

RILIAM

HUSRB/1602/41/0012

IPARI ROBOTOK PROGRAMOZÁSA



TARTALOMJEGYZÉK

Tartalomjegyzék.....	1
1 A robotok részei	3
1.1 Vezérlő.....	3
1.2 Robot	4
1.3 Teach Pendant.....	4
2 Robotok mozgatása	6
2.1 Program kiválasztása	7
2.2 Program elindítása	8
2.3 A Tool Center Point (TCP).....	9
2.3.1 A TCP beállítása.....	11
3 Programozási eszközök	15
3.1 Új program létrehozása.....	15
3.2 Pozíciók rögzítése.....	15
3.3 Program tesztelése manuális módban.....	17
3.4 Mozgás változtatása.....	18
3.5 Megjegyzések beszúrása.....	18
3.6 User Frame tanítása	19
3.7 Offset és Position Register	21
3.8 Sorok törlése a programban.....	23
3.9 For To függvény	24
3.10 If függvény.....	27
3.11 Jump Label.....	27
3.12 FINE és CNT függvények	28
3.13 Wait függvény.....	28
3.14 Input/Output	28
3.15 Több program párhuzamos futtatása.....	31
3.16 Különböző STOP típusok	31
4 Programozás ROBOGUIDE környezetben.....	33
4.1 A beszerzett robotok modelljei.....	33
4.2 A beszerzett robotok alkalmazási területei	34

4.3	A ROBOGUIDE felhasználofelületének elemei	34
4.4	Menüpontok.....	35
4.5	A cellák elemei	42
4.5.1	Robot vezérlők	43
4.5.2	Egyéb elemek	44
4.5.3	Szenzor egységek	47
4.6	Programok létrehozása	48
4.6.1	Teach Pendant programok.....	48
4.6.2	Szimulációs programok.....	49

1 A ROBOTOK RÉSZEI

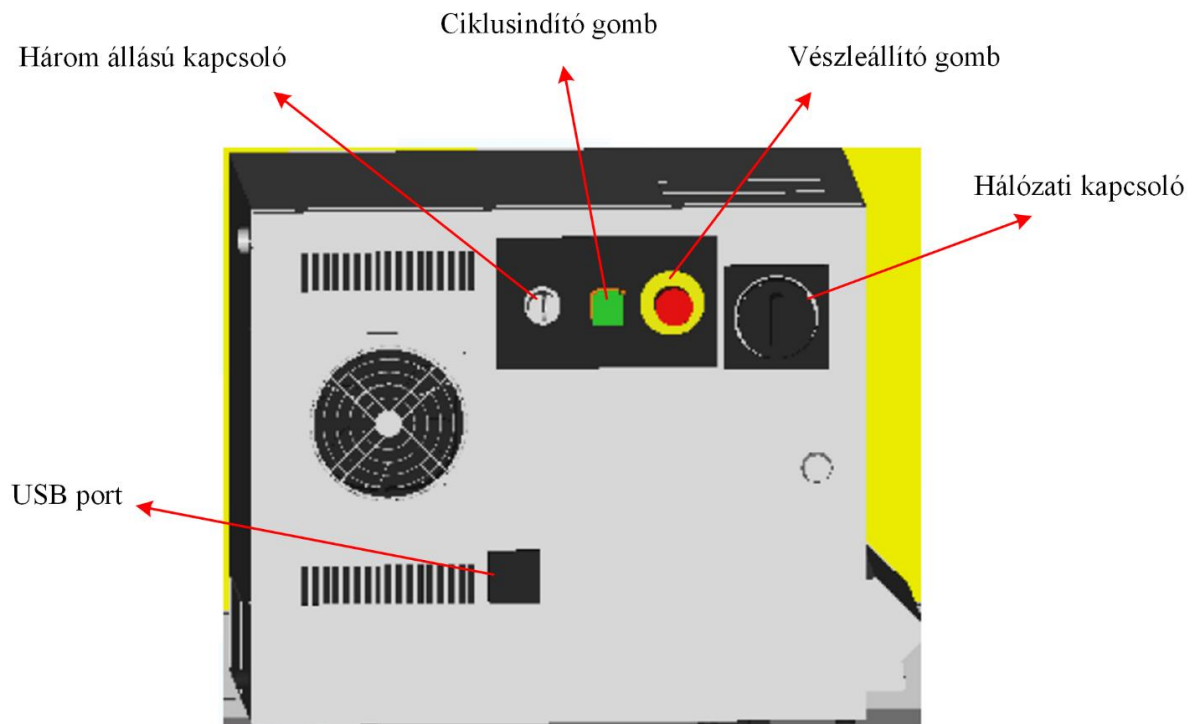
A robotok három fő egységből állnak:

- Vezérlő
- Robot
- Teach Pendant

1.1 VEZÉRLŐ

A vezérlő irányítja a robotot és ebben található a hardver és a szoftver, így ez nevezhető a robot „agyának”. A vezérlőhöz kell csatlakoztatni az alkalmazott robotot és a Teach Pendant-et. Ebben az egységben találhatóak még a kommunikációs kártyák csatlakozói és a különböző kimenetek és bemenetek csatlakozói is.

A vezérlőben található szoftver határozza meg a robot irányításához szükséges paramétereket, irányítja a robotban található motorokat, olvassa a szenzorokat, vezérli az egyéb elemeket, stb.



A vezérlőn található eszközök szerepe:

- Hálózati kapcsoló – ezzel tudjuk áram alá helyezni a rendszert.
- Három állású kapcsoló – üzemmód választására használható. Választható üzemmódok:
 - Automatikus – teljes sebességgel futtatható a program ciklikusan. Ebben az esetben nincs lehetőség a robot manuális mozgására.
 - Csökkentett sebességű manuális (T1) – manuálisan mozgatható a robot vagy tesztelhetőek programok, de csak csökkentett maximális sebességgel.
 - Teljes sebességű manuális (T2) – szintén manuális mozgásra vagy programok tesztelésére használható, viszont teljes sebesség is elérhető.

- Vészleállító gomb – vészhelyzet esetén bármikor megszakítható a robot működése.
- Ciklusindító gomb – automatikus üzemmód esetén ezzel a gombbal elindítható a program ciklikus futtatása.
- USB port – pen drive csatlakoztatására használható, például backup készítésére vagy program másolására.

1.2 ROBOT

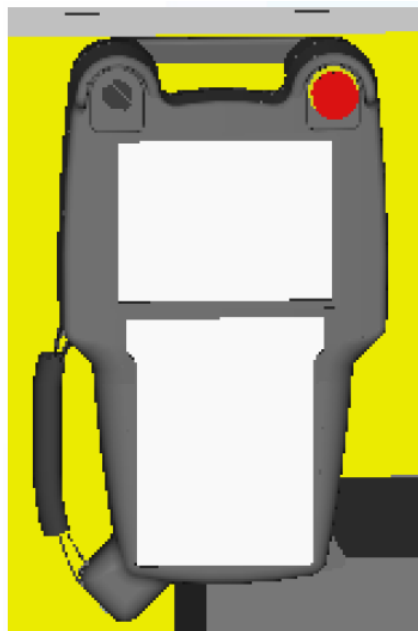
A robotok szegmensekből és csuklókból álló mechanikai eszközök, amelyekhez valamilyen effektor csatlakozik. Az effektor lehet valamilyen megfogó, hegesztő, festő, stb.

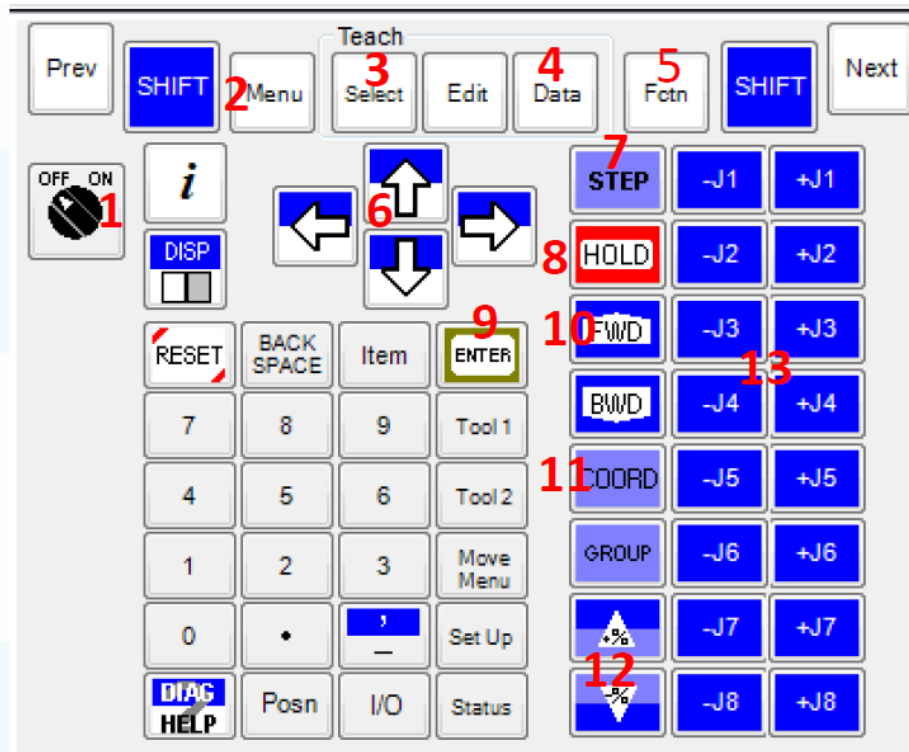
A csuklók lehetnek rotációs és transzlációs csuklók, melyeket általában szervó motorok hajtanak, és egymástól függetlenül mozgathatók. A transzlációs csuklók a csukló tengelye mentén végeznek haladó mozgást, míg a rotációs csuklók forgó mozgást tesznek lehetővé a csukló tengelye körül.

1.3 TEACH PENDANT

A Teach Pendant a robot kezeléséhez és programozásához használható, tehát ez az interfész a robot és a felhasználó között. Lehetővé teszi a robot manuális mozgását, programok írását és programok tesztelését.

A Teach Pendant tartalmaz egy képernyőt, vészleállító gombot, illetve a mozgáshoz és programozáshoz felhasználható gombokat.





A Teach Pendant-en található gombok funkciója:

1. Teach Pendant Enable kapcsoló – Teach Pendant engedélyezése.
2. Menu gomb – képernyőmenü megnyitása.
3. Select gomb – program lista megnyitása.
4. Data gomb – regiszter lista megnyitása.
5. Function (Fctn) gomb – funkció menü megnyitása.
6. Cursor gombok – kurzor mozgatása.
7. Step gomb – léptető mód bekapcsolása/kikapcsolása.
8. Hold gomb – program megállítása.
9. Enter gomb – belépés a kiválasztott pontba.
10. Forward (FWD) gomb – a program léptetése a következő sorba.
11. Coordinate (COORD) gomb – koordinátarendszer váltása.
12. Override gombok – sebesség változtatása.
13. Jog gombok – robot mozgatása teach módban.

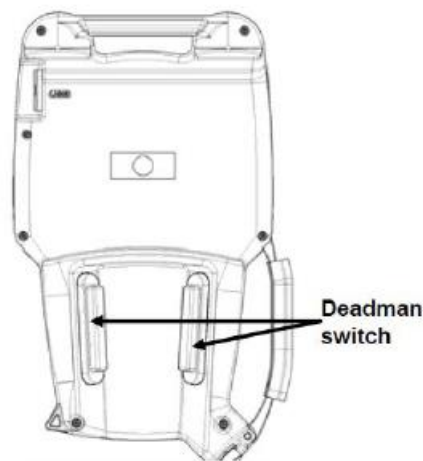
A kék gombok csak a Shift gomb lenyomása esetén lesznek aktívak, míg a félig fehér félig kék gomboknál megváltozik az adott gombok funkciója. Például a menüben a kurzor gombok segítségével egyesével tudunk mozogni a Shift gomb lenyomása nélkül, míg a Shift lenyomásával lapozni tudunk a menüben.

