



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Óbudai Egyetem

Bánki Donát Gépész és Biztonságtechnikai Mérnöki Kar

Fröccsöntési folyamat robottal való kiszolgálásának tervezése

**OE-BGK
2022.**

Hallgató neve:
Hallgató törzskönyvi száma:

**Nagy László
T008659/FI12904/B**



ÓBUDAI EGYETEM
ÓBUDA UNIVERSITY

Óbudai Egyetem
Bánki Donát Gépész és Biztonságtechnikai Mérnöki Kar
Anyag- és Gyártástudományi Intézet

SZAKDOLGOZAT FELADATLAP

Hallgató neve: Nagy László

Szaktervezési száma: SZD22052513064139

Törzskönyvi száma: T008659/FI12904/B

Neptun kódja: XXXXXXXXXX

Szak: gépészmérnöki

Specializáció: gépészmérnöki - CAD-CAM-CNC

A dolgozat címe: Fröccsöntési folyamat robotos kiszolgálásának tervezése

A dolgozat címe angolul: Design of the robotic support of an injection moulding process

A feladat részletezése:

1. Ismertesse a robotos gépkiszolgálás lehetőségeit és elemeit!
2. Mutassa be a kijelölt terméket és a gyártási folyamatát!
3. Tervezen robotmegfogót a termék szerszámból való eltávolítására, mutassa be a tervezési folyamatot!
4. Készítsen folyamattervet és robotprogramot a robotos kiszolgálás végrehajtására!

Intézményi konzulens neve: Dr. Mikó Balázs

A kiadott téma elévülési határideje: 2024. 05. 15.

Beadási határidő: 2022. 12. 15.

A szakdolgozat: Nem titkos.

Kiadva: Budapest, 2022. 10. 21.



Intézetigazgató

A dolgozatot beadásra alkalmasnak találom:

belső konzulens

külső konzulens



ÓBUDAI EGYETEM
OBUDA UNIVERSITY

Óbudai Egyetem

Bánki Donát Gépész és Biztonságtechnikai Mérnöki Kar

HALLGATÓI NYILATKOZAT

Alulírott hallgató kijelentem, hogy a szakdolgozat/diplomamunka saját munkám eredménye, a felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Az elkészült szakdolgozatban/diplomamunkában található eredményeket az egyetem és a feladatot kiíró intézmény saját céljára térítés nélkül felhasználhatja, a titkosításra vonatkozó esetleges megkötések mellett.

Budapest, 20.22.12.13.....

.....
hallgató aláírása

Tartalomjegyzék

1.	Bevezetés	5
2.	A robotos kiszolgálás elemei	6
2.1	Az ipari robotok kialakulása	6
2.2	Az ipari robot kiválasztásának paraméterei	7
2.3	Az ipari robotok csoportosítása.....	8
2.4	A robot effektorok.....	11
3.	Műanyag alkatrész gyártása robotos kiszolgálással.....	16
3.1	A fröccsöntés folyamata.....	16
3.2	A vákuumos megfogó egység tervezése	18
4.	A robot program készítése	27
4.1	A fröccsöntőgép és a robot közti kommunikáció.....	29
4.2	A mozgási utasítás felépítése	32
4.3	A robot program leírása	37
5.	Összefoglalás	49
6.	Summary.....	51
7.	A felhasznált irodalom.....	52
8.	Mellékletek	53

1. Bevezetés

Az elmúlt évtizedekben az automatizálás kiemelkedően fontos szempont lett a versenyképes gyártásban az egész világon. Az ipari robotok alkalmazása napról napra ugrásszerűen növekszik. Napjainkra a gyártás számos területén felváltotta a robotizáció az emberi erőforrást, és piaci jelentőségük egyre csak nő. Felhasználásuk sokrétű, a legtöbb iparágban megtalálhatók. Ezen szakdolgozat témája egy fröccsöntés folyamat robotkarral való kiszolgálása és a hozzá szükséges megfogó egység megtervezése, valamint a robot program készítése.

A szakmai gyakorlatomat a Fanuc Hungary Kft.-nél töltöttem, mint robot szervíztechnikus gyakornok, ahol bepillantást nyerhettem az ipari robotok világába. A Fanuc egy japán cégcsoport, amely automatizáláshoz kapcsolódó termékeket biztosít az egész világon legyen az ipari robot, vagy épp CNC vezérlés. Ezen felül szerszámgépeket is kínál úgy, mint marógépeket, huzalszikra forgácsolókat és fröccsöntőgépeket. A Fanuc a legnagyobb ipari robotgyártó cég az egész világon, robotjaik mind Japánban készülnek a Fuji hegy lábánál.

Szakmai gyakorlatom során rálátásom volt számtalan hazai cég gyártósorára, amik egytől egyig lenyűgöztek. Mivel már mérnöki tanulmányaim elejétől kezdve nagy érdeklődéssel fordulok az automatizálás felé, szerettem volna ezzel kapcsolatban gyakorlati tapasztalatokat is szerezni.

Szakdolgozatom témájának kiválasztásakor kiemelkedő figyelmet fordítottam arra, hogy egy számomra izgalmas feladattal foglalkozhassak és kamatoztathassam a szakmai gyakorlatomon szerzett tapasztalataimat.

Szakdolgozatom első felében a felhasznált szakirodalmak feldolgozásával foglalkozom. Elsődlegesen csoportosítom a különböző ipari robotokat kialakításuk, valamint felépítésük szerint, foglalkozom a robotok történelmével, és a robot effektorok csoportosításával is.

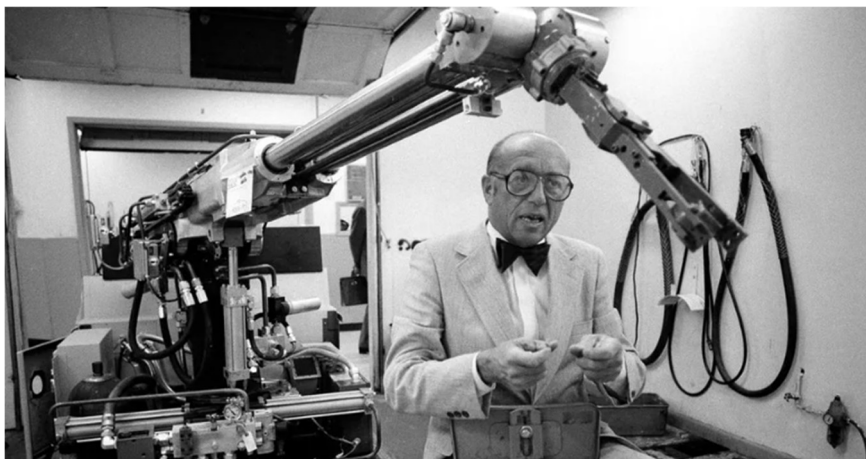
Projektem egyik fő feladata egy megfogó egység tervezése a robotkar végére, amely lehetővé teszi a robot számára, hogy kölcsönhatásba lépjen a műanyag alkatrészekkel.

Másik fő feladatomban egy robot program készítése, amely biztosítja a fröccsöntőgép kiszolgálását a robot által. Fontos szempont volt a két gép közötti megfelelő kommunikáció kialakítása.

2. A robotos kiszolgálás elemei

2.1 Az ipari robotok kialakulása

Az ipari robotokat gyakran a 21. század újításaihoz sorolják, azonban a gyökerei egészen az 1950-es évekig nyúlnak vissza. George Devol fejlesztette az első ipari robotot, egy 2 tonnás gépet, ami autonóm módon tárgyakat helyezett át egy pontról egy másik pontra, mindezt hidraulikus aktuátorokkal (2.1. ábra:). Azóta az érzékelők, számítógépes rendszerek és elektronika fejlődésével az ipari robotok képesek lettek komplex feladatok elvégzésére, mint az összeszerelés, hegesztés, festés, palettázás és ezeket mind nagy sebességgel, pontossággal és megismételhetőséggel. [1]



2.1. ábra: Az első ipari robot [1]

Először összeszerelő sorokon használták őket a korai 1960-as években. Legtöbbször hidraulikus vagy pneumatikus karok voltak, amiket elsődlegesen nehéz tárgyak mozgatására alkalmaztak. Annak ellenére, hogy kezdetlegeseek, illetve korlátozottan voltak programozhatók, mégis biztosították a várt darabszám növekedést.

A késő, 1970-es évek végére a robotok képességei még tovább bővültek, és olyan feladatokra is kiterjedtek, mint az anyagátvitel, festés, és az ívhegesztés. Elkezdték az emberre nézve veszélyes feladatokat ellátni a gyártó létesítményekben. Acél gyárakban például robotokat használtak alkatrészek és anyagok mozgatására, olyan magas hőmérsékletű környezetben, amely az emberek számára körülményes és veszélyes volt.

Az 1980-as évek közepén növekedett meg az érdeklődés az ipari robotok iránt és lettek a „jövő gépei”. A mérnökök ekkor tették le a ma ismert ipari robotok alapját úgy, hogy ötvözni kezdték a szenzorokat, illetve a kezdetleges gépi látórendszereket. [1]

2.2 Az ipari robot kiválasztásának paraméterei

Alapvetően a robot kiválasztásának kritériumait mindig az általa elvégezni kívánt feladat, illetve a környezete határozza meg. Ezen kritériumok figyelembevételével először a robot típusát kell kiválasztani. Egy adott típusú robot különböző konfigurációkban érhető el, ezzel is biztosítva az optimális megoldást az adott feladatra. Az első paraméter amit meg kell vizsgálni, az a robot maximális terhelhetősége, vagy más néven a hasznos teher, amit a katalógusokban tömegként adnak meg és a mértékegysége kilogramm. A hasznos teher az a legnagyobb tömeg, amit a robot képes a munkaterületén manipulálni. A hasznos teherbe beletartozik az adott alkatrész tömege, valamint a robotkar végén található effektor tömege is. Egy másik paraméter a robot vezérelt tengelyeinek száma. A vezérelt tengelyek száma közvetlenül összefügg a robot szabadságfokával. Egyszerű alkalmazásra elegendő lehet egy négy vezérelhető tengellyel rendelkező robot, ami palettázásra ideális választás. Azonban, ha a robotnak kis munkaterületen kell elvégezni a feladatát és a robotkarnak sokat kell csavarodnia akkor egy hat vagy akár hét szabályozható tengellyel rendelkező robot az ideális. A következő paraméter, amit figyelembe kell venni, az a robot munkavégzési távolsága. A robot maximális függőleges munka távolsága az általa elért legalacsonyabb, illetve megmagasabb pont közötti távolság. A maximális vízszintes munkatávolság pedig a robottalp középpontja és az elérhető legtávolabbi pont közötti távolság. Ezt a paramétert érdemes úgy kiválasztani, hogy legyen rajta egy biztonsági ráhagyás, ami a robot élettartamát hosszabbítja meg. Fontos paraméter az ismétlési pontosság, ami a robot azon képessége, hogy milyen pontosan tud a robot ugyanabba a pontba visszatérni. Jellemzően ez az érték egy tartomány, amit a térben kell elképzelni és mindig az adott alkalmazáshoz kell megválasztani. Egy meglehetősen durva alkalmazáshoz nem feltétlen szükséges a magas ismétlési pontosság. A robot tömege szintén nem elhanyagolható. Egy gyártócella tervezésénél, ha a robot egy sínen vagy valamilyen emelvényen, esetleg a levegőben valamilyen konzolhoz erősítve van, akkor a tömege elengedhetetlen a megfelelő méretezés érdekében.

A robot maximális sebessége a gyártók számára nagyon fontos paraméter, ugyanis ez az érték nagyban befolyásolja az elérhető ciklusidőt. A sebességet többnyire tengelyenként

határozza meg a gyártó %/s-ban. Fontos azonban, hogy a sebesség mellett a robot által végzett mozgások milyensége is nagyban befolyásolja a ciklusidőt.

Alapvetően a legtöbb ipari robot összes tengelye rendelkezik fékrendszerrel. Ez alól vannak kivételek, mint például a scara robot, aminek kialakításából következően erre nincs szükség. [2]

Különleges esetekben a robotot a falra vagy fejjel lefelé kell elhelyezni a gyártó cellában, ilyenkor a robot talpzata úgy van méretezve, hogy eltudja viselni a rá ható terheléseket az elhelyezésből következően. Úgynevezett tisztateres környezetben a robotnak is teljesítenie kell a rá vonatkozó előírásokat. Ezek főleg az élelmiszeriparban, gyógyszeriparban vagy elektronikaiparban a jellemzőek a robotok pedig ennek megfelelően vannak konfigurálva.

2.3 Az ipari robotok csoportosítása

Az általánosságban vett robotokat jellemzően az alkalmazási terület vagy az intelligenciaszint szerint csoportosítják. Alkalmazási területek szerint vannak ipari, tengeralatti kutató, űrkutató illetve gyógyászati robotok, de mára megjelentek a háztartásokban is. A következőkben kizárólag az ipari robotokkal foglalkozom. Az alkalmazást tekintve az ipari robotokat további két alkalmazási területre lehet bontani. Vannak a valamilyen technológiai műveletet ellátó ipari robotok és vannak az anyagmozgást megvalósító ipari robotok. A robotok által végzett technológiai műveletek között főként a hegesztés és a festés az értejedt, de ezek mellett alkalmazzák csiszolásra, ragasztásra, sorjábanításra és ezeken kívül még számos műveletre képesek a robotok. Anyagmozgatás során elsődlegesen valamilyen gépet szolgálnak ki, legyen az fröccsöntőgép, forgácsoló szerszámgép vagy épp egy hőkezelő berendezés. Ezeken felül fontos szerepük van a gépek, gépsorok közötti anyagáramlásban és a palettázásban is. A másik csoportosítási lehetőséget a robot intelligenciaszintje adja. A robot intelligenciáját a vezérlés, valamint a vezérlőegység feldolgozó képessége biztosítja. A mai modern ipari robotok már összetett feladatokat képesek ellátni, mindezt a környezet jellemzőinek észlelésére alkalmas érzékelők és a nagy teljesítményű számítógépek segítségével. [3]

Számos gyártó használ ipari robotot, hogy automatizálja a gyártási folyamatokat, növelje a darabszámot, fokozza a munkások biztonságát és ezekkel együtt csökkentse a költségeket. Mivel az ipari robotok egyre elterjedtebbé váltak, az iparban megnövekedett az igény a különféle típusú robotok iránt, melyek megfelelnek az adott alkalmazásnak és iparágnak. [4]

Mechanikai felépítésük alapján az ipari robotokat öt csoportba lehet sorolni: csuklós robotok, lineáris vagy portál robotok, scara robotok, delta vagy pók robotok, és kollaboratív robotok. A mechanikai felépítésükön kívül még lehet őket csoportosítani a mozgás vezérlésük vagy a fizikai jellemzőik alapján is.

A csuklós robot az egyik legelterjedtebb ipari robot (2.2. ábra:). Mechanikai felépítésében az emberi kart mintázza. Ez a kar egy alaphoz van kötve, amihez képest el tud fordulni. Három, négy, - vagy akár tíz csuklóval is rendelkezhet és minden egyes csukló egy különálló szabadság fokot biztosít. A csuklók lehetnek párhuzamosak, illetve ortogonálisak. A legtöbb csuklós robot 6 szabadság fokkal rendelkezik, ugyanis ez biztosítja azt a sokoldalúságot, ami miatt a legtöbb helyen alkalmazzák. Habár a legtöbb feladatot el tudja látni -köszönhetően a rugalmasságának-, nem mindig egy csuklós robot az optimális megoldás minden feladatra. A csuklós robotok nagy sebességre képesek, akár 4 m/s-ra is. Az elfoglalt térükhöz képest nagy munkatérrel rendelkeznek és csuklóik számának függvényében szűk helyen is elférnek vagy szűk helyekre is benyúlhatnak. Összetettségükből következően saját robotvezérlőt igényelnek, valamint a programozásukhoz oktatás szükséges. Az ipar számos területén alkalmazzák, mint például alkatrész illetve anyag kezelés, ívhegesztés, festés, gépkiszolgálás, össze,- és rész összeszerelések során is.

A portál robotok olyan ipari robotok, amelyek 3 lineáris tengelyen mozognak (2.3. ábra:). A Descartes-i koordináta rendszert használják, ami azt is jelenti, hogy egyszerre csak az egyik tengelyen képesek mozogni. Képtelenek bonyolultabb műveleteket végrehajtani, ennek ellenére gyakran választják őket a gyártók az egyszerű működtetés és a kedvező vételár miatt. Egyszerű felépítésük folytán kezdetleges programozási készségeket igényelnek. Nagy ismétlési pontosságra képesek. Kialakításuk eredményeképp nagy a teherviselésük nagy pontosság mellett, valamint a testreszabhatóságuk is kedvező a vételáruk mellett. Telepítésük összetett, nagy helyet

foglalnak el a munkaterükhöz képest. Gépkezelésre, alkatrész kezelésre alkalmazzák a legtöbb esetben.



