

# PixelPrint

(Raziskovalna naloga)

**Mentorji:**

Franc Vrbančič, uni. dipl. ing. el.

Tadej Vidovič, dijak, SSI, elektrotehnik

Jakob Gaberc Artenjak, SSI, tehnik računalništva

**Avtorji:**

Niko Ivezić, SSI, elektrotehnik

Tomaž Šešerko, SSI, elektrotehnik

Aljaž Šešo, SSI, tehnik računalništva



## **AVTORSTVO**

V naši skupini sodelujemo 4-je dijaki, to smo Aljaž Šešo, Tadej Vidovič (mentor), Niko Ivezić, Jakob Gaberc Artenjak in Tomaž Šešerko. Da smo se lažje organizirali smo si delo razdelili, še predno smo pa začeli delati, smo morali najprej dobiti idejo, kaj bomo naredili. Po dolgih dneh razmišljanja, smo prišli na idejo, da bi naredili tako imenovani »Pixel printer«. To je tiskalnik ki bo tiskal sliko po piksljih. Ko smo ugotovili, kako bo celotna stvar delovala, smo se takoj lotili dela. Razdelili smo se na dva dela, na programsko-mehanski del in na testiranje-dokumentiranje. Aljaž Šešo, Tadej Vidovič in Jakob Gaberc so prevzeli programsko-mehanski del, Niko Ivezić in Tomaž Šešerko pa sta prevzela testiranje in dokumentiranje.

## **ZAHVALA**

Posebna zahvala gre mentorju Francu Vrbančiču, ki nam je pomagal pri idejah in nas vzpodbudil, da smo se lotili raziskovalnega dela. V veliko pomoč nam je bil z nasveti za oblikovanje raziskovalne naloge. Zahvala gre tudi našim staršem, ki so nam vsa leta stali ob strani in razredničarki Barbari Jarh Ciglar, ki nam je omogočila razvoj robota med časom pouka.

Hvala tudi vsem tistim, ki so kakorkoli pomagali pri nastajanju izdelka in pisanju poročila.

## **POVZETEK**

Namen naše naloge je Pixel printer, ki bo tiskal slike po vsakem pikslu posebej. Pri delu smo uporabili razne mehanske postopke obdelave pri katerih smo uporabili lego kocke. Za to nalogo smo se navdušili v lanskem šolskem letu, ter smo se odločil, da izdelamo svoj izdelek. Namen našega izdelka je v veliki meri pridobivanja novega znanja in dokaz, da lahko tudi dijaki storimo kaj produktivnega z malo sredstvi. Namenjen je tiskanju portretov in slik z umetniškim pridihom. V primeru finančne podpore in daljšega časa izdelave, bi izdelali večjo in boljšo verzijo takšnega tiskalnika, saj je ta namenjen zgolj za učenje programiranja in izdelave v tej smeri.

## **SUMMARY**

The purpose of this research project is creation of Pixel printer. This is a printer that will print a picture with a pixel at a time. For the making of this printer we used different mechanical procedures for which we used lego blocks. We got inspired for this project at our school, then we decided that we will make our own product. Purpose of our product is to gain new knowledge and to proof, that we can do something productive with limited resources. The purpose of this product is to print portraits and pictures with artistic touch. In case of good financial support and longer development time, we would create larger and better version of such printer. This printers purpose is mainly for education, to learn how to program and build in that direction.

## Kazalo vsebine

AVTORSTVO .....	1
ZAHVALA .....	2
POVZETEK.....	3
SUMMARY.....	3
Kazalo slik.....	5
Kazalo tabel.....	5
STROKOVNA TERMINOLOGIJA, TABELA NEZNANIH BESED .....	6
1. UVOD .....	8
2. OPREDELITEV RAZISKOVALNEGA PROBLEMA – RAZISKOVALNO VPRAŠANJE .....	9
2.1. RAZISKOVALNO VPRAŠANJE(TRDITEV) Z OBRAZLOŽITVIJO .....	9
2.2. Razlaga .....	9
3. Pretvorba iz slike v datoteko za robota .....	11
4. Funkcijski bloki .....	12
5. Pisanje programa.....	14
5.1. Vezalni načrt.....	14
5.1.1. Uporabljeni elementi (krmilnik, aktuatorji, senzorji) .....	14
5.1.2. Shema.....	15
5.2. Program .....	16
5.2.1. Opis programa:.....	18
7. Konstrukcija.....	20
7.1. Glavni nosilec .....	21
7.2. Mehanizem za premik lista.....	22
7.3. Pisalna glava.....	23
8. POSEBNO POGLAVJE - STROKOVNA POVEZAVA Z UPORABO (AKTUALNOST PROBLEMA).....	24
9. REZULTATI Z VREDNOTENJEM.....	25

## **Kazalo slik**

Slika 1: ikona orodja LEGO NXT.....	14
Slika 2: shema vezave elektronskih elementov.....	15
Slika 3: program.....	16
Slika 4: program razdeljen na odseke.....	17
Slika 5: Zgradba tiskalnika.....	20
Slika 6: Glavni nosilec, pogled od zadaj.....	21
Slika 7: Premik papirja.....	22
Slika 8: Pisalna glava.....	23

## **Kazalo tabel**

Tabela 1: Strokovna terminologija, tabela neznanih besed.....	6
---	---

## STROKOVNA TERMINOLOGIJA, TABELA NEZNANIH BESED

Pojem / strokovna beseda	Prevod	Pomen
Lego Mindstroms NXT	NXT pametna kocka	Lego Mindstroms NXT je univerzalen krmilnik s pomočjo katerega se programira in na katerega se programira.
Mikrokrmilnik	/	Mikrokrmilnik ali mikrokontroler je čip, ki vsebuje skoraj vse sestavine mikroročunalnika (procesor, notranji pomnilnik, vmesnike ...). Za popoln mikroročunalnik, mikrokrmilniku manjkajo le vhodno-izhodne enote (tipke, senzorji, elektromotorji, žarnice ...), ki niso primerne za vgradnjo v čip. Mikrokrmilnike srečamo v večini modernih elektronskih naprav. Na primer v mobilnem telefonu, televiziji, v DVD-predvajalniku, v mikrovalovni pečici, v pralnem in pomivalnem stroju... V našem primeru je to NXT pametna kocka.
Robot	/	Robot je stroj, ki ga nadzoruje računalnik in ga lahko programiramo, da samostojno opravlja določeno opravilo. Robote pogosto uporabljamo v industriji za prenašanje materiala ali za izvajanje ponavljajočih se opravil.
Čip	Vezje	Čip ali integrirano vezje je mikrovezje, sestavljeno iz množice elektronskih elementov, ki so na skupnem vezju iz polprevodniškega materiala med seboj povezani v električno vezje.
Funkcijski bloki	/	Funkcijski bloki so orodja, ki se uporabljajo v grafičnih načini programiranja. Vsak blok ima svoj pomen in upravlja točno določeno funkcijo. Funkcijski bloki so v programu označeni z kvadrati in povezani z puščico, ki označuje smer in zaporedje izvajanja programa.
Actuator	Aktuator	To so naprave, ki izvršujejo dejanja, kot so premikanje ali upravljanje mehanizma ali sistema.
Podatkovne linije	/	Podatkovne linije so linije, ki v grafičnem načinu programiranja povezujejo funkcijske bloke. Med seboj se ločijo po funkcionalnosti, ki jo določimo z barvo linije (rumena za številke, oranžna za tekst).
System	Sistem	Sistem je sestavljen iz preprostejših delov, ki je samostojen torej deluje neodvisno o drugih faktorjev. Računalniški sistem je sestavljen iz strojne opreme in programske opreme.
Hardware	Strojna oprema	Strojna oprema so sestavni deli sistema, ki so fizični torej se jih lahko dotaknemo. V računalniku so to diski, procesor, matična plošča,... v našem tiskalniku pa ohišje, motorčki...



