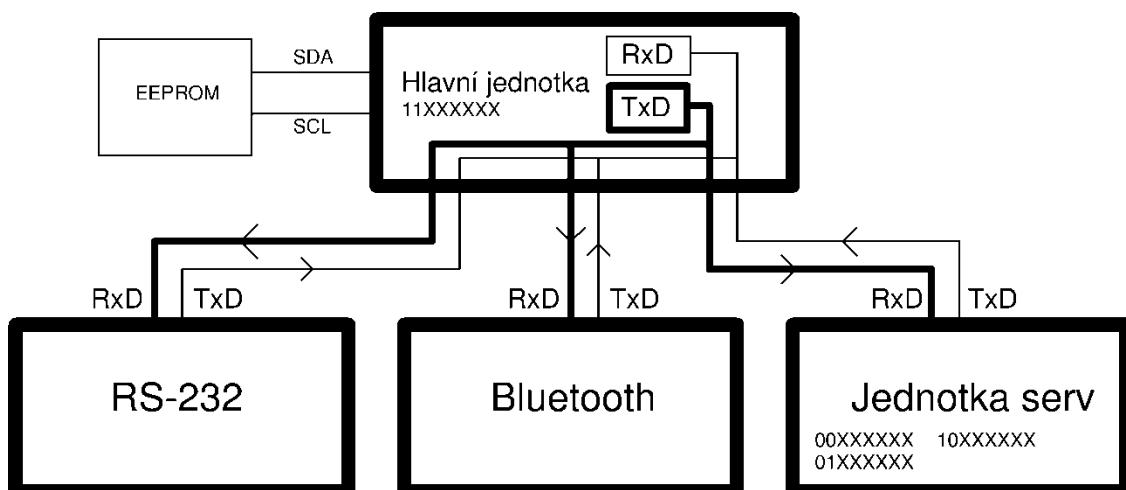
Obrázek 6: Základní zapojení mikrokontroléru AT89C2051

2.2 Komunikace

Komunikaci mezi jednotlivými bloky zajišťuje plně duplexní sériový kanál. Všechny bloky jsou propojeny pomocí dvou vodičů TxD a RxD. Komunikace mezi externí pamětí EEPROM a hlavní řídicí jednotkou využívá sběrnici I2C. Komunikace mezi zařízením a okolím (např.: PC) je zprostředkovávána pomocí Bluetooth modulu nebo sériového portu RS-232.

2.2.1 Komunikace mezi jednotlivými bloky

Všechny bloky mimo paměť EEPROM jsou navzájem propojené sériovou linkou UART, avšak veškerá komunikace chodí přes hlavní řídicí jednotku (mikrokontrolér AT89C4051). Komunikace probíhá dvou bajtové, to znamená, že jsou vysílány dva bajty za sebou. První dva bity v prvním bajtu určují komu je informace určena, ale zároveň i informaci k čemu jsou data určené. Pokud jsou první dva bity 11, data jsou určeny pouze pro hlavní jednotku, ale pokud jsou jiné, hlavní jednotka je znova odešle a data přijme nejen jednotka serv, ale i ostatní bloky. Závěrem lze říci, že hardwarově lze označit architekturu této komunikace jako STAR (komunikace probíhá přes uzel), ale softwarově jako RING (data přijímají všechny bloky – pouze pro data, které nemají v prvním bajtu na začátku 11).



Obrázek 7: Komunikace mezi jednotlivými bloky

2.2.2 Sériové rozhraní UART

Sériové rozhraní UART je součástí mikrokontroléru AT89C2051. Jak již bylo dříve zmíněno, jedná se o plně duplexní kanál. Tento kanál lze nastavit to dvou režimů a to synchronní a asynchronní - asynchronní se dále dělí na 8 bitový a 9 bitový. K tomuto nastavení slouží registr modu a řízení SCON.

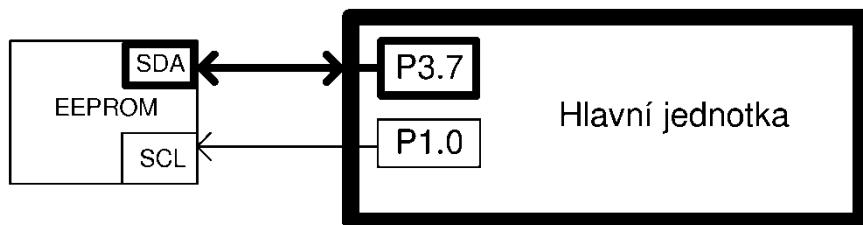
Pro komunikaci mezi jednotlivými bloky byl vybrán 8 bitový asynchronní mód [4]. Tento mód se vyznačuje tím, že má programovatelnou rychlosť a vysílá se 10 bitů (start bit, datová bity a stop bit). Registr SMOD slouží k zdvojnásobení rychlosti, dále je nutné nastavit časovač/čítač 1, který určuje přenosovou rychlosť. Rychlosť, kterou jednotlivé bloky komunikují, je 62500baud.

$$v_p = \frac{2^{SMOD}}{32} \cdot \frac{f_{osc}}{12 \cdot (256 - TH1)} = \frac{2^1}{32} \cdot \frac{12 \cdot 10^6}{12 \cdot (256 - 255)} = 62500 \text{ bit/s}$$

kde V_p - přenosová rychlosť [bit/s]
SMOD - zdvojnásobení rychlosť
 f_{osc} - frekvence oscilátoru [Hz]
TH1 - registr časovače/čítače 1

2.2.3 Sériová sběrnice I2C

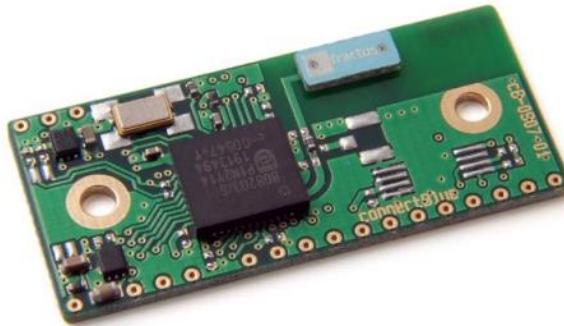
Sběrnice I2C [5] je použita pouze pro komunikaci paměti a hlavní řídicí jednotky. Mikrokontroler AT89C2051 tuto sběrnici neobsahuje, avšak je ji možné softwarově nahradit. Sběrnici tvoří dva vodiče a to vodič SDA, který přenáší sériové data a SCL přenos hodinového signálu.



Obrázek 8: Propojení EEPROM paměti a hlavní jednotky

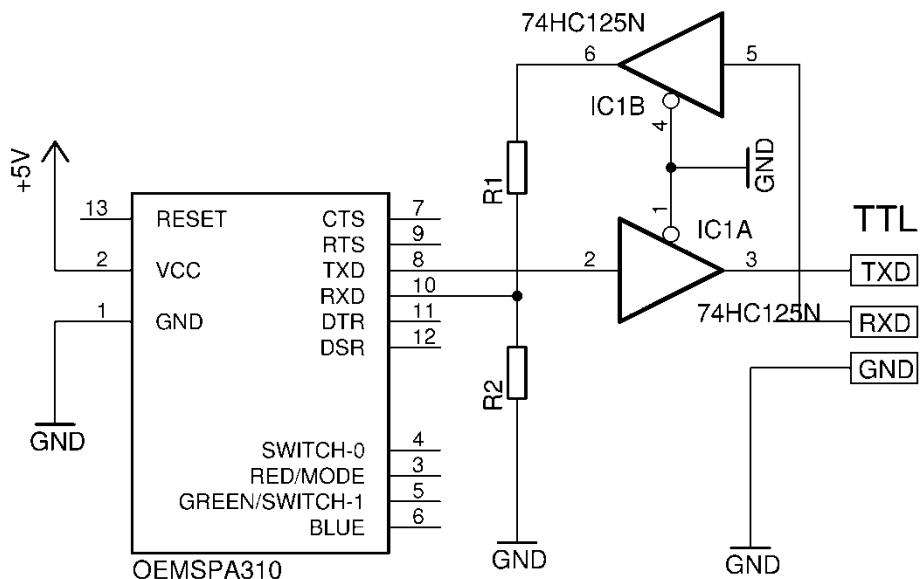
2.2.4 Bluetooth

Jedním ze způsobů komunikace mezi tímto zařízením a okolním (například PC), může být bluetooth komunikace. Jedná se o bezdrátovou komunikaci, kterou zajišťuje bluetooth modul OEMSPA310 od firmy connectBlue.



Obrázek 9: Bluetooth modul

Modul bluetooth [6] je 2. Třídy o výkonu 2mW, přibližný dosah by měl být okolo 20 metrů. Napájecí napětí modulu je 3V – 6V. Logické úrovně tohoto modulu jsou CMOS, což znamená, že logická jednička je 3,3V. Převod na TTL logiku zajišťuje obvod 74HC125. Zapojení je převzato ze zdroje [7].