2.2. ábra: Fanuc
R2000iC csuklós robot
[5]



2.3. ábra: Portál robot [6]

A scara robotoknak felépítésüknek köszönhetően sajátos felhasználásuk van (2.4. ábra:). Nem olyan sokoldalúak, mint például a csuklós robotok, mindazonáltal kiválóak azokhoz a feladatokhoz, amikhez tervezve lettek. Fánk alakú munkatérrel rendelkeznek, amit 2 párhuzamos csuklóval és egy függőlegesen elhelyezett tengellyel biztosítanak. A függőleges tengely a tengelyvonal körül forgómozgásra, illetve lineáris mozgásra képes. Nagy sebességre képesek kiváló ismétlési pontossággal, valamint elfoglalt területükhöz képest nagy munkatérrel rendelkeznek. Saját robotvezérlőt igényelnek, ezen felül kialakításuk miatt körülményesebb a programozásuk. Teherviselésük nem túl nagy és csak sík felületű munkatérre alkalmazhatók. Össze, - illetve részösszeszerelésekre, valamint palettázásra alkalmazzák főként.[7]

A Delta robotokat gyakran párhuzamos kinematikájú robotoknak is nevezik, mivel párhuzamos csuklós összeköttetésekből állnak, amelyek közös alappal vannak összekötve (2.5. ábra:). Kupola alakú munkatérrel rendelkeznek és gyakran a munkatér felett helyezkednek el. Sajátos felépítésüknek köszönhetően nagyon gyorsan valósítanak meg bonyolult mozgásokat. Egyszerre képesek összetett pályákat nagy sebességgel megvalósítani kiváló ismétlési pontossággal. Saját robot vezérlőt igényelnek, ezen felül komplex felépítésük miatt programozásuk nehézkes. Főként csomagolásra, illetve palettázásra alkalmazzák őket gyorsaságuk miatt.[7]

A kollaboratív robotok felépítésükben nem különböznek a csuklós robotoktól (2.6. ábra:). Ugyan úgy az emberi kart mintázzák, azonban az a nagy különbség, hogy az emberrel közös munkatérben dolgozhatnak, köszönhetően a beépített erőmérőknek, amik a modernkori kollaboratív robotokban az összes csuklónál megtalálhatók. Nincs szükség biztonsági kerítésekre mert a robot érzékeli, ha hozzáérnek és azonnal megáll. Telepítésük egyszerű és gyors. Kialakításukból következően emberrel közös térben tud dolgozni. A nem kollaboratív csuklós robotoknál számottevően lassabbak és a teherbírásuk is alacsonyabb. Összetettségükből következően saját robotvezérlőt igényelnek, valamint a programozásukhoz oktatás szükséges. [7]



2.4. ábra: Fanuc SR-6iA Scara robot [5]



2.5. ábra: Fanuc M-1iA/1H delta robot [5]

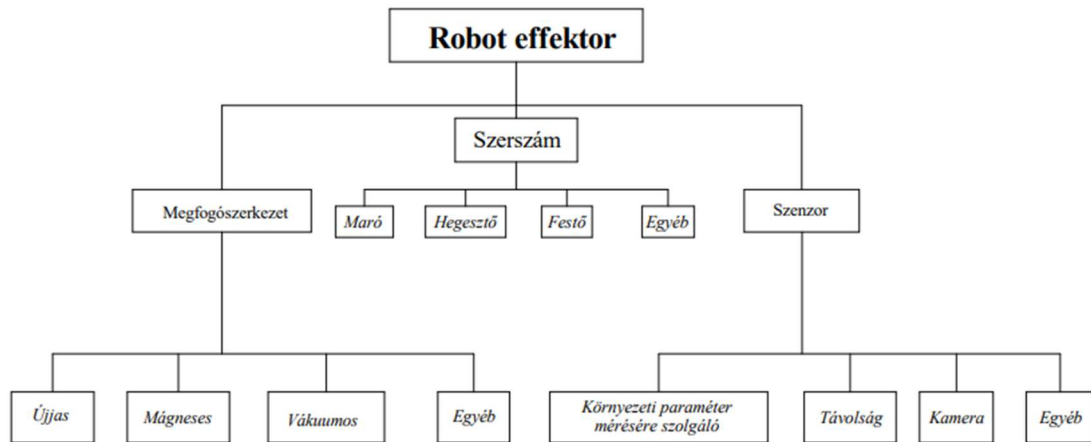


2.6. ábra: Fanuc CRX-20iA/L kollaboratív robot [5]

2.4 A robot effektorok

Az effektorok olyan készülékek, amelyek a robot csuklójához kapcsolódnak, lehetővé téve a robot számára, hogy kölcsönhatásba lépjen a környezetével. A legtöbb robot effektor mechanikus vagy elektromechanikus és megfogó szerkezetként, szerszámként vagy érzékelőként funkcionál. Ezek a legegyszerűbb kétujjas megfogótól az összetett érzékelő rendszerekig terjednek. [8]

Az effektorokat három fő csoportba lehet sorolni: megfogó szerkezetek, szerszámok és érzékelők (2.7. ábra:).



2.7. ábra: Az effektorok csoportosítása

A megfogó szerkezetek a leggyakoribb effektorok. Elsődlegesen alkatrészek megfogására, mozgatására valamit helyezésére alkalmazzák, mindezeket nagy pontossággal. Lehetővé teszik az alkatrészek, anyagok felvételét és kezelését. Anyagmozgatáshoz, összeszereléshez, illetve gépkiszolgáláshoz ideálisak. Az ipar számos területén veszik igénybe. [8]

A mechanikus megfogók a leggyakrabban alkalmazott megfogó szerkezetek. Nagy választékban kaphatók mind a méretet, tömeget továbbá megfogási erőt tekintve.

Kiválasztásukkor több szempontot vehetünk figyelembe. Egyik szempont a hasznos teherbírásuk, mely mindig egy maximális érték és a megfogott alkatrész tömege ennél nem lehet nagyobb. Fontos szempont lehet még a fogásszélességi tartomány, mely megmondja, hogy maximum és minimum mekkora alkatrészt képes megfogni az adott megfogó. Ezeken kívül számos jellemzőjük van, mint például a fogóerő, fogási sebesség, fogási idő és ismétlési pontosság. [8]

A legegyszerűbb mechanikus megfogó egy két pofás megfogó (2.8. ábra:), aminek a pofái egyszerűen testre szabhatók az alkatrészek pontos, illetve szilárd megfogása érdekében. Belső, illetve külső átmérőn is képes az alkatrész biztos megfogására. Elérhetők még 3,4 de akár 5 ujjas konfigurációban is, ennél többre is van lehetőség, de az ritkán szükséges. Lehetnek pneumatikus, hidraulikus, valamint elektromos működtetésűek ennek köszönhetően a megfogási erő nagysága szabályozható. A mechanikus megfogók ritkán alkalmasak lapos alkatrészek megfogására, valamint nyomot hagyhatnak a kényesebb alkatrészekben.

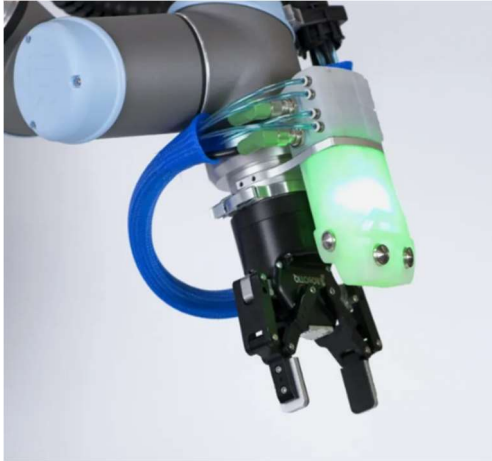
Léteznek adaptív mechanikus megfogók, amik képesek különféle formájú alkatrészek megfogására, így nem szükséges megfogót cserélni egy-egy termék átállásnál. Kezdetleges kialakításuk miatt még nem terjedtek el annyira az iparban, valamint sok esetben egy egyszerű kétpofás megfogó olcsóbb és biztosítani tudja a tökéletes megfogást.

Számottevően sok gyártó tervezett egyedi megfogót a robotjaira, melynek megvannak az előnyei, továbbá a hátrányai is. Egy egyedi megfogó segítségével akár több munkadarabot is manipulálni képes a robot, csökkentve ezzel a ciklus időt és egyszerűsítve a robot mozgását. Arra is van lehetőség egyedi megfogó esetén, hogy többféle alkatrészt legyen képes megfogni a robot, például egy csomagolásnál képes legyen a raklapot is, valamint a csomagolni kívánt tárgyat is manipulálni. Hátrányuk a magas tervezési költség, illetve az idő, amely az alkalmazástól függően kifizetődik.

A vákuumos megfogók szintén gyakran használt megfogó szerkezetek (2.9. ábra:). A levegő áramlását használják fel negatív nyomású tér létrehozására a tapadókorong belsejében. A szívóerő nagysága szabályozható az áramoltatott levegő mennyiségével. A vákuumos megfogók sokoldalúsága azonban nem a szívóerőből, hanem a tapadókorongok sokféleségéből és rugalmasságából fakad. Ideálisak olyan feladatokra, ahol a nagy ismétlésipontosság nem fontos vagy máshogy van megoldva. Általában nem porózus felületű alkatrészeket, anyagokat mozgatnak vele. Ezek lehetnek könnyű műanyag alkatrészek, valamint nehéz acéllemezek is. [8]

A vákuumos megfogókat a tartó erőt nézve jól kiegészítik a mágneses megfogók. Nagyrészt állandó vagy gerjesztett mágnes biztosítja a tartó erőt. Az állandó mágnes esetében a munkadarabot tartó fej felépítése egyszerű azonban a letevéshez további elemek szükségesek. Az állandó mágnes csak akkor engedi el a munkadarabot, ha az rögzítve van a letevés helyén, vagy a megfogó szerkezeten van egy kiegészítő mechanikai elem, ami kellő mértékben eltávolítja a munkadarabot a mágnestől. Problémát jelent az állandó mágnes által a megfogóra feltapadt fémes szennyeződés. Ezek miatt a gerjesztett mágnesű megfogók terjedtek el. Csak ferromágneses anyaghoz lehet alkalmazni, a felületegységre eső erők nagyok, gyors megfogást biztosítanak, egyszerű felépítésűek és az élettartamuk viszonylag hosszú. A mágneses

megfogószerkezetek elsősorban acéllemezek kezelésére alkalmasak. Tartóerejük a távolság növekedésével drasztikusan csökken. [3]



2.8. ábra: Példa két pofás megfogóra [9]



2.9. ábra: Példa vákuumos megfogóra [9]

A következő csoport az effektorokon belül a szerszámok. Ezek olyan eszközök, amiket meghatározott gyártási folyamatoknál használnak legyen az sorjázás, hegesztés, festés, csiszolás vagy épp ragasztás.

Sorjázás során a robot egy maró szerszám segítségével eltávolítja a sorját egy alkatrésztől, például öntött acél elemről. A szerszámba épített erőmérő rendszer segítségével a különböző egyenletlenségek kiküszöbölhetők, ugyanis a robot adaptívan alakítja a szerszám pályáját. A hegesztés, továbbá a pont hegesztés nagyon fontos része az ipari robotok felhasználásának. Az emberi kéz számára fárasztó és monoton feladat, valamint nem is képes minden esetben tökéletes varratokra. Ezzel szemben egy csuklós robot rendelkezik a megfelelő pontossággal és rugalmassággal, ami a hegesztéshez kell. Az autópárházban használják leggyakrabban. A különböző ipari robot gyártók külön a hegesztéshez, illetve ponthegeztéshez optimális kialakítású csuklós robotokat terveznek.

Az utolsó csoportja az effektoroknak az olyan eszközök, amiket a termékek ellenőrzésére vagy a robot irányítására alkalmaznak. Szenzorokat használnak a nyomás, erő, hőmérséklet vagy más, -a gyártás során fontos- paraméter változásának mérésére. Ellenőrző eszközként is használhatók. Az egyes folyamatok minőségének nyomon követésével csökkenthető a selejt mennyisége. A különböző kamerarendszerek lehetővé

teszik a véletlenszerűen elhelyezett tárgyak kezelését a robot számára. A 3D-s kamera rendszerek mélységinformációkat is képesek biztosítani. A szenzorokat ritkán alkalmazzák önmagukban egy roboton, többnyire egy másik effektorral ötvözve optimális a felhasználásuk.

3. A műanyag alkatrész gyártása robotos kiszolgálással

Szakdolgozatomban egy műanyag alkatrész fröccsöntési folyamatának robotos kiszolgálása a megoldandó feladat. A megfelelő fröccsöntőgép kiválasztása után egy vákuumos megfogó egységet tervezek, az adott műanyag alkatrész biztos kivételére.

3.1 A fröccsöntés folyamata

A fröccsöntés egy szakaszos művelet. Egy fröccsöntési ciklus elején a gépbe épített záróegység összezárja az alakadó szerszámegységet, majd az úgynevezett plasztikáló egység -amely egy hengerből és egy csigából áll- a megolvasztott műanyagot a hűtéssel ellátott szerszámba juttatja. A bejutatott olvadt műanyag felveszi a szerszámüreg alakját, majd lehűl és megszilárdul. Ezután a szerszám nyit, a késztermék eltávolításáról pedig a kidobó rendszer, vagy egy robot gondoskodik. Amíg a szerszámba juttatott olvadt műanyag hűl, a plasztikáló rendszer előállítja a következő ciklushoz a kellő mennyiségű olvadt műanyagot. [10]

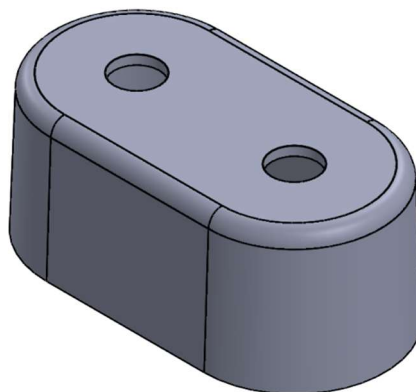
Az alakadó szerszám egység két félből áll. Az egyik fele a fröccsöntőgép mozgó lapjára a másik fele pedig az álló lapjára van felerősítve. A mozgó lap a minél rövidebb ciklusidő elérése érdekében a lehető leggyorsabban zár, azonban az álló laptól egy bizonyos távolságra csökken a sebessége, ezzel védve a szerszámot. Miután ezzel a kis sebességgel összezár a szerszám, felépül a záróerő, ami meggátolja a két fél szétfeszülését. [10]

Miután a szerszám összezárt, a plasztikáló rendszer előre mozog, hogy a végén található kis átmérőjű nyílás -az úgynevezett fűvóka- elérje az álló felfogó lapot. A plasztikáló rendszerben megolvadt műanyagot a fűvókán keresztül a szerszámüregbe nyomja a csiga, mely során a műanyag felgyorsul, ezzel elérve az optimális szerszámkitöltést. A fűvóka előre mozgását biztosító hidraulikus, vagy elektromechanikus meghajtásnak ellen kell állnia a fröccsöntés közben keletkező magas nyomásnak. [10]

A fröccsöntött alkatrészek gyártása során komoly probléma, hogy a műanyagok fajtérfogata nagy mértékben változik a hőmérséklet ingadozásával. Ez azt jelenti, hogy a műanyag megolvadása során fajtérfogata nő, lehűlésekor csökken. Ezt kompenzálva a befröccsöntés után utónyomásra van szükség, ami eredményeként méretpontos alkatrészek keletkeznek. [10]

Az utánnymás után veszi kezdetét a hűlés, aminek idejét úgy kell megválasztani, hogy a műanyag alkatrész kellően megszilárduljon, hogy a szerszámból való kivétel ne tudja deformálni azt. A hűlés ideje alatt a plasztikáló rendszer felkészül a következő ciklusra, valamint, ha szükséges, a fúvóka hátra mozdul. A hűlési időt döntően az alkatrész falvastagsága határozza meg. Az utánnymást követően a csiga forogni kezd, amivel a gyűjtő tartályban található műanyag granulátumot a szerszám felé szállítja. A fellépő súrlódás és a beépített fűtőtestek miatt a granulátum megolvad. Ezt nevezzük plasztikálási folyamatnak. Ha a hűlési idő túl hosszú, a fúvókát hátra szükséges járatni, nehogy a benne lévő olvadt műanyag megdermedjen. Ha az alkatrész megfelelően lehűlt -azaz a hűlési ideje lejárt- a szerszám nyit, kezdetben lassan, -ezzel is kímélve a szerszámot-, majd a ciklus idő redukálása érdekében gyorsan, és a teljes nyitás előtt szintén lassan. Ezután a szerszám kidobó rendszere, vagy egy robot, eltávolítja a kész munkadarabot a formaadó betétekből és elindul újra a ciklus. [10]

A fröccsönteni kívánt munkadarab egy pohár szerű alkatész, befoglaló méretei 75x40x30 milliméter, anyaga: polietilén-tereftalát, vagy ahogy ismertebb pet (3.1. ábra:). Ez egy szívós, kemény anyag, aminek jó a mérettartása, valamint kiváló a folyóképessége, ami fröccsöntésnél előnyös, nehezen gyűjthető meg, de ha meg is gyűjtják, elalszik. Erős savnak vagy lúgnak nem képes ellenállni. A fröccsöntő szerszám négy fészkes, hidegsatornás és temperált, befoglaló méretei: 550x280x260 milliméter.