Software	Programska oprema	Programska oprema (program) je tisti del sistema, ki upravlja s strojno opremo. Programske opreme se ne moremo dotakniti ali je videti, razen na monitorju.
Senzor	/	Senzor je naprava, ki zaznava zunanje vplive in podatke posreduje v sistem. Senzorji so lahko analogni ali digitalni. Lahko zaznavajo svetlobo, pritisk, oddaljenost... V našem primeru smo za senzorje uporabili kar tipko.
Pixel	Piksel	Piksel je točka, ki nima oblike ali določene velikosti, le informacije o karakteristiki slike. Na zaslonu pa je piksel omejen z obliko in velikostjo in predstavlja tudi mersko enoto, s katero izražamo ločljivost prikazane slike, iz česar je možno izpeljati tudi njeno fizično velikost.
Algorithm	Algoritem	Algoritem je navodilo, s katerim rešujemo nek problem. Običajno je zapisan kot seznam/zaporedje korakov/postopkov, ki nas pripeljejo do rešitve problema.
Binary	Binarno	Binarni jezik, je strojni jezik, se pravi jezik katerega razumejo računalniki in vse naprave. Binarna datoteka je datoteka katera vsebuje le številke 0 in 1.

## **1. UVOD**

Robot je stroj, ki ga nadzoruje računalnik in ga lahko programiramo, da samostojno opravlja določeno opravilo. Poznamo stacionarne ter mobilne robote, ki opravljajo svojo zastavljeno nalogo po zamisli človeka. Vse to, da bi človeku olajšali življenje. Najrazličnejši roboti že obstajajo denimo v avtomobilski industriji, tovarnah, kjer človek ne more dostopati itd. Tako smo se odločili izdelati stacionarnega robota, ki bo tiskal slike po vsakem pikslu posebej. To mora storiti v čim krajšem času, ter čim bolj natančno. Za vse potrebe ima vgrajene stikala in motorčke. Ker mora opravljati točno določeno nalogo smo si za programiranje izbrali Lego Mindstorms NXT. Do te naloge nas je pripeljala želja po izdelavi nekaj svojega, želja po raziskovanju. Izdelava je zahtevala veliko teoretičnega znanja, predvsem programiranja ter gradnja.

## **2. OPREDELITEV RAZISKOVALNEGA PROBLEMA – RAZISKOVALNO VPRAŠANJE**

Tehnika iz dneva v dan napreduje, se razvija, zato moramo tudi mi slediti tehnologiji. Imamo znanje LEGO sistema NXT, programiranja in mehatronskega delovanja sistemov. Naša želja je, da naredimo sistem, ki bo natisnil ljudem prepoznavno fotografijo.

### **2.1. RAZISKOVALNO VPRAŠANJE(TRDITEV) Z OBRAZLOŽITVIJO**

Vprašanje: »Ali je z obstoječimi materiali in dosegljivim znanjem možno narediti tiskalnik, ki bo tiskal ljudem prepoznavne slike?«

Uspešno je takrat (hipoteze):

- Ko bodo vse komponente, vključno z programsko in strojno opremo, delovale kot en Lego sistem.
- Ko bomo uspešno pretvorili slikovno datoteko v tekstovno, katero bo naš robot zmožen prebrati.
- Ko bo program na robotu deloval glede na podatke v datoteki.
- Takrat, ko bo natisnjena slika prepoznavna ljudem, se pravi, da bo 10 ljudi prepoznalo vseh 10 fotografij.

Trditev: Narediti delujoči mehatronski sistem (ploter), ki bo tiskal črno-bele portrete.

TRDITEV BO PRAVILNA:

- Ko bo mehatronski sistem narejen in bo deloval ter ko bo znan algoritem ki bo datoteko pretvarjal v risbo na papirju.
- Ko bo znan algoritem pretvorbe slike v datoteko prepoznavno našemu programu.

### **2.2. Razlaga**

V osnovi je robot v celoti zgrajen iz »LEGO« kock. Izjema je le pisalo z široko konico. Ima tri premikajoče se dele. Premikanje lista naprej – nazaj, premikanje glave levo – desno in premikanje pisala gor – dol.

Da bi pa robot narisal željeno sliko jo je sprva potrebno pretvoriti v njemu prepoznavno obliko. Ta oblika je datoteka s številčkama 1 in 0, 1 za črno barvo, 0 za belo bravo. Sliko pretvorimo tako, da jo z programom za obdelavo slik naredimo črno-belo, potem pa reguliramo kontrast ter svetlost, da se znebimo sivin in dobimo čisto črno-belo sliko brez sivin. Nato ji določimo velikost, ki mora biti v skladu s programom na robotu. Ko smo vse to storili jo s pomočjo programa pretvorimo v znake in nato v 1 in 0.

Ko dobljeno datoteko naložimo na robota jo ta prebere in številke v določenem zaporedju pošilja do obdelave in nato krmili motorčke (pomikanje lista, glave in pisala). Pred začetkom pisanja robota ročno kalibriramo. To je edini postopek pri celotni izdelavi slike, ki ga mora narediti človek, če izvzamemo list papirja.

Program na robotu deluje tako, da bere podatke iz datoteke. Če prebere število 0, potem to pomeni da je prostor (piksel) prazen zato se pomakne dalje in pisala ne spušča. Če pa prebere število 1 potem pomeni, da je na tem mestu črna barva, zato spusti pisalo in naredi piko. Pri sliki velikosti 100x100 ponovi postopek 10000. Ko prebere 100 števil je končal eno vrstico in pisalo se premakne v naslednjo vrstico. Premik navzdol tudi ponovi stokrat, vsakič ko se vrstica zaključi. Za izdelavo ene slike potrebuje približno od vsaj 30 minut.