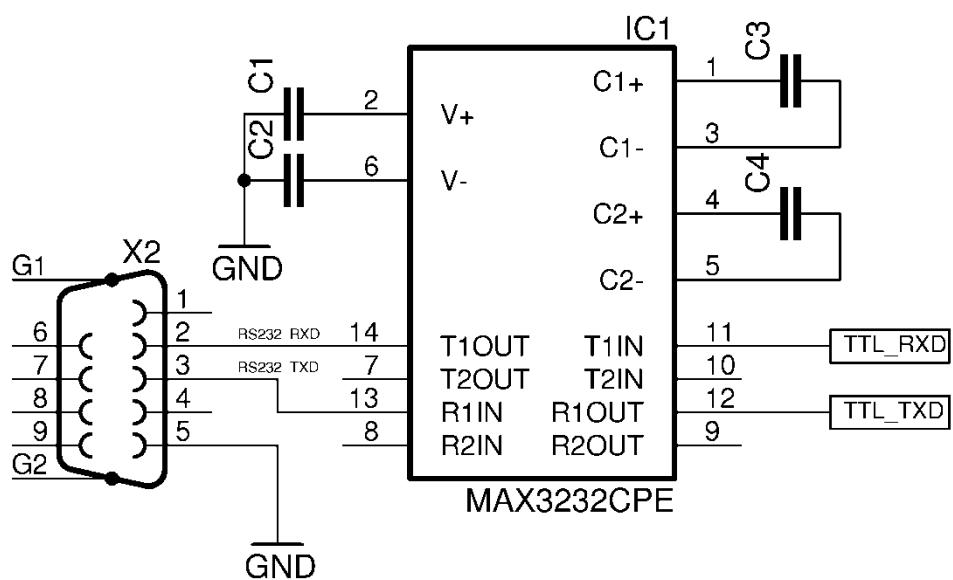


Obrázek 10: Převod 3,3V na TTL

Bluetooth je propojen s řídicí deskou pomocí sériové linky, která disponuje rychlostí 62500baud.

2.2.5 Komunikační rozhraní RS232

Další možností připojení je pomocí sériového portu RS-232. Ovšem nejprve je nutné zajistit kompatibilitu logických úrovní. Logické úrovně RS-232 jsou: logická nula je -3V až -15V a logická jednička je +3V až +15V. Mikrokontrolér, který zpracovává příchozí signál, pracuje v TTL logice, proto je potřeba použít převodník. [8]



Obrázek 11: Převodník RS232 na TTL

2.3 Řídicí jednotka pro serva

Úkolem této jednotky je generovat signály pro jednotlivá serva. Tato jednotka přijímá povely od hlavní jednotky. Základem této jednotky je mikrokontrolér AT89C2051. Jednotka je schopná pracovat samostatně i bez hlavní řídicí jednotky. Osm výstupních signálů je posíláno na port P1 a zbylé dva na port P3.7 a P3.5.

Tabulka 2: Seznam příkazů pro ovládání řídicí jednotky serv

Příkaz	1. bajt	2. bajt	Poznámka
Nastavení natočení serv	00XX XXXXB	Nastavení natočení serv (od 1 do 255)	Nastavení natočení serv. Serva určují bity X, kombinace bitů X je v následující tabulce.
Nastavení rychlosti serv	01XX XXXXB	Rychlosť serva (od 0 do 255)	Nastavení rychlosti serv.
Zapnutí všech serv	1000 0001B	0011 0011B	Zapne generování signálu pro všechny serva. Druhý bajt je potvrzovací.
Vypnutí všech serv	1000 0010B	0011 0011B	Vypne generování signálů pro všechny serva. Druhý bajt je potvrzovací.
Obnovení do výchozí polohy	1000 0011B	0011 0011B	Obnovení natočení serv do výchozí polohy. Druhý bajt je potvrzovací.
Získání aktuální polohy serva	1000 0100B	00XX XXXXB	Získání aktuální polohy serva. Bity X můžou adresovat pouze jedno servo. Hodnota je poslána v jednom bajtu.
Získání aktuální rychlosti serva	1000 0101B	00XX XXXXB	Získání aktuální rychlosti serva. Bity X můžou adresovat pouze jedno servo. Hodnota je poslána v jednom bajtu.

Následující tabulka popisuje adresování jednotlivých serv (bit y₅ v Tabulce 1). Bit y₅ určuje o jakou skupinu serv se jedná (0 – serva prstů, 1 – serva konstrukce). Bity y₀ až y₄ určují jednotlivá serva – najednou lze ovládat až 5 serv.

Tabulka 3: Seznam adres příslušných serv

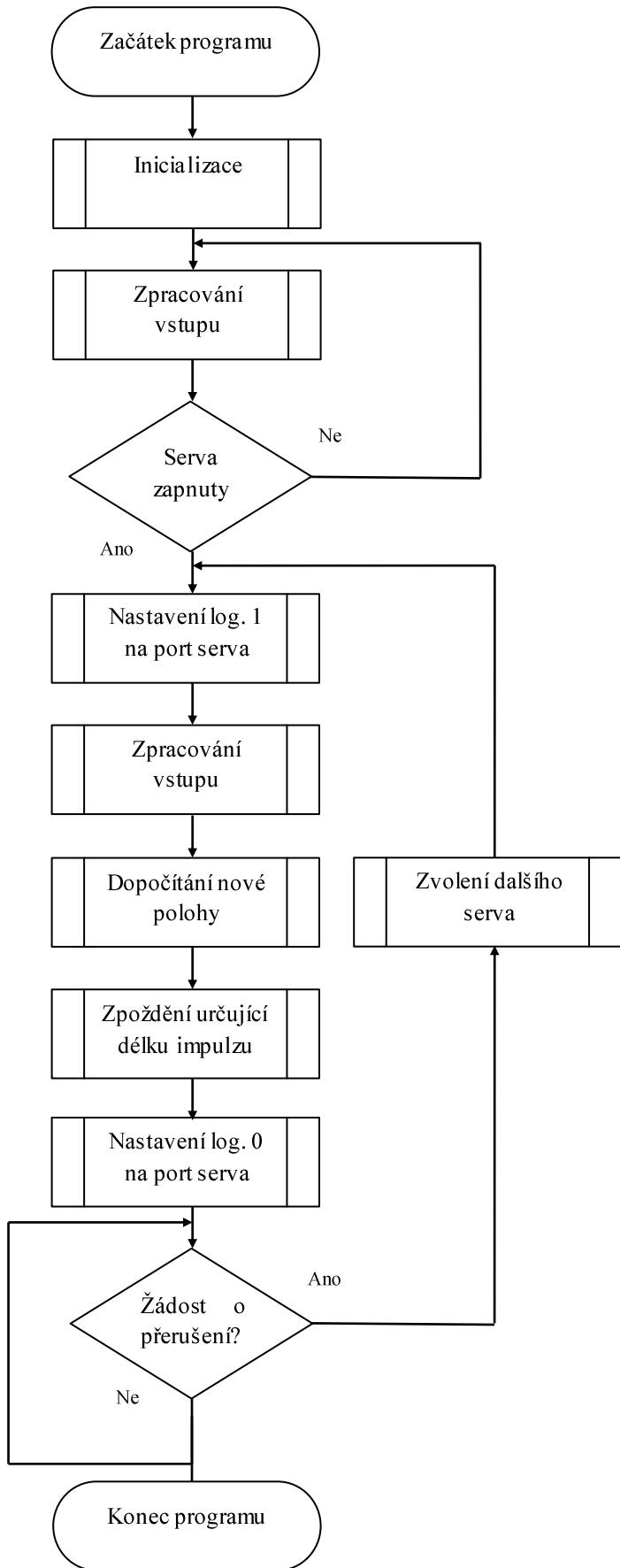
Název serva	Adresa serva	Kombinace pro více serv
Ramenní otáčecí	10 1000B	Adresa s posledním bitem 1 označuje serva konstrukce. Tyto adresy lze skládat dohromady: například 11 1111B adresa všech pěti serv, 10 1010B adresa ramenních serv.
Ramenní zvedací	10 0010B	
Loketní otáčecí	11 0000B	
Loketní zvedací	10 0001B	
Zápěstní zvedací	10 0100B	
Malíček	00 0001B	Adresa s posledním bitem 0 označuje serva prstů. Tyto adresy lze skládat dohromady jako v předchozím případě.
Prsteníček	00 0010B	
Prostředníček	00 0100B	
Ukazováček	00 1000B	
Palec	01 0000B	

2.3.1 Generování signálu pro servo

Generování signálů začíná po přijetí příkazu „Zapnutí všech serv“. Délka jednotlivých impulzů je v rozmezí 513µs až 2,563ms. Natočení serva je udáváno na osmi bitech, z čehož plyne, že změna o jeden LSB změní generovaný signál o 8µs. Jednotka ovládá deset serv a každý signál může mít maximální délku impulzu 2,563ms, tudíž opakovací frekvence je 39Hz. O přepínání serv se stará časovač 0, který přeteče každých 2,563ms a po přetečení je obsluhováno následující servo.

Každé servo má v paměti kontroléru uloženy tři parametry: požadovaná poloha, aktuální poloha a rychlosť pohybu. Požadovaná poloha je nastavována příkazy z hlavní řídicí jednotky. Aktuální polohu si mikrokontrolér dopočítává sám a to na základě předchozí polohy a rychlosti pohybu. Generovaný signál se generuje na základě aktuální polohy.

Rychlosť pohybu je číselná hodnota uložená v paměti. Pokud je tato hodnota menší jak 128, je jednou za 25,64ms (opakovací frekvence 39Hz) přičtena/odečtena od aktuální polohy. Tohle je opakováno, dokud se aktuální poloha nerovná požadované poloze. Pokud je hodnota větší než 128 je přičtena/odečtena jednička od aktuální polohy a to jednou za 1/(hodnota-128) * 25,65ms.