2 ROBOTOK MOZGATÁSA

A robotok mozgatása (jogging) a Teach Pendant segítségével történik.

A folyamat során a Teach Pendant engedélyezve kell, hogy legyen, és a vezérlőn található háromállású kapcsolót manuális üzemmódra kell állítani (T1 vagy T2 állás). Az üzemmódok közül természetesen a csökkentett sebességű mód (T1) a biztonságosabb.

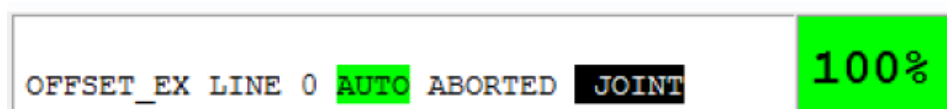
A mozgatás a jog gombok segítségével történik, viszont a Teach Pendant hátsó oldalán található ún. Deadman Switch-ek közül az egyik folyamatosan be kell, hogy legyen nyomva. Ezek a Deadman Switchek háromállású gombok, melyeknél csak a középső állás teszi aktívvá a motorokat. Ennek biztonsági szerepe van, mivel a gomb elengedésével gyorsan tudunk reagálni vészhelyzet esetén, a benyomott állás pedig akár valamilyen vészhelyzet hatására létrejött görcsöt is jelenthet a felhasználónál. A gomb elengedésével a robot hiba módba kerül, melyet a Reset gomb segítségével tudunk megszüntetni.



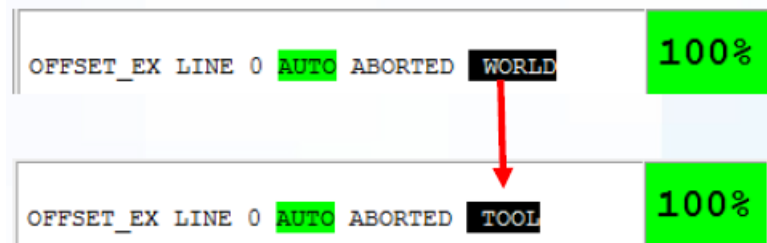
A mozgatáshoz a jog gombok mellett a Shift gombot is állandóan benyomva kell tartani.



A mozgatás történhet több koordinátarendszer szerint. A JOINT koordinátarendszer csuklónkénti mozgatást tesz lehetővé. Ebben az esetben például a +J1 gomb pozitív irányban mozgatja az első csuklót, a -J1 pedig negatív irányba.



A többi koordináta-rendszer-nél az első három sor gomb segítségével (J1-J3) az X, Y és Z mentén végzünk haladó mozgást, míg a következő három sor (J4-J6) forgó mozgást tesz lehetővé a három tengely körül. A koordináta-rendszerek pozíciójukban különböznek. A világkoordináta-rendszer-nél (WORLD) a robot alapjánál felvett koordináta-rendszer szerint mozgunk, míg például TOOL koordináta-rendszer esetén a szerszám alapján felvett koordináta-rendszer szerint történik a mozgás.



A USER koordináta-rendszereket a felhasználó önállóan határozza meg. Ilyen koordináta-rendszerekből maximum 9 definiálható. Ha nincs definiálva egy ilyen koordináta-rendszer sem, akkor a robot a WORLD koordináta-rendszert fogja alkalmazni. A koordináta-rendszerek között a COORD gomb megnyomásával tudunk váltani.

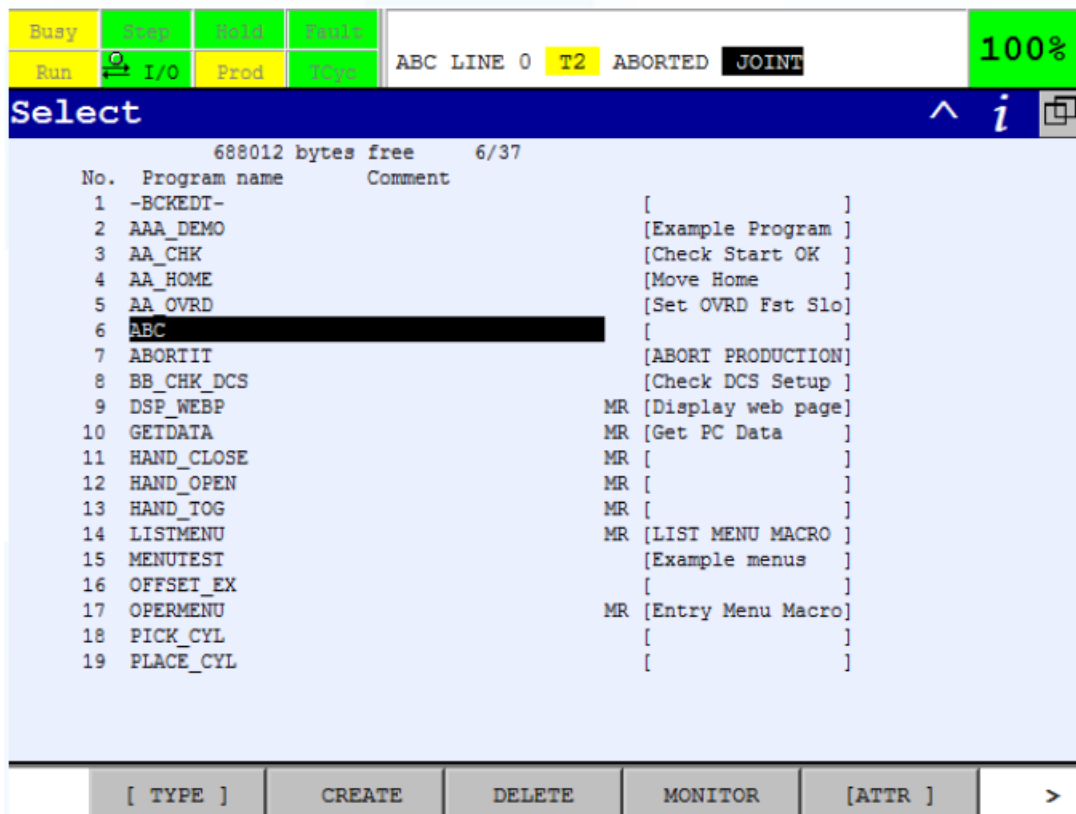


2.1 PROGRAM KIVÁLASZTÁSA

Adott program megnyitásához először a program listát szükséges előhozni, melyet a Select gomb segítségével tehetünk meg.



A listán a kurzor gombok segítségével mozoghatunk.



A kiválasztott program megnyitásához pedig az Enter gomb megnyomása szükséges, amely után a Teach Pendant képernyőjén megjelenik a program. Automatikus indításra pedig ez válik kiválasztott programmá.



2.2 PROGRAM ELINDÍTÁSA

A programok auto módban való elindítása előtt fontos, hogy eltüntessünk minden akadályt a robot környezetéből.

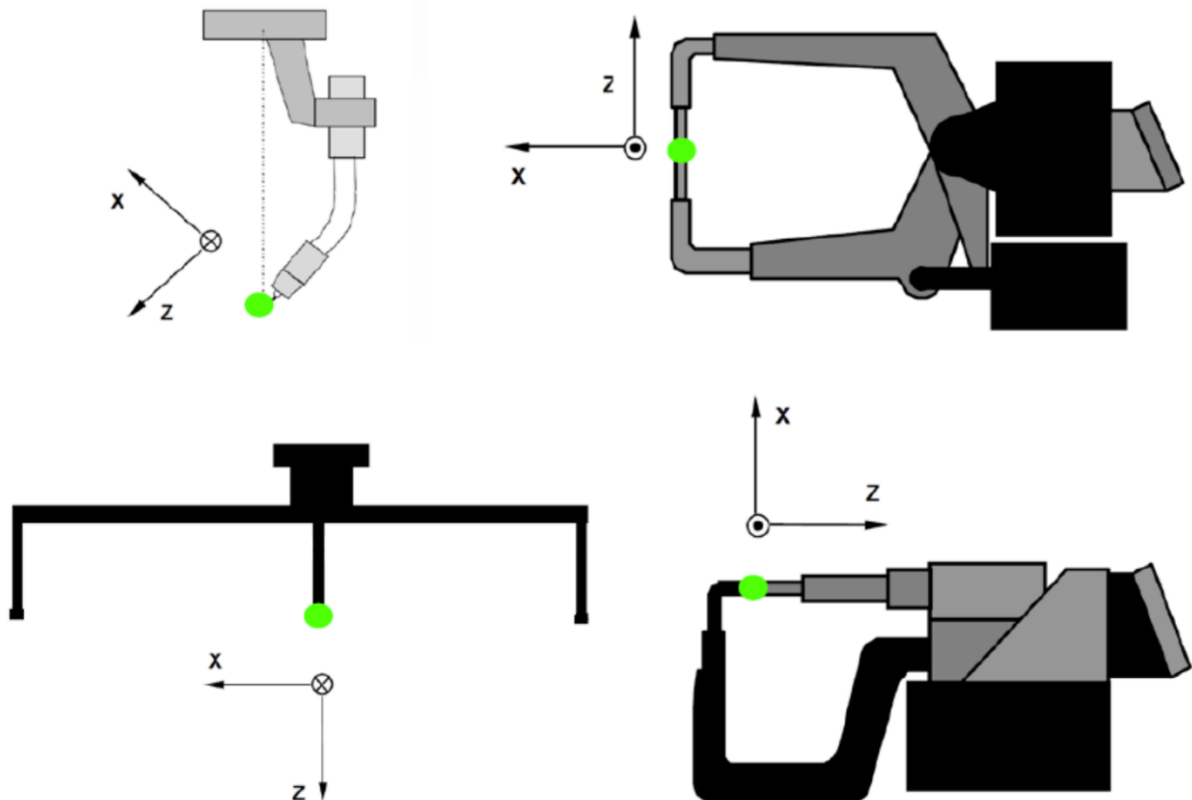
A program indításához először be kell állítani az auto módot, melyhez a vezérlőn a három állású kapcsolót automatikus üzemmódba kell állítani, a Teach Pendant-en az eszközt nem engedélyezett módra a kapcsoló segítségével.

A vezérlőn a ciklusindító gomb megnyomásával pedig elindul a kiválasztott program.

A program adott beállítás esetén megkérdezheti, hogy minden eszköz a megfelelő helyen van-e. Ebben az esetben csak akkor fog elindulni ha az F2 gombot megnyomjuk a Teach Pendant-en. Az F2 gomb segítségével ugyanúgy meg is állíthatjuk a program futását a ciklus után.