### 3. Pretvorba iz slike v datoteko za robota

Da bi lahko robot »videl« sliko jo moremo prvo spremeniti v številke. To naredimo po naslednjih korakih.

- Poiščemo ali narišemo poljubno sliko
- Sliki spremenimo velikost na okrog 100x100 pikslov.
- Sliki prilagodimo barvo kontrast in svetlost tako da bo v željenem stanju (poljubno, pomembno je, da nimamo sivin, samo črno in belo barvo). Velikost slike moramo nastaviti v skladu s podatki iz programa.



- Nato s pomočjo programa »ImageJ« sliko pretvorimo v znake, katere nato zamenjamo z 1 in 0.
- Tako smo le v nekaj postopkih sliko spremenili v številke, ki jih lahko prebere program na robotu.

## 4. Funkcijski bloki

Funkcijski bloki so elementi, uporabljeni za izdelavo programa. Vsak ima v ozadju kompleksno kodo, ki določi za kaj se funkcijski blok uporablja.

1. **Keep Alive:** S tem blokom lahko preprečimo našemu robotu da bi prešel v stanje pripravljenosti ali pa se samodejno izklopil. Če se v to malo poglobimo, pa mu lahko tudi določimo kdaj preide v stanje pripravljenosti. Izbiramo lahko med 2, 5, 10, 30, in 60 minutami.



2. **Loop:** S tem blokom robotu določimo kolikokrat naj postopek, ki je v tem delu, ponovi. Čas ponavljanja mu lahko določimo glede na število ponovitev, senzor, logično stanje, ... Dodamo mu lahko tudi števec, katerega lahko v nadaljnje uporabimo.



3. **Wait:** Je eden najpreprostejših blokov. Določimo mu naj počaka tako dolgo, da se izteče čas, ali pa dobimo ukaz od poljubnega sensorja.



4. **Motor:** S tem blokom lahko upravljamo z motorčkom, pogosto na izhodih 1, 2 in 3. Določamo lahko kateri motorček želimo upravljati, lahko izberemo le en motorček na enkrat. Določimo mu smer, moč, čas vrtenja v obratih, sekundah, ali pa neskončno. Določimo mu lahko tudi ali se naj ob koncu ukaza takoj ustavi ali pa se odvrta dokler ga trenje ne ustavi.



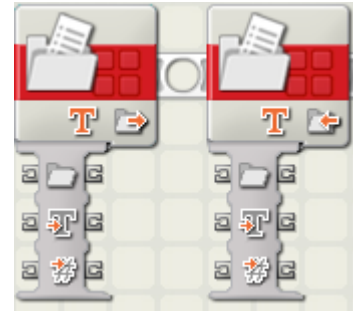
5. **Display:** Ali prikazovalnik. S to kocko lahko določamo kaj naj se prikaže na prikazovalniku. Izbiramo lahko med številnimi simboli. Preko naprednih povezav pa lahko tudi sami nekaj napišemo oziroma narišemo.



6. **Switch:** S tem blokom lahko izbiramo med dvema stanjema (je ali pa ni). To nam pomaga še posebej pri senzorjih, ali pa pri spremenljivkah. Lahko mu določimo tudi da stanje oceni glede na primerjavo besedila in števil.



7. **File Access:** S pomočjo tega bloka lahko beremo besedilo ali številke iz mape, katero smo naložili na robota. S pomočjo naprednih povezav lahko določimo kdaj, od kje, in kolokokrat naj robot prebere znake iz označenega dokumenta. Nato moramo postaviti še eno kocko s pomočjo katere beremo podatke, ki jih je prej poslala prva kocka, in jih s pomočjo naprednih povezav pošiljamo naprej.



## 5. Pisanje programa

Program za naš tiskalnik smo napisali z programskim orodjem LEGO MINDSTORMS Education NXT. Programiranje v tem razvojnem okolju poteka z funkcijskimi bloki (ang. function blocks), torej podobno kot programiranje nekaterih industrijskih PLC-jev (Programmable Logic Controller). Tak način programiranja nam omogoča hiter razvoj kompleksnega programa.

### 5.1. Vezalni načrt

Preden začnemo programirati potrebujemo vezalni načrt, shemo kako so senzorji in aktuatorji priklopljeni na naš krmilnik oz. NXT kocko. NXT kocka ima 9 priključkov:

- 1 priključek za napajanje
- 1 USB priključek tipa B
- 3 izhode z NXT priključki
- 4 vhode z NXT priključki

NXT priključki so podobni RJ11 in RJ12 priključkom, vendar niso med sabo združljivi.

#### 5.1.1. Uporabljeni elementi (krmilnik, aktuatorji, senzorji)

V vezalnem načrtu so naslednji elementi:

- NXT kocka:

krmilnik, zbira podatke iz senzorjev in podatke, ki mu jih ki podamo ter glede na program upravlja z aktuatorji oz v našem primeru z motorčki

- trije motorji:

uporabljeni so LEGO motorčki, priloženi v kompletu, motorčki premikajo mehanizem za dvig in spust pisala, premikajo pisalo, ter premikajo list

- končno stikalo:

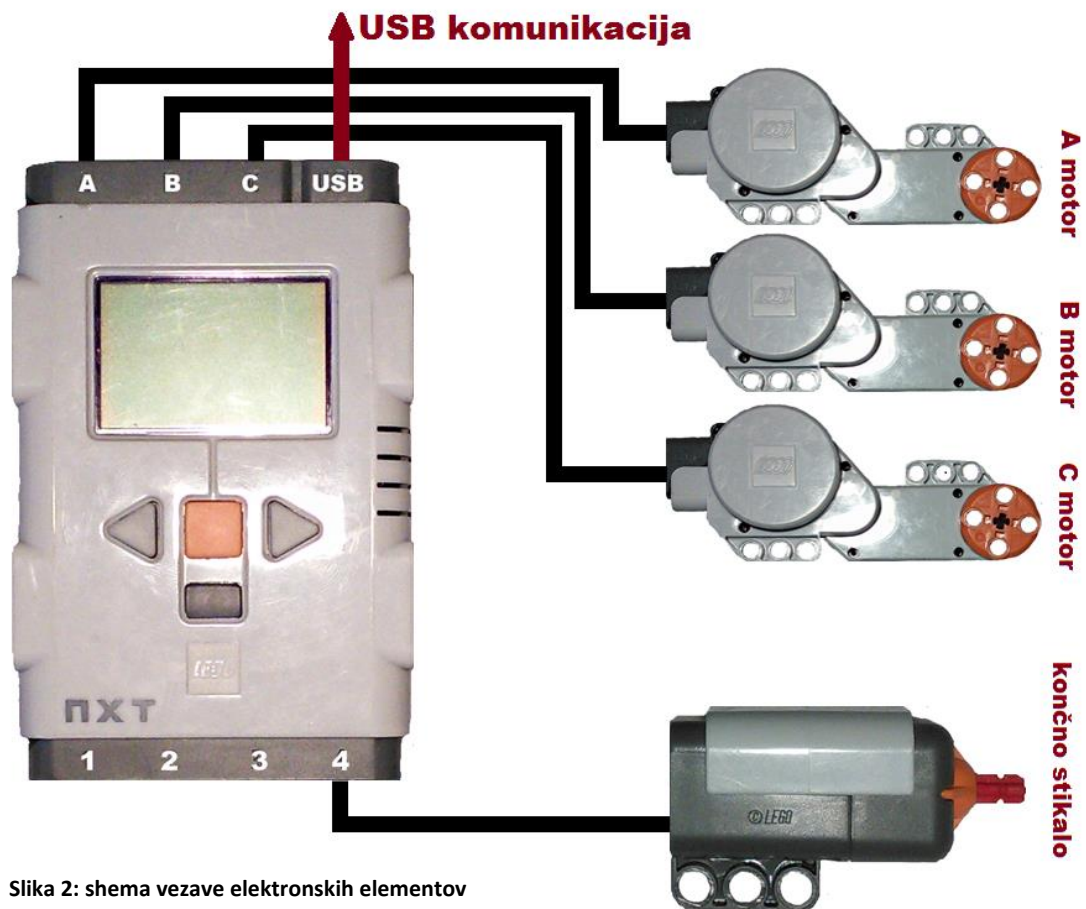
tudi uporabljeno stikalo je produkt podjetja LEGO