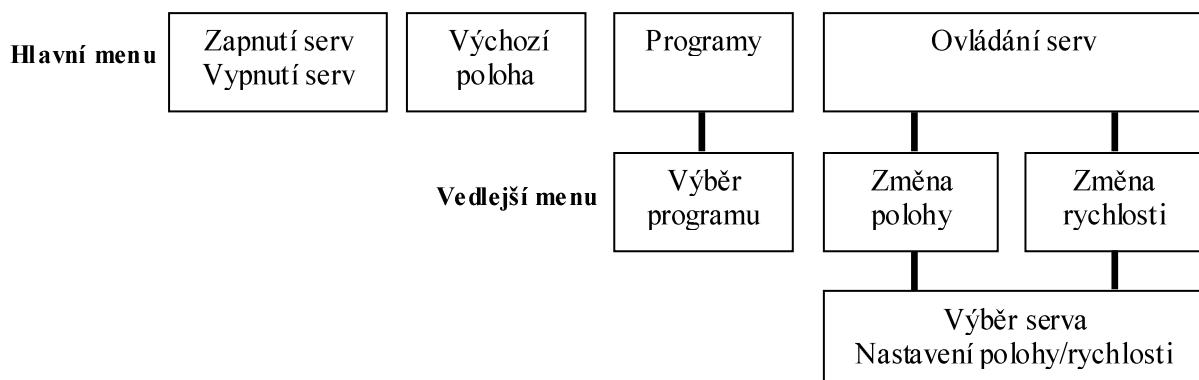
Obrázek 12: Algoritmus jednotky generování signálu pro serva

2.4 Hlavní řídící jednotka

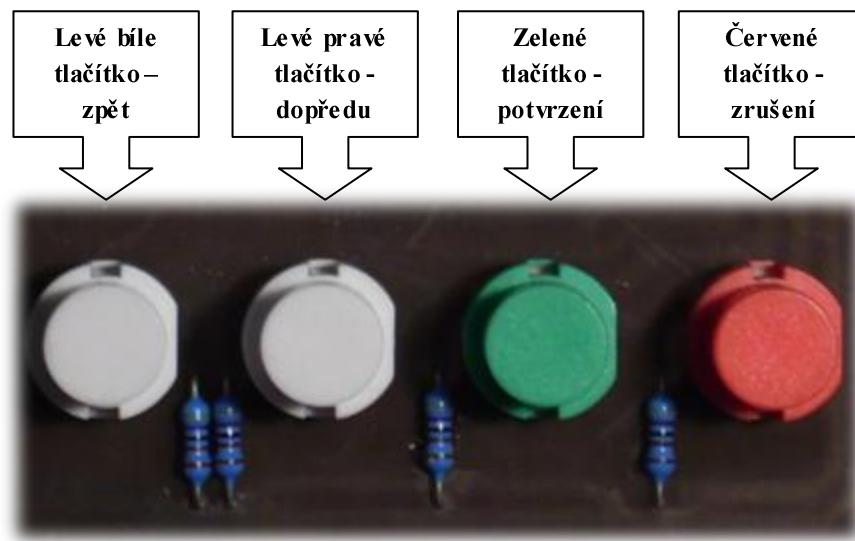
Tato řídící jednotka zajišťuje komunikaci mezi bluetooth/sériovým portem a řídící jednotkou serv. Zároveň také zprostředkovává komunikaci mezi uživatelem pomocí dvourádkového displeje.

Jako komunikační prostředí mezi uživatelem a hlavní jednotkou bylo vytvořeno čtyři prvkové menu, které obsahuje základní funkce. Prvním prvkem je zapnutí nebo vypnutí serv (generování signálu), dalším je obnovení do výchozí polohy. Tyto dvě funkce neobsahují podmenu. Třetím prvkem hlavního menu jsou programy, tato funkce umožňuje zvolit si program, který bude ruka provádět samostatně. Posledním prvkem je ovládání serv, pomocí něhož lze ovládat natočení (polohu) nebo rychlosť jednotlivých serv. Menu je znázorněno na obrázku č. 11. V menu se uživatel pohybuje pomocí čtyř tlačítek, tyto tlačítka jsou popsána na obrázku č. 12.

Obrázek 13: Znázornění uživatelského menu v hlavní jednotce



Obrázek 14: Ovládací tlačítka hlavní jednotky



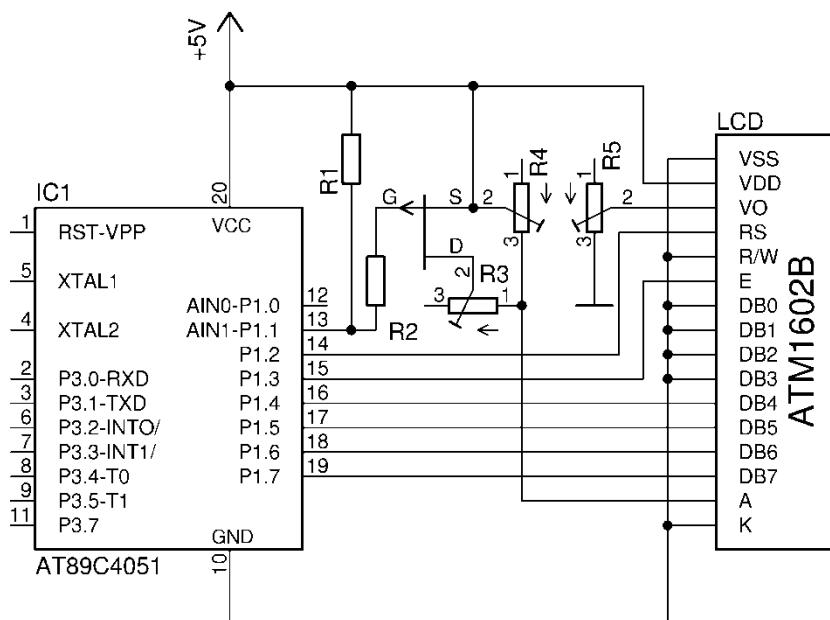
Stejně jako řídicí jednotka serv, má i hlavní jednotka své ovládací příkazy, které jsou určené pouze pro ni. Tyto příkazy jsou popsány v tabulce č. 3.

Tabulka 4: Seznam příkazů pro ovládání hlavní jednotky

Příkaz	1. bajt	2. bajt	Poznámka
Programování paměti EEPROM	1100 0001B	1100 1100B	Přepne hlavní jednotku do režimu programování paměti. Následně se posílají data po jednom bajtu. Programování začíná od prvního bajtu paměti.
Čtení obsahu paměti EEPROM	1100 0001B	1100 0011B	Po odeslání tohoto povolení, hlavní jednotka posílá zpět data z paměti EEPROM a to od prvního bajtu v paměti.

2.4.1 Display s řadičem S6A0069 (kompatibilní s HD44780)

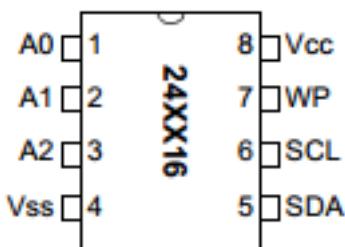
Jako hlavní zobrazovač byl použit dvou řádkový, šestnácti znakový LCD displej MC1602E-SYL/H s řadičem S6A0069 [9]. Display komunikuje po paralelní sběrnici, a to v osmibitovém nebo čtyřbitovém režimu. Z důvodu nedostatku volných pinů mikrokontroléru, byl použit právě čtyřbitový režim. V tomto režimu se první posílají horní čtyři bity slova a poté dolní čtyři bity.



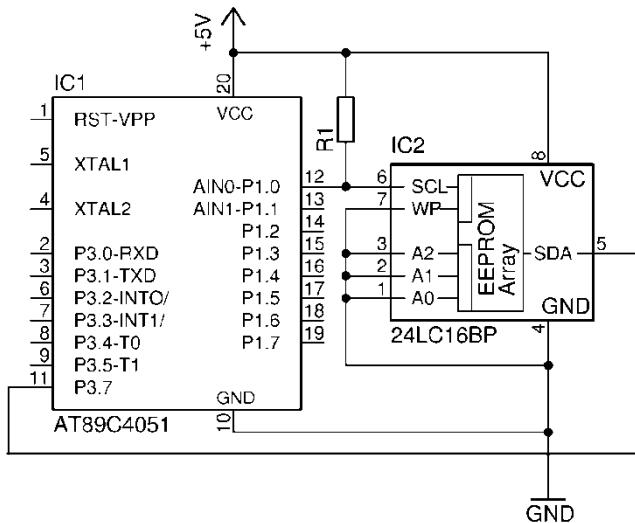
Obrázek 15: Zapojení displeje ve 4 bitovém módu

2.4.2 Paměť EEPROM

Jednotlivé programy (automatické pohyby) jsou uloženy v externí paměti EEPROM. Tuto paměť lze programovat pomocí bluetooth. Paměť je realizována obvodem 24LC16B, který komunikuje pomocí sériové sběrnice I2C [10]. Mikrokontrolér AT89C2051 tuto sběrnicu hardwarově neobsahuje a proto byla řešena softwarově [11]. Jedná se o 16Kbit paměť, kterou je možné programovat přes bluetooth pomocí hlavní jednotky a PC. Paměť se programuje pomocí PC programu, kde je možné jak z paměti číst, tak i nahrávat.