2.3 A TOOL CENTER POINT (TCP)

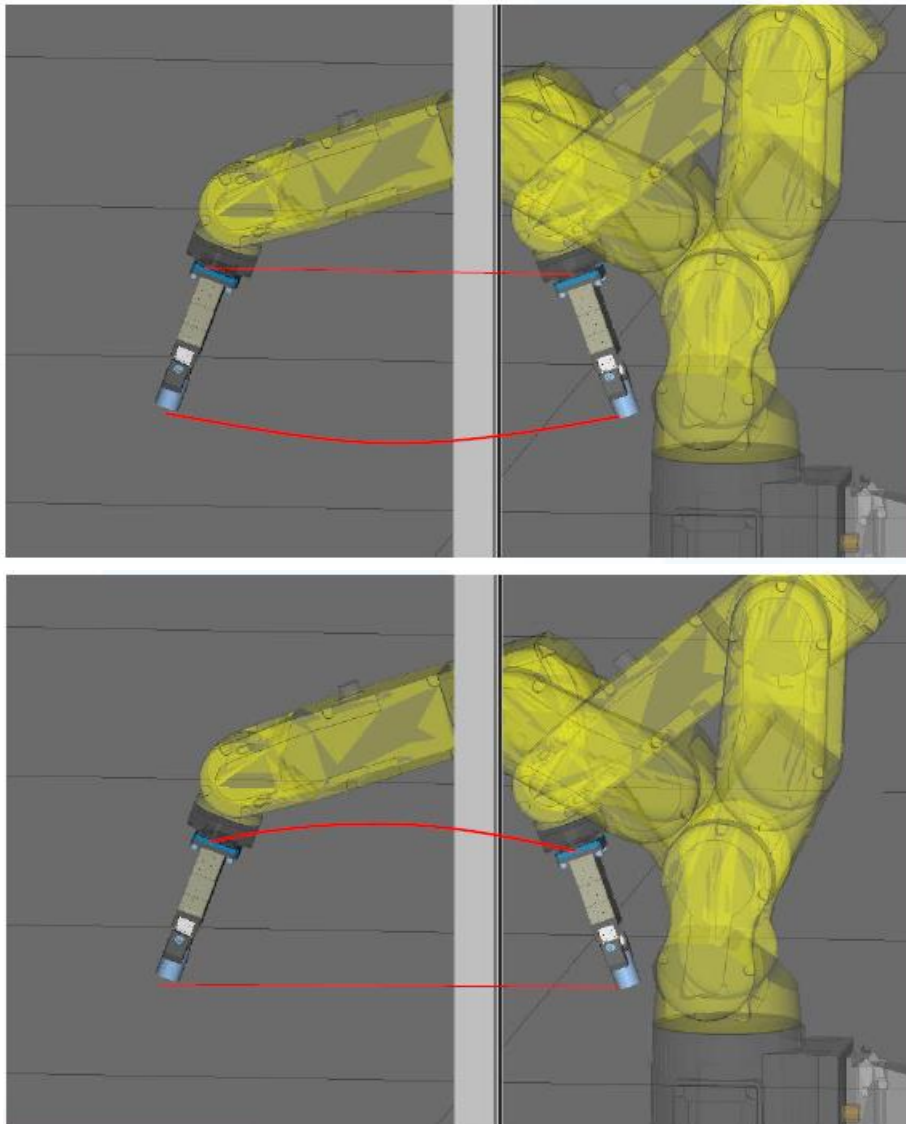
A könnyebb mozgás érdekében fontos a Tool Center Point (TCP), azaz a szerszám középpontjának a beállítása. Ez a pont lényegében egy relatív pozíció az utolsó csukló végéhez viszonyítva. Általában az effektor „munkapontját” definiáljuk ennek a pontnak, például egy megfogó vagy egy ívhegesztő pisztoly középpontját. A következő ábrán néhány példa látható a TCP helyzetére (a koordinátarendszerek a TCP koordinátarendszerét reprezentálják, míg a zöld pontok jelölik a TCP helyét).



Fontos tulajdonsága, hogy egyszerűsíti a pontok felvételének folyamatát. Például, lehetőségünk van a végpont körüli forgatásra, ami definiált TCP esetén a szerszám végpontja körül fog történni.

Jelentős szerepe van még a mozgás tervezésénél is. Az egyenes vonalú, vagyis lineáris mozgás esetén a szerszám végpontja az esetek többségében görbe mentén mozogna, mivel a mozgás tervezése a TCP alapján történik.

Lineáris mozgás tervezésére nem beállított és bállított TCP esetén a következő ábrákon láthatóak példák.



A TCP helye kihat a futás közbeni mozgásra is, mivel a vezérlő szoftvere ez a pont alapján határozza meg a sebességeket. Ha nem lenne beállítva, akkor a szerszámunk végpontjának a sebessége nem lenne lineáris, mivel a robot végpontját alkalmazná a tervezés során.

2.3.1 A TCP BEÁLLÍTÁSA

A TCP megfelelő beállítása biztosítja, hogy a szerszám koordinátarendszere a szerszám tengelyeinek megfelelően van meghatározva.

Mozgatás előtt mindig le kell ellenőrizni, hogy be van-e állítva a megfelelő TCP. A pillanatnyi TCP a SETIND funkcióval állítható be.

The screenshot shows a CNC control interface with the following elements:

- Top status bar: Busy, Stop, Hold, Fault, Run, I/O, Prod, Stop, AAA_DEMO LINE 0, T2, ABORTED, WORLD, 100%
- Header: SETUP Frames
- Table of Tool Frames:

Tool Frame	X	Y	Z	Comment
1	0.0	0.0	115.0	[EGP40]
2	-0	38.6	228.5	[Eoat2]
3	0.0	0.0	0.0	[Eoat3]
4	0.0	0.0	0.0	[Eoat4]
5	0.0	0.0	0.0	[Eoat5]
6	0.0	0.0	0.0	[Eoat6]
7	0.0	0.0	0.0	[Eoat7]
8	0.0	0.0	0.0	[Eoat8]
9	0.0	0.0	0.0	[Eoat9]
10	0.0	0.0	0.0	[Eoat10]

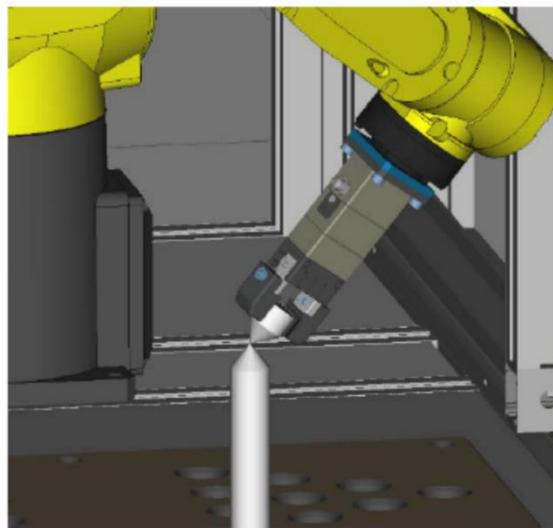
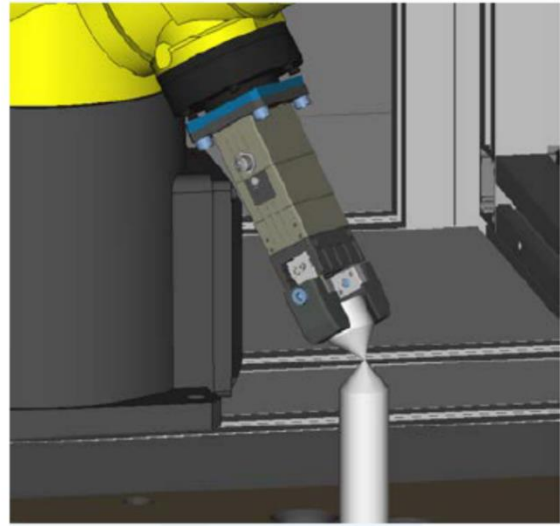
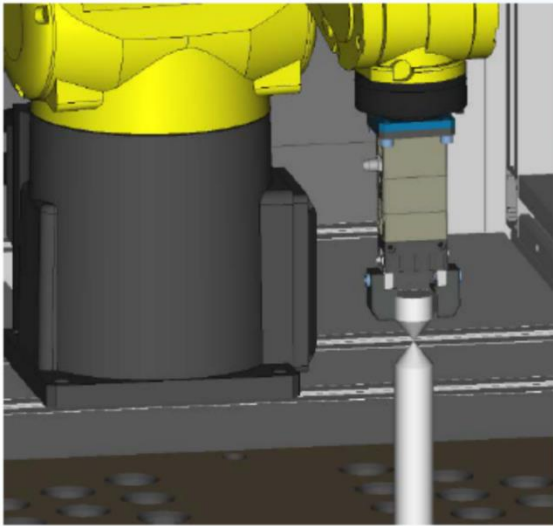
Active TOOL \$MNUTOOLNUM[1] = 1

Bottom buttons: [TYPE], DETAIL, [OTHER], CLEAR, SETIND (circled in red)

A TCP tanítására több módszer létezik.

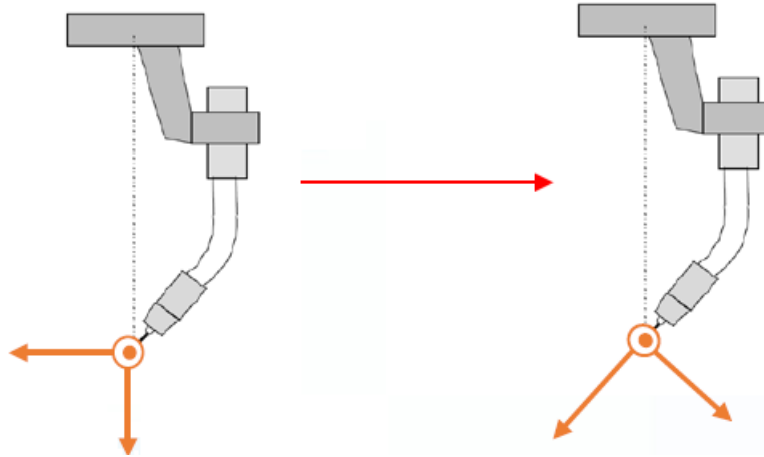
HÁROM PONTOS MÓDSZER

A TCP beállításának legegyszerűbb módja a három pontos módszer. Ennek lényege, hogy a pontot három különböző irányból érintjük. A pontok felvételére a következő ábrán látható egy példa.