3.1. ábra: A műanyag alkatrész

A fröccsöntőgép kiválasztása során a fő szempont, hogy biztosítani tudja a megfelelő záróerőt. A szükséges záróerőt a fröccsönteni kívánt alkatrész vetített területéből, a

fröccsnyomásból és a fészekszámból számítottam ki. A műanyag alkatrész vetített területe 2500 mm^2 .

$$F = A * P * t \text{ [10]}$$

, ahol az F a szükséges záróerő

, az A az alkatrész vetített területe

, a P a fröccsöntéskor fellépő belső nyomás

, a t pedig a szerszám fészkeinek száma

$$F = 2500 \text{ mm}^2 * 100 \frac{\text{N}}{\text{mm}^2} * 4 \text{ db} = 1.000.000 \text{ N} = 100 \text{ t}$$

Azaz a választott fröccsöntőgépnek legalább 100 tonnás záróerővel kell rendelkeznie.

Fontos szempont a Fröccsöntőgép meghajtásának módja. Vannak hidraulikus és elektromotoros meghajtású berendezések. Jelen esetben egy villamos hajtású az optimális választás, ugyanis nem tartalmaz olajat míg a hidraulikus társaik igen. Választásom a Fanuc Roboshot α -s130iB típusú fröccsöntőgépre esett, mely 130 tonnás záró erővel rendelkezik és elektromos meghajtású (3.2. ábra:). Munkaterének szélessége 400 milliméter.



3.2. ábra: A Fanuc Roboshot α -s130iB típusú fröccsöntőgép [5]

3.2 A vákuumos megfogó egység tervezése

A vákuumos megfogó egység tervezése során a fröccsöntőszerszámban található formaadó betétek méreteit, valamint a választott fröccsöntőgép munkaterének nagyságát használtam fel kiindulási alapnak. Tervezéskor fontos szempont volt a minél alacsonyabb tömeg elérése, az oldható kötések használata az állíthatóság érdekében, valamint a szűk munkatér miatt a kis méret.

Amit először ki kellett választanom, az a vákuumos tappancs anyaga. A Festo katalógusa alapján lehet: fluorkaucsuk, nitril gumi, polioximetilén ésszilikon és poliuretán. Fő szempontként a hőállóságot vettem alapul, ami alapján a fluorkaucsukot választottam. Az alkatrész méreteit figyelembe véve 20 milliméteres köralakú tappancsot véltem a legjobbnak. Névleges üzemi nyomása -0,7 bár. Tartóereje névleges üzemi nyomáson 16N, keresztirányú tartó ereje pedig 15N, ami elegendő az alkatrész tömegét nézve. Csatlakozása a tartóhoz egy M6-os külső meneten történik. A kör alakú szívótappancs szükséges átmérője a következő egyenletből becsülhető meg, feltételezve a lassú függőleges irányú mozgást.

$$d = 11.3 \sqrt{\frac{m * S}{\sigma_u * z * \mu}} = [mm] \quad [11]$$

, ahol a d a szükséges szívótappancs átmérő [mm]

az m a munkadarab tömege [kg]

az S a biztonsági tényező

a σ_u az alkalmazott nyomás a szívótappancsban [bar]

a z a szívó tappancsok száma

a μ pedig a súrlódási együttható

$$d = 11.3 \sqrt{\frac{0,02 \text{ kg} * 3}{0,9 \text{ bar} * 1 * 0,5}} = 4,2 \text{ mm}$$

Annak ellenére, hogy a gyártó megadja a szívótappancs emelési erejét, a biztos méretezés érdekében, a következő egyenlettel kiszámolható a tappancs statikus tartóereje:

$$F = (\sigma_o - \sigma_u) * A = [N] \quad [11]$$

, ahol a d a szükséges szívótappancs átmérő [mm]

az F a szívótappancs statikus tartóereje [N]

az σ_o a környezeti légnyomás $\left[\frac{kN}{m^2}\right]$

a σ_u az alkalmazott nyomás a szívótappancsban $\left[\frac{kN}{m^2}\right]$

az A a hasznos terület a szívótappancs és a munkadarab között $[m^2]$

$$F = \left(100 \frac{kN}{m^2} - 10 \frac{kN}{m^2}\right) * \frac{0,02^2 m * \pi}{4} = 28 N$$

A számított statikus tartó ereje a szívótappancsnak nagyobb, mint a gyártó által közölt érték. Amint a tappancs sikeresen megfogta a munkadarabot, a kezelő szerv -jelen esetben a robot- megkezdí a munkadarab manipulálását, ami miatt fellépnek a mozgatásból származó erők. A legtöbb szívótappancs típustól független deformálódik működés közben. A mozgásból származó erők, illetve a tappancs alakváltozása redukálja az emelő erő nagyságát [11]. A redukált emelőerő nagyságát a következő egyenlet segítségével számoltam ki:

$$F = (\sigma_o - \sigma_u) * A * n * \eta * z * \frac{1}{S} - m(g + a) = [N] \quad [11]$$

, ahol az F a szívótappancs statikus tartóereje $[N]$

az σ_o a környezeti légnyomás $\left[\frac{kN}{m^2}\right]$

a σ_u az alkalmazott nyomás a szívótappancsban $\left[\frac{kN}{m^2}\right]$

az A a hasznos terület a szívótappancs és a munkadarab között $[m^2]$

az n az alakváltozási együttható

az η a rendszer hatékonysága

a z a szívótappancsok száma

az S a biztonsági tényező

a g a gravitációs gyorsulás $\left[\frac{m}{s^2}\right]$

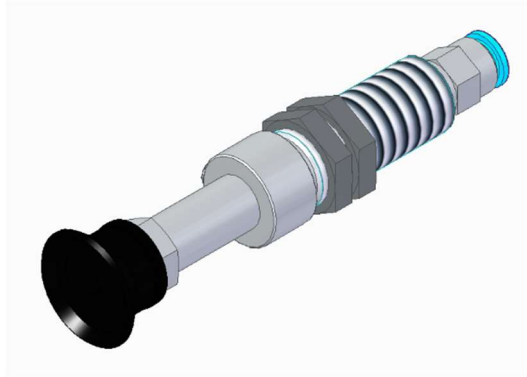
az a a kezelő szerv maximális gyorsulása $\left[\frac{m}{s^2}\right]$

$$F = \left(100 \frac{kN}{m^2} - 10 \frac{kN}{m^2}\right) * \frac{0,02^2 m * \pi}{4} * 0,6 * 0,9 * 1 * \frac{1}{3} - 0,02 \left(9,817 \left[\frac{m}{s^2}\right] + 1,25 \left[\frac{m}{s^2}\right]\right) = 4,8 N$$

A redukált erő értéke pozitív maradt az alakváltozás, illetve a mozgás miatt fellépő erők figyelembevételével is, így a választott szívótappancs megfelelő.

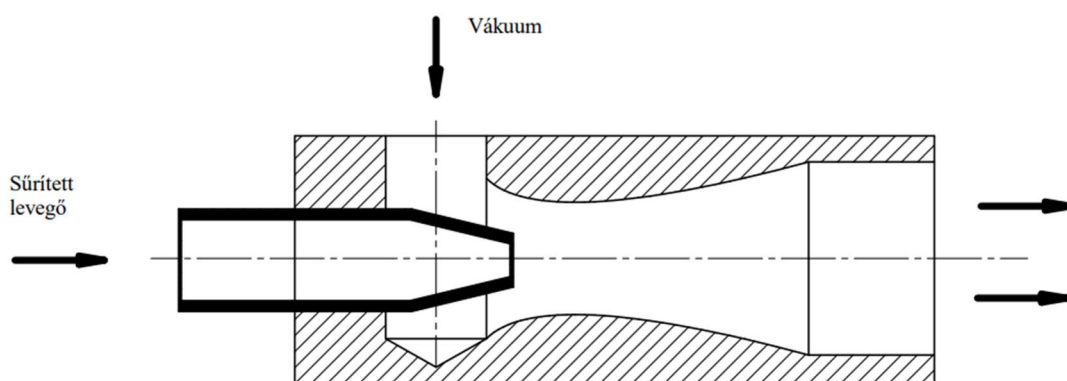
A vákuum tappancshoz kellett választanom egy megfelelő szívókorong tartót, amihez kapcsolódik és ami rendelkezik levegő csatlakozóval. A levegő csatlakozó lehet a tartó tetején, illetve az oldalán. Jelen esetben ideálisabb, ha a tartó tetején van a csatlakozó,

így ezt választottam. Rendelkezhetnek beépített magasság kiegyenlítő rendszerrel, ami segít az alkatrész kíméletes kezelésében. A robotos kiszolgálás során kedvező ennek a rendszernek a megléte. Ezeket figyelembe véve a tartó felső csatlakozású, magasság kiegyenlítéses, a tappancs pedig fluorkaucsuk anyagú, kör alakú, 20 milliméter átmérőjű, ezek alkotják a vákuumos alegységet (3.3. ábra:).



3.3. ábra: A vákuumos alegység [12]

A vákuumos alegység összeállítása után a vákuum előállításának módját kellett megválasztanom. Szempont volt a minél egyszerűbb működés, kis méret és az egyszerű beépítés. Ezek alapján a vákuum ejektor mellett döntöttem. A vákuum ejektor a Venturi-elv alapján működik, ebből következik, hogy csak sűrített levegő kell a működéséhez, aminek a rendszere a legtöbb ipari létesítményben kiépített (3.4. ábra:).

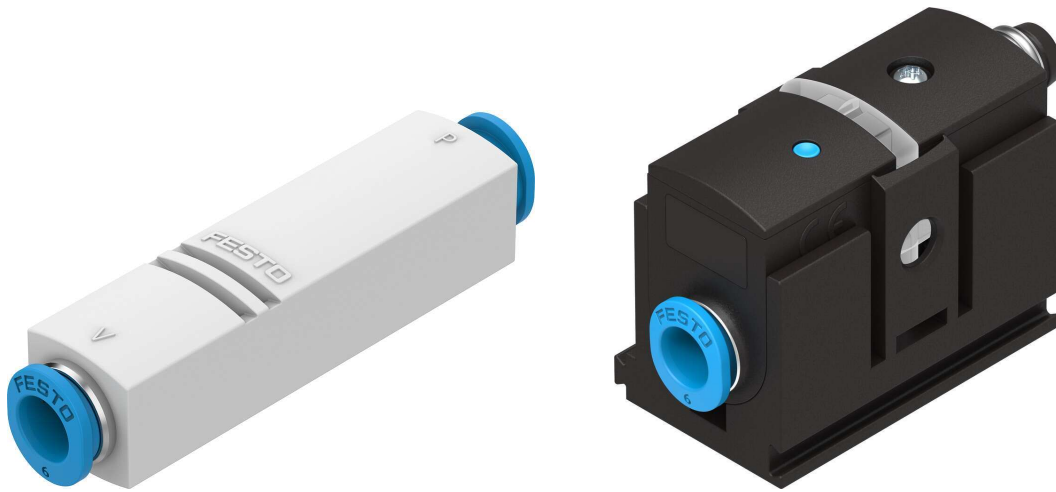


3.4. ábra: Venturi csatornás vákuum ejektor [3]

Egyszerűen és gyorsan valósítja meg a vákuumot mindezt kedvező áron és karbantartás mentesen. Választásom a Festo VN-07-M-I3-PQ2-VQ típusú vákuum ejektorra esett, ami soros bekötésű, kis méretű, alacsony tömegű és 5,8 bar nyomás mellett 86%

maximális vákuumra képes. Kérdés volt, hogy az alkalmazást tekintve hány vákuum ejektorra van szükség (3.5. ábra:). Mivel a fröccsöntőszerszám négy fészkes, így négy vákuumos alegységet kell használnom. A négy vákuumos alegységet egy nagyobb teljesítményű vákuum injektor is képes lenne ellátni. Ezzel csökkennének a költségek, egyszerűbb lenne az összeszerelés. Azonban, ha a négy tappancs közül valamilyen okból az egyik nem tapad a munkadarabra, akkor a másik három sem fog, ugyanis nem alakul ki a vákuum a nyitott rendszer miatt, vagy ha ki is alakul, akkor sem lesz biztos a megfogás. Így a négy vákuumos alegység mindegyikét külön vákuum ejektorral láttam el, mégpedig sorba kötve.

Egy fröccsöntési folyamat során adódhat olyan probléma, hogy a fröccsöntött alkatrész valamilyen okból nem távozik a szerszámból. Ezt a fröccsöntőgép nem tudja figyelni, így újra összezárja a szerszámot, majd megpróbálja bejuttatni az olvadt műanyagot, ami nem szerencsés, hiszen akár meghibásodáshoz is vezethet. Ha egy operátor szolgálja ki a fröccsöntőgépet akkor ő látja a bent ragadt munkadarabot és le tudja állítani a gépet ezzel elkerülve az esetleges meghibásodást. Azonban a robotos kiszolgáláskor erre nincs lehetőség, mert nem áll senki a gép mellett. Ezt kiküszöbölendő, elláttam a vákuum rendszert egy úgynevezett nyomásérzékelővel. A nyomásérzékelő képes az adott levegőrendszer nyomását mérni egy bizonyos tartományon belül. Konfigurációtól függően több előre betanított nyomásérték mérésekor elektromos jelet ad ki magából, amit a különböző vezérlőegységek feltudnak használni. Jelen esetben az imént felvázolt probléma megoldására elég egy előre betanítható nyomásértékkel rendelkező nyomásérzékelő, ami akkor ad jelet, ha felépült a vákuum. Ezért egy előre betanítható nyomásértékkel rendelkező nyomásmérőt alkalmazok, minden vákuumos alegységhez egyet. Az általam választott Festo SDE5-B2-O1-Q6-P-M8 típusú nyomásérzékelő, amely egy előre betanítható nyomás értékkel rendelkezik, nyomásmérési tartományának kezdőértéke mínusz egy bar, végértéke egy bar, soros bekötésű és hat milliméteres gyorscsatlakozóval rendelkezik (3.6. ábra:). A szenzort a vákuumos alegység és a vákuum ejektor közé építettem be, sorosan kötve.



3.5. ábra: A választott vákuum ejektor [12]

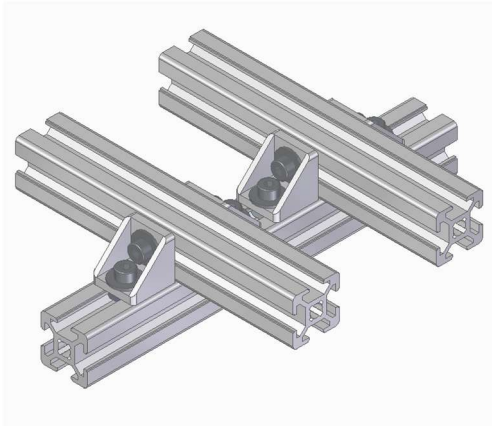
3.6. ábra: A választott nyomásérzékelő [12]

Miután a megfogó szerkezet vákuumos alegységei megfogják a négy munkadarabot, a robot kiveszi a fröccsöntőgépből, majd leteszi valahova. Tehát a vákuumra csak akkor van szükség, amikor a robotnak manipulálnia kell az alkatrészeket. Ezt az mágnes kapcsolóval oldom meg, ami lényegében egy elektromosan vezérelhető útváltó szelep. Adott lenne egy egyszerű 2/2-es útváltó, ami alapesetben zárt, ha jelet kap, akkor pedig nyitott. Azonban, ebben az esetben a vákuum nem szűnne meg, a tappancsokban ugyanis a külső környezetből nem tud beáramlani a levegő a rendszerbe. Ezért kell egy 3/2-es útváltó szelep, ami ha jelet kap, biztosítja a táplevegőt a megfogó egység számára, és miután nincs szükség a levegőre, elzárja a táplevegőt és biztosítja a környezeti levegőt a vákuum megszüntetésére. Választásom a Festo MHE3-MS1H-3/2G-QS-6 típusú mágnesszelepre esett, ami egy 3/2 útszelep, 6 milliméteres levegő csatlakozókkal és alapesetben zárt (3.7. ábra:).