Obrázek 17: Pouzdro paměti 24LC16B

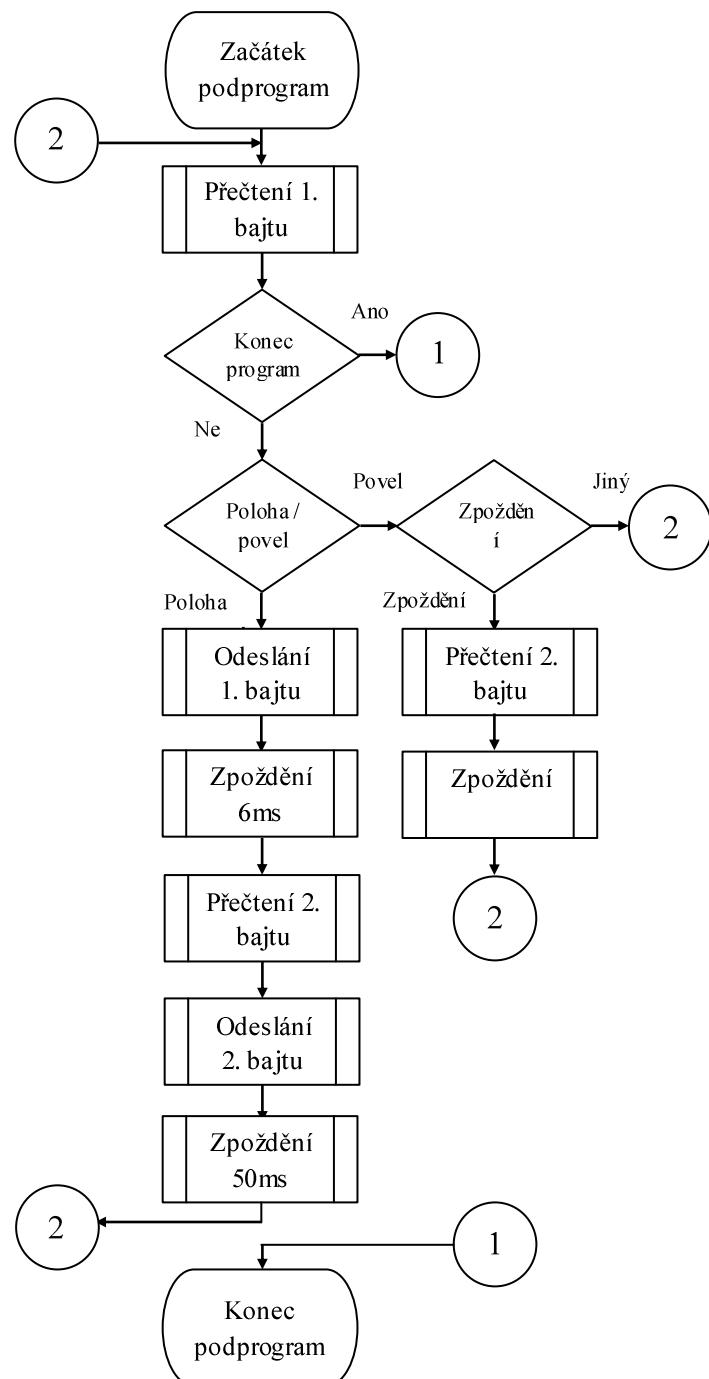


Obrázek 16: Připojení paměti EEPROM k mikrokontroléru

Piny A0, A1, A3 jsou adresové, avšak v tomhle typu paměti (24XX16) nejsou zapojené. Pin WP je ochrana proti zápisu do paměti, pokud je uzemněn je možné zapisovat. Piny SCL a SDA jsou piny sběrnice I2C. Napájecí napětí této paměti je 5V. Komunikační rutina, která simuluje sběrnicu I2C byla použita přímo od výrobce. Do této paměti by se teoreticky mělo vejít až 1000 poloh.

2.4.3 Automatické provádění programu

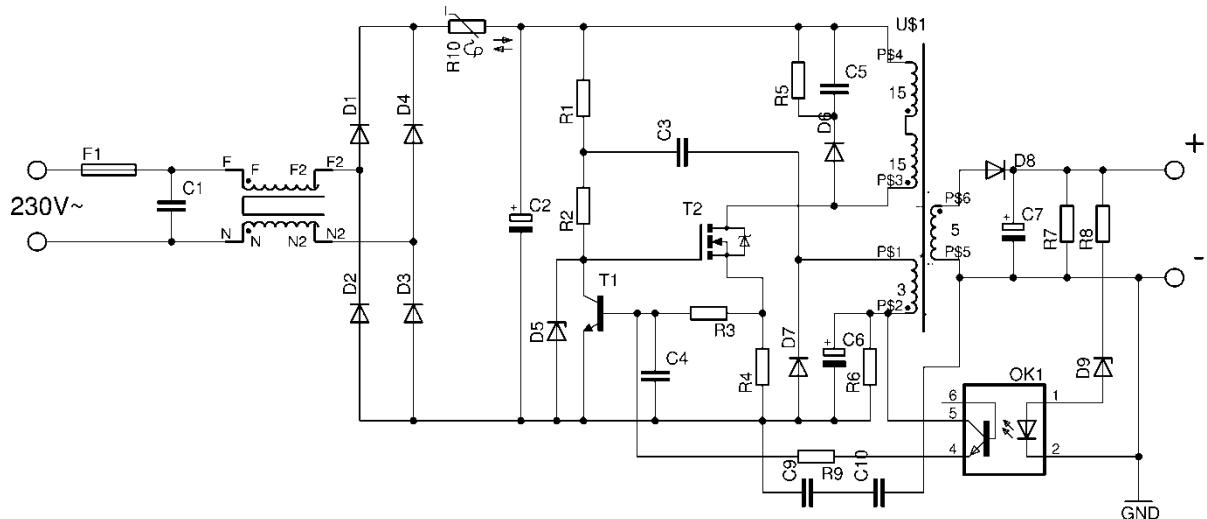
V hlavním menu pod položkou programy lze vybrat program, který se bude automaticky provádět. Programy se načítají z paměti EEPROM. K této volbě není zapotřebí žádné externí zařízení, ani počítač. Vše provádí pouze hlavní řídicí jednotka. Po výběru programu, který se má provádět, se postupně načítají jednotlivé polohy z paměti. Kromě poloh lze přaprogramovat i rychlosť serv a také zpoždění mezi jednotlivými polohami. Algoritmus provádění programu je znázorněn na obrázku č. 16. Programování pohybů je uvedeno v kapitole č. 5.



Obrázek 18: Algoritmus provádění programu

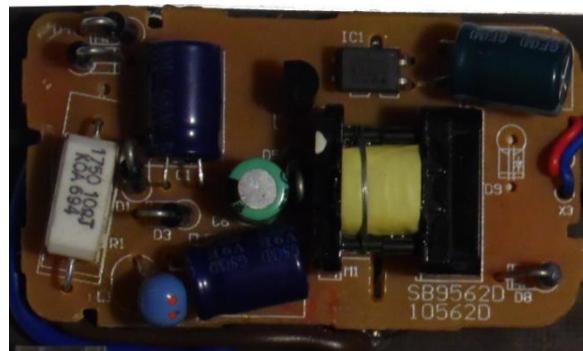
2.5 Napájecí zdroj

Modelářská serva lze napájet napětím v rozmezí 5 až 7V. Serva této konstrukce jsou napájeny napětím 6V, tohle napětí zajišťuje spínaný zdroj napětí. Proudové zatížení, které by měl tento zdroj trvale unést je okolo 5A. Zapojení bylo převzatou ze [12] a upraveno pro požadované napětí.



Obrázek 19: Schéma spínaného zdroje pro modelářská serva

Napájení řídicích obvodů zajišťuje druhý spínaný zdroj, který dodává napětí okolo 5V. Maximální proud, který je možné z tohoto zdroje odebírat je 0,8A. Tento zdroj pochází z nabíječky pro telefon.



Obrázek 20: Spínaný napájecí zdroj pro řídicí obvody

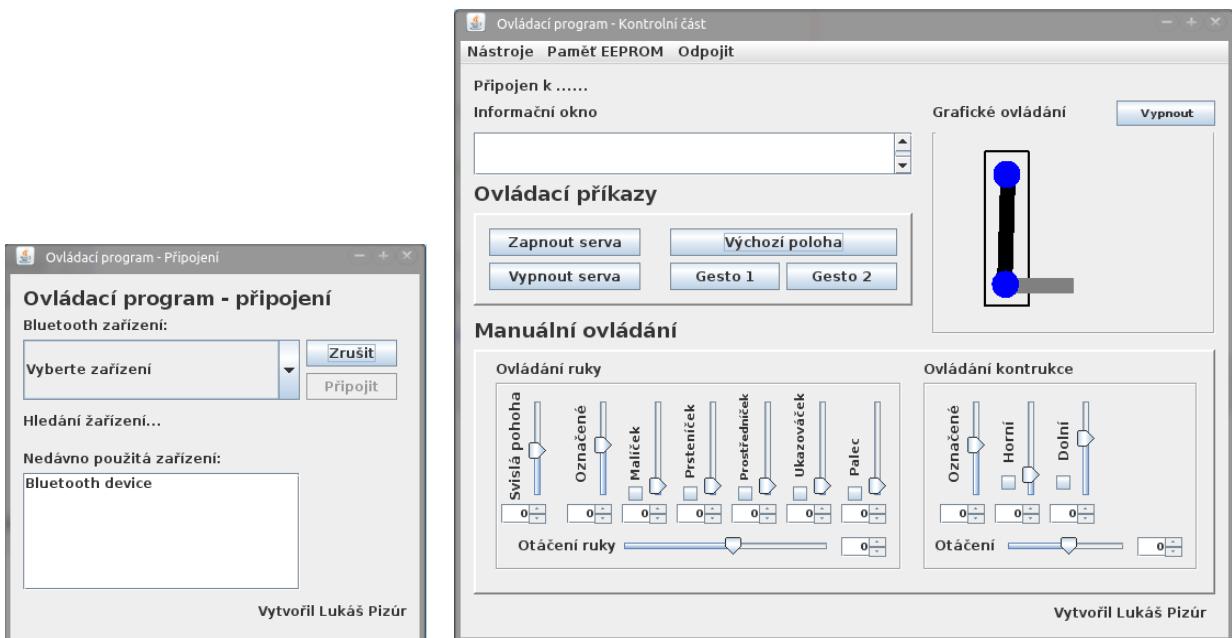
Oba zdroje byly umístěny na jednu desku. Kromě spínaných zdrojů lze zařízení napájet i z 6V akumulátoru, který je možné připojit pomocí konektoru na desce. Po připojení akumulátoru se za pomoci relé odpojí spínaný zdroj.

3. Software pro operační systém

Jak již bylo dříve řečeno, robotické rameno lze ovládat i pomocí externích zařízení. Byly naprogramovány dva základní programy: pro PC (odzkoušeno na systémech Linux a Windows) a pro systém Android. Oba tyto programy byly naprogramovány v jazyce Java [13] a jsou obsaženy včetně zdrojových kódů na přiloženém CD.

3.1 Ovládací program pro PC

Tímto programem lze ovládat veškeré funkce ramene. Oproti programu pro Android, lze upravovat a programovat pohyby nahrané v paměti EEPROM. Komunikace probíhá pomocí bluetooth a je zapotřebí mít bluetooth adaptér připojený k PC.