Slika 1: ikona orodja LEGO NXT



## 5.1.2. Shema

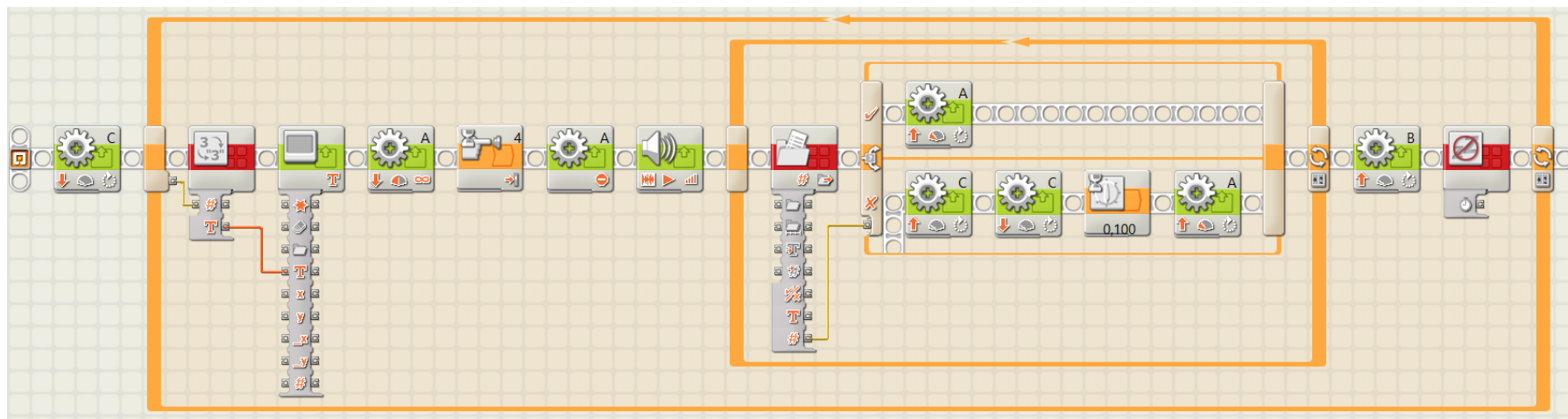


Slika 2: shema vezave elektronskih elementov

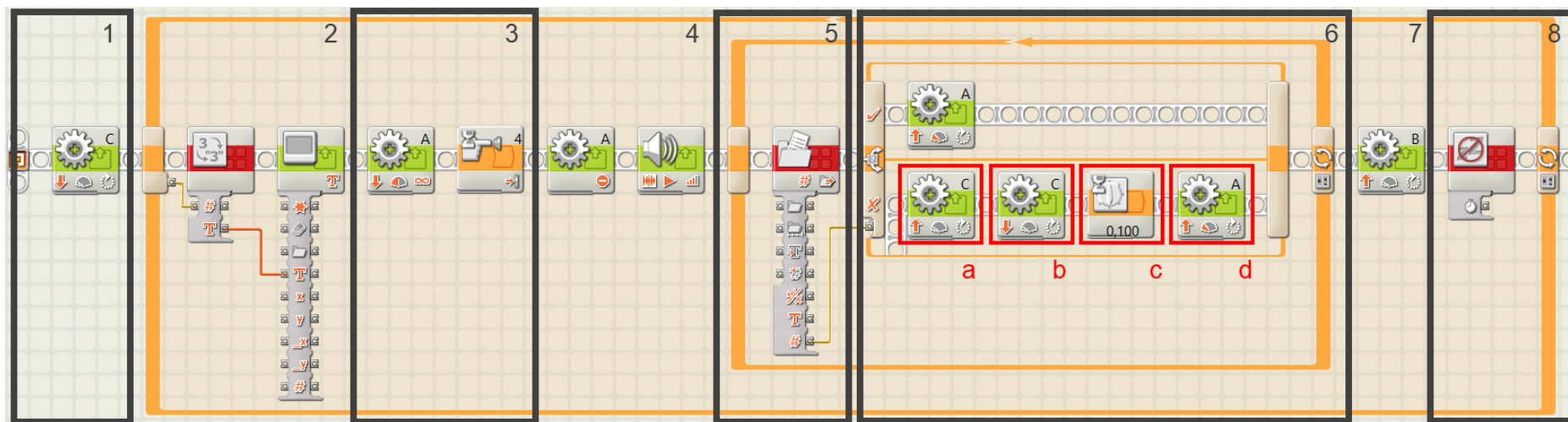
Na sliki imamo prikazane elektronske povezave med krmilnikom in elementi. A motorček je uporabljen za premikanje pisala levo in desno, B motorček za premikanje lista, C motorček za spust in dvig pisala, končno stikalo pa za kalibracijo ob prehodu v novo vrstico. Funkcije posameznih motorčkov in stikala so opisane v poglavju o konstrukciji.

## 5.2. Program

Na sliki imamo napisan program kot ga vidimo v LEGO NXT razvojnem programu.