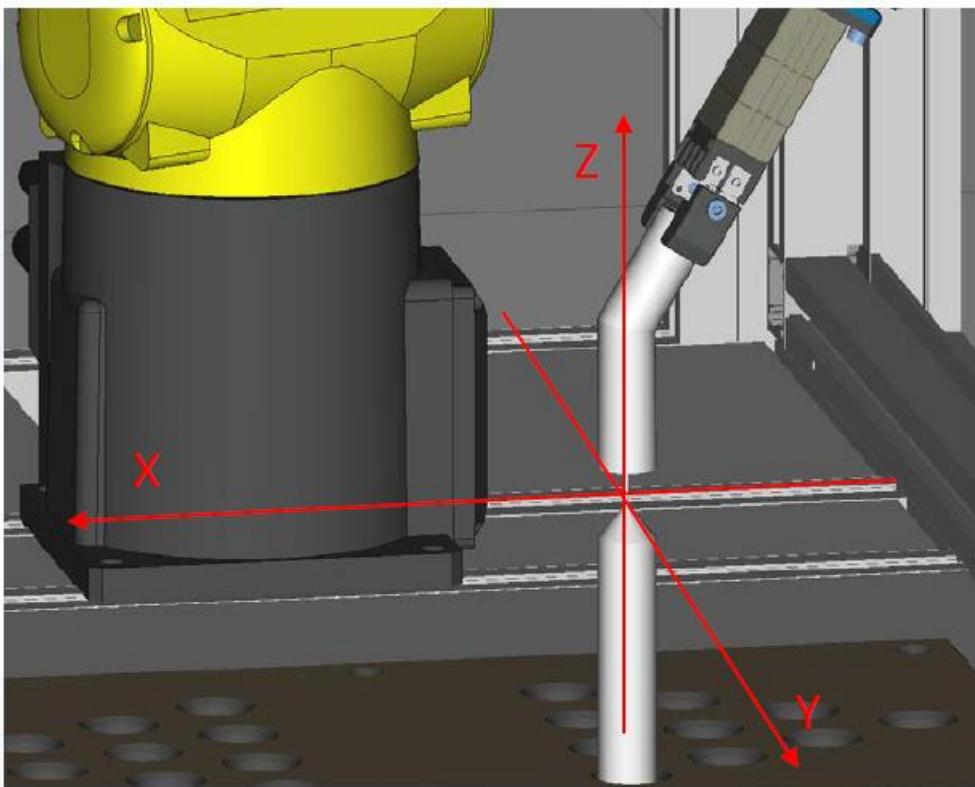


HAT PONTOS MÓDSZER

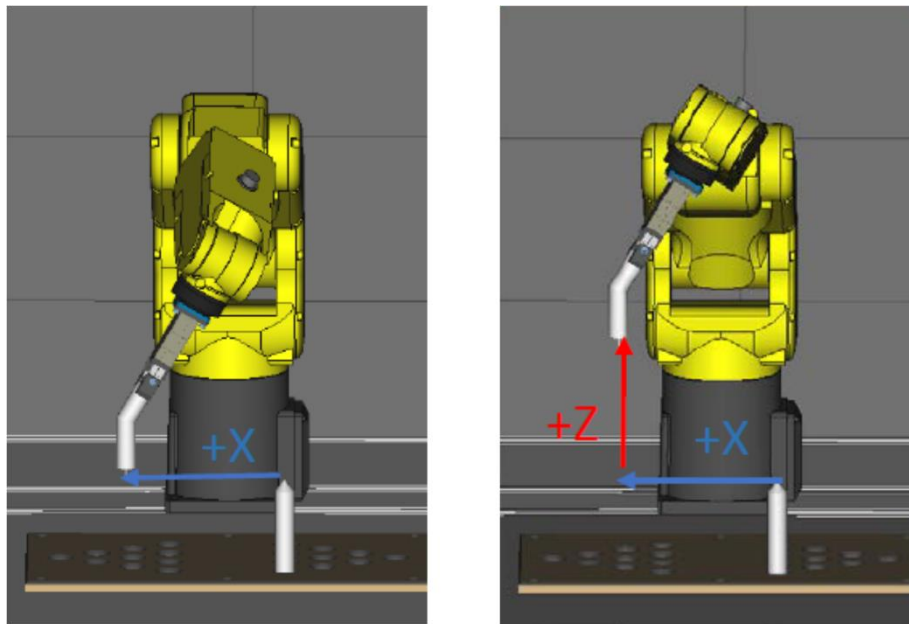
A három pontos módszer csak akkor használható, ha a szerszám egyenes és egy vonalban van az utolsó csukló fedőlapjával. Ellenkező esetben viszont a szerszám koordinátarendszere nem lenne megfelelő, mivel a tengelyek nem lennének a szerszám végpontjához igazítva.



A hat pontos módszer segítségével a koordinátarendszer orientációja is megadható. Ennél az eljárásnál az első három pont megegyezik az előző módszernél alkalmazottakkal. A többi pont rögzítéséhez először be kell állítani a szerszám végpontját úgy, hogy a világoordinátarendszer X tengelyére merőleges legyen.



Ezután azokban az irányokban kell mozogni, amelyekben szeretnénk, hogy a koordinátarendszer tengelyei legyenek. Először X irányban mozdítjuk el és veszünk fel egy pontot, majd a Z tengely irányában.



DIRECT LIST MÓDSZER

Ennél a módszernél a TCP koordinátarendszerének meghatározásához minden pozíciót és orientációt manuálisan adunk meg. Az előző eljárásokhoz képest ez sokkal gyorsabb és egyszerűbb, viszont ismernünk kell hozzá a szerszám dimenzióit.

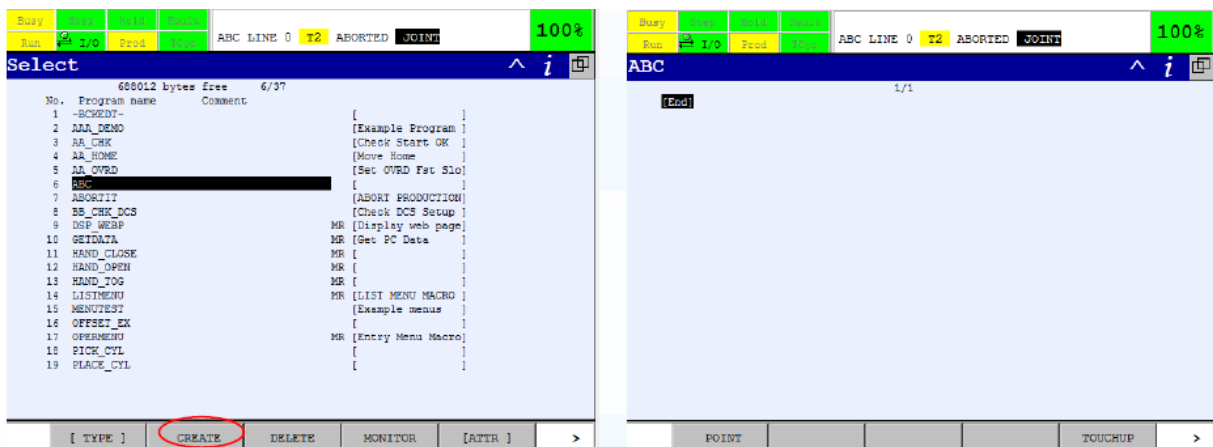
3 PROGRAMOZÁSI ESZKÖZÖK

Ipari feladatok esetén általában az a cél, hogy a robotok ugyanazt a mozgássorozatot végezzék újra és újra. Így, egyszer készítünk egy programot, majd a robot állandóan ezt a munkát végzi. Ez a munkát nagyon kiszámíthatóvá és egyben nagyon hatékonyá teszi.

3.1 ÚJ PROGRAM LÉTREHOZÁSA

Új program létrehozásához először mindenképp manuális üzembe helyezzük a robotot (célszerű a T1 módot alkalmazni), majd engedélyezzük a Teach Pendant működését.

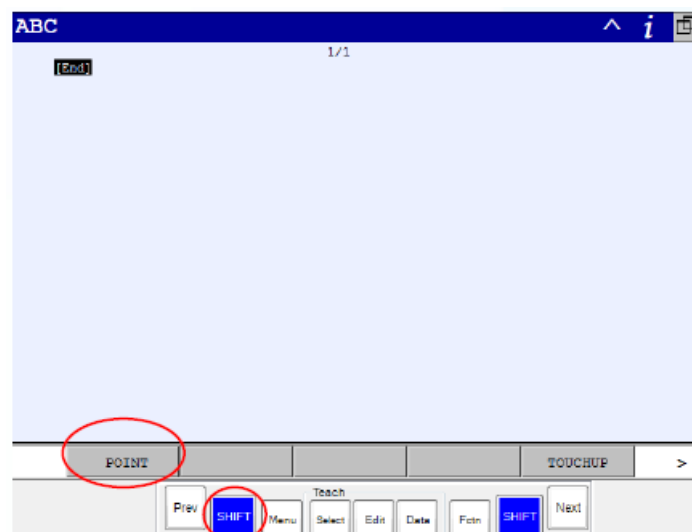
Ezután a Select gomb segítségével megnyitjuk a programlistát, majd megnyomjuk az F2 „Create” funkciógombot. Először be kell írni a program nevét, majd megnyomni az Enter gombot, amely után megjelenik az új üres programunk.



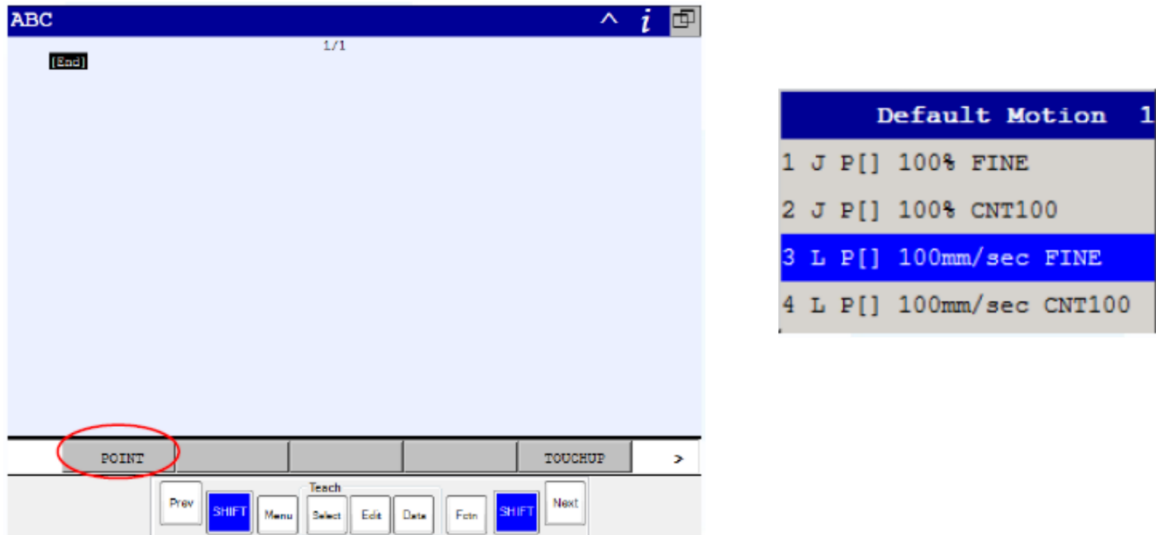
3.2 POZÍCIÓK RÖGZÍTÉSE

Két lehetőség van pozíció rögzítésére miután megfelelő helyre mozgattuk a robotot.

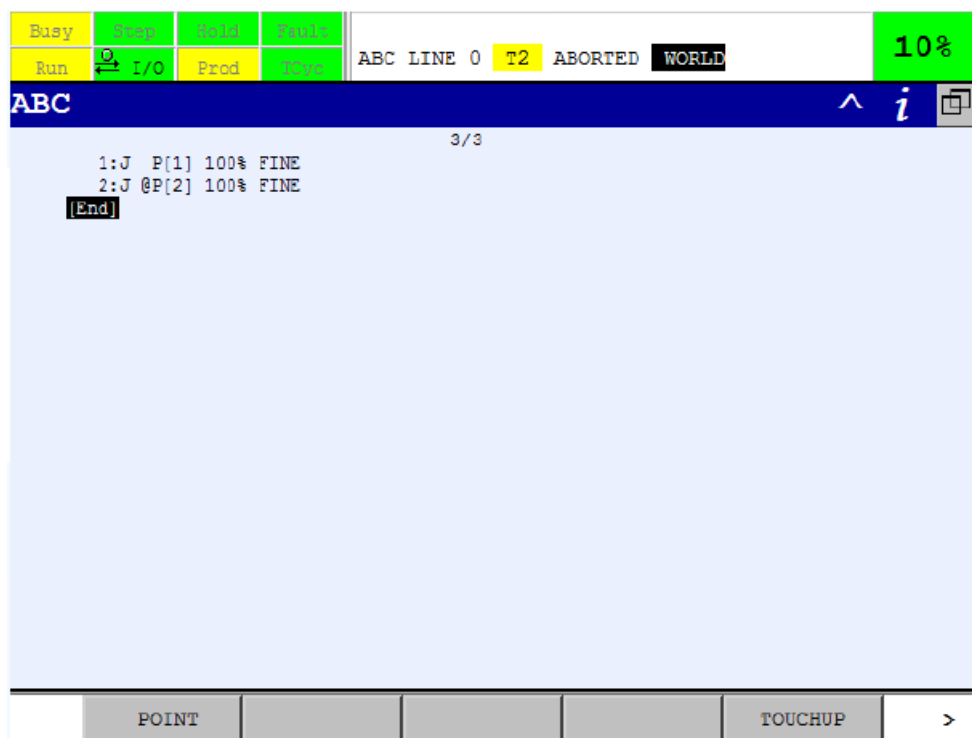
Az első lehetőség a Shift gomb és a Point funkciógomb egy időben történő megnyomásával lehetséges.