Obrázek 21: Připojovací a ovládací okno PC aplikace

3.1.1 Ovládání ramene

Rameno lze ovládat třemi způsoby: pomocí posuvníků, grafického rozhraní nebo klávesnice. Dále jsou zde ovládací tlačítka: zapnout serva, vypnou serva, výchozí poloha (uveďe serva do výchozí polohy), gesto 1 (provede nastavené gesto), gesto 2.

Ovládání pomocí posuvníků. Na obrázku č. 20 je znárodněné okno ovládacího rozhraní. V sekci manuální ovládání se nachází dvanáct posuvníků sloužící k ovládání serv. Každé servo má svůj vlastní posuvník, kterým se nastavuje natočení serva. Dále má jeden přepínačí seznam, pomocí něhož se nastavuje rychlosť serva.

Ovládání pomocí grafického režimu, tímto způsobem lze ovládat pouze dvě serva a to loketní a ramení zdvihačí. Nejprve se pomocí myši označí část, kterou chceme ovládat (lze vybrat i obě části). Poté sunutím myší lze měnit polohu části.

Ovládaní pomocí klávesnice, program dokáže snímat stisknuté klávesy a následně podle toho měnit natočení serv. Klávesnicové zkratky jsou obsaženy v tabulce č. 5.

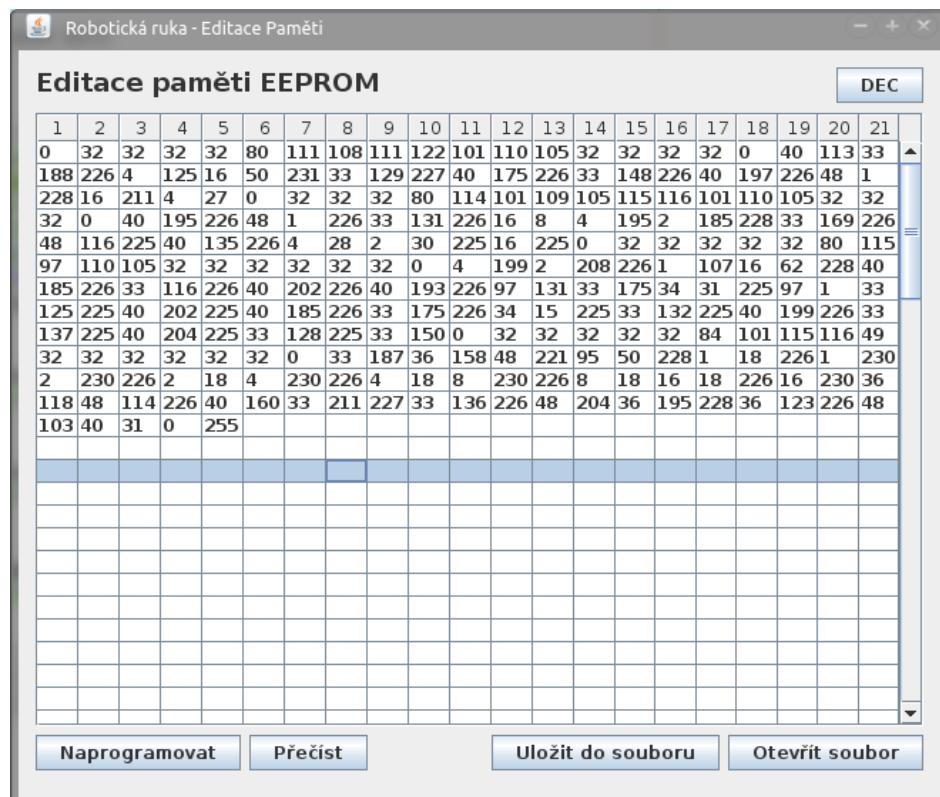
Tabulka 5: Klávesnicové zkratky v PC programu

Servo	Klávesa		Poznámka
	Natočení doleva nebo nahoru	Natočení doprava nebo dolu	
Ramenní otáčecí	J	L	doleva - doprava
Ramenní zvedací	I	K	nahoru - dolu
Loketní otáčecí	A	D	doleva - doprava
Loketní zvedací	W	S	nahoru - dolu
Zápěstní zvedací	Q	E	nahoru - dolu
Zrychlení	CTRL		Podržením klávesy CTRL a jiné klávesy se pohyb zrychluje.
Zpomalení	ALT		Podržením klávesy ALT a jiné klávesy se pohyb zpomaluje.
Malíček – Palec (Pokrčení)	C, V, B, N, M		Pokrčení prstu.
Malíček – Palec (Natažené)	SHIFT + C, V, B, N, M		Podržením klávesy SHIFT a klávesy příslušného prstu dochází k natažení.

3.1.2 Programování paměti EEPROM

Externí paměť EEPROM lze programovat pomocí programu pro PC a to bezdrátově přes bluetooth. K tomuto účelu bylo vytvořeno prostředí, kde je možné, jak paměť programovat, tak z ní i číst. Toto prostředí lze nalézt v záložce „Paměť EEPROM – Editace paměti“.

Okno se skládá z jedné tabulky a pěti tlačítek. Tlačítko (přepínač) DEC/HEX převádí data v tabulce do požadované soustavy, a to do hexadecimální nebo dekadické. Tlačítko „Přečíst“ přečte obsah paměti EEPROM a vloží ho do tabulky. Tlačítko „Naprogramovat“ naprogramuje obsah tabulky do paměti EEPROM. Data tabulky lze načítat ze souboru nebo je do něj ukládat.

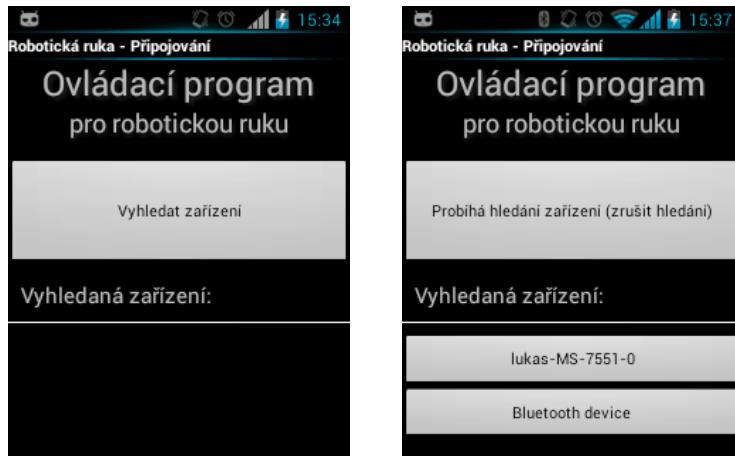


Obrázek 22: Prostředí pro programování paměti

3.2 Ovládací program pro operační systém android

Tímto programem lze ovládat veškeré pohyby serv, avšak není možné prostřednictvím tohoto programu programovat paměť EEPROM nebo nastavovat rychlosť serv. Program je naprogramován v jazyce Java [14].

První je zapotřebí připojit se k zařízení. Podobně jako u PC programu k tomu slouží první okno a tlačítko „Vyhledat zařízení“, po kliknutí na tohle tlačítko se začnou objevovat vyhledaná bluetooth zařízení. Následně pak stačí, kliknout na dané zařízení.



Obrázek 23: Připojovací okno android aplikace

Po připojení k zařízení se objeví ovládací okno, pomocí něhož lze robotické rameno ovládat. Ve spodní části lze vidět dva joysticky. Pomocí červeného joysticku se ovládá natočení loktu a zvedání loktu. Pravým (zeleným) joystickem se ovládá otáčení celého ramene a zvedání. Jednotlivé serva prstů lze ovládat pomocí pěti posuvátek umístěných vlevo nahore. Dalšími důležitými funkcemi jsou zapnutí/vypnutí serv a obnovení polohy. Aplikace byla testována na telefonu Sony Ericsson Xperia x10 mini a verze androidu byla 4.1.2 Jelly Bean.



Obrázek 24: Ovládací okno android aplikace

4. Programování pohybů

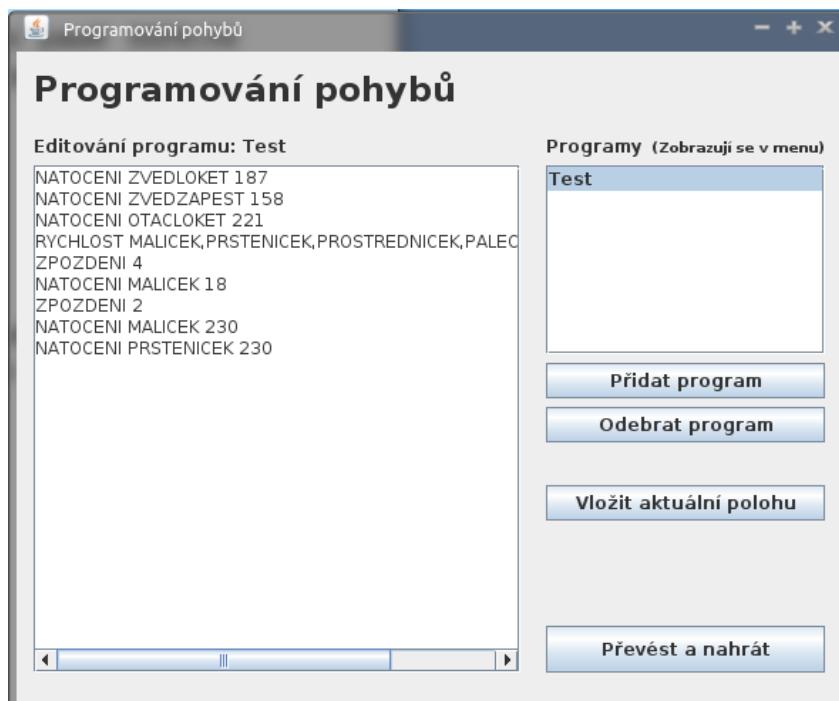
Jak již bylo dříve řečeno, lze robotickou ruku, přesněji externí paměť EEPROM programovat a později z ní nahrávat naprogramované pohyby do hlavní jednotky. Tyto pohyby lze později vybrat v menu hlavní jednotky.