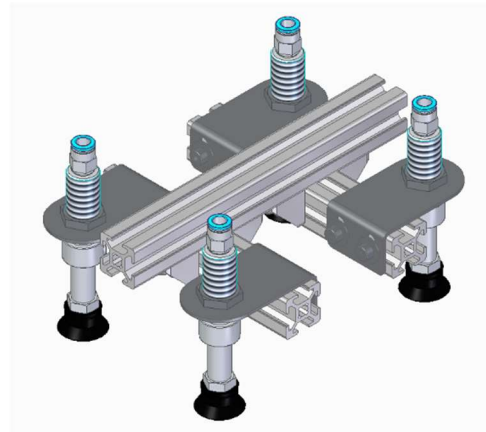


3.7. ábra: A választott mágnesszelep [12]

Következő lépésben terveznem kellett egy keretet, amihez rögzíteni lehet a szívókorong tartókat. A fröccsöntő szerszám formaadó lapjaiból indultam ki, mind a keret alakját, mind a méreteit tekintve. A keret alakját tekintve „X” és „H” alakú lehet. A „H” alak egyszerűbb állítást biztosít, valamint egyszerűbb az összeszerelése is. Az „X” alak tömege pedig kisebb. Mérlegelve tulajdonságaikat végül „H” alakú keretet terveztem, amihez a Bosch Rexroth alumínium profiljait használtam. Ezek a profilok anyaguk miatt könnyűek, ennek ellenére biztosítják a megfelelő merevséget. A kis munkatér miatt 20x20 milliméter keresztmetszetű profilokat használtam. A „H” alak kialakításához két darab 120 milliméter, illetve egy darab 160 milliméter hosszú elemet használtam, egymáshoz való rögzítésüket pedig kalapácsfejű anyákkal, sarok összekötő elemekkel, illetve belső kulcsnyílású csavarokkal oldottam meg (3.8. ábra:). A sarok összekötő elemeket úgy helyeztem el, hogy az esetleges állításokhoz ne keljen leszerelni a megfogót a robotkarról. A szívókorong tartókat a hozzájuk tartozó felfogó szögelemek segítségével rögzítettem a kialakított „H” alakhoz (3.9. ábra:).



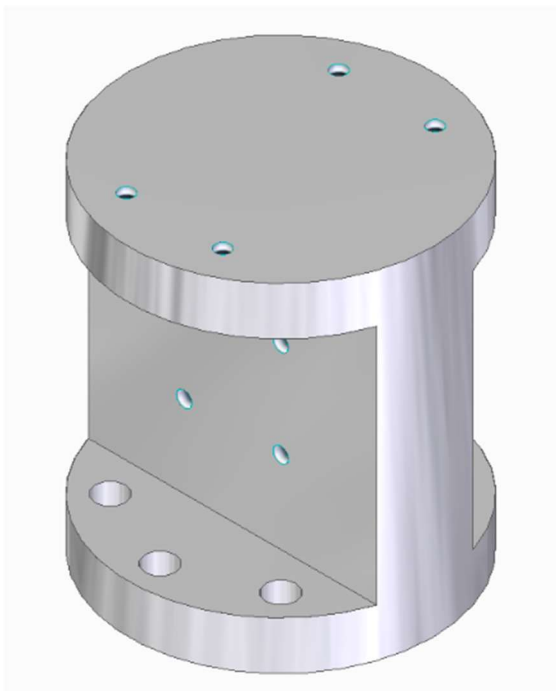
3.8. ábra: A kialakított "H" alak



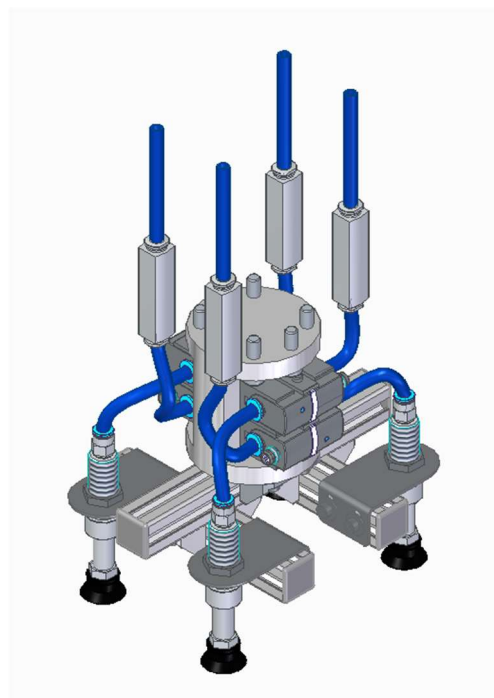
3.9. ábra: A Vákuumos tappancsok rögzítése

Miután a vákuum tappancs tartót rögzítettem, a „H” alakú kerethez terveznem kellett egy köztes alkatrészt, amihez csatlakoztatni tudom a vákuumos egységet és amit rögzíteni lehet a robotkarhoz. Szempont volt a minél kisebb tömeg, ugyanis ezt fogja közvetlen mozgatni a robot. Valamint célom volt az is, hogy legyen rajta egy kialakított hely ahová rögzíteni lehet a négy nyomásmérő szenzort. Fontos volt a minél kisebb befoglaló méret is a szűk munkatér miatt.

Anyagnak alumíniumot választottam az acélnál alacsonyabb sűrűsége miatt, ennek ellenére biztosítja az alkalmazáshoz szükséges merevséget. A robotkar végén egy Ø63 milliméteres felfogó lap található, amihez lehet rögzíteni a különböző effektorokat. Ezért a köztes darab szintén Ø63 milliméteres rúdanyagból készül, amibe szimmetrikusan két elől hátul nyitott zsebet terveztem egyrészt ahova rögzíteni lehet a négy nyomásmérő szenzort másrészt, hogy a robotkarhoz rögzítés során legyen honnan behelyezni a hat darab alacsony fejű, belső kulcsnyílású, M6-os csavart (3.10. ábra:). A vákuumos egységet kalapácsfejű anyákkal, sarok összekötő elemekkel, illetve belső kulcsnyílású csavarokkal rögzítettem a köztes alkatrészhez (3.11. ábra:). A megfogó robbantott ábrája megtalálható az 1. sz. mellékletben.



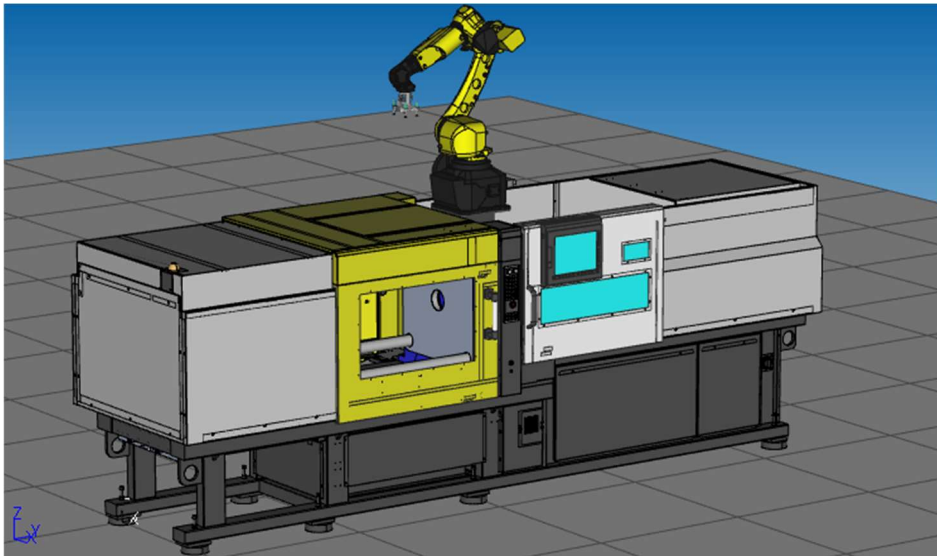
3.10. ábra: A köztes alkatrész



3.11. ábra: A vákuumos megfogó egység

4. A robot program készítése

A robot program elkészítése előtt meg kellett találnom az optimális robotot az adott feladatra. Először a robot típusát kellett kiválasztanom. Fröccsöntési folyamatra a gyakorlatban csuklós, illetve portál robotot alkalmaznak leggyakrabban a gyártók. Jelen esetben csuklós robotot választottam, mivel jelentősen kevesebb helyet foglal el a munkatérből, mint egy portál robot, valamint összetettebb feladatokra képes. Az alkalmazást tekintve a legfontosabb paraméter a robot maximális munkatávolsága. A robot maximális terhelése nem fontos szempont a felhasználást nézve, ugyanis a manipulálni kívánt munkadarab még a megfogóval együtt is elhanyagolható tömegű a robot méretéhez képest. A robot ismétlési pontossága sem elhanyagolható, de az alkalmazást tekintve a mai modern robotok mindegyike képes biztosítani a megfelelő pontosságot. A robotot a fröccsöntőgép tetején kialakított fogadó helyre telepítem, így célszerű a lehető legkisebb tömegű robotot választani ezzel is kímélve a fröccsöntőgép szerkezetét (4.1. ábra:).



4.1. ábra: A robot elhelyezése

A felsorolt paraméterek alapján a Fanuc M-10iD/10L típusú csuklós robotot választottam, mely hosszított kialakítású, így maximális munkatávolsága 1636 milliméter (4.2. ábra:). Maximális teherbírása 10 kilogramm, a robot saját tömege pedig 150 kilogramm. Ismétlési pontossága $\pm 0,03$ milliméter.



4.2. ábra: A Fanuc M-10iD/10L típusú csuklós robot [5]

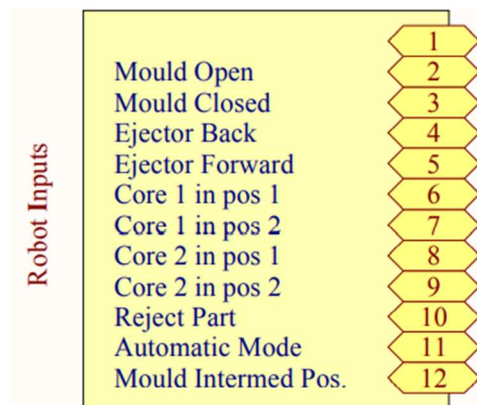
A robot program készítésekor több szempontot vettem figyelembe. Első szempont volt az átláthatóság, azaz, hogy a programon belüli egybe tartozó információkat tömbösítsem. A tömbök elé mindig jelöltem az adott tömbben szereplő utasítások szöveges leírását. Ugyanis, ha valamilyen módosítást kell végre hajtani a programon, például, ha az összes pontot újra kell tanítani, akkor a szöveges leírásokból egyértelműen visszakövethető az adott program felépítése, működése. A következő szempont a felléphető hibákra való felkészítés, azaz, hogy a robot képes legyen észlelni valamennyi potenciális hibát. Mindezek mellett fontos a lehető legkisebb ciklusidő elérése is. Ez egy fröccsöntési folyamat esetében azt jelenti, hogy a fröccsöntési folyamat ciklus idejénél nem lehet több a robot ciklus ideje. Egyszerűen fogalmazva a fröccsöntő gép nem várhat a robotra csak a robot a fröccsöntőgépre. A robotot kézi üzemmódban az úgynevezett Teach Pendant-tal -továbbiakban TP- lehet irányítani, mozgatni és programozni is (4.3. ábra:). A robot program elkészítéséhez a Fanuc Roboguide szimulációs programját használtam fel. A teljes program megtalálható az 1. sz. mellékletben.



4.3. ábra: A Teach Pendant [5]

4.1 A fröccsöntőgép és a robot közti kommunikáció

Egy fröccsöntési folyamat során a fröccsöntőgépnek, illetve a robotnak muszáj kommunikálnia egymással. A két gép közötti kommunikáció miatt kerülhetők el a különböző balesetek, meghibásodások. A kommunikációt a digitális be-, és kimenetekkel valósítom meg. A robot úgynevezett DI -mint Digital Input- jeleket fogad a fröccsöntőgép felől. Ezek a jelek a fröccsöntőgép állapotát leíró jelek, például arról, hogy a szerszám nyitva van. A robot működése közben ezekre a DI jelekre vár. A fröccsöntőgép összesen 12 DI jelet tud küldeni a robotnak (4.4. ábra:). Ezekből hét jelre lesz szükség jelen esetben. Ezeket a jeleket a robot felhasználói felületén definiáltam, valamint leírást készítettem hozzá az egyszerűbb programozás érdekében (4.5. ábra:).



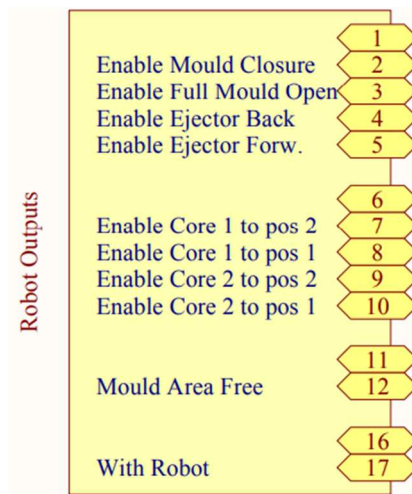
4.4. ábra: A 11 DI jel [5]

#	SIM	STATUS	2/512
DI[2]	U	OFF	[mould open]
DI[3]	U	OFF	[mould closed]
DI[4]	U	OFF	[ejektor back]
DI[5]	U	OFF	[ejektor forward]
DI[6]	U	OFF	[]
DI[7]	U	OFF	[]
DI[8]	U	OFF	[]
DI[9]	U	OFF	[]
DI[10]	U	OFF	[reject]
DI[11]	U	OFF	[mm auto]
DI[12]	U	OFF	[mould interned p]

4.5. ábra: A definiált és leírással ellátott DI jelek

A DI[2] jel a fröccsöntő szerszám nyitásakor lesz aktív, egész pontosan, ha teljesen kinyitott a szerszám, nem meglete nem jelenti, hogy a szerszám nyitva van. A DI[3] jel a szerszám összezárásakor, - a DI[4] jel a kilökök visszatérésével lesz aktív. A fröccsöntőgép a mért fröccsöntési paraméterekből észleli, hogy az épp elkészült alkatrész selejt-e, ha selejt akkor a DI[10]-es jellel jelzi azt. Fontos, hogy az esetleges DI[10] jel csak a szerszám zárásáig aktív, így érdemes valamilyen módon eltárolni azt. A DI[11] jel a fröccsöntőgép automata üzemmódba kapcsolását jelzi: ha a gép kézi üzemmódban van, akkor inaktív, ha automata üzemmódban van, akkor pedig aktív. A DI[12] jel a fröccsöntőgép munkaterének szabadságát jelzi. A robot programban ezen jelek aktivitása, illetve inaktivitása is szükséges információ.

A fröccsöntőgép felől érkező DI jelek a két gép közötti kommunikációban csak az egyik csatornát biztosítják. A másik csatorna az úgynevezett DO -Digital Output- jelek amelyeket a robot küld a fröccsöntőgépnek. Ezek a jelek engedélyező jelek egy kivételével. Az engedélyező jelek a robot programban be, illetve ki kapcsolhatók és utasításokat tartalmaznak például, hogy a szerszám csukódjon be. Így elmondható, hogy a két gép által végzett folyamatot a robot irányítja. A robot összesen tíz DO jelet tud küldeni a fröccsöntőgépnek (4.6. ábra:). Ezekből öt jelre lesz szükség jelen esetben. Ezeket a jeleket a robot felhasználói felületén definiáltam, valamint leírást készítettem hozzá az egyszerűbb programozás érdekében (4.7. ábra:).



4.6. ábra: A 11 DO jel [5]

#	SIM	STATUS	2/512
DO[2]	U	OFF	[enable mould clo]
DO[3]	U	OFF	[enable mould ope]
DO[4]	U	OFF	[enable ejector b]
DO[5]	U	OFF	[enable ejektor f]
DO[6]	U	OFF	[
DO[7]	U	OFF	[
DO[8]	U	OFF	[
DO[9]	U	OFF	[
DO[10]	U	OFF	[
DO[11]	U	OFF	[
DO[12]	U	OFF	[mold area free]

4.7. ábra: A definiált és leírással ellátott DO jelek

A DO[2] jel ha aktív, akkor a fröccsöntőgép engedélyt kap a szerszám nyitására. Fontos, hogy ezen jel meglétéig a szerszámot nem csukja össze a fröccsöntőgép. A DO[3] jel a szerszám összecsukására ad engedélyt, ha aktív hiába küld DO[2] jelet a robot a szerszám nem fog nyitni. A DO[4] jel a kilökő rendszer működésbe lépését engedélyezi. Meglétéig a kilökő rendszer nem húzza vissza a kilökőket. A DO[5] jel a kilökők visszahúzását engedélyezi. A DO[12] jel az egyetlen kivétel, amely nem engedélyező jel, nem utasítást közöl, hanem a robotkar helyzetét jellemzi. Ha aktív, a robotkar a fröccsöntőgép munkaterén kívül tartózkodik, ha inaktív akkor pedig benne. Mint látható nem elég kizárólag az DO jeleket alkalmazni a programban, ugyanis az utasítások kiadása után nem tudja a rendszer, hogy az utasítások végre lettek-e hajtva. Erre szolgálnak a DI jelek, például, ha a robot a DO[2] jellel utasítja a fröccsöntőgépet a szerszám nyitására és az ki nyílik akkor a fröccsöntőgép visszajelez a DI[2] jellel, hogy a szerszám tényleg ki is nyílt.