Slika 3: program



Slika 4: program razdeljen na odseke

### 5.2.1. Opis programa:

1. Program se prične tako da zavrti motor C za 40°. S tem dvignemo pisalo okrog 1,5 cm od podlage na katero rišemo. Preden program zaženemo moramo pisalo ročno spustiti na risalno podlago ravno toliko, da pusti odtis. S tem kalibriramo dvig in spust pisala.
2. Prva zanka (ang. loop). Zanki lahko nastavimo koliko krat se ponovi. Tej zanki nastavimo, da se ponovi tolikokrat, kolikor je visoka slika, ki jo hočemo natisniti. Torej če imamo 200px (pikslov) visoko sliko nastavimo, da se zanka ponovi 200 – krat. Tako se vse znotraj te zanke ponovi 200 – krat. Zanka se konča v osmem odseku, v osmem odseku pa tudi nastavimo število ponovitev te zanke. V tem odseku imam zanka tudi izhod za števec. Ta izhod nam pove kolikokrat se je zanka ponovila in posledično katero vrstico tiskalnik tiska. To število preko rumene podatkovne linije prenesemo na naslednji funkcijski blok (rdeč, number to text), ki nam pretvori številko v tekstovno obliko, ki je primerna za izpis na LCD zaslonu. Iz funkcijskega bloka za pretvorbo dobimo oranžno podatkovno linijo, ki prenaša tekst. Naslednji funkcijski blok (zelen, display block) sprejme podatke iz te linije in jih izpiše na LCD ekran. Tako na ekranu dobimo podatek katero vrstico trenutno naprava tiska.
3. Funkcijski blok za gibanje motorčka (zelen, motor block) ukaže motorčku, da se vrti v smeri proti končnemu stikalu z močjo 50%. Tako se pisalo začne pomikati proti začetnemu položaju. Naslednji blok (oranžen, wait) je nastavljen tako, da ustavi program dokler se ne sproži končno stikalo na vhodu 4. Ko se stikalo sproži se program nadaljuje.
4. Ko se je stikalo sprožilo se je program nadaljeval in funkcijski blok (zelen, motor block) nam ustavi motor A. Tako smo pisalo dobili v začetni položaj oz. smo ga kalibrirali. Takoj ko se motor A ustavi se sproži zvočni signal iz zvočnika NXT kocke. To nam naredi drugi funkcijski blok v četrtem odseku (zelen, sound).
5. V tem odseku se prične druga zanka. Število ponovitev te zanke je enako širini slike. Torej če je slika široka 100px nastavimo, da se ta zanka ponovi sto krat. Število ponovitev nastavimo na koncu šestega odseka, kjer se zanka konča. Prvi funkcijski blok v tej zanki (rdeč, file acces) nam omogoča dostop do datotek v spominu NXT kocke. Nastavljen je tako, da odpre izbrano datoteko, v tem primeru slika.txt, in začne brati številke iz nje. Blok deluje tako, da vsakič ko se sproži, prebere številko v naslednji vrstici. Številko nato odda na izhod oz. na rumeno podatkovno linijo.
6. Prvi element v tem odseku je funkcija switch. Ta funkcija sprejme na vhod rumeno podatkovno linijo iz prejšnjega funkcijskega bloka za dostop do datotek. Funkcija je nastavljena tako, da prebere številko iz te linije in jo primerja z pogoji, ki smo jih mi zapisali. Če prebere število 0 se začne izvajati program napisan za znakom kljukica. V tem primeru se motor A premakne naprej za 10 stopinj z močjo 30% in se nato ustavi. S tem se pisalo v zraku premakne 2 milimetra naprej. Potem se program nadaljuje in zanka se ponovi. Če pa funkcija prebere število 1 se začne izvajati program za znakom križec. Ta program pa spusti pisalo, da naredi odtis nato dvigne pisalo in ga premakne naprej za 2 milimetra. Blok v rdečem okvirčku »a« (motor block) zavrti motor C za 40°, s tem se pisalo spusti in pusti odtis na risalni površini.

Blok v okvirčku »b« (motor block) zavrti motor C v drugo smer za 40° in s tem dvigne pisalo v začetni položaj. Blok iz okvirčka »c« (wait) je nastavljen, da ustavi program za 100 milisekund.

V tem času se konstrukcija tiskalnika umiri in tako preprečimo morebitne napake v delovanju.

Naslednji blok v okvirčku »d« (motor block) zavrti motor A za 10° z 30% močjo in ga nato ustavi. Tako se pisalo premakne v naslednjo pozicijo. Program se potem nadaljuje in zanka se ponovi, razen če se je že dovolj krat ponovila. Če se je že dovolj krat ponovila se program nadaljuje v naslednji odsek.

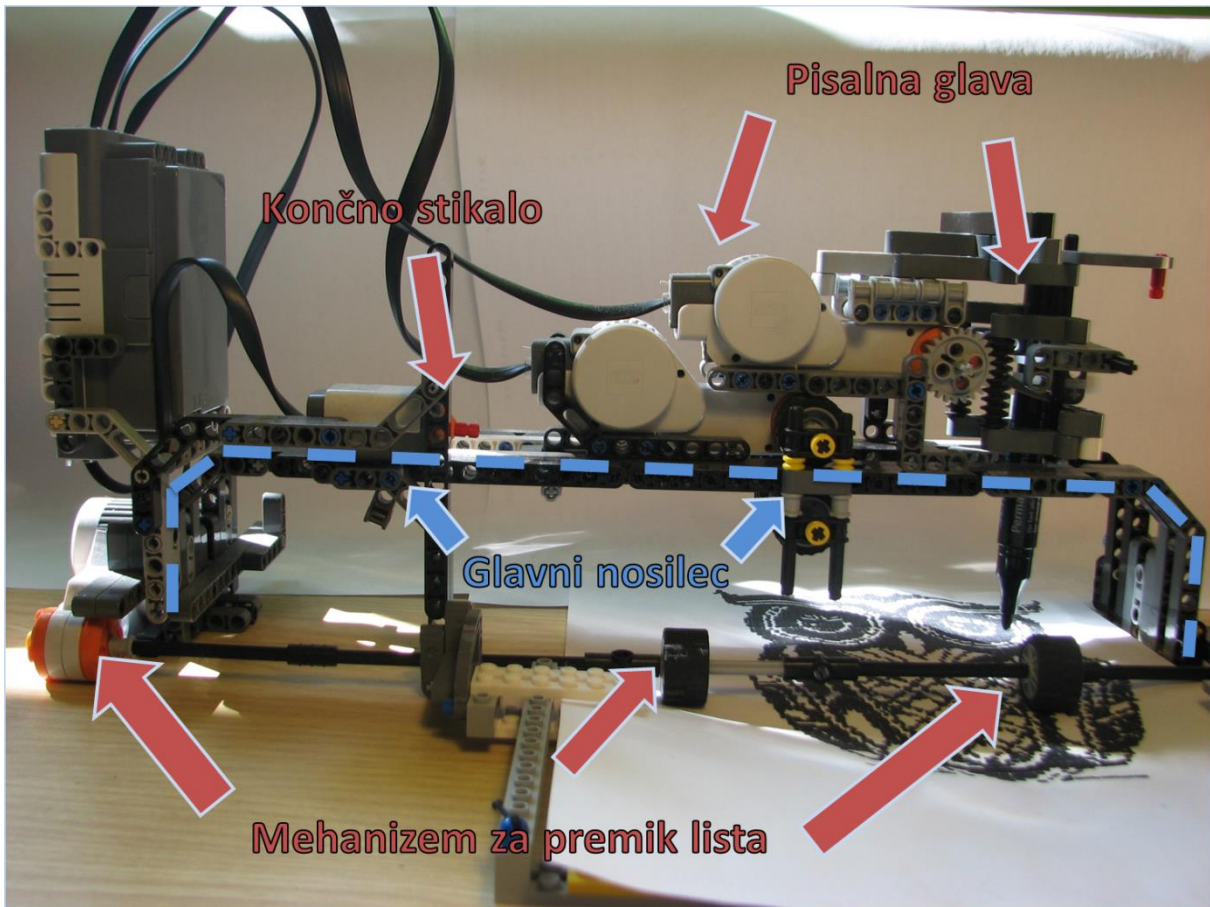
7. Funkcijski blok (motor block) v tem odseku premakne motor B za 10° z 15% močjo. Motor B premika list, zato smo tudi uporabili tako nizko moč. Z nizko močjo smo povečali natančnost, saj tako mehanizem, ki premika list ne zdrsava po površini lista. List se premakne vsakič ko se konča vrstica oz. vsakič, ko se druga zanka konča.