4.1 Popis programování

Aby bylo programování z hlediska uživatele lehké, byl vytvořen jednoduchý nástroj, který umožnuje pohyby programovat v jazyce s podobnou syntaxí, jako má assembler. Lze ho nalézt v záložce „Nástroje – Vytvoření pohybů“.

Vytváření pohybu (programu) začíná stisknutím tlačítka „Přidat program“, po stisknutí se objeví okno, kde se zadává název programu (tentotéž název se zobrazuje na displeji v sekci „Programy“). Dále má každý program své vlastní kroky, které se zapisují do pole „Editování programu“.

Syntaxe psaní programu je následující. První se píše instrukce (NATOCENI, RYCHLOST, ZPOZDENI). U instrukcí NATOCENI a RYCHLOST je nutné uvést serva, o které se jedná. Poté je zadána poloha/rychlosť, u ZPOZDENI se přidá jen zpoždění v sekundách. Jednotlivé operandy je nutné oddělit mezerou. Jednotlivá serva se oddělují čárkou a to bez mezery. Princip je lépe viditelný na obrázku č. 24.



Obrázek 25: Prostředí pro vytváření pohybů

4.2 Ukázky naprogramovaných pohybů

Zde je přidána ukázka pohybu aktuálně naprogramovaného v paměti. Pohyb začíná zvednutím loktu, zápěstí a zároveň otáčením loktu. Následně je nastavena vyšší rychlosť serv ovládajúcich prsty. Poté sú prsty postupne pokrčovány. Ďalej sa ruka otočí a provede pári ďalších pohybów.

NATOCENI ZVEDLOKET 187
NATOCENI ZVEDZAPEST 158
NATOCENI OTA CLOKET 221
RYCHLOST MALICEK,PRSTENICEK,PROSTREDNICEK,PALEC,UKAZOVACEK 50
ZPOZDENI 4
NATOCENI MALICEK 18
ZPOZDENI 2
NATOCENI MALICEK 230
NATOCENI PRSTENICEK 230
ZPOZDENI 2
NATOCENI PRSTENICEK 18
NATOCENI PROSTREDNICEK 230
ZPOZDENI 2
NATOCENI PROSTREDNICEK 18
NATOCENI UKAZOVACEK 230
ZPOZDENI 2
NATOCENI UKAZOVACEK 18
NATOCENI PA LEC 18
ZPOZDENI 2
NATOCENI PA LEC 230
NATOCENI ZVEDZAPEST 118
NATOCENI OTA CLOKET 114
ZPOZDENI 2
NATOCENI OTA CRAMEN 160
NATOCENI ZVEDLOKET 211
ZPOZDENI 3
NATOCENI ZVEDLOKET 136
ZPOZDENI 2
NATOCENI OTA CLOKET 204
NATOCENI ZVEDZAPEST 195
ZPOZDENI 4
NATOCENI ZVEDZAPEST 123
ZPOZDENI 2
NATOCENI OTA CLOKET 103
NATOCENI OTA CRAMEN 31

Závěr

Od začátku bylo mým cílem sestrojit robotické rameno, které by bylo ovládané pomocí PC. Začal jsem s jednotkou generování signálů pro serva a následně přidal hlavní jednotku a to z důvodu provádění automatických pohybů. Poté jsem začal přemýšlet, kam bych mohl přeprogramované pohyby ukládat, což jsem vyřešil přidáním externí paměti EEPROM. Následně jsem naprogramoval aplikaci určenou pro systém Android. Tato aplikace byla naprogramována z důvodu lepšího (jednoduššího) ovládání konstrukce. Ovládání probíhá pomocí dotykového displeje, což je pro uživatele značně přívětivější.

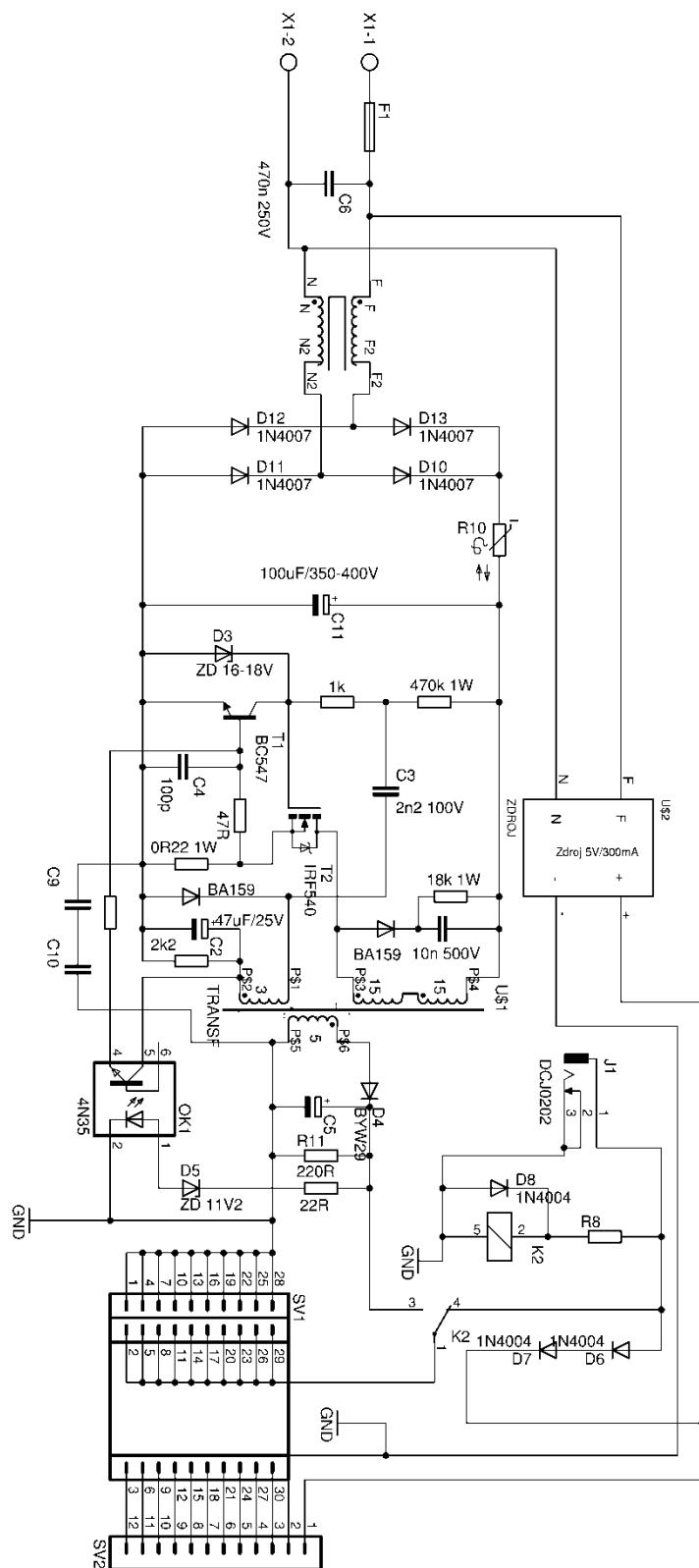
Na základě získaných znalostí při tvorbě této práce bych v budoucnu řešil tento problém z mnoha hledisek odlišněji. Například první problém, který bych řešil jinak, by byla jiná volba pohonů pro konstrukci. Na místo použitých servomotorů bych zvolil krokové motorky. Tyto motorky by se na rozdíl od servomotorů ovládaly poněkud složitěji, avšak nabídly by lepší výkon a následně i lepší stabilitu celé konstrukce. Další změna by byla ve volbě mikrokontroléra a to hlavně kvůli komunikaci. AT89C2051 neobsahuje sériovou sběrnici I2C. Tato sběrnice by značně usnadnila komunikaci mezi jednotlivými bloky. Pomocí sběrnice UART bylo velmi obtížné realizovat komunikaci mezi více než dvěma bloky. Další důraz, který by byl kladen na výběr mikrokontroléra, by bylo, aby obsahoval paměť EEPROM, která by nahradila externí paměť.

Další, co by stalo za zmínění je, že v budoucnu plánuji k tomuto robotickému rameni přidělat další zařízení, které už jsem z důvodu časové tísni nestihl dodělat. Prvním nápadem je „ovládací rukavice“, která by snímala polohu (pokrčení) prstů na lidské ruce a následně ji přivedla a poslala ji hlavní řídicí jednotce. Ta by následně změnila polohu prstů na ruce robotické. Další možností vylepšení je přidání snímačů, které by zjišťovali přítomnost předmětu anebo působení jednotlivých sil, at' už na serva nebo na uchopený předmět.