A DI és DO jeleket közvetítő vezetékek a robot vezérlőegységébe közvetlenül vannak bekötve. Azonban a robotkar végén lévő megfogó el van látva négy nyomásmérő szenzorral és egy mágnesszeleppel, ezen érzékelők kivezetéseit nem lehet közvetlen a vezérlőegységbe kötni, ugyanis a kar mozog a térben. Ezért a robotgyártók ellátták a robotkart robot típustól függően egy, vagy kettő RS485 csatlakozóval, ahová be lehet kötni a robotkaron lévő visszajelző szenzorokat, illetve az irányító eszközöket, például mágnesszelepet. Egy csatlakozón nyolc vezérlő, illetve nyolc bemeneti jelet képes kezelni a robot. A mágnesszelepet irányító jelet RO[1] -mint Robot Out- kimeneti jelre

definiáltam, illetve leírással láttam el (4.8. ábra:). A RO[1] kimeneti jel ha aktív, akkor a mágnesszelep megnyitja a levegő útját a vákuumos megfogó egység felé, ha inaktív akkor elzárja. Nyomásmérő érzékelőből négyet alkalmazok a megfogón, amik négy független jelet adnak, ha a velük sorosan kötött tappancsban felépült a vákuum. Ezt a négy visszajelző jelet a RI [1], [2], [3] és [4]-re definiáltam és leírással láttam el (4.9. ábra:). Az RI [1], [2], [3] és [4] jelek, ha aktívak, az azt jelenti, hogy a tappancsokban kialakult a vákuum.

#	SIM	STATUS	1/8
RO[1]	U	OFF	[vacuum on]
RO[2]	U	OFF	[]
RO[3]	U	OFF	[]
RO[4]	U	OFF	[]
RO[5]	U	OFF	[]
RO[6]	U	OFF	[]
RO[7]	U	OFF	[]
RO[8]	U	OFF	[]

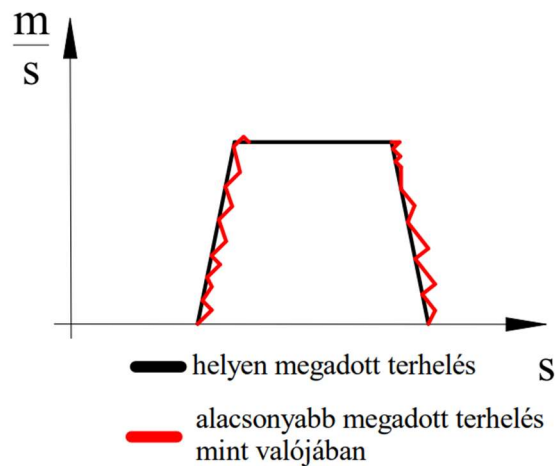
4.8. ábra: A definiált és leírással ellátott RO [1] jel

#	SIM	STATUS	1/8
RI[1]	U	OFF	[Vacuum1 ok]
RI[2]	U	OFF	[vacuum2 ok]
RI[3]	U	OFF	[vacuum3 ok]
RI[4]	U	OFF	[vacuum4 ok]
RI[5]	U	OFF	[]
RI[6]	U	OFF	[]
RI[7]	U	OFF	[]
RI[8]	U	OFF	[]

4.9. ábra: A definiált és leírással ellátott RI jelek

4.2 A mozgási utasítás felépítése

A különböző jelek definiálása után megadtam a robot terhelési adatait. Terhelés adatok alatt a vákuumos megfogó egység tömegére kilogrammban, valamint a tömegközéppontjának távolságát x, y és z irányban a robotkar végétől mérve, illetve a tehetetlenségét is. Ezen adatokat a robot figyelembe veszi a gyorsulási-lassítási paramétereinek kiszámításához. Rosszul vagy nem megadott adatok esetén a gyorsulási görbéje a robotnak egyenletlen lesz, ami rossz hatással van a robot élettartamára (4.10. ábra:).



4.10. ábra: Gyorsulási görbe [13]

Fontos megemlíteni, hogy nem csak a vákuumos megfogó egység tömege számít, hanem az általa megfogott munkadarab vagy munkadarabok. Így két külön tömeget kell definiálni. A gyártó ajánlása szerint csak abban az esetben kell beleszámolni a munkadarabot, ha annak tömege meghaladja az üres megfogó tömegének az 5%-át. Összesen tíz különböző terhelési adat adható meg (4.11. ábra:). Jelen esetben a műanyag alkatrész tömege elhanyagolható a megfogó tömegéhez képest, így csak a megfogó adatait adtam meg (4.12. ábra:). Mivel általános esetben szükség van több beállított hasznos teherre így az ezek közötti váltás programból megoldható.

No.	PAYLOAD[kg]	Comment
1	.93	[vakuumos megfogó]
2	20.00	[]
3	20.00	[]
4	20.00	[]
5	20.00	[]
6	20.00	[]
7	20.00	[]
8	20.00	[]
9	20.00	[]
10	20.00	[]

Active PAYLOAD number = 1

4.11. ábra: A tíz megadható terhelés

Group 1		
1 Schedule No[1]:	[vakuumos megfogó]	
2 PAYLOAD	[kg]	0.93
3 PAYLOAD CENTER X	[cm]	6.00
4 PAYLOAD CENTER Y	[cm]	0.00
5 PAYLOAD CENTER Z	[cm]	5.08
6 PAYLOAD INERTIA X	[kgfcm ²]	2.00
7 PAYLOAD INERTIA Y	[kgfcm ²]	1.00
8 PAYLOAD INERTIA Z	[kgfcm ²]	1.00

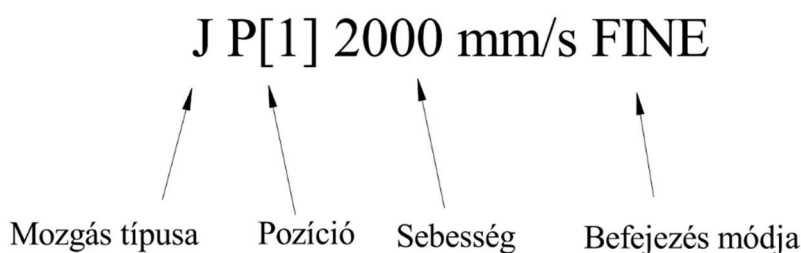
4.12. ábra: Az üres vákuumos megfogó terhelési adatai

Van lehetőség kiegészítő teher megadására is, ami egy hegesztő robot esetén lehet a huzaltoló tömege, amit a robotkarra szerelnek vagy épp egy nagyobb szelepsziget. Ezek a kiegészítők többlet terhet jelentenek a robot számára, amely tömeg a helyes mozgási pálya kiszámítása érdekében elengedhetetlen. A kiegészítő teher tömege rendszerint nem változik, így ez az érték programból nem szerkeszthető.

A robot mozgásához mozgási utasításokat kell rögzíteni a programban majd ezeket a mozgási utasításokat be kell tanítani. A mozgási utasítások betanításához a robotkar végén található TCP pontot-mint Tool Centre Point, továbbiakban szerszámközéppont- az általunk kívánt térbeli pontba kell mozgatni kézi üzemmódban majd elmenteni azt. Így a programban lévő mozgási utasítások tulajdonképpen paraméterekkel ellátott pontok a térben. Kézi üzemmódban három koordináta rendszerben, valamint tengelyenként van lehetőség mozgatni a robotot. Tengelyenkénti mozgatás során a robot tengelyeit önállóan lehet mozgatni, ez alól kivétel a kettes tengely, ugyanis Fanuc robot esetén a kettes tengely mozgatása közben a hármas tengely automatikusan kompenzálja azt úgy, hogy a felső kar orientációja ne változzon. Ez a kompenzáció szoftveresen kikapcsolható. A robot betanításakor hasznos mozgásforma az ütközések elkerülése érdekében azonban kissé körülményes. Gyárilag a robot rendelkezik egy alap koordináta rendszerrel, amelyet nem lehetséges szerkeszteni, ez az úgynevezett World koordináta rendszer -továbbiakban világ koordináta rendszer-. A világ koordináta rendszer egy Descartes-i koordináta rendszer, elhelyezése függőleges, origója a robot alapjában helyezkedik el. Ebben a koordináta rendszerben már képesek vagyunk a szerszámközéppontot X, Y, illetve Z tengelyek mentén lineárisan mozgatni, valamint a tengelyek körül forgatni. Ideális a felhasználása szűk terekben, például egy fröccsöntőgép szerszámában, ahol az ütközés elkerülése érdekében szükséges a lineáris mozgás. Ezen felül még két koordináta rendszerben lehet mozgatni a robotot, melyeket előzetesen meg kell tanítani. A Tool-Frame koordináta rendszer -továbbiakban szerszám-koordináta rendszer - kartezián koordináta rendszer, mely meghatározza az épp alkalmazott effektor -legyen az szerszám vagy megfogó- középpontját és helyzetét a felső kar végének a középpontjához képest. A szerszám koordináta rendszert érdemes úgy felvenni, hogy középpontja az adott effektor középpontja, a Z tengelye pedig az effektor tengelye legyen. Például egy vákuumos megfogó esetében a szerszámközéppont a tappancs közepe, a Z tengely pedig a vákuum tappancs tartójának a tengelye, ebben az esetben képesek vagyunk bármilyen felülethez képest merőlegesen tartani és mozgatni a vákuumos megfogót. Ettől természetesen van lehetőség eltérni. Az utolsó koordináta rendszer az úgynevezett User-Frame koordináta rendszer - továbbiakban felhasználói koordináta rendszer. A felhasználó koordináta rendszer szintén egy kartezián koordináta rendszer, amelyet a felhasználó határoz meg egy

sajátos munkaterülethez. Legyen az egy paletta vagy épp egy adagoló. Előnyös abban az esetben, ha az adott munkaterület tengelyei nem párhuzamosak a robot világ koordináta rendszerével. Felvételével kézi mozgatás során egy a világ koordináta rendszerrel nem párhuzamos elhelyezésű paletta egyszerűen bejárható. [13]

A már említett mozgási utasítás tehát egy pont paraméterekkel ellátva, amely paraméterek mindegyike befolyásolja az eredményezett mozgást (4.13. ábra:). Az első paraméter a mozgás típusa, ami lehet tengelyes, lineáris és köríves. Tengelyes mozgás során az adott pontba a robot egyszerre indul meg az összes tengelyével és áll is meg, ebből következik, hogy a szerszámközépponttal leírt út nem lesz lineáris. Elmondható, hogy a tengelyes mozgás nagyban növeli a robot élettartamát, valamint használatával csökkenteni lehet a ciklusidőt, ugyanis a robot karjai a lehető legkevesebbet mozognak. Abban az esetben, ha az adott feladat során elengedhetetlen a szerszám középpont egyenes vonal menti mozgatása, akkor a lineáris mozgás a megfelelő. A szerszámközéppont lineáris mozgatása kedvezőtlenül hat a robot élettartamára, valamint a ciklus időre is. Köríves mozgás során a robot három pontból számít körívet, azonban maximum csak egy fél kört képes kiszerkesztetni, egy teljes kör leírásához kettő köríves mozgási utasítás szükséges. Fontos, hogy a mozgási utasítások felvételekor a pontokban eltárolódnak a felvételkor érvényes felhasználói, illetve szerszám koordináta rendszerek. [13]



4.13. ábra: A mozgási utasítás felépítése [13]

A második paraméter maga a pozíció. A robot a mozgáshoz szükséges térbeli pontokat sorszámozott pontokként kezeli. Ezekre a pontokra a programban később lehet hivatkozni, így nem szükséges újra tanítani őket. Ezen felül meg lehet nézni a robot tengelyeinek szögértéket az adott pontban (4.14. ábra:), valamint a pont elhelyezkedését a felhasználói koordináta rendszerben (4.15. ábra:). Ebben az ablakban lehetőség van az

adatok szerkesztésére, így például, ha tudjuk egy paletta kiosztásának méreteit akkor elég lehet a pontok koordinátáit átírni és nem szükséges az összeset betanítani, feltéve, hogy a palettához készítettünk felhasználói koordináta rendszert. Eben az ablakban jelzi a rendszer, hogy az adott pont milyen szerszám, illetve felhasználói koordináta rendszerben lett betanítva. A mozgási utasítás csak abban az esetben fut le a programban, ha a felvételkor érvényes koordináta rendszerek az aktívak, ellenkező esetben nem futnak le. Ez fordítva nem teljesen igaz, ha mozgási utasítás felvételekor rossz koordináta rendszerek aktívak, akkor a mozgási utasítás lefut azonban a robot teljesen más pozícióba juttatja a szerszámközpontot.

P[12]	UF:0	UT:1		
J1	0.000	deg	J4	0.000 deg
J2	-60.000	deg	J5	0.000 deg
J3	0.000	deg	J6	0.000 deg

4.14. ábra: Pozíció tengelyes leírása

P[12]	UF:0	UT:1		CONF:NUT 000
X	400.840	mm	W	180.000 deg
Y	-0.000	mm	P	-90.000 deg
Z	645.000	mm	R	0.000 deg

4.15. ábra: Pozíció koordináta leírás

Fontos paraméter a mozgás sebessége. Lineáris és köríves mozgás esetén a mozgás sebessége megadható különböző mértékegységekben, ezen felül megadható a mozgás időtartama is (4.16. ábra:). Tengelyes mozgás során azonban nincs lehetőség konkrét sebességet megadni, kizárólag a leggyorsabb tengely motorjának a fordulatszámát százalékban, valamint a mozgás időtartamát (4.17. ábra:).

1 mm/sec
2 cm/min
3 inch/min
4 deg/sec
5 sec
6 msec

4.16. ábra: Lineáris és köríves mozgás során választható sebesség mértékegységek

1 %
2 sec
3 msec

4.17. ábra: Tengelyes mozgás során választható sebesség mértékegységek

A mozgás befejezése két féle lehet. Az első, amikor a robot a betanított pozíciót elérve megáll, majd kis idő múlva, ha szükséges elindul. Ezt nevezik FINE befejezési módnak. Ezt a befejezési módot választva a robot mozgása darabos lesz, ez a robot élettartamát drasztikusan csökkentheti, azonban ideális olyan pozíciókban, ahol szükséges a pontos

mozgáspálya tartása. A másik befejezési mód az úgynevezett CNT -mint continuous- mód, amikor is a robot nem áll meg a betanított pontot elérve, hanem a következő pozíció felé folytatja az útját. Ettől a robot mozgása egyenletes lesz, amivel az élettartam növelhető. Fontos megjegyezni, hogy CNT utasítás használatával a robot csak megközelíti az adott pozíciót. A megközelítés mértéke állítható nulla és száz között. A CNT0 működése megegyezik a FINE-al. CNT100 használatakor a lehető legnagyobb elkerülőpályát teszi meg a robot. Ezen felül az elkerülés függ a robot sebességétől. Nagyobb sebesség mellett kevésbé közelíti meg a pozíciót a robot. Programozás során a CNT paramétert kizárólag tapasztalati úton lehet meghatározni. [13]

4.3 A robot program leírása

A program tervezése előtt felvázoltam a folyamatot lépésről lépésre (4.18. ábra:). A robot a ciklus kezdetekor vár a kész munkadarabra. Miután a fröccsöntési folyamat lezajlott kész az alkatrész. Ezután a fröccsöntőgép nyitja a szerszámot melyből a robot kiveszi az elkészült alkatrészt. Miután a robot elhagyta a fröccsöntőgép munkaterét a szerszám összezár és indul az újabb fröccsöntési ciklus. Eközben a robot a lerakási pontba juttattja az alkatrészt majd alap pozícióban várja az újabb elkészült alkatrészt.