8. V zadnjem odseku imamo funkcijski blok keep alive. NXT kocka je vnaprej sprogramirana, da se samodejno izklopi po določenem času. S tem blokom preprečimo, da se samodejno ugasne. Zadnja funkcija pa je konec prve zanke, tukaj določimo število ponovitev te zanke oz. višino natisnjene slike. Ko se ta zanka dovolj krat ponovi se program konča in slika je natisnjena.

## 7. Konstrukcija

Tiskalnik je zgrajen iz kock Lego Technik. Zgradba je precej preprosta (slika 5), saj je sestavljen iz treh glavnih delov:

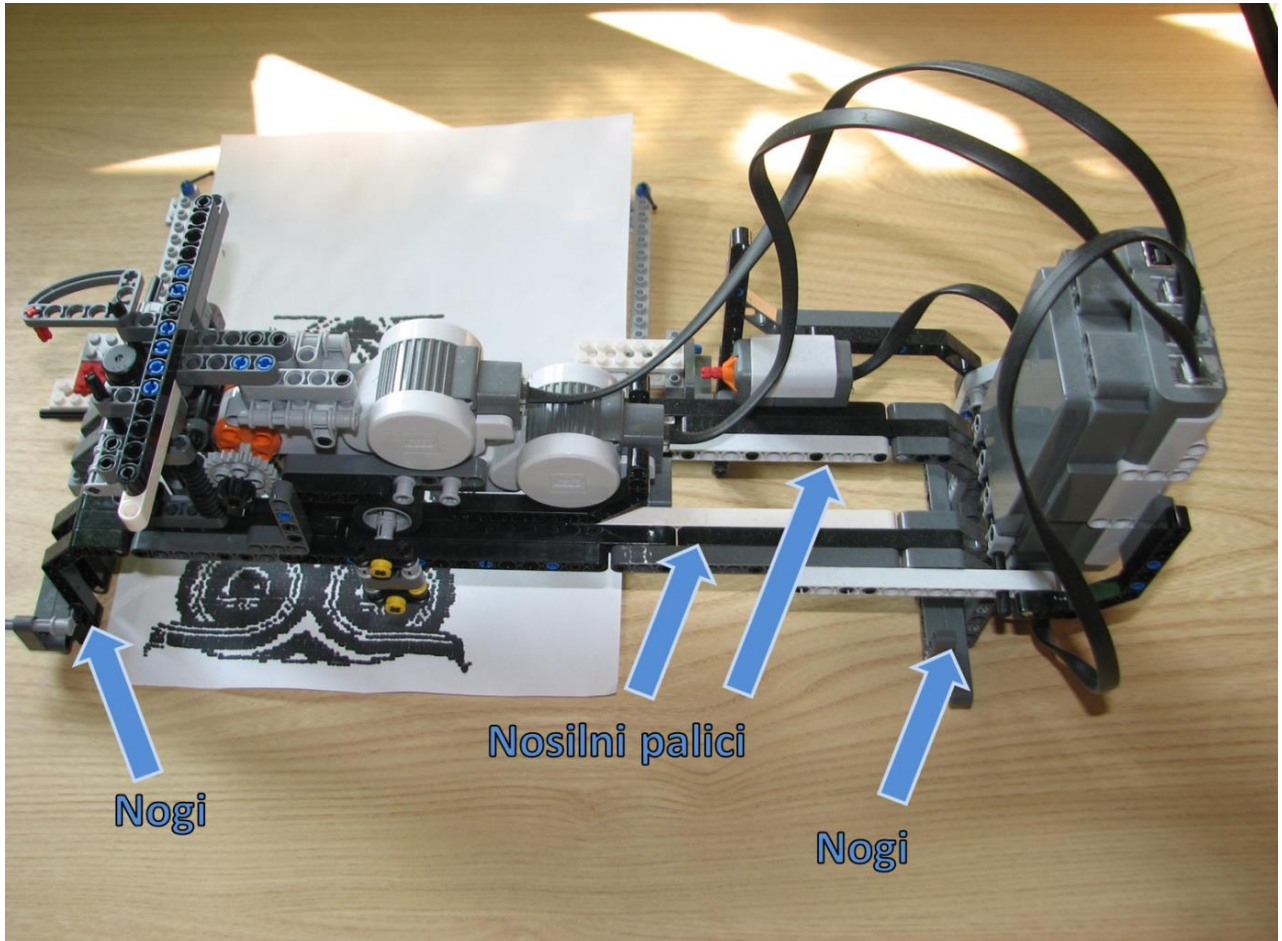
- glavni nosilec
- mehanizem za premik lista
- pisalna glava.