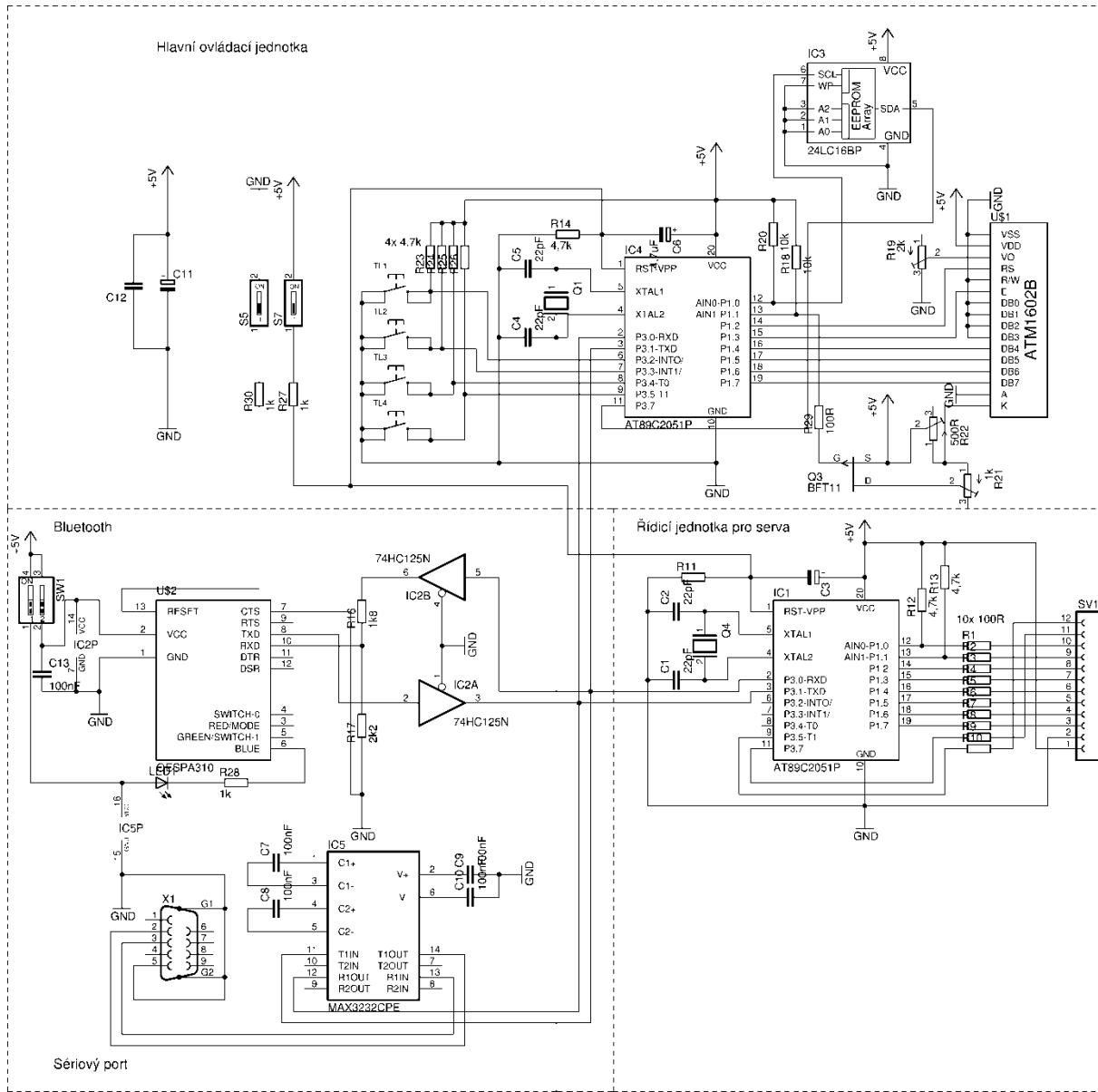
Seznam použitých zdrojů

- [1] Jiří Ježerský, Řízení serva – teorie [cit. 2014-03-23]
<http://www.serva.cz/rizeni-serva-teorie/> [online]
- [2] Datasheet k AT89C2040 [cit. 2014-03-23]
<http://www.gme.cz/img/cache/doc/432/200/at89c4051-24pu-datasheet-1.pdf> [online]
- [3] Datasheet k AT89C4040 [cit. 2014-03-23]
<http://www.gme.cz/img/cache/doc/432/199/at89c2051-24pu-datasheet-1.pdf> [online]
- [4] David Hankovec, DH servis – Sériový kanál [cit. 2014-03-23]
<http://www.dhservis.cz/serial.htm> [online]
- [5] Wikipedia, I2C [cit. 2014-03-23]
<http://cs.wikipedia.org/wiki/I%C2%BA%C2%BC> [online]
- [6] Datasheet k bluetooth OEMSPA-310
http://www.spezial.cz/pdf/em_ds_oemspa_310.pdf [online]
- [7] Matěj Kubíčka, Použití Bluetooth v robotice (I. díl)
<http://www.matejk.cz/zdroje/Dil-I-Bluetooth-a-robotika.pdf> [online]
- [8] Wikipedia, RS-232
<http://cs.wikipedia.org/wiki/RS-232> [online]
- [9] Datasheet S6A0069
http://www.topwaydisplay.com/Pub/IC_DataSheet/S6A0069.pdf [online]
- [10] Datasheet k 24LC16B
<http://www.gme.cz/img/cache/doc/414/047/24lc16b-i-p-datasheet-1.pdf> [online]
- [11] David Hankovec, Připojování sériové paměti EEPROM [cit. 2014-03-23]
<http://www.dhservis.cz/eeprom.htm> [online]
- [12] Danyk.cz, Jednoduchý impulzní zdroj 12V 5A
<http://danyk.cz/impulz3.html> [online]
- [13] Wikibooks, Java Programming
http://en.wikibooks.org/wiki/Java_Programming [online]
- [15] Android Developers – API Guides
<http://developer.android.com/guide/index.html> [online]

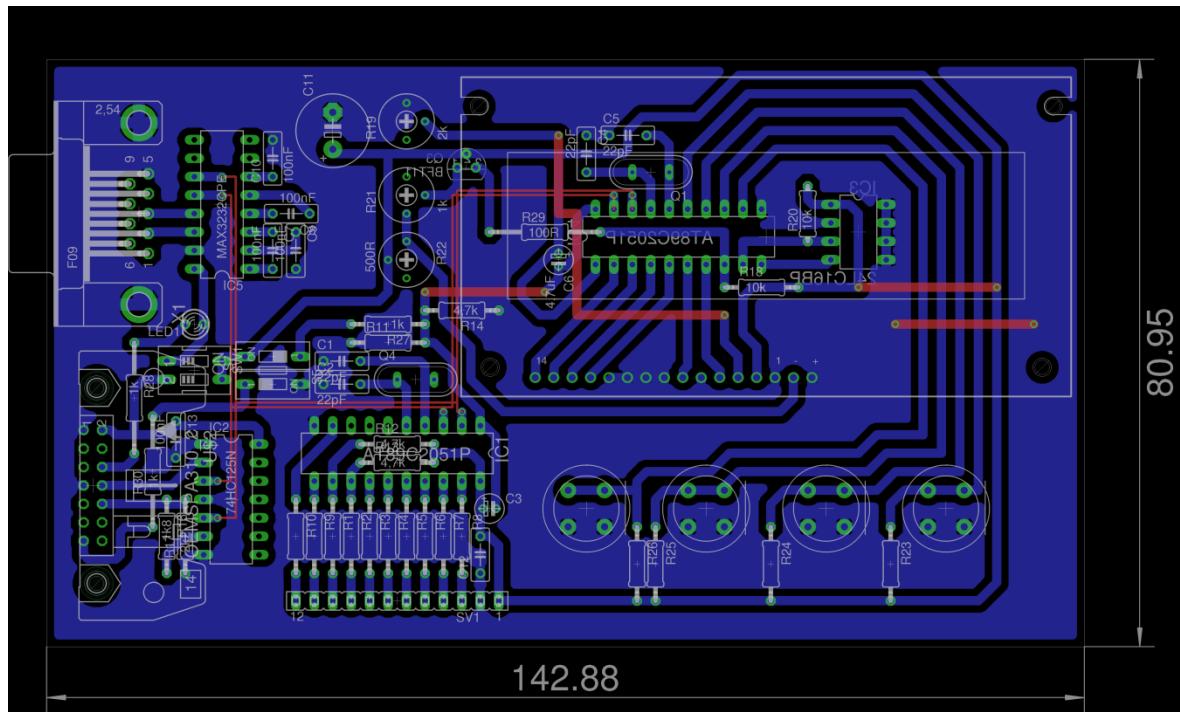
Příloha



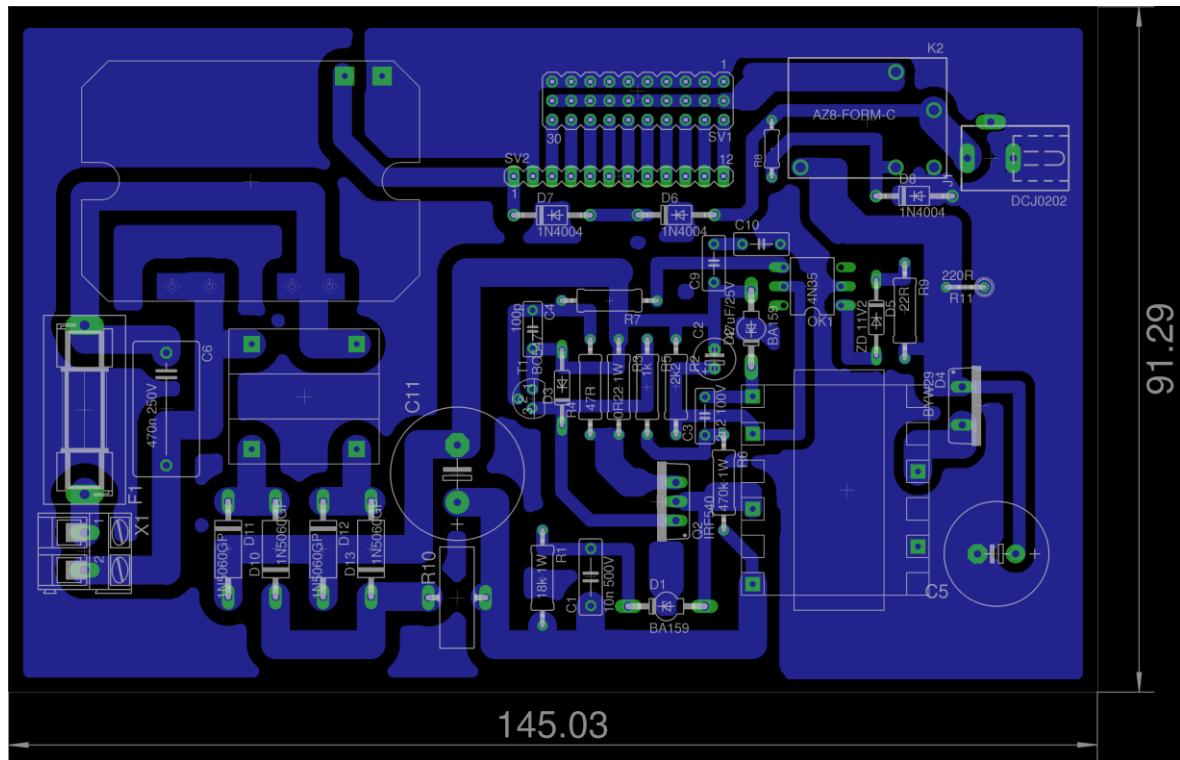
Obrázek 26: Schéma kompletního spínaného zdroje