4.18. ábra: A folyamat terv

Általánosan egy robot program úgy működik, hogy a programban szereplő paramétereket a robot beolvassa soronként, majd teljesíti azt. Azt, hogy épp mely sort olvassa a robot egy fekete kurzorral jelöli a sorszámmon. A programozás elkezdésekor első lépésben felvettem egy alap pozíciót (4.19. ábra:). Az alap pozíciót úgy választottam meg, hogy a lehető legkisebb teret foglalja el munkaterületén belül, paramétereit tekintve tengelyes mozgást, maximális sebességet, illetve FINE befejezési módot választottam. A robot ebbe a pozícióba tér vissza, míg vár a fröccsöntési folyamat teljesülésére, valamint, ha leállás van, ebben a pozícióban marad. Az alap pozíció elé beszúrta a vákuumos megfogó egység terhelési adatait aktiváló parancsot. Ezután egy úgynevezett WAIT –továbbiakban várakozás- parancsot használtam melynek segítségével a fröccsöntőgép felől érkező DI [3] illetve DI [11] jelek aktivitására vár a robot (4.20. ábra:). Alapesetben a robot addig vár amíg meg nem jelennek a várt jelek. Lehetőség van időkorlátot beállítani a várakozáshoz, hogy egy bizonyos idő után a robot valamit reagáljon. A várakozási idő egy rendszerváltozóban van eltárolva század másodpercben, amit lehet manuálisan, valamint programból is módosítani. Programból való módosításhoz először definiálni kell úgynevezett

REGISTER-eket -továbbiakban változók- (4.21. ábra:). A változók tulajdonképpen manuálisan és programból is módosítható számértékek. Jelen esetben manuálisan megadtam az R [1] -es változó értékét 3000-re melyet a program 16. sorában beillesztettem a várakozási idő rendszer változóba, így a várakozási idő 30 másodperc lesz. Utána beillesztettem egy úgynevezett TIMEOUT,LBL[10] parancsot, amely akkor teljesül, ha időtúllépés lép fel. Lehetőség van a program sorok között váltani, erre szolgál a JUMPLBL és a LBL -továbbiakban ugró címke és címke-. Mindkét parancs a program sorok közötti váltásra szolgál. Jelen esetben úgy alakítottam ki a várakozást, hogy időtúllépés során a program kurzora lépjen a tízes címkét tartalmazó sorra, ahol a hibakezelő része található a programnak, amit a későbbiekben részletezek.

```
10: !Alap pozicio
11: PAYLOAD[1:vakuumos megfogo]
12: J @P[1:Alap pozicio] 100% FINE
```

4.19. ábra: Az alap pozíció mozgási utasítása

```
14: !varakozas a DI3 es DI11-re
15: R[10:hibakod]=1
16: $WAITTMOUT=R[1:ciklus start ido]
17: WAIT DI[3:mould closed]=ON AND
: DI[11:mm auto]=ON
: TIMEOUT,LBL[10]
```

4.20. ábra: Várakozás a DI [3] és Di [11] jelek aktivitására

```
R[ 1:ciklus start ido]=3000
R[ 2:keszenlet ido ]=1000
R[ 3:mould felett ido]=300
R[ 4:mouldban ido ]=300
```

4.21. ábra: Az előre definiált változókra példa

Abban az esetben, ha a várt jeleket észleli a robot, akkor tovább fut a program és a robot felveheti a készenléti pozícióját, amit úgy választottam meg, hogy minél közelebb legyen a fröccsöntő szerszámhoz, de ne érjen be a gépbe. A mozgási parancs tengelyes mozgású, maximális sebességű, valamint FINE befejezési módú (4.22. ábra:). Ebben a pozícióban a robot a DO [3] jelet aktivizálja, azaz engedélyezi a fröccsöntőgépnek a szerszám nyitást (4.23. ábra:). Ezután a várakozás parancsot alkalmaztam, amely a DI [2] jelnek az aktivitására vár (4.24. ábra:). Azonban főleg egy új fröccsöntő szerszám beüzemelésakor kézi üzemmódban végeznek pár próba öntést, amely során finom hangolják a fröccsöntési paramétereket. Így a várakozást úgy alakítottam ki, hogy vagy a DI [2] jel aktivitására, vagy a DI [11] jel inaktivitására várjon. Amennyiben

valamelyik feltétel teljesül, a robot felveheti a fröccsöntő szerszám feletti pozíciót, mely már a fröccsöntőgépterében található. A mozgási utasítás ebben az esetben már lineáris mozgású, maximális sebességű, valamint CNT1 befejezési módú, annak érdekében, hogy egy egyenes vonalú, gyors és folyamatos mozgást érjen el a lehető legkisebb elkerülő pályával (4.25. ábra:).

```
19: !keszenleti pozicio
20: J @P[2:keszenleti poz] 100% FINE
```

4.22. ábra: A készenléti pozíció mozgási utasítása

```
22: !szerszamot nyito jel bekapcsol
23: DO[3:enable mould open]=ON
```

4.23. ábra: Szerszám nyitás engedélyezése

```
25: !mould bejarata
26: R[10:hibakod]=2
27: $WAITTMOUT=R[2:keszenlet ido]
28: WAIT DI[2:mould open]=ON OR
: DI[11:mm auto]=OFF
: TIMEOUT, LBL[20]
```

4.24. ábra: Várakozás a DI [2] jel aktivitásra vagy a DI [11] inaktivitására

```
30: !szerszam feletti pozicio
31: L @P[3:mould fellette] 2000mm/sec
: CNT1
```

4.25. ábra: Szerszám feletti pozíció mozgási utasítása

Ezután külön a DI [11] jelet vizsgálom az IF -továbbiakban HA – paranccsal (4.26. ábra:). A fent említett próba öntések okán, ha a fröccsöntőgép kézi üzemmódban van, akkor a robot térjen vissza az alap pozíciójába és kizárólag indító jel hatására induljon újra a program. Ezt egy kettes ugró címkével valósítom meg, mely csak akkor kerül alkalmazásra, ha a DI [11] jel inaktív. A DI [11] jel inaktivitásakor a kurzor a kettes címkére lép, amely a program második sorában található. Utána egy alap pozíció mozgási utasítása található, majd a PAUSE utasítás, amely megállítja a program futását és kizárólag indító jel hatására folytatja azt (4.27. ábra:). Abban az esetben, ha fröccsöntőgép automata üzemmódban van, a HA utasítást átlépi a kurzor. Majd a DO [2] és D [3] jeleket kikapcsolom a szerszám zárás elkerülése érdekében, valamint a későbbi szerszám csukás sikeressége érdekében (4.28. ábra:).


```
33: !ha nincs auto vissza alap
34: IF DI[11:mm auto]=OFF, JMP LBL[2]
```

4.26. ábra: Ha parancs alkalmazása DI [1] aktív feltételével

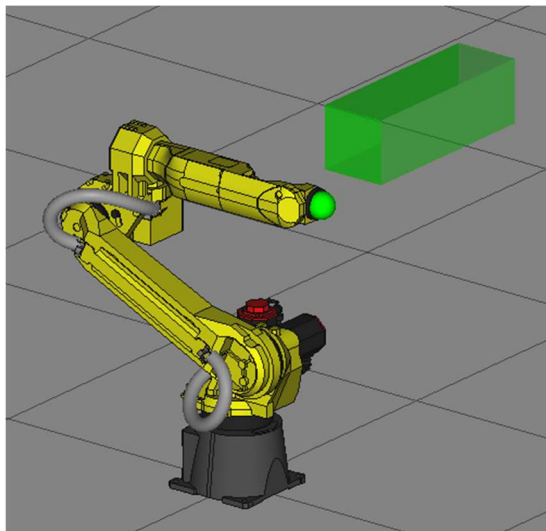
```
1: !hibavagy DI(1)off eseten varakoz
2: LBL[2]
3: J @P[1:Alap pozicio] 100% FINE
4:
5: RO[1:vacuum on]=OFF
6:
7: !csak start jelre indul ujra
8: PAUSE
```

4.27. ábra: A HA parancs teljesülése

```
36: !szerszam csuko jel kikapcsolas
37: DO[2:enable mould close]=OFF
38:
39: !szerszmoz nyito jel kikapcsolasa
40: DO[3:enable mould open]=OFF
```

4.28. ábra: DO [2] és DO [3] jelek kikapcsolása

Elsődleges ellenőrzésként tökéletes a várakozás paranccsal figyelni a fröccsöntő szerszám állapotát, azaz, hogy zárva vagy csukva van. Másodlagos ellenőrzésként alkalmaztam egy úgy nevezett SPACE FUNCTION -továbbiakban üres tér- funkciót. Az üres térrel létre lehet hozni különböző méretű tereket, amelyekbe való belépéshez, mint a robotnak, mint a fröccsöntőgépnek engedélyező jelekkel kell rendelkezniük (4.29. ábra:). [13]



4.29. ábra: Példa az üres tér funkció alkalmazására

Összesen tíz teret lehet létrehozni, a tereket engedélyezni pedig manuálisan, valamint programból is lehet (4.30. ábra:). Mivel csak a fröccsöntőgéppel van közös tere a

robotnak, így egy teret definiáltam mely a fröccsöntőgép munkatere. Amíg a robot szerszámközepontja a téren kívül tartózkodik, addig a DO [12] jel aktív, amit a fröccsöntőgép észlel, így összecsukhatja a szerszámot. A szerszám összecsukásakor a fröccsöntőgép kikapcsolja a DI [12] jelet, ami tiltó jel a robot számára, így a térbe való behatolásakor letilt. Amennyiben a fröccsöntőgép nyitja a szerszámot a DI [12] jel aktív lesz, a robot be tud hatolni a térbe, és belépésének pillanatában DO [12] jel inaktív lesz, aminek hatására a fröccsöntőgép tiltja a szerszám zárását (4.31. ábra:). Ezzel a funkcióval másodlagos védelemmel láttam el a rendszert a balesetek elkerülése érdekében.

No.	Enb/Dsbl	Comment	Usage
1	ENABLE	[geppel ko>]	Common Space
2	DISABLE	[	Common Space
3	DISABLE	[	Common Space
4	DISABLE	[	Common Space
5	DISABLE	[	Common Space
6	DISABLE	[	Common Space
7	DISABLE	[	Common Space
8	DISABLE	[	Common Space
9	DISABLE	[	Common Space
10	DISABLE	[	Common Space

4.30. ábra: A tíz készíthető tér

SPACE :1	GROUP :1
USAGE :	Common Space
1 Enable/Disable:	ENABLE
2 Comment:	[geppel ko>]
3 Output Signal:	DO [12]
4 Input Signal:	DI [12]
5 Priority:	High
6 Inside/Outside:	Inside
7 Common Space Num:	0

4.31. ábra: Az általam definiált tér részletei

Miután megbizonyosodott a robot mind a programból mind az üres tér funkcióból a fröccsöntő szerszám állapotáról, azaz, hogy nyitva van, aktiválja az RO [1] jelet, amivel megnyitja a levegőt a vákuumos megfogó egység felé (4.32. ábra:). Azért nem közvetlen a megfogási pont előtt aktiválom, hogy legyen ideje felépülni a vákuumnak. ezt követően egy mozgási utasítást illesztettem be, mely a fröccsöntőszerszámban található pozíció, amely közvetlen a megfogási pozíció előtt van. A mozgási utasítás mindenképp lineáris kell, hogy legyen a szűk tér miatt, maximális sebességű és FINE befejezésű, ugyanis a szűk hely miatt nem megengedhető a legkisebb elkerülő pálya sem. A mozgási utasítás után szükség van a vákuum tappancsok ellenőrzésére, értem ezalatt, hogy nem került semmilyen tárgy a tappancsokra. Ezt várakozás paranccsal ellenőrzöm. Ha bármelyik tappancsra felkerült bármi, akkor abban a tappancsban kialakul a vákuum és adott tappancs nyomásmérő szenzora jelet küld a robotnak. A várakozó paranccsban ezeket a jeleket figyelem idő túllépéssel (4.33. ábra:).

```

42: !vakuum bekapcsolasa
43: RO[1:vacuum on]=ON
44:
45: !megfogas elotti pozicio
46:L P[4:megfogas elotti ]
   : 2000mm/sec FINE

```

4.32. ábra: A vákuum bekapcsolása, illetve a megfogás előtti pont mozgási utasítása

```

48: !tappancsok ellenorzesse
49: R[10:hibakod]=4
50: $WAITTMOUT=R[3:mould felett ido]
51: WAIT RI[1:Vacuum1 ok]=OFF AND
   : RI[2:vacuum2 ok]=OFF AND
   : RI[3:vacuum3 ok]=OFF AND
   : RI[4:vacuum4 ok]=OFF
   : TIMEOUT,LBL[40]

```

4.33. ábra: Várakozás parancs a tappancsok ellenőrzésére

Amennyiben egyik tappancsban sem épült fel a vákuum, a kurzor tovább halad. A tappancsok ellenőrzését követően beillesztettem a megfogási pozíció mozgási utasítását, mely csak lineáris mozgás lehet, viszonylag alacsony sebességű és kizárólag FINE befejezésű (4.34. ábra:). Eben a pozícióban a tappancsok hozzáérnek a munkadarabokhoz, de csak annyira, hogy kialakuljon a vákuum. Itt történik meg a munkadarabok megfogása. Ezt követően a robot aktiválja a DO [5] jelet, amely engedélyezi a kilökök előre jövetelét. A kilökök útját a tappancs tartóba épített magasság kiegyenlítő egység biztosítja. Ezután a robot meggyőződik a kilökök előre jöttéről, a várakozás paranccsal szintén idő túllépéssel (4.35. ábra:). Amennyibe a DI [5] jel aktív, a DO [5] jelet inaktíválja a robot a későbbi kilökő visszahúzás érdekében.

```

53: !megfogasi pont
54:L P[5:megfogasi pont] 100mm/sec
   : FINE
55:
56: !kilokok előre
57: DO[5:enable ejektor forward]=ON

```

4.34. ábra: A megfogási pont mozgási utasítása, illetve a DO [5] jel bekapcsolása

```

59: !kiloko jel varas
60: R[10:hibakod]=5
61: $WAITTMOUT=R[3:mould felett ido]
62: WAIT DI[5:ejector forward]=ON
   : TIMEOUT,LBL[40]
63:
64: !kilokojel torles
65: DO[5:enable ejektor forward]=OFF

```

4.35. ábra: Várakozás parancs a DI [5] jel aktivitására, valamint a DO [5] jel kikapcsolása

Ezután várakozás paranccsal meggyőzőik a robot a vákuum kialakulásáról (4.36. ábra:). Csak abban az esetben halad tovább a program, ha RI [1], RI [2], RI [2] és RI [4] jelek egytől egyik aktívak. Abban az esetben, ha bármelyik jel inaktív, akkor a kurzor a program hibakezelő részére ugrik, ugyanis valószínűleg munkadarab ragadt a fröccsöntő szerszámba, ami meghibásodáshoz vezetne. Ha az összes jel aktív folytatódik a program. A fröccsöntési folyamat során a fröccsöntő gép a mért fröccsöntési paraméterekből meg tudja határozni, hogy az épp öntött munkadarab selejt-e vagy sem. Ezt a DI [10] jellel közli a robot felé, mely jel csak a szerszám zárásáig

elérhető. Így ezt a jelet eltárolom egy úgynevezett FLAG-ben -továbbiakban zászló- (4.37. ábra:). A zászlók két állásúak, aktív és inaktív. Így a későbbiekben, ha a zászló aktív, akkor az épp manipulált alkatész selejt, ha inaktív akkor pedig jó. Az eltárolást követően aktiválom a DO [4] jelet, mellyel engedélyezem a kilökök visszahúzását, ezzel csaknem egyidejűleg beszúrtam a megfogás előtti pozíció mozgási utasítását lineáris mozgással, de csökkentett sebességgel.

```
67: !vakuum ok?
68: R[10:hibakod]=6
69: $WAITTMOUT=R[3:mould felett ido]
70: WAIT RI[1:Vacuum1 ok]=ON AND
   : RI[2:vacuum2 ok]=ON AND
   : RI[3:vacuum3 ok]=ON AND
   : RI[4:vacuum4 ok]=ON
   : TIMEOUT,LBL[50]
```

4.36. ábra: Várakozás a vákuum kialakulására

```
72: !selejt rogzites
73: F[1:selejt-e]=(DI[10:reject])
74:
75: !kiloko vissza jel bekapcs
76: DO[4:enable ejector back]=ON
77:
78: !vissza a megfoasi pont ele
79:L P[4:megfogas elotti ] 100mm/sec
   : FINE
```

4.37. ábra: A selejt jel rögzítése

Ezt követően várakozás paranccsal a kilökök visszatérését vizsgálja a robot idő túllépéssel (4.38. ábra:). Amennyiben a DI [4] jel aktív, a program fut tovább és egyúttal kikapcsolja a DO [4] jelet a későbbi kilökö előre jövetel érdekében. Majd szerszám feletti pozícióba juttattam a robotot lineáris, maximális sebességű FINE befejezésű mozgási utasítással (4.39. ábra:). Ebben a pozícióban zárható a szerszám, így a DO [2] jelet aktiválom. Várakozás paranccsal vizsgálom a szerszám tényleges zárását, melyhez időtúllépést rendeltem. Amennyiben a DI [1] jel aktív, a szerszám ténylegesen bezárt.

```
81: !varakozas kilokok hatul
82: R[10:hibakod]=7
83: $WAITTMOUT=R[3:mould felett ido]
84: WAIT DI[4:ejektor back]=ON
   : TIMEOUT,LBL[30]
85:
86: !kiloko jel vissza kikapcs
87: DO[4:enable ejector back]=OFF
```