Ilyenkor a pozíciót egy adott mozgáshoz adja hozzá, amely Joint mozgásnak lesz beállítva. A másik lehetőségnél csak a Point gombot nyomjuk meg, és megnyílik egy ablak amelyben kiválaszthatjuk a mozgás típusát.



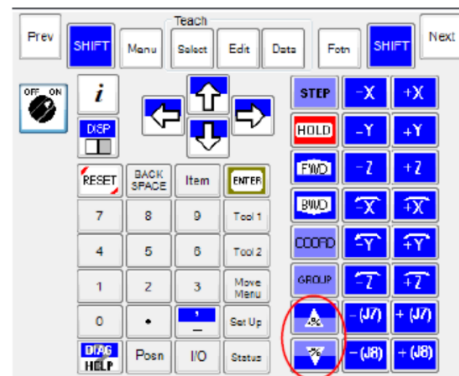
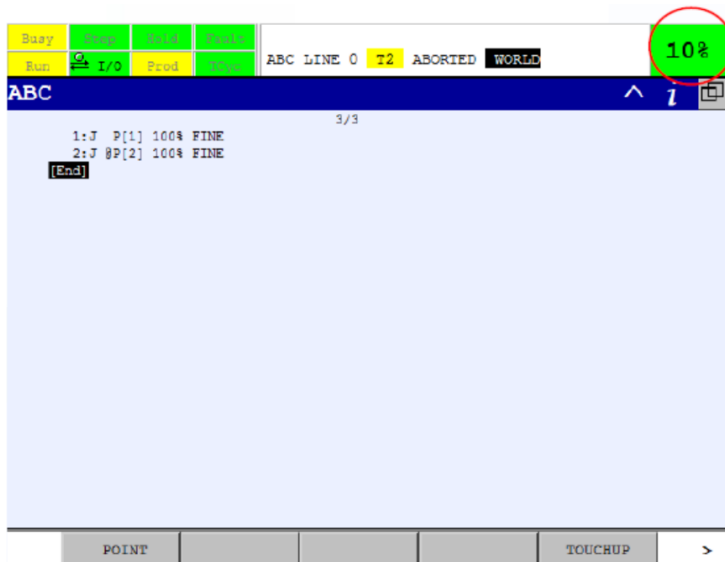
Két pont felvételével például a következő ábrán látható egyszerű programot kapjuk.



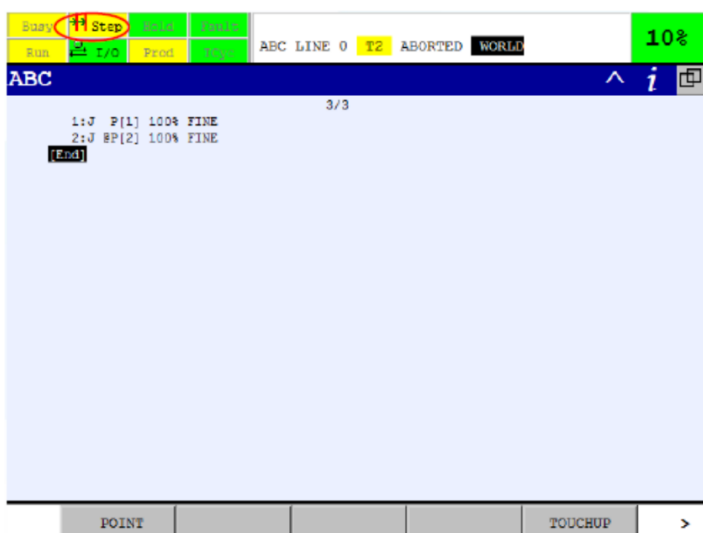
A kódban található „@” azt jelöli, hogy a robot jelenleg abban a pozícióban található.

3.3 PROGRAM TESZTELÉSE MANUÁLIS MÓDBAN

Az előző két mozgásból álló program teszteléséhez célszerű először egy alacsonyabb sebességhatárt beállítani az Override gombok segítségével, hogy nem várt mozgás esetén a robot ne okozzon kárt. Alacsonyabb sebességnél a felhasználónak több ideje van reagálni és megállítani a robotot.



A biztonságosabb teszteléshez fontos még a lépésenkénti futtatási mód beállítása. Ezt a Teach Pendant-en található Step gomb segítségével tehetjük meg. Ilyen módban a program futása lépésenként, gombnyomásra történik.



A futtatáshoz a Deadman Switch-et be kell nyomni megfelelő állásba és úgy tartani, illetve a Shift gombot szintén folyamatosan benyomva tartva megnyomni a FWD gombot.

Ha úgy véljük, hogy a program biztonságosan fut, akkor megnövelhetjük a sebességet és visszaállíthatjuk a lépésenkénti módot is folyamatosra.

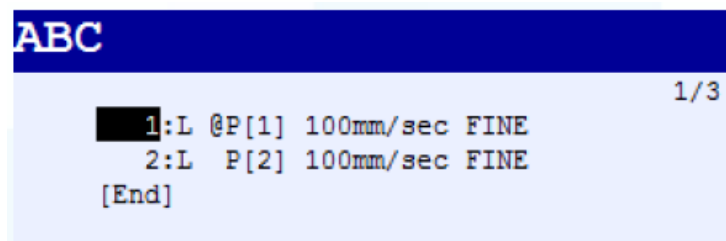
3.4 MOZGÁS VÁLTOZTATÁSA

Alapesetben a mozgás joint (J) típusú. Ez úgy módosítható, hogy a kurzort az adott mozgáshoz visszük, majd megnyomjuk a „Type” funkció gombot. Így négy különböző mozgástípus közül választhatunk.

A joint mozgás során az útvonal nem feltétlenül egyenes, a szerszám orientációja mozgás közben pedig nincs meghatározva.

A mozgás átállítható lineáris (L) mozgásra. Ilyen mozgás esetén a TCP egy egyenes mentén mozog, és orientációja nem változik.

Ha a kurzorral a mozgás sebességére mozgunk, akkor a Type gomb segítségével a sebesség módosítására van lehetőségünk. Joint mozgás esetén %-ban adható meg a sebesség, míg lineáris mozgásnál mm/s-ban.



```
ABC  
1/3  
1:L @P[1] 100mm/sec FINE  
2:L P[2] 100mm/sec FINE  
[End]
```

KÖR ALAKÚ MOZGÁS

A joint és linear mozgások mellett létrehozhatunk kör alakú mozgást is (Circular motion). Ebben az esetben szükségünk van egy segédpontra is az a pont mellett amelybe el szeretnénk jutni. A szoftver ilyenkor egy körív alakú pályát fog tervezni úgy, hogy a kezdőponttól a segédpontra keresztül haladva jussunk a végpontba.

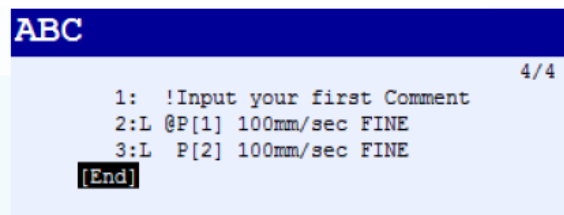
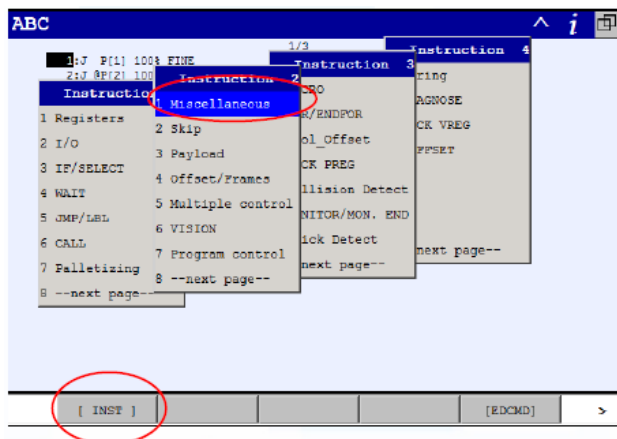
Ilyen mozgás létrehozására először egy előzőekben említett mozgástípust kell létrehozni, majd azt módosítani Circular (C) típusú mozgásra. Mivel így a segédpont még definiálatlan, szükséges még a második sorban található segédpont beállítása is.

3.5 MEGJEGYZÉSEK BESZÚRÁSA

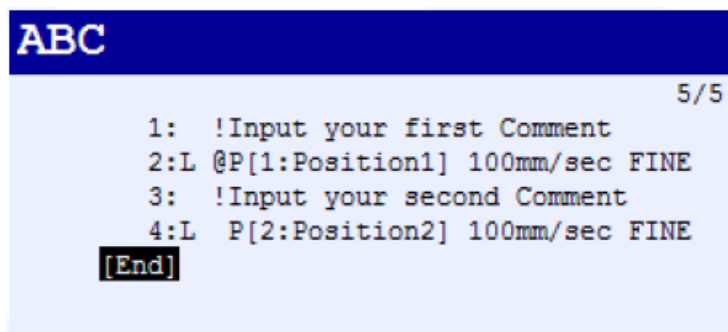
Annak érdekében, hogy mások könnyebben megértsék az általunk készített programot, fontos a kódot megfelelő megjegyzésekkel ellátni.

Először egy új sor beszúrására van szükség, melyhez először az F6 (Next), majd az F5 (EDCMD) gombokat kell megnyomni, és végül kiválasztani az „Insert” opciót. Ekkor a Teach Pendant-en megjelenik egy kérdés, hogy hány sort szeretnénk beszúrni. Az érték megadása után az Enter gomb hatására megjelennek az új sorok.

Ez a folyamat után az F1 (INST) gombot kell megnyomni, majd kiválasztani a „Miscellaneous” menüpontot a felugró menü második lapján. Az újabb lehetőségek közül a „Remark”-ot kell választani. Az Enter megnyomása után be kell írni a megjegyzést, majd a becsukáshoz újra az Enter-t kell megnyomni. A megjegyzéseket tartalmazó sorok felkiáltójellel (!) kezdődnek.