Slika 5: Zgradba tiskalnika

## 7.1. Glavni nosilec

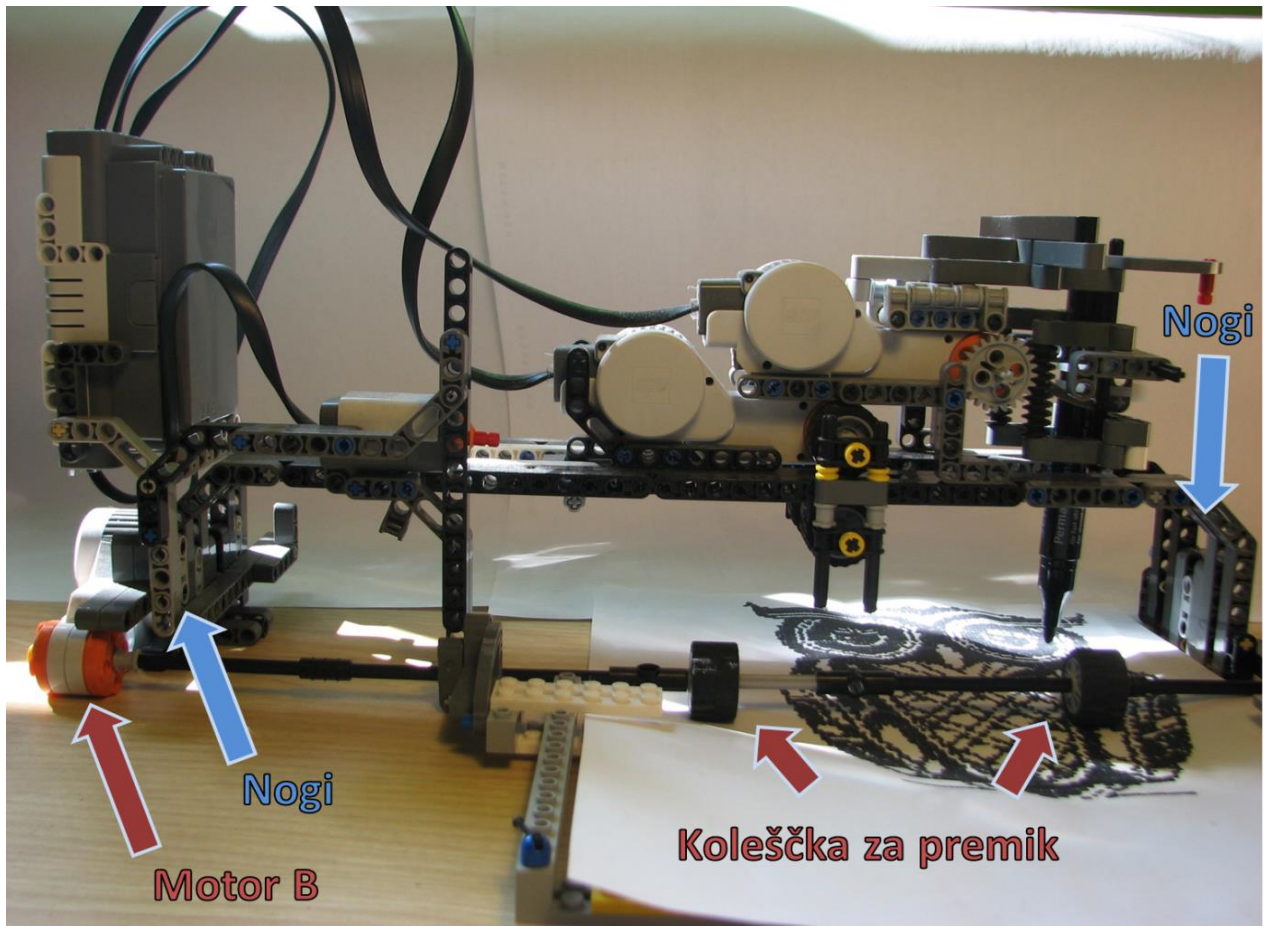
Glavni nosilec (slika 6) je oblikovan iz dveh večjih delov oz. nosilnih palic, ki nudita podpora za pisalno glavo, NXT pametno kocko in mehanizmu za premikanje lista. Vsaka palica ima na vsaki strani navzdol zavrt del oz. nogo (slika 6, slika 7), tako da sta palici v zraku. Te noge nosijo celoten tiskalnik.



Slika 6: Glavni nosilec, pogled od zadaj

## 7.2. Mehanizem za premik lista

Mehanizem za premik lista (slika 7) je narejen iz motorja, osi in dveh gumijastih koles. Deluje tako, da motor B zasuče os za določeno število stopinj, kolesa pritrjena na os pa so v kontaktu z papirjem, zato ga premaknejo. K temu mehanizmu tudi spada del za izravnavo lista, ki preprečuje rotacijo lista. Ta del je sestavljen iz dveh ravnih dolgih ravnih kock, ena je pritrjena na desno nogo, ena pa ima posebej nosilec, ki je pritrjen na nosilno palico. Zgradba je dimenzionirana za A4 list.