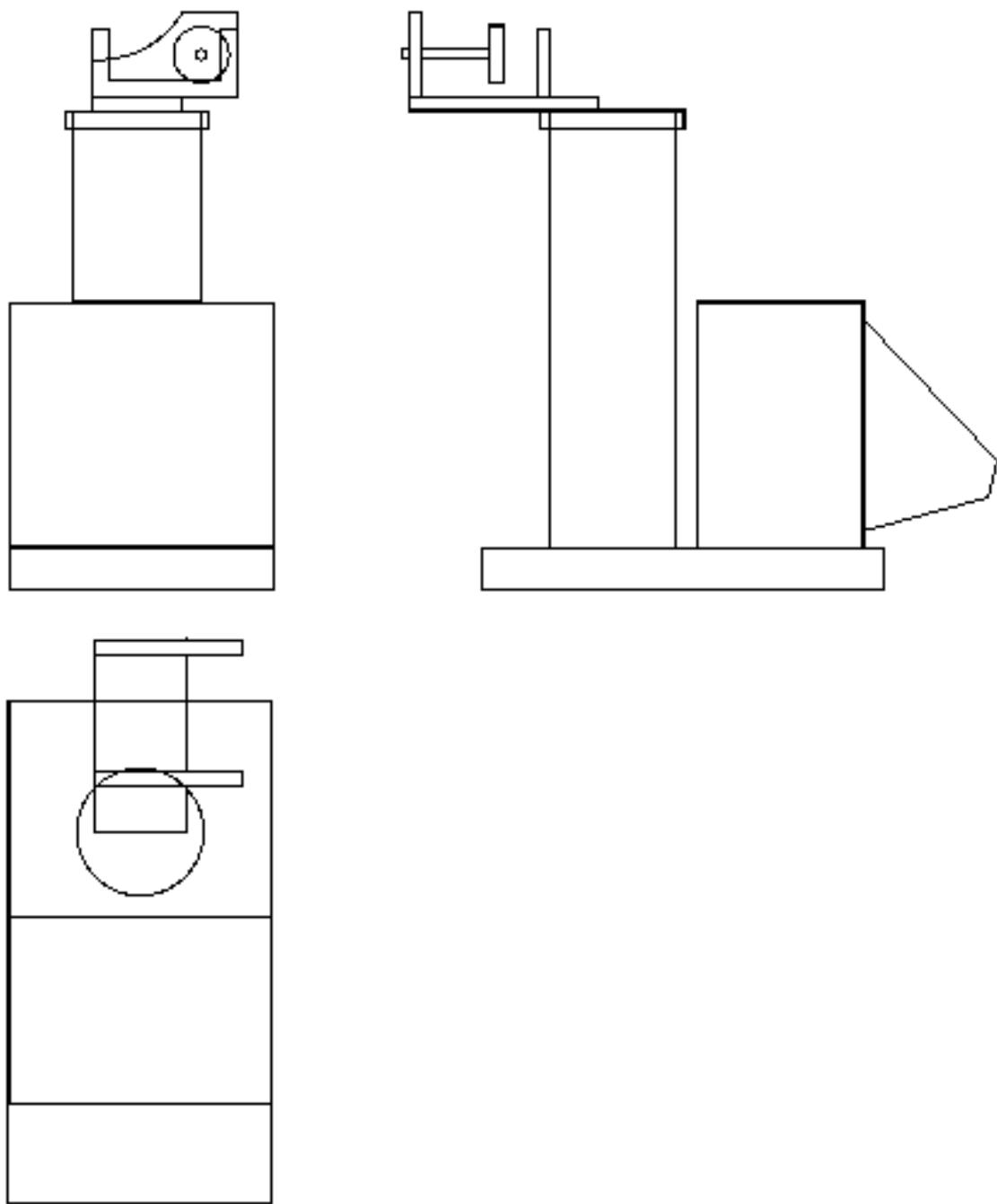
Obrázek 27: Kompletní schéma řídicí jednotky



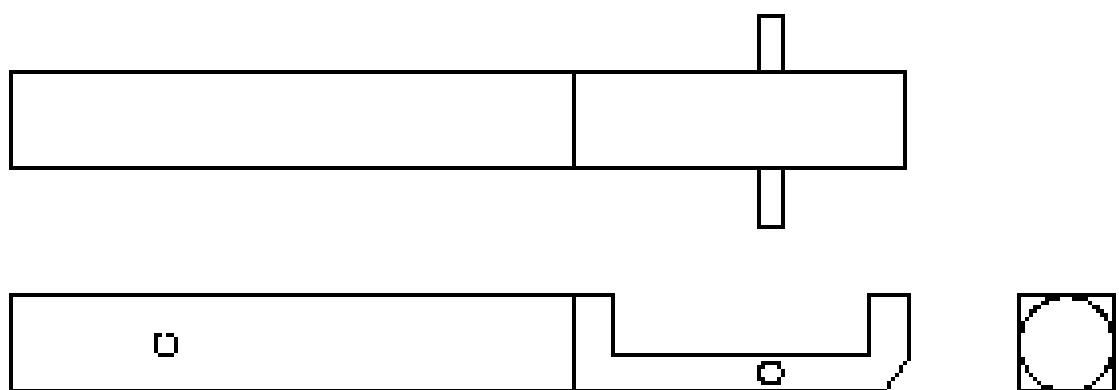
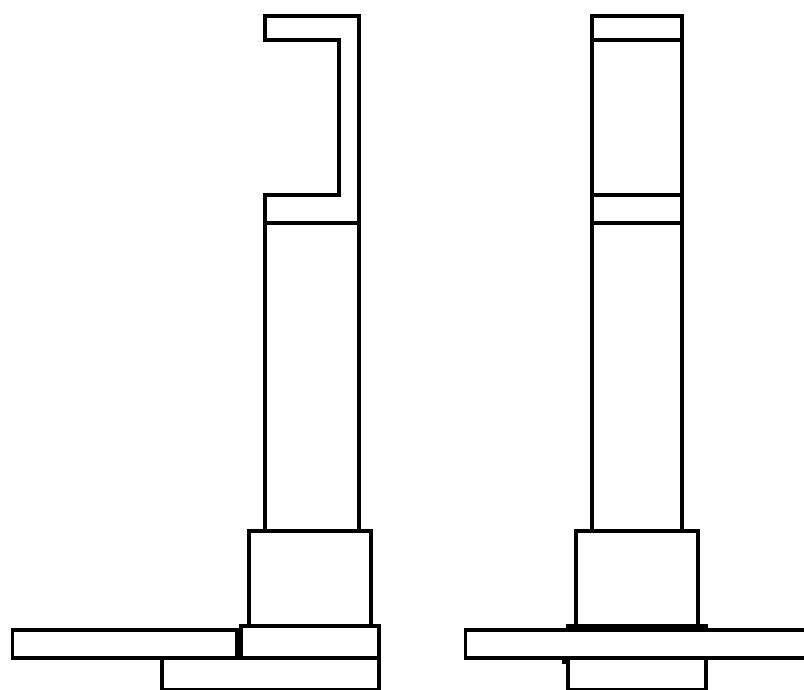
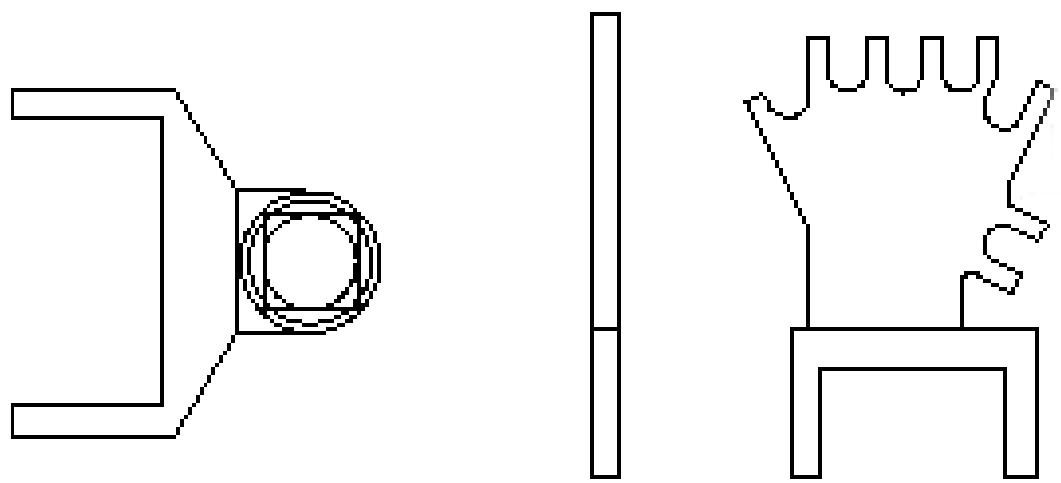
Obrázek 28: Osazovací plán desky zdrojů



Obrázek 29: Osazovací plán řídící desky



Obrázek 30: Výkres podstavce pro robotické rameno (měřítko 1:4)



Obrázek 31: Výkres robotického ramene (měřítko 1:2)