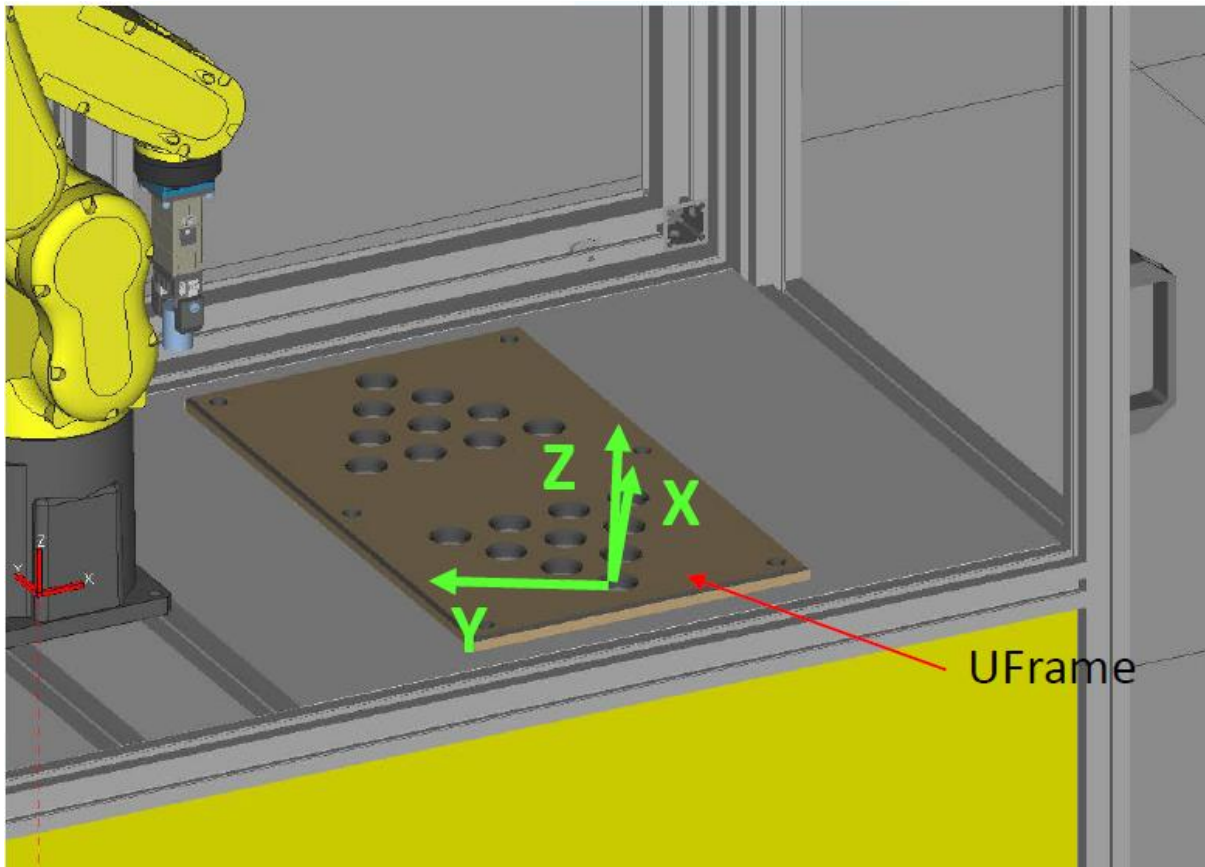


Megjegyzéseket a pozíciókhoz is hozzáadhatunk, amely szintén megkönnyítheti a kód működésének későbbi megértését. Ilyenkor a kurzort a pozícióhoz visszük, majd megnyomjuk az Enter-t, beírjuk a megjegyzést, és újra megnyomjuk az Enter-t.



3.6 USER FRAME TANÍTÁSA

A User Frame (UFrame) egy derékszögű koordinátarendszer amely szintén a robot mozgását és a pontok felvételét könnyíti meg. Ezt a koordinátarendszert a robot munkaterületén kell felvenni. Különösen akkor fontos, ha a felület amin dolgozunk nem párhuzamos a világkoordinátarendszerrel. A User Frame meghatározása és beállítása után a robot lineáris mozgása a megadott koordinátarendszer szerint történik.



A User Frame megadására három pont felvétele szükséges. Először a koordináta-rendszer origójába kell mozdítani a TCP-t. A második pontot a tervezett koordináta-rendszer X tengelyén kell felvenni, a harmadikat pedig az X és Y tengelyek által meghatározott síkon. Fontos, hogy mindhárom pont felvétele ugyanabban a magasságban legyen, tehát, a tervezett koordináta-rendszer szerinti Z tengelyen mindhárom esetben nulla magasságban legyünk. A vezérlőn futó szoftver ennek a három pontnak a segítségével egyértelműen meg tudja határozni a koordináta-rendszer elhelyezkedését a világkoordináta-rendszerhez képest. Szintén fontos, hogy a minél nagyobb pontosság elérése érdekében használjunk nagy távolságokat a mozgások során.

A User Frame felvétele után az F5 (SetInd) gomb megnyomásával tudjuk kiválasztani, hogy ez legyen az aktív User Frame.

A programok készítése során is szükséges a User Frame-ek váltása. A programba való beillesztéshez először az F1 (INST) gombot kell megnyomni, majd az „Offset/Frames” opciót kiválasztani, végül pedig az „UFRAME_NUM”-ot, és a programba beírni a számát amit tanítás során megadtunk.

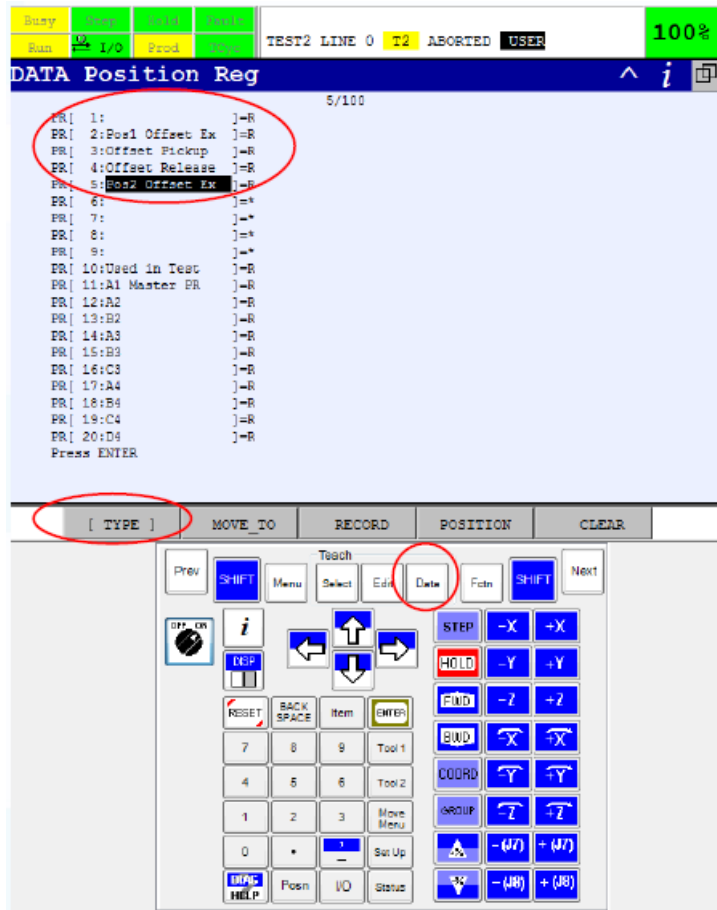
```
PAUSED 1/29
1: !Example
2: !Defining UFrame
3: UFRAME_NUM=6
4:L P[1:Above Pick 1] 250mm/sec FINE
5:L P[2:Pick Pos 1] 250mm/sec FINE
6: !Gripping the Pin
7: RO[7:Open Gripper]=OFF
8: !Making sure the gripper engages
9: WAIT .30(sec)
10:L P[1:Above Pick 1] 250mm/sec FINE
11:L P[3:Above Pick 2] 250mm/sec FINE
12:L P[4:Release Pos 2] 250mm/sec FINE
13: !Releasing the Pin
14: RO[7:Open Gripper]=ON
15:L P[3:Above Pick 2] 250mm/sec FINE
16: !Calling a predefined program
17: CALL AA_HOME
18: WAIT 2.00(sec)
19:L P[3:Above Pick 2] 250mm/sec FINE
20:L P[5:Pick Pos 2] 250mm/sec FINE
```

[INST]

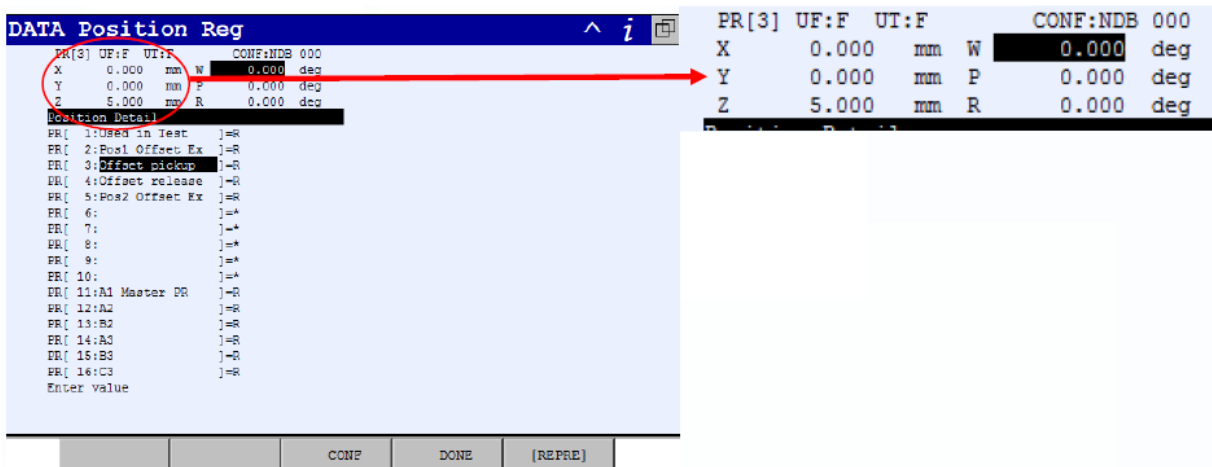
3.7 OFFSET ÉS POSITION REGISTER

Az offset valamilyen eltolást ad hozzá egy adott pozícióhoz. A Position Register (PR) egy változókat tartalmazó memória, amelybe a pozíciók és offsetek helyezhetők.

A pozíciók megadása történhet az értékek manuális megadásával, vagy a robot adott pozíciójának mentésével. A PR-be mentéshez először a Data gombot kell megnyomni, majd a [Type]-nál Position Reg-et kell beállítani, és végül kiválasztani a regisztert amelybe a pozíciót szeretnénk menteni.



Például amikor szeretnénk felemelni egy tárgyat a robot segítségével, akkor több pontra van szükségünk ugyanazokkal az X és Y koordinátákkal, és csak a Z-ben különböznek. Ilyen esetre nagyon hasznos az offset használata. Az offset-et szintén a PR-ben adjuk meg manuálisan. Az említett példa esetén csak a Z koordinátánál szükséges értéket megadni, a többinél pedig nullát kell beállítani.