Slika 7: Premik papirja

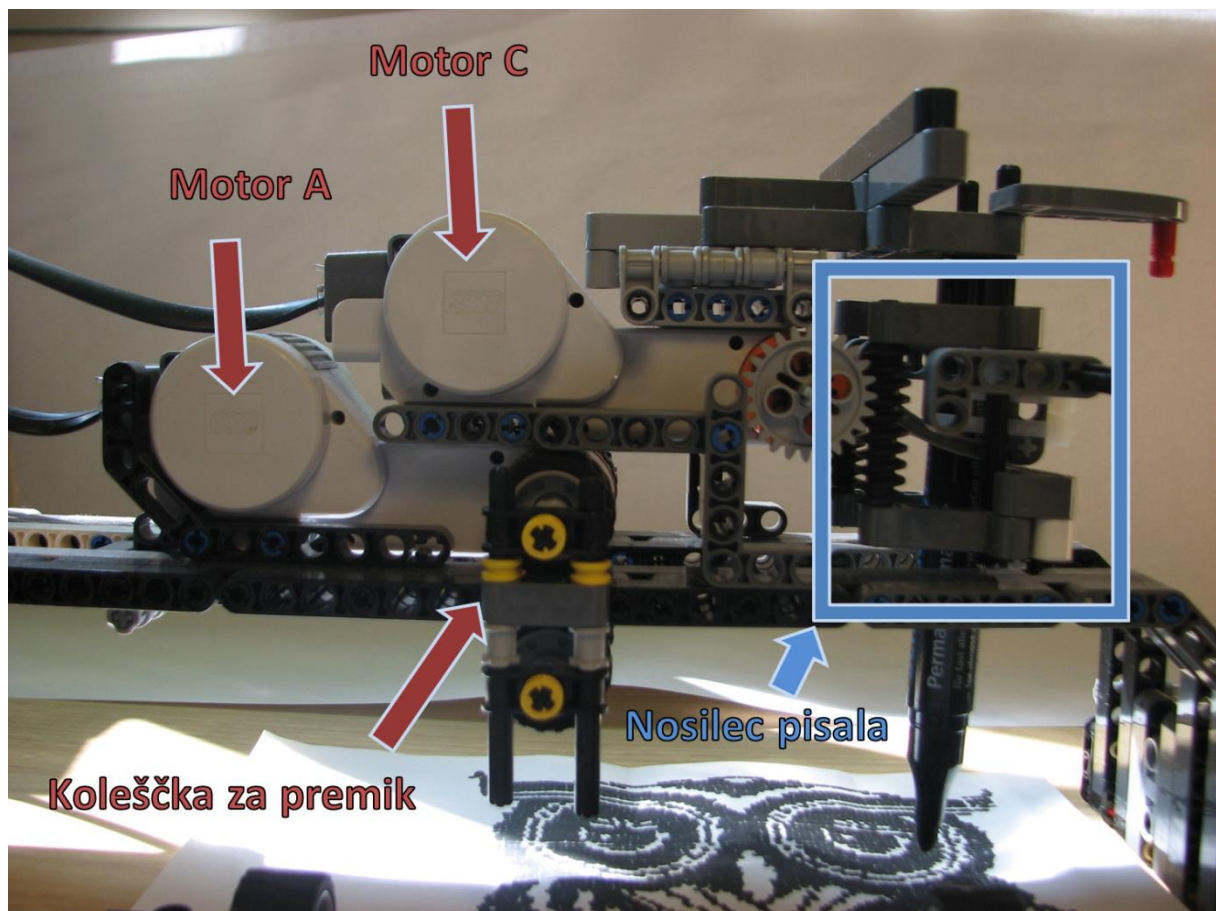


### 7.3. Pisalna glava

Pisalna glava (slika 8) je sestavljena iz dveh glavnih mehanizmov: mehanizem za premik glave naprej/nazaj in mehanizem za dvig/spust pisala. Najprej mehanizem za premik pisalne glave.

Mehanizem za premik pisalne glave je sestavljen iz motorja A in dveh parov koles. Vsak par koles ima svojo os. Zgornji par koles poganja motor A, ta kolesa premikajo pisalno glavo po nosilnih palicah. Spodnji par koles pa preko distančnika vleče zgornja kolesa navzdol ter s tem ustvarja več trenja, kot rezultat tega pa zgornja kolesa ne drsijo po površini in omogočajo natančne premike. Distančnik, ki vleče kolesa skupaj je pritrjen na osi zgornjih in spodnjih koles. Distančnik je uporabljen na obeh straneh osi, kot vidimo na sliki 6 in sliki 8, uporabljen pa je tudi za proženje končnega stikala, kar vidimo na sliki 5.

Mehanizem za dvig in spust pa je sestavljen iz motorja C, ki preko dveh zobnikov dviga in spušča nosilec pisala. Na os motorja C je na vsaki strani nameščen en zobnik, ta zobnika pa dvigata in spuščata spirali na nosilcu pisala. Spirali sta nameščeni tako, da se ne vrtita, zato pa se zaradi vrtenja zobnika dvigata in spuščata. Spirali smo uporabili zato, ker nam zgradba ni omogočala uporabiti nazobčane letve. Na spirali so nameščeni štirje nosilci med katere damo pisalo. Pisalo pa nato pritrdimo z uporabo vsaj dveh vezic.



Slika 8: Pisalna glava

## **8. POSEBNO POGLAVJE - STROKOVNA POVEZAVA Z UPORABO (AKTUALNOST PROBLEMA)**

Torej z našim znanjem je mogoče narediti veliko večje projekte, kot je bil ta. Ta projekt nas je omejeval z sredstvi. Z več materiali bi lahko izdelali človeku zanimive in dan danes popularne predmete/stroje, kot so:

- 3D tiskalnik, saj smo že naredili 2D tiskalnik. 3D tiskalniki so trenutno zelo popularni in se zelo uporabljajo v industriji. Uporabljajo se pa zato, ker lahko tako hitreje ustvariš sestavne dele, katere lahko kasneje uporabiš za izgradnjo sistema.
- Avtonomni vrtalniki – avtonomni vrtalnik, bi bil vrtalnik, ki bi poiskal vijake na določeni ploskvi, in bi jih sam privil.
- Rezalnik papirja – stroj, v katerega bi dal zelo velik list papirja. Ta stroj, bi ti ta papir narezal na manjše formate, kot so A4.
- Sortiranje tablet – stroj, kateremu bi na določeno podlago razsipali ogromno tablet, ta bi z senzorji prepoznal tablete in jih ustrezno sortiral, tako bi prikrajšali delu farmacevtu.
- Prepoznavna obraza – to bi bila spojina stroja z naprednim algoritmom. Najprej bi slikal obraz neke osebe (npr. osumljenca), ga shranil v podatkovno bazo, nato pa primerjal, če je to slika osumljenca. Če slika ne bi bila prepoznana kot slika osumljenca, bi se lahko v primeru kršitve zasebnosti samodejno izbrisala.

To je samo nekaj zanimivih konceptov, katere bi lahko naredili, če nas seveda nebi prikrajševala sredstva. Vse te ideje, ki so naštet, bi se lahko uporabljale v industriji, zdravstvu, trgovinah, servisih, uradnih ustanovah ali celo pomagale policiji pri varovanju javnosti.

## 9. REZULTATI Z VREDNOTENJEM

Rezultati so bili vidni že na samem začetku. Najprej smo imeli težave z prepoznavo slike, saj algoritem ni bil izpopolnjen. Trudili smo se težavo odpraviti in nam je v roku 15minut uspelo. Ko smo izpopolnili naš algoritem za pretvorbo slike, je bil samo še čas, čakanja. Počakati smo morali, da robot »natisne« sliki in, da jo potem ljudje prepoznajo.

Hipoteze:

- Ko bodo vse komponente, vključno z programsko in strojno opremo, delovale kot en Lego sistem.
- Ko bomo uspešno pretvorili slikovno datoteko v tekstovno, katero bo naš robot zmožen prebrati.
- Ko bo program na robotu deloval glede na podatke v datoteki.
- Takrat, ko bo natisnjena slika prepoznavna ljudem, se pravi, da bo 10 ljudi prepoznalo vseh 10 fotografij.

Hipoteza 1 je bila pravilna, saj so vse komponente vključno z programsko in strojno opremo, delovale kot en Lego sistem.

Hipoteza 2 je bila pravilna, saj smo uspešno izpopolnili algoritem pretvorbe slike v binarno datoteko.

Hipoteza 3 je bila pravilna saj je program na robotu deloval glede na dane podatke, ki jih je bral iz binarne datoteke.

Hipoteza 4 je bila pravilna, saj je deset od desetih ljudi, prepoznalo sliko. Na sliki je bila miki miška.

Naš izdelek, je aktualen, saj sedaj uporabljajo podobne algoritme našemu, za prepoznavo obraza in za prepoznavo oseb na sliki/video posnetku. Datoteka je bila pretvorjena iz slikovne v binarno, katero je nato naš robot izrisal in so ljudje prepoznali kaj/kdo je na sliki.