4.38. ábra: Várakozás a kilökök visszatérésére majd a DO [4] jel kikapcsolása

```
89: !vissza a szerszam fele
90:L P[3:mould fellelte] 2000mm/sec
   : FINE
91: !szerszam zaras
92: DO[2:enable mould close]=ON
93: !vrakozas szerszam bezarasara
94: R[10:hibakod]=8
95: $WAITTMOUT=R[3:mould felett ido]
96: WAIT DI[3:mould closed]=ON
   : TIMEOUT,LBL[50]
```

4.39. ábra: A szerszám feletti pont mozgási utasítása, a szerszámzárás engedélyezése, valamint várakozás a szerszám zárásra

Miután a fröccsöntő szerszám zárt, indul az újabb fröccsöntési ciklus, ami közben a robot a szerszámból kivett munkadarabokat eljuttatja a lerakási pozícióba. Előtte azonban ki kell olvasni az F [1] zászlóban tárolt selejt jelet, ami alapján eldől az adott munkadarab lerakási pontja. Ehhez az IF THEN -továbbiakban- HA Akkor- parancsot

használtam. A HA AKKOR parancs abban tér el az egyszerű HA parancstól, hogy az adott feltétel teljesülése után több következményt lehet rendelni a parancshoz, majd le lehet azt zárni. Abban az esetben, ha a feltétel nem teljesül a kurzor átugrik a parancs utánra. Az első Ha AKKOR parancs feltételének az F [1] zászló kikapcsolt állapotát rendeltem, mely feltétel teljesülésekor a megfogott munkadarab nem selejt. Így a kurzor belép a parancs belsejébe, ahol először mozgási utasításokkal a lerakási pontba manipulálja az alkatrészeket (4.40. ábra:). Majd a RO [1] jelet kikapcsolja, aminek hatására a vákuumos megfogó elengedi az alkatrészeket. A lerakást ellenőrizve várakozó parancsot használtam, mely az RI [1], RI [2] RI [3] valamint az RI [4] jeleket figyelő idő túllépéssel. Amennyiben az összes jel inaktív, a lerakás sikeresen megtörtént a kurzor pedig tovább megy. Ezután előre definiált változó alkalmazásával számolom a jó munkadarabok darabszámát programból. A szerszám négy fészkes, így lerakásonként négygel növelem a változó értékét.

Abban az esetben, ha az első HA AKKOR parancs feltétele nem teljesül, a kurzor átugrik a második HA AKKOR parancsra. Amelynek feltétele az F [1] zászló aktivitása (4.41. ábra:). Ez a parancs fogja kezelni a selejtes munkadarabok manipulálását. Mozgási utasításokkal eljuttatja a selejtet a lerakási pontjához majd az RO [1] jel kikapcsolásával a vákuumos megfogó lerakja a selejtes darabokat. A lerakást ellenőrizve itt is szükség van várakozó parancsra, mellyel az RI [1], RI [2] RI [3] valamint az RI [4] jeleket figyelő. Ha mind inaktív, a lerakás sikeresen megtörtént. Ezt követően ide is beépítettem egy selejt darabszám számlálót, melyhez előre definiált változót használtam. A szerszám négy fészkes, így lerakásonként négygel növeltem a változó értékét.


```

98: !ha jo akkor ell es vissza homeba
99: IF (F[1:selejt-e]=OFF) THEN
100:J P[2:keszenleti poz] 100% CNT60
:
101:J P[6:jo md lerakas el] 100%
: CNT60
102:J P[7:jo md lerakasi] 100% FINE
:
103: !vakuum kikapcs, ellenorzes
104: RO[1:vacuum on]=OFF
105: R[10:hibakod]=9
106: $WAITTMOUT=R[3:mould felett ido]
107: WAIT RI[1:Vacuum1 ok]=OFF AND
: RI[2:vacuum2 ok]=OFF AND
: RI[3:vacuum3 ok]=OFF AND
: RI[4:vacuum4 ok]=OFF
: TIMEOUT,LBL[40]
108:
109: R[13:jo darabszam]=
: R[13:jo darabszam]+4
110: ENDIF

```

4.40. ábra: A jó alkatrészek kezelése

```

112: !selej le vissza home
113: IF (F[1:selejt-e]=ON) THEN
114:J P[2:keszenleti poz] 100% CNT60
:
115:J P[8:selej md le elot] 100%
: CNT60
116:J P[9:selej le pozici] 100% FINE
:
117: !vakuum kikapcsolasa ellenorzes
118: RO[1:vacuum on]=OFF
119: R[10:hibakod]=9
120: $WAITTMOUT=R[3:mould felett ido]
121: WAIT RI[1:Vacuum1 ok]=OFF AND
: RI[2:vacuum2 ok]=OFF AND
: RI[3:vacuum3 ok]=OFF AND
: RI[4:vacuum4 ok]=OFF
: TIMEOUT,LBL[40]
122:
123: R[12:selej darabszam]=
: R[12:selej darabszam]+4
124: ENDIF

```

4.41. ábra: A selejt alkatrészek kezelése

A lerakást követően még beszúrtam egy ciklus számlálót, mely akkor is számlál, ha selejt és akkor is, ha nem selejt a kész munkadarab. Ezenkívül beszúrtam egy változókból álló számlálót, mely a jó darabszámot osztja el az összes darabszámmal, az eredményt pedig minden ciklus után frissíti (4.42. ábra:). A számlálók után egyes ugró címkét használtam, mely a kurzort visszadobja a program kilencedik sorába és a ciklus újraindul (4.43. ábra:).

```

126: !ciklus szamlalo
127: R[11:ciklus szamlalo]=
: R[11:ciklus szamlalo]+1
128:
129: !jo munkadarab % szamlalo
130: R[15:hany db md]=
: R[11:ciklus szamlalo]*4
131: R[14:hany % jo ]=
: R[13:jo darabszam]/
: R[15:hany db md]
132: R[14:hany % jo ]=
: R[14:hany % jo ]*100
133:
134: !cikls ujra
135: JMP LBL[1]

```

4.42. ábra: A különböző számlálók, illetve a ciklus újraindítása

```

9: LBL[1]
10:
11: !Alap pozicio
12: PAYLOAD[1:vakuumos megfogo]
13:J P[1:Alap pozicio] 100% FINE
14:
15: !varakozas a DI3 es DI11-re
16: R[10:hibakod]=1
17: $WAITTMOUT=R[1:ciklus start ido]
18: WAIT DI[3:mould closed]=ON AND
: DI[11:mm auto]=ON
: TIMEOUT,LBL[10]
19:
20: !keszenleti pozicio
21:J P[2:keszenleti poz] 100% FINE

```

4.43. ábra: Az újraindult ciklus

A program eddigi része a folyamat rendeltetésszerű működéséért felelős, azaz nem lép fel semmilyen rendellenesség. Azonban célom volt felkészíteni a robotot a legtöbb

várható meghibásodásra, így készítettem egy hibakezelő részt a programon belül, mely kizárólag akkor kerül olvasásra, ha valamelyik várakozó parancs során időtullépés történik. Például, ha a fröccsöntő szerszám zárását várva letelik a előre definiált három másodperc, akkor vélhető, hogy a szerszám valamilyen külső tényező miatt nem csukódik be. Ebben az esetben a várakozás parancsba épített ugró címke a kurzort a program hiba kezelő részére dobja. A hibakezelő rész első fele a robot pozíciójával foglalkozik. Ugró címkék és a már használt mozgási parancsok segítségével összeállítottam egy program részt, ami akármilyen pozícióban az alap pozícióba vezeti a robotot ütközés nélkül. Tehát például, ha a robot a megfogási pontban van, és a kilökök nem jönnek előre, akkor a robot azt észleli a 61. sorban található várakozó parancssal és mivel a parancs időtullépéssel van ellátva, egy előre definiált idő után a 40-es ugró címkével a kurzort a program 143. sorába dobja (4.44. ábra:) (4.45. ábra:). Ezt követően a megfogás előtti pozíciót veszi fel a robot, majd a legvégén az alap pozíciót mindezt ütközés nélkül.

```
59: !kiloko jel varas
60: R[10:hibakod]=5
61: $WAITTIMEOUT=R[3:mould felett ido]
62: WAIT DI[5:ejector forward]=ON
   : TIMEOUT,LBL[40]
63:
64: !kilokojel torles
65: DO[5:enable ejektor forward]=OFF
```

4.44. ábra: Várakozás a DI [5] jelre

```
142: !hiba eseten haza program
143: LBL[40]
144:L P[4:megfogas elotti ] 100mm/sec
   : FINE
145: LBL[30]
146:L P[3:mould fellelt] 100mm/sec
   : FINE
147: LBL[20]
148:J P[2:keszenleti poz] 100% FINE
   :
149: LBL[10]
150:J P[1:Alap pozicio] 100% FINE
```

4.45. ábra: Alap pozícióba vezető
programrész

A program így már képes meghibásodás esetén alap pozícióba állítani a robotot. Azonban ezt követően a gépkezelőnek fel kell tárnia a hiba forrását. Az munkáját könnyítve „hibakód” néven definiáltam az R [10] változót. A változó értékét programból állítom, mégpedig az összes várakozó parancs előtt. Tehát az első várakozó parancs előtt az R [10] változó értékét egyre állítom, a második várakozó parancs előtt pedig kettőre és így tovább. Így akármelyik várakozó parancsnál akad meg a folyamat az R [10] változó kiolvasásával tudom azt, hogy melyiknél. A kiolvasáshoz a SELECT - továbbiakban szelektál- parancsot alkalmaztam (4.46. ábra:). A szelektál parancssal a R [10] változót vizsgálom, egész pontosan az értékét. Abban az esetben, ha az értéke 1,

akkor a 12-es ugró címkével a kurzor a program 167. sorára ugrik, és onnan folytatódik, ahová elhelyeztem két úgynevezett MESSAGE -továbbiakban üzenet- parancsot (4.47. ábra:). Az üzenet paranccsal a robot felhasználói felületére lehet megjegyzéseket írni, akár a program futása közben is. Így az egyes hibakód esetén a programba épített üzenet parancsok kiírják a robot felhasználói felületére például, hogy a „szerszám nincs zárva” vagy, hogy „nincs automata üzemmódban a gép”. Az üzenet parancsok meghatározásához a várakozó parancsokban található feltételeket használtam, ugyanis, ha valamilyen hiba lép fel, akkor biztosan az adott várakozó parancson belül nem teljesült valamelyik feltétel. Miután az adott hibakódhoz tartozó üzenet parancs lefutott a kettes ugró címkével, a kurzor a program elejére ugrik, ahol a robot alap pozícióban vár az indító jelre.

```

155: !hibakod kiolvassa
156: SELECT R[10:hibakod]=1,
:   JMP LBL[11]
157:     =2, JMP LBL[12]
158:     =3, JMP LBL[13]
159:     =4, JMP LBL[14]
160:     =5, JMP LBL[15]
161:     =6, JMP LBL[16]
162:     =7, JMP LBL[17]
163:     =8, JMP LBL[18]
164:     =9, JMP LBL[19]

```

4.46. ábra: A szelektál paranccsal olvasott hiba kódok

```

167: LBL[11]
168: MESSAGE[szerszam nincs zarva]
169: MESSAGE[gep nincs autoban]
170: JMP LBL[2]
171:
172: LBL[12]
173: MESSAGE[szerszam nincs nyitva]
174: JMP LBL[2]

```

4.47. ábra: Példa az üzenetek parancsra

5. Összefoglalás

Szakdolgozatomban egy fröccsöntés folyamat robotos kiszolgálása volt a megoldandó feladat. A dolgozat elején az ipari robotok csoportosítását fejtettem ki hosszabban, valamint a különböző típusú robot effektorokat igyekeztem kategorizálni a lehető leglogikusabb módon. A robotos kiszolgáláshoz először is terveznem kellett egy megfogót, amivel a robot képes kölcsönhatásba lépni a műanyag alkatrészekkel. Az effektorok csoportosításában is említett megfogó szerkezetek közül az ujjas, illetve a vákuumos megoldás jöhetett szóba. Végül vákuumos megfogó egységet terveztem, ugyanis egyszerűbb kivitelezni a megfogás sikerességének a visszajelzését. Nyomásmérő szenzorokat alkalmaztam a visszacsatolás megvalósításához, melyek karbantartás mentesek és gazdaságosak. A vákuumos megfogó egység tervezésekor fontos szempont volt a csekély tömeg, állíthatóság és a kis méret. Megfelelő számításokkal méreteztem a vákuumos tappancs átmérőjét, anyagát pedig a magas munkahőmérséklethez választottam. Az alacsony tömegeket, valamint az állíthatóságot az iparban oly gyakran használt extrudált alumínium profilokkal és a hozzá tartozó kiegészítőkkel értem el (pl.: kalapácsanya). A kis méret miatt az összeszerelés kissé nehézkes, ugyanis a nyomásmérő szenzorokat csak az után lehet rögzíteni a megfogón, hogy a szerkezetet felszereltük a robotkarra. Ezen kívül elégedett vagyok a végleges megfogó egységgel.

A megfogó egység tervezése után a robot programját készítettem el, mégpedig a Fanuc Roboguide szimulációs szoftverén. Céлом volt a program átláthatósága, így minden részfolyamatot leírással láttam el. A robot és a fröccsöntőgép közötti kommunikációra nagy hangsúlyt fektettem. A program elkészítésében nagy hasznát vettem a szakmai gyakorlatomon szerzett tudásnak. A robot program biztosítja a folyamat üzemszerű működését, valamint a lehetséges hibák észlelését és megfelelő kezelését. A program a szimulációs környezetben megfelelően működik azonban sajnos valós körülmények között nem volt lehetőségem tesztelni. Ennek hiányában is azt gondolom, hogy szakdolgozatomban megfogalmazott feladatokat sikerült teljesítenem.

A robotos gépkiszolgálást egy nagyon izgalmas témának találom, mellyel szeretnék a jövőben is foglalkozni. Szakdolgozatom elkészítése során lehetőségem volt arra, hogy

bővítsem az e témával kapcsolatos tudásom, mely a jövőbeni munkámban is biztosan előnyömre válik majd.

6. Summary

In my thesis, the task to be solved was robotic support of an injection molding process. At the beginning of the thesis, I explained the grouping of industrial robots at length, and I tried to categorize the different types of robot effectors in the most logical way as possible. For the robotic support, I first had to design a gripper with which the robot can interact with the plastic parts. Among the gripping structures mentioned in the grouping of effectors, the finger and vacuum solutions could be considered. Finally, I designed a vacuum gripping unit, because it is easier to implement the feedback of the success of the gripping. I used pressure sensors to implement the feedback, which are maintenance-free and economical. Low weight, adjustability and small size were important considerations when designing the vacuum gripper unit. With appropriate calculations, I sized the diameter of the vacuum cup, and I chose its material for the high working temperature. I achieved the low weights and the adjustability with the extruded aluminum profiles that are so often used in the industry and the related accessories (e.g. hammer nut). Due to the small size, assembly is somewhat difficult, as the pressure sensors can only be attached to the gripper after the structure has been mounted on the robot arm. Other than that, I'm pleased with the final grip unit.

After designing the gripper unit, I created the program for the robot using Fanuc's Roboguide simulation software. My aim was to make the program transparent, so I described each sub-process. I put a lot of emphasis on the communication between the robot and the injection molding machine. In preparing the program, I made great use of the knowledge gained during my professional internship. The robot program ensures the normal operation of the process, as well as the detection and appropriate handling of potential errors. The program works properly in the simulation environment, but unfortunately I did not have the opportunity to test it in real conditions. Even in the absence of this, I think that I managed to fulfill the tasks set out in my thesis.

I find robotic machine supporting to be a very exciting topic that I would like to continue to deal with in the future. During the preparation of my thesis, I had the opportunity to expand my knowledge which will certainly benefit me in my future work.