Programozás során használhatóak a PR-ben található értékek. Először rögzíteni kell egy pontot az eddigiekben említett módon, majd a kurzorral a mozgásban található pozícióra kell mozogni.

```
1:L @P[4] 100mm/sec FINE
[End]
```

1/2

Ezután a Type-ot át kell állítani PR-re, és végül a Choice funkció gomb segítségével kiválasztani mely értéket szeretnénk használni.

```
1:L @P[1] 100mm/sec FINE
[End]
```

Motion Modify 1

- 1 P[]
- 2 PR[]
- 3
- 4
- 5
- 6
- 7
- 8 --next page--

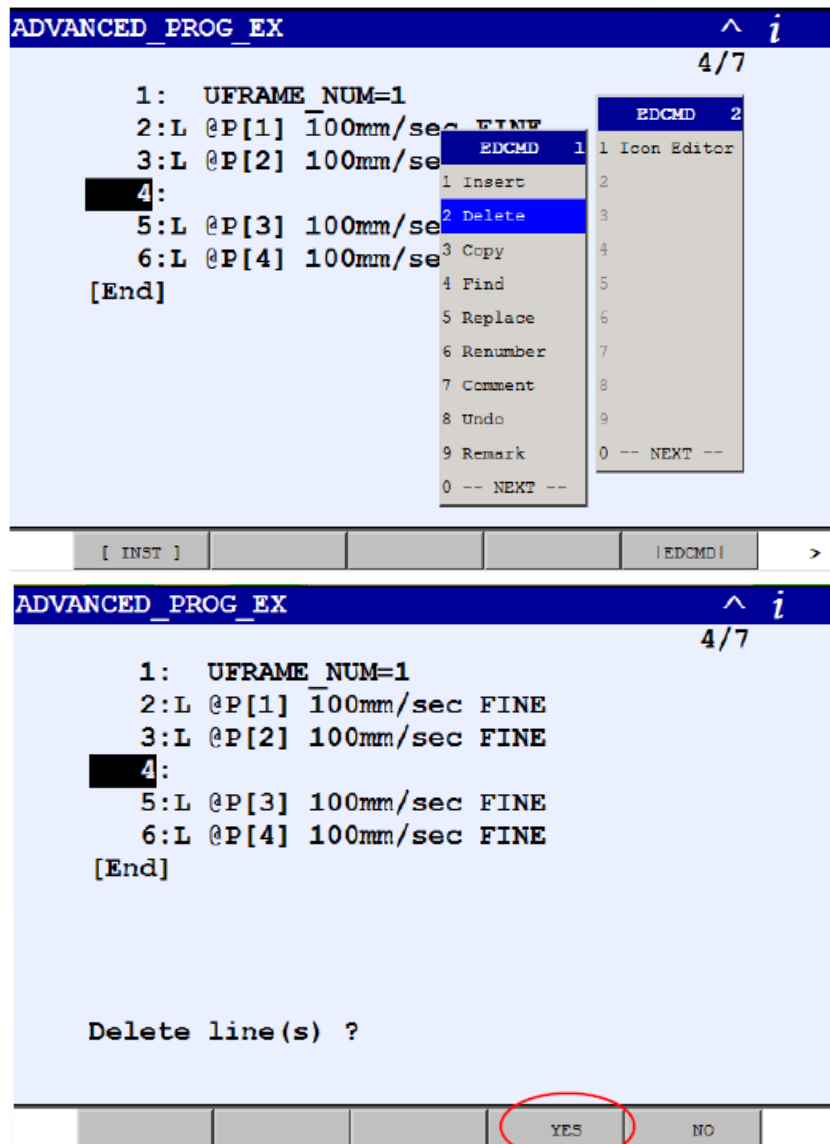
```
1:L @P[1] 100mm/sec FINE
```

[CHOICE]

3.8 SOROK TÖRLÉSE A PROGRAMBAN

Egy sor törléséhez a törölni kívánt sorra mozgunk a kurzorral, majd az EDCMD (F5) gombot megnyomjuk, és kiválasztjuk a Delete pontot. A program erre megkérdezi, hogy biztosan szeretnénk-e törölni a sort, és ha igen, akkor ezt a YES (F4) gombbal tehetjük meg.

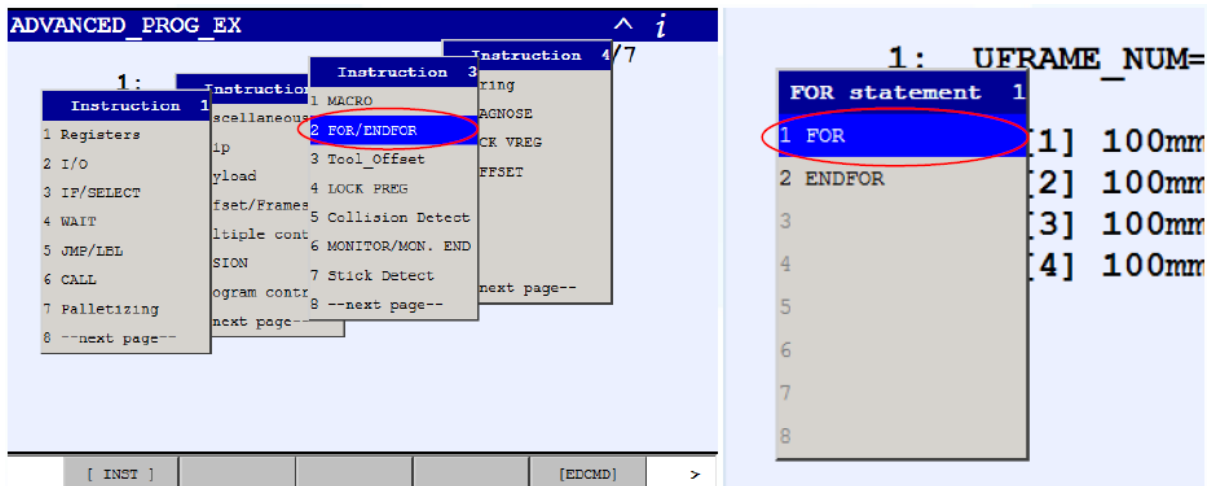
A következő két ábrán üres sor törlésének folyamatára látható példa. Az üres sor nem hat ki a programunkra, mivel nem tartalmaz parancsot.



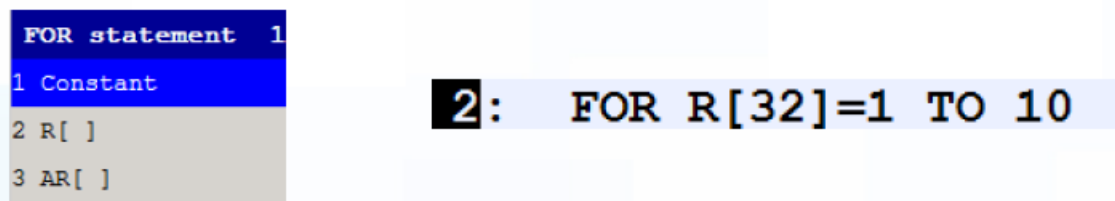
3.9 FOR TO FÜGGVÉNY

Ahhoz, hogy programunk egy mozgást vagy műveletet többször is elvégezhessen egymás után, használhatjuk a „For to” függvényt.

Először az ismételni kívánt sorok fölé berakunk egy üres sort, majd megnyomjuk az INST gombot, kiválasztjuk a FOR/ENDFOR pontot a menüben, majd a FOR opciót.



Ezután kiválasztunk egy regisztert, amelybe a ciklusváltozó értékét fogjuk helyezni. Vigyáznunk kell, hogy a regisztert ne használja egyik programunk sem. Szükséges még két érték megadása, amelyek meghatározzák, hogy a változó mely értéktől mely értékig változzon. Mindkettőnél adhatunk meg konstansot vagy valamilyen regisztert amely tartalmazza a használni kívánt értéket.



Az ábrán látható példában a regiszter értéke 1-től 10-ig fog változni, és minden ismétlésnél 1-el fog növekedni a változó értéke.

A regisztereknél szintén megadható valamilyen azonosító, amely a regiszterlistában és a programban is megjelenik. Ennek beállításához először a Data gomb segítségével megnyitjuk a regiszterlistát, majd átnevezzük a regisztert.

```

R[ 30: ]=-1
R[ 31: ]=0
R[ 32: ForTo Value Ex ]=30
R[ 33: ]=5
R[ 34: ]=0

```

ADVANCED_PROG_EX ^ i

3/7

```

1: UFRAME_NUM=1
2: FOR R[32:ForTo Value Ex]=1 TO 10
3:L @P[1] 100mm/sec FINE
4:L @P[2] 100mm/sec FINE
5:L @P[3] 100mm/sec FINE
6:L @P[4] 100mm/sec FINE
[End]

```

[INST] [EDCMD] >

A „For to” ciklus működéséhez még szükség van egy ENDFOR beillesztésére is, amely a parancssorozat végét fogja mutatni. A hozzáadás menete megegyezik a FOR beillesztésével. A regiszterek pillanatnyi értékét működés közben is figyelhetjük. Ehhez célszerű a képernyőt két részre osztani, mely a Shift és a Disp gombok egyidejű megnyomásával lehetséges. Az aktív képernyőt a hozzáadás után a Disp gomb segítségével változtathatjuk.

ADVANCED_PROG_EX	DATA Registers
1: UFRAME_NUM=1	R[22:]=0
2: FOR R[32:ForTo Value Ex]=1 TO 10	R[23:]=0
3:L @P[1] 100mm/sec FINE	R[24:]=0
4:L @P[2] 100mm/sec FINE	R[25:]=0
5:L @P[3] 100mm/sec FINE	R[26:]=0
6:L @P[4] 100mm/sec FINE	R[27:]=0
7: ENDFOR	R[28:]=0
[End]	R[29:]=-1
	R[30:]=-1
	R[31:]=0
	R[32: ForTo Value Ex]=10
	R[33:]=0
	R[34:]=0
	R[35:]=0
	R[36:]=0
	R[37:]=0
	R[38:]=0
	R[39:]=1
	R[40:]=1
	R[41:]=0
	R[42:]=0

Press ENTER

[INST] [EDCMD] >

3.10 IF FÜGGVÉNY

Az „if” függvény segítségével feltételt helyezhetünk a programunkba. Ha a feltétel igaz, akkor a program végrehajtja a feltételben megadott műveletet, ha nem, akkor pedig a következő parancssorra ugrik.

Például létrehozhatunk egy programot amely az előző részben bemutatott „For to” ciklusváltozójának páratlan értékére meghív egy adott másodlagos programot, páros érték esetén pedig egy másikat. Ehhez szükségünk van egy másik regiszterre is, amelybe a ciklusváltozó kettővel való osztás utáni maradékát helyezzük, ugyanis ez felváltva fog nullás és egyes értékeket eredményezni.

```
ADVANCED_PROG_EX
8/12
1: !UFrame Set
2: UFRAME_NUM=1
3: !Loop Instruction
4: FOR R[32:ForTo Value Ex]=1 TO 10
5: !Setting R33
6: R[33:If Value Ex]=R[32:ForTo Value Ex] MOD 2
7: !If Commands
8: IF R[33:If Value Ex]=0,CALL ADVANCED_PROGRAM_PART1
9: IF R[33:If Value Ex]=1,CALL ADVANCED_PROGRAM_PART2
10: !End of For loop
11: ENDFOR
[End]
```

3.11 JUMP LABEL

Programozás során lehetőségünk van ugrások alkalmazására jump parancsok segítségével. A Jump LBL parancs egy a programban definiált címkéhez ugrik.

A parancs beillesztéséhez az F5-öt (INST) kell megnyomni, majd a JMP/LBL opciót kiválasztani.

```
ADVANCED_PROGRAM_PART1
11/11
1: JMP LBL[1]
2: LBL[2]
3:L PR[45:Pos4 Adv Prog Ex] 100mm/sec FINE
4: JMP LBL[3]
5: LBL[1]
6:L PR[44:Pos3 Adv Prog Ex] 100mm/sec FINE
7:L PR[43:Pos2 Adv Prog Ex] 100mm/sec FINE
8:L PR[42:Pos1 Adv Prog Ex] 100mm/sec FINE
9: JMP LBL[2]
10: LBL[3]
[End]
```

3.12 FINE ÉS CNT FÜGGVÉNYEK

A programok felgyorsíthatók ha a mozgásokban a CNT függvényt alkalmazzuk a FINE helyett. Ennek hatására a robot nem fog elmenni teljesen a megadott pontig, csak annak környezetéig, majd megállás nélkül folytatja a következő mozgást.

A CNT melletti paraméter értéke 0-100 tartományban lehet, és azt jelképezi, hogy mennyire közelítse meg a robot a pontot. A CNT0 működése megegyezik a FINE működésével. A CNT100 esetén pedig úgy fogja megtervezni a pályát, hogy maximális sebességgel tudjon végig mozogni a pályán. A CNT hatása tehát függ a sebességtől. Minél nagyobb a sebesség, annál nagyobb lesz a hatása.

```
ADVANCED _PROGRAM _PART1
5/5
1:L PR[42:Pos1 Adv Prog Ex] 100mm/sec CNT100
2:L PR[43:Pos2 Adv Prog Ex] 100mm/sec CNT100
3:L PR[44:Pos3 Adv Prog Ex] 100mm/sec CNT100
4:L PR[45:Pos4 Adv Prog Ex] 100mm/sec CNT100
[End]
```

3.13 WAIT FÜGGVÉNY

A Wait parancs segítségével lehetőségünk van a programunkba várakozást is berakni. Ez a függvény egy regiszter vagy konstans segítségével megadott időt fog várni, majd a program a következő sorban található paranccsal fogja folytatni a futást.

```
ADVANCED _PROGRAM _PART1
6/6
1:L PR[42:Pos1 Adv Prog Ex] 1000mm/sec CNT100
2:L PR[43:Pos2 Adv Prog Ex] 1000mm/sec CNT100
3:L PR[44:Pos3 Adv Prog Ex] 1000mm/sec CNT100
4:L PR[45:Pos4 Adv Prog Ex] 1000mm/sec CNT100
5: WAIT 2.00(sec)
[End]
```

3.14 INPUT/OUTPUT

A I/O-k a robotoknál más eszközökkel való kommunikációra használhatóak, így nagyon fontosak ha a robot nem egyedül, hanem perifériákkal együtt működik. A robot információt kaphat szenzoroktól vagy kameráktól a bemeneteken keresztül, vagy parancsot küldhet

motoroknak vagy más robotoknak a kimeneteken keresztül. Például a robot küldhet egy jelet a másik robotnak miután befejezte a munkát egy adott munkadarabon.

A I/O-knak több típusa van:

- Digital I/O – digitális jelek.
- Group I/O – csoportosított jelek. Ezek kettő vagy több digitális jelet tartalmaznak. Segítségükkel például küldhető vagy fogadható bináris formába alakított szám.
- Analog I/O – analóg jelek.

A I/O-k listájának megnyitásához először a menüt kell megnyitni, majd az I/O opciót kiválasztani. A kimenetek és bemenetek listája között az F3 (IN/OUT) funkció gomb segítségével válthatunk. Az elemek egy adott értéktől, például 101-től indulnak. Az Item gombbal egy adott elemre ugorhatunk.



A kimenetek értékét ezen a képernyőn változtathatjuk, a bemenetek esetén pedig figyelhetjük a változásokat.

A bemeneteket szimulálhatjuk is. Ehhez a kurzort a Sim-re kell vinnünk, majd megnyomni az F4 Simulate gombot.

A kimeneteket és bemeneteket természetesen a programokban is használhatjuk. A következő ábrán egy példa látható amelyben „If” függvények segítségével a program különböző bemenetek hatására különböző műveleteket végez.

```
I _ O _ EX ^ i  
1/16  
1: !Set UFrame  
2: UFRAME_NUM=1  
3: LBL[1]  
4: OVERRIDE=10%  
5: IF DI[101:Switch 1]=ON, CALL ADVANCED_PROGRAM_PART1  
6: IF DI[102:Switch 2]=ON, CALL ADVANCED_PROGRAM_PART1  
7: IF DI[103:Switch 3]=ON, CALL ADVANCED_PROGRAM_PART2  
8: IF DI[104:Switch 4]=ON, CALL ADVANCED_PROGRAM_PART2  
9: IF DI[105:Switch 5]=ON, CALL AA_HOME  
10: IF DI[106:Switch 6]=ON, CALL AA_HOME  
11: IF DI[107:Switch 7]=ON, JMP LBL[2]  
12: IF DI[108:Switch 8]=ON, JMP LBL[2]  
13: WAIT .10(sec)  
14: JMP LBL[1]  
15: LBL[2]  
[End]
```

[INST] [EDCMD] >

A kimeneteket pedig a következő ábrán látható példa szerint kell használni, melyben az összes használt kimenetet kikapcsolt módba állítjuk.

```
DO_SET_OFF ^ i  
9/9  
1: !Setting Outputs OFF  
2: DO[102:LED 2]=OFF  
3: DO[103:LED 3]=OFF  
4: DO[104:LED 4]=OFF  
5: DO[105:LED 5]=OFF  
6: DO[106:LED 6]=OFF  
7: DO[107:LED 7]=OFF  
8: DO[108:LED 8]=OFF  
[End]
```

[INST] [EDCMD]

3.15 TÖBB PROGRAM PÁRHUZAMOS FUTTATÁSA

Lehetőségünk van több program párhuzamos futtatására, mely nagyon hasznos lehet ha például a robot mozgatása mellett külső felszerelést is szeretnénk vezérelni.

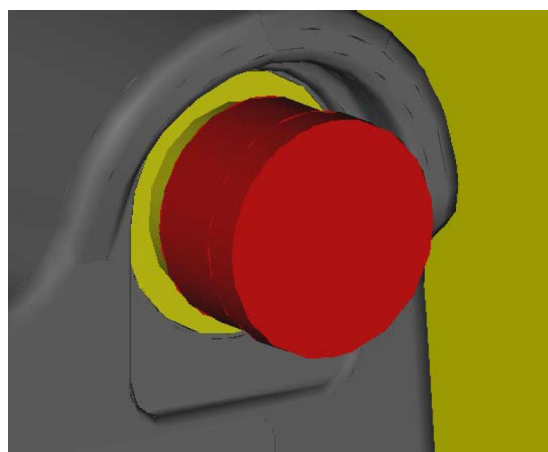
Ennek beállításához először a programunkba be kell illeszteni egy Run parancsot, amely majd a másik programot elindítja. Ehhez az INSTR gombot kell megnyomni, majd a lista második lapján az ötödik elemet, azaz a Multiple Control-t kiválasztani, és beállítani a másik program nevét.

Ha ez a lépés után elindítjuk a programunkat, akkor a Teach Pendant-en a következő hibát fogjuk látni: „PROG-040 Already locked by other task”. Ennek megoldásához a Run parancsban használt programok Group Mask-ját kell módosítani „0”-ról „*”-ra. Ez az F2 Detail gomb megnyomásával lehetséges.

```
Program name:
1 REGISTER_TO_BINARY
2 Sub Type:      [None      ]
3 Comment:      [          ]
4 Group Mask:   [0,*,*,*,*,*,* ]
5 Write protect: [OFF       ]
6 Ignore pause: [OFF       ]
7 Stack size:   [ 500     ]
```

3.16 KÜLÖNBÖZŐ STOP TÍPUSOK

A FANUC robotoknál különböző STOP-ok állnak rendelkezésre, amelyeket különböző elvek alapján kell használni.



A Stop-oknak 3 kategóriája van (az IEC 60204-1-t követve):

- 2. kategóriás stop – kontrollált stop, amelynél a stop után az aktuátorokon (esetünkben motorokon) továbbra is rendelkezésre áll a táp.

- 1. kategóriás stop – kontrollált stop, amelynél a stop után az aktuátorokon nincs tovább táp.
- 0. kategóriás stop – azonnali stop, amely elveszi a tápot az aktuátorokról.

Az 1. és 0. kategóriás stopoknál a folytatáshoz el kell távolítani a stop okát (elengedni az E-Stop gombot, becsukni a kaput, stb.) és meg kell nyomni a reset gombot.

VÉSZLEÁLLÍTÓ (E-STOP)

Vészhelyzetekben kell használni, például ha ember van veszélyben vagy a robot ütközni fog más tárgyakkal. Tehát, a fő célja károk és sérülések elkerülése. Ez 0. kategóriás stop.

DEADMAN SWITCH ELENGEDÉSE

A robot mozgatása vagy a program tesztelése során a Deadman Switch-nek középső állásban kell lennie. Mivel ennek a gombnak a célja szintén a felhasználó és a környező eszközök megvédése, ez is 0. kategóriás stop.

NYITOTT KERÍTÉS

A robotok általában egy cellában találhatóak, így körbe vannak kerítve valamilyen kerítéssel. Ezek lehetnek valódi kerítések, de akár fénysorompók is. Amikor ezek segítségével érzékeljük, hogy valaki a robot munkaterébe lépett, akkor a robotnak meg kell állnia. Viszont ilyen esetben a robotnak van valamennyi ideje a megállásra, mivel a személynek szükséges valamennyi idő amíg olyan közel ér a robothoz ahol az már el is éri. A normák szerint ez az idő 1 másodperc. Ez is 1. kategóriás stop, mivel a cellába való belépés nem jelent direkt veszélyt arra a személyre.

HOLD

A Teach Pendant-en található Hold gombot használhatjuk kontrollált megállásra annak érdekében, hogy például a robot beállításában kicseréljünk valamit.

A Hold megnyomásának hatására a robot megáll, viszont nem ad hibaüzenetet, így elindítható abból a pozícióból ahol megállítottuk.

Ez 2. kategóriás stop.

4 PROGRAMOZÁS ROBOGUIDE KÖRNYEZETBEN

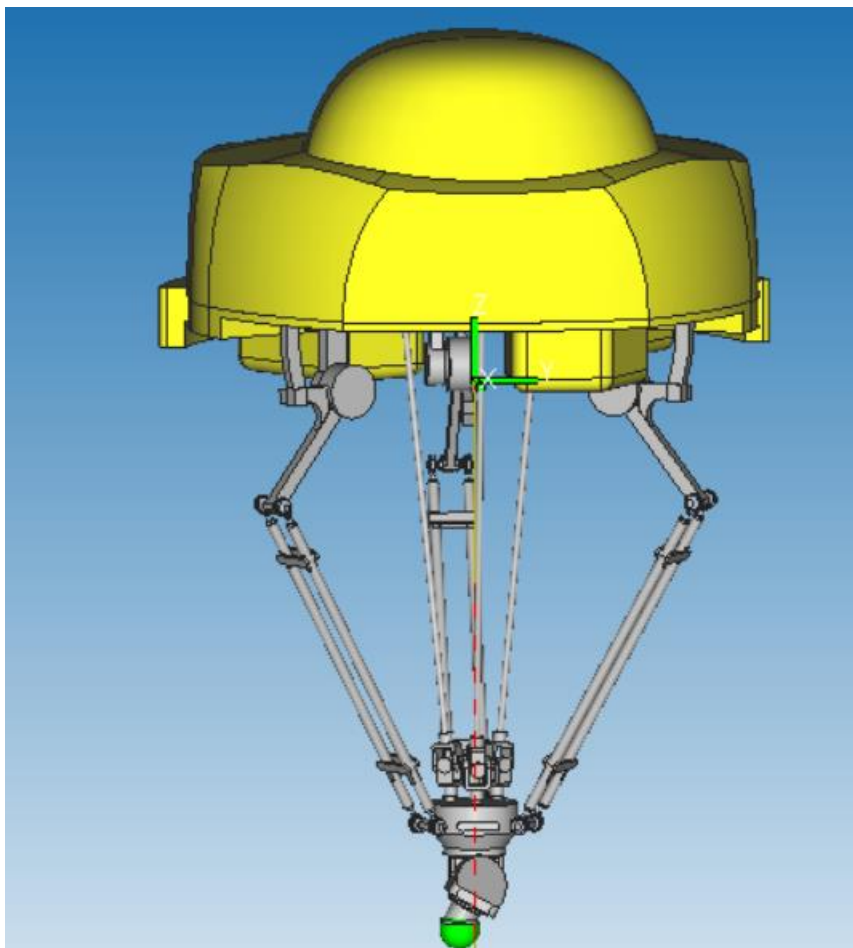
FANUC robotokhoz a programok készítésére és a szimulációk végzésére rendelkezésre áll a ROBOGUIDE környezet. Ennek a szoftvercsomagnak a segítségével teljes robotcellákat valósíthatunk meg akár több robottal, futószalagokkal, szenzorokkal, kamerákkal, környezeti elemekkel, stb.