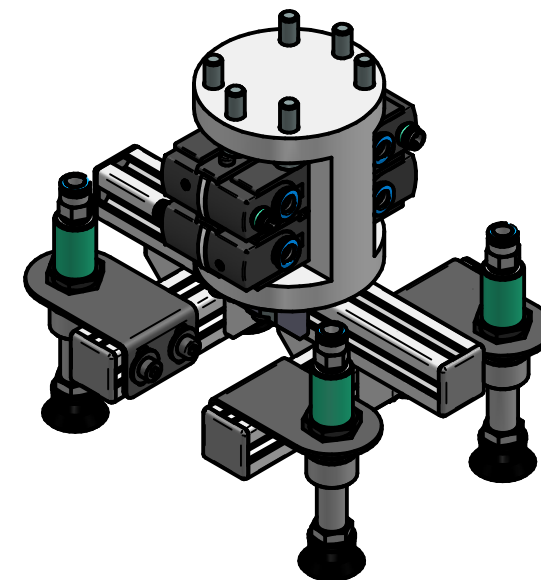
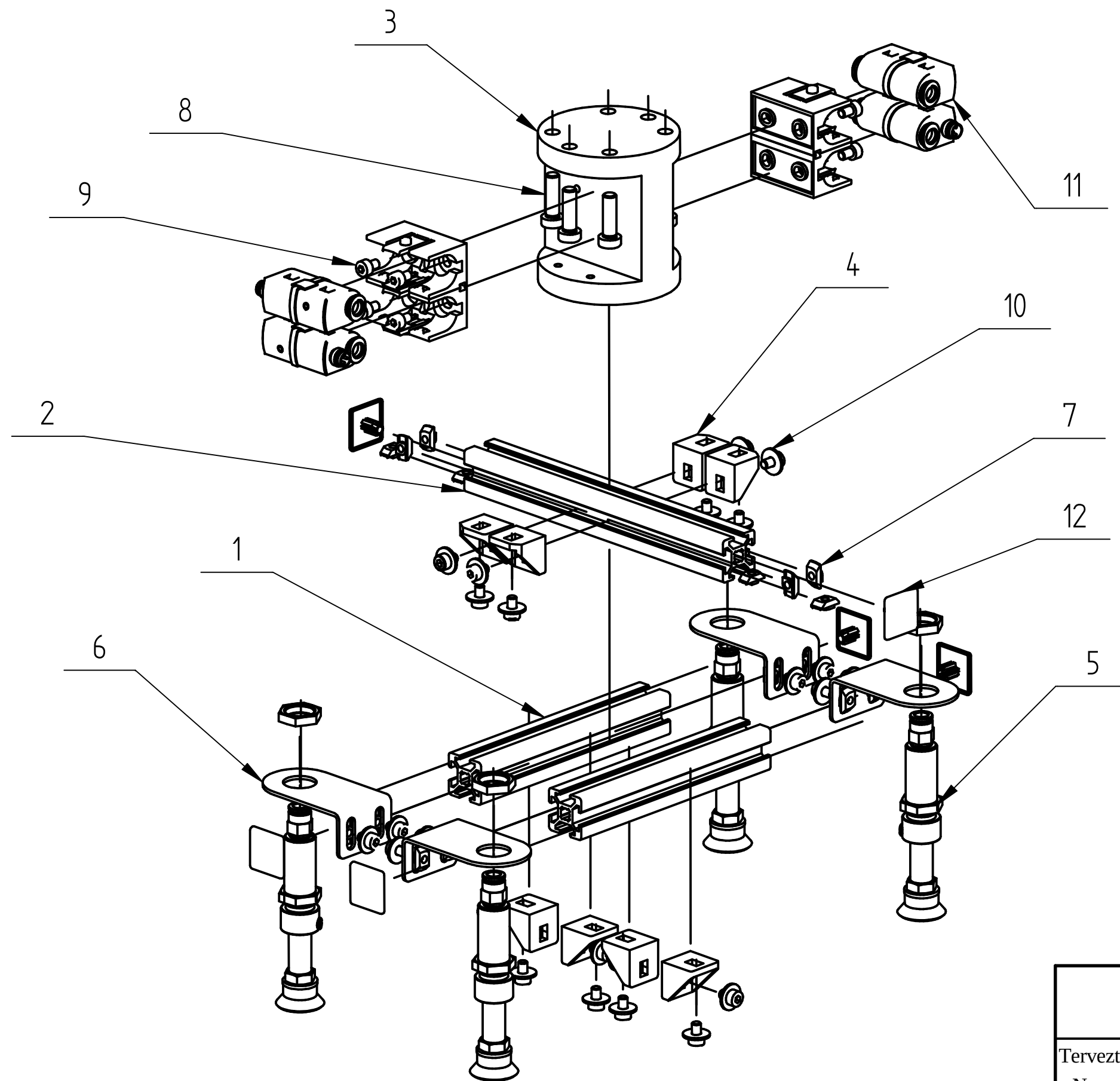
7. A felhasznált irodalom

- [1] Alex Misiti: A history of Industrial Robots 2020
<https://www.wevolver.com/article/a-history-of-industrial-robots>
- [2] Mathieu Bélanger-Barette: How to choose the right industrial robot? 2021.
<https://blog.robotiq.com/how-to-choose-the-right-industrial-robot>
- [3] Dr. Helm László: Ipari robotok. Műszaki könyvkiadó. Budapest, 1983. p. 16-54., ISBN:963 10 4740-7
- [4] Process Solution: What are the different types of industrial robots and their applications? 2018.
<https://processsolutions.com/what-are-the-different-types-of-industrial-robots-and-their-applications/>
- [5] <https://www.fanuc.eu/hu/hu>
- [6] <https://www.gudel.com/>
- [7] 6 major types of industrial robots used in the global manufacturing. 2018
<https://blog.technavio.org/blog/major-types-of-industrial-robots>
- [8] What is an end effector and how do you use one? 2022.
<https://www.automate.org/news/what-is-an-end-effector-and-how-do-you-use-one>
- [9] Kel Guerin: Automation 101: End-of-arm-Tooling, 2022.02.11.
<https://www.wevolver.com/article/automation-101-end-of-arm-tooling>
- [10] Dunai Antal, Dr. Macskási Levente: Műanyagok fröccsöntése. Lexica Kft. Budapest 2003. p. 1-11., ISBN 963 206 550 6
- [11] Gareth J. Monkman, Stefan Hesse, Ralf Steinmann, Henrik Schunk: Robot Grippers. Wiley-VCH. Darmstadt 2004., p. 169-182., ISBN 978-3-527-40619-7
- [12] <https://www.festool.hu/>
- [13] Fanuc: Oktatási kézikönyv, Robot működtetés és programozás 8. verzió 2020.

8. Mellékletek

1. sz. melléklet: A vákuumos megfogó egység robbantott ábrája

2. sz. melléklet: A teljes robot program



Darabjegyzék		
Tételszám	Megnevezés	Darabszám
1.	120x20x20 Bosch Rexroth profil	2
2.	160x20x20 Bosch Rexroth profil	1
3.	Ø63x70 köztes alkatrész	1
4.	20x20 szögidom	8
5.	Vákuum tappancs tartó	4
6.	Vákuum tappancs tartó lemez	4
7.	M4 kalapácsanya	24
8.	M6x20 belsőkulcsnyílású csavar	6
9.	M4x6 belsőkulcsnyílású csavar	32
10.	Ø12 acél alátét	24
11.	Nyomásmérő szenzor	4
12.	20x20 fedősapka	6

A vákuumos megfogó egység robbantott ábrája

Tervezte: Nagy László	Méretarány: 1:2	Tömeg: 0,93 kg
Dátum/aláírás: 2022.12.01		Rajzsám: Szd001

2. sz. melléklet: A teljes robot program

```

1: !hibavagy DI(1)off eseten varakoz ;
2: LBL[2] ;
3: J P[1:Alap pozicio] 100% FINE ;
4: ;
5: RO[1:vacuum on]=OFF ;
6: ;
7: !csak start jelre indul ujra ;
8: PAUSE ;
9: LBL[1] ;
10: ;
11: !Alap pozicio ;
12: PAYLOAD[1:vakuumos megfogo] ;
13: J P[1:Alap pozicio] 100% FINE ;
14: ;
15: !varakozas a DI3 es DI11-re ;
16: R[10:hibakod]=1 ;
17: $WAITTMOUT=R[1:ciklus start ido] ;
18: WAIT DI[3:mould closed]=ON AND DI[11:mm auto]=ON TIMEOUT,LBL[10] ;
19: ;
20: !keszenleti pozicio ;
21: J P[2:keszenleti poz] 100% FINE ;
22: !szerszamot nyito jel bekapcsol ;
23: DO[3:enable mould open]=ON ;
24: ;
25: !mould bejarata ;
26: R[10:hibakod]=2 ;
27: $WAITTMOUT=R[2:keszenlet ido] ;
28: WAIT DI[2:mould open]=ON OR DI[11:mm auto]=OFF TIMEOUT,LBL[20] ;
29: ;
30: !szerszam feletti pozicio ;

```

```

31:L P[3:mould fellette] 2000mm/sec CNT1 ;
32: ;
33: !ha nincs auto vissza alap ;
34: IF DI[11:mm auto]=OFF,JMP LBL[2] ;
35: ;
36: !szerszam csuko jel kikapcsolas ;
37: DO[2:enable mould close]=OFF ;
38: ;
39: !szerszmoz nyito jel kikapcsolasa ;
40: DO[3:enable mould open]=OFF ;
41: ;
42: !vakuum bekapcsolasa ;
43: RO[1:vacuum on]=ON ;
44: ;
45: !megfogas elotti pozicio ;
46:L P[4:megfogas elotti ] 2000mm/sec FINE ;
47: ;
48: !tappancsok ellenorzes ;
49: R[10:hibakod]=4 ;
50: $WAITTMOUT=R[3:mould felett ido] ;
51: WAIT RI[1:Vacuum1 ok]=OFF AND RI[2:vacuum2 ok]=OFF AND
RI[3:vacuum3 ok]=OFF AND RI[4:vacuum4 ok]=OFF TIMEOUT,LBL[40] ;
52: ;
53: !megfogasi pont ;
54:L P[5:megfogasi pont] 100mm/sec FINE ;
55: ;
56: !kilokok elore ;
57: DO[5:enable ejektor forward]=ON ;
58: ;
59: !kiloko jel varas ;
60: R[10:hibakod]=5 ;
61: $WAITTMOUT=R[3:mould felett ido] ;

```



```

62: WAIT DI[5:ejector forward]=ON TIMEOUT,LBL[40] ;
63: ;
64: !kilokojel torles ;
65: DO[5:enable ejektor forward]=OFF ;
66: ;
67: !vakuum ok? ;
68: R[10:hibakod]=6 ;
69: $WAITTMOUT=R[3:mould felett ido] ;
70: WAIT RI[1:Vacuum1 ok]=ON AND RI[2:vacuum2 ok]=ON AND RI[3:vacuum3
ok]=ON AND RI[4:vacuum4 ok]=ON TIMEOUT,LBL[50] ;
71: ;
72: !selejt rogzites ;
73: F[1:selejt-e]=(DI[10:reject]) ;
74: ;
75: !kiloko vissza jel bekapcs ;
76: DO[4:enable ejektor back]=ON ;
77: ;
78: !vissza a megfoasi pont ele ;
79: L P[4:megfogas elotti ] 100mm/sec FINE ;
80: ;
81: !varakozas kilokok hatul ;
82: R[10:hibakod]=7 ;
83: $WAITTMOUT=R[3:mould felett ido] ;
84: WAIT DI[4:ejektor back]=ON TIMEOUT,LBL[30] ;
85: ;
86: !kiloko jel vissza kikapcs ;
87: DO[4:enable ejektor back]=OFF ;
88: ;
89: !vissza a szerszam fele ;
90: L P[3:mould fellette] 2000mm/sec FINE ;
91: !szerszam zaras ;
92: DO[2:enable mould close]=ON ;

```

```

93: !varakozas szerszam bezarasara ;
94: R[10:hibakod]=8 ;
95: $WAITTMOUT=R[3:mould felett ido] ;
96: WAIT DI[3:mould closed]=ON TIMEOUT,LBL[50] ;
97: ;
98: !ha jo akkor ell es vissza homeba ;
99: IF (F[1:selejt-e]=OFF) THEN ;
100: J P[2:keszenleti poz] 100% CNT60 ;
101: J P[6:jo md lerakas el] 100% CNT60 ;
102: J P[7:jo md lerakasi] 100% FINE ;
103: !vakuum kikapcs, ellenorzes ;
104: RO[1:vacuum on]=OFF ;
105: R[10:hibakod]=9 ;
106: $WAITTMOUT=R[3:mould felett ido] ;
107: WAIT RI[1:Vacuum1 ok]=OFF AND RI[2:vacuum2 ok]=OFF AND
RI[3:vacuum3 ok]=OFF AND RI[4:vacuum4 ok]=OFF TIMEOUT,LBL[40] ;
108: ;
109: R[13:jo darabszam]=R[13:jo darabszam]+4 ;
110: ENDIF ;
111: ;
112: !selej le vissza home ;
113: IF (F[1:selejt-e]=ON) THEN ;
114: J P[2:keszenleti poz] 100% CNT60 ;
115: J P[8:slejt md le elot] 100% CNT60 ;
116: J P[9:selejt le pozici] 100% FINE ;
117: !vakuum kikapcsolasa ellenorzes ;
118: RO[1:vacuum on]=OFF ;
119: R[10:hibakod]=9 ;
120: $WAITTMOUT=R[3:mould felett ido] ;
121: WAIT RI[1:Vacuum1 ok]=OFF AND RI[2:vacuum2 ok]=OFF AND
RI[3:vacuum3 ok]=OFF AND RI[4:vacuum4 ok]=OFF TIMEOUT,LBL[40] ;
122: ;

```

```

123: R[12:selejt darabszam]=R[12:selejt darabszam]+4 ;
124: ENDIF ;
125: ;
126: !ciklus szamlalo ;
127: R[11:ciklus szamlalo]=R[11:ciklus szamlalo]+1 ;
128: ;
129: !jo munkadarab % szamlalo ;
130: R[15:hany db md]=R[11:ciklus szamlalo]*4 ;
131: R[14:hany % jo ]=R[13:jo darabszam]/R[15:hany db md] ;
132: R[14:hany % jo ]=R[14:hany % jo ]*100 ;
133: ;
134: !ciklus ujra ;
135: JMP LBL[1] ;
136: ;
137: ;
138: ;
139: ;
140: ;
141: ;
142: !hiba eseten haza program ;
143: LBL[40] ;
144:L P[4:megfogas elotti ] 100mm/sec FINE ;
145: LBL[30] ;
146:L P[3:mould fellette] 100mm/sec FINE ;
147: LBL[20] ;
148:J P[2:keszenleti poz] 100% FINE ;
149: LBL[10] ;
150:J P[1:Alap pozicio] 100% FINE ;
151: ;
152: !hibakod kezelo ;
153: CALL CLEARUSER ;
154: ;

```

```

155: !hibakod kiolvassa ;
156: SELECT R[10:hibakod]=1,JMP LBL[11] ;
157:     =2,JMP LBL[12] ;
158:     =3,JMP LBL[13] ;
159:     =4,JMP LBL[14] ;
160:     =5,JMP LBL[15] ;
161:     =6,JMP LBL[16] ;
162:     =7,JMP LBL[17] ;
163:     =8,JMP LBL[18] ;
164:     =9,JMP LBL[19] ;
165: ;
166: ;
167: LBL[11] ;
168: MESSAGE[szerszam nincs zarva] ;
169: MESSAGE[gep nincs autoban] ;
170: JMP LBL[2] ;
171: ;
172: LBL[12] ;
173: MESSAGE[szerszam nincs nyitva] ;
174: JMP LBL[2] ;
175: ;
176: LBL[13] ;
177: MESSAGE[szerszam nincs nyitva] ;
178: JMP LBL[2] ;
179: ;
180: LBL[14] ;
181: MESSAGE[felepult a vakuum ] ;
182: MESSAGE[tappancson van valami] ;
183: J P[10:vizsgalati pont ] 50% FINE ;
184: WAIT RI[1:Vacuum1 ok]=OFF AND RI[2:vacuum2 ok]=OFF AND
RI[3:vacuum3 ok]=OFF AND RI[4:vacuum4 ok]=OFF ;
185: WAIT 10.00(sec) ;

```

```

186: JMP LBL[2] ;
187: ;
188: LBL[15] ;
189: MESSAGE[kiloko nincs elol] ;
190: JMP LBL[2] ;
191: ;
192: LBL[16] ;
193: MESSAGE[nem epult fel a vakuum] ;
194: JMP LBL[2] ;
195: ;
196: LBL[17] ;
197: MESSAGE[kiloko nincs hatul] ;
198: JMP LBL[2] ;
199: ;
200: LBL[18] ;
201: MESSAGE[szerszam nem csukodik] ;
202: JMP LBL[2] ;
203: ;
204: LBL[19] ;
205: MESSAGE[md nem esett le] ;
206: J P[10:vizsgalati pont ] 50% FINE ;
207: WAIT RI[1:Vacuum1 ok]=OFF AND RI[2:vacuum2 ok]=OFF AND
RI[3:vacuum3 ok]=OFF AND RI[4:vacuum4 ok]=OFF ;
208: WAIT 10.00(sec) ;
209: JMP LBL[2] ;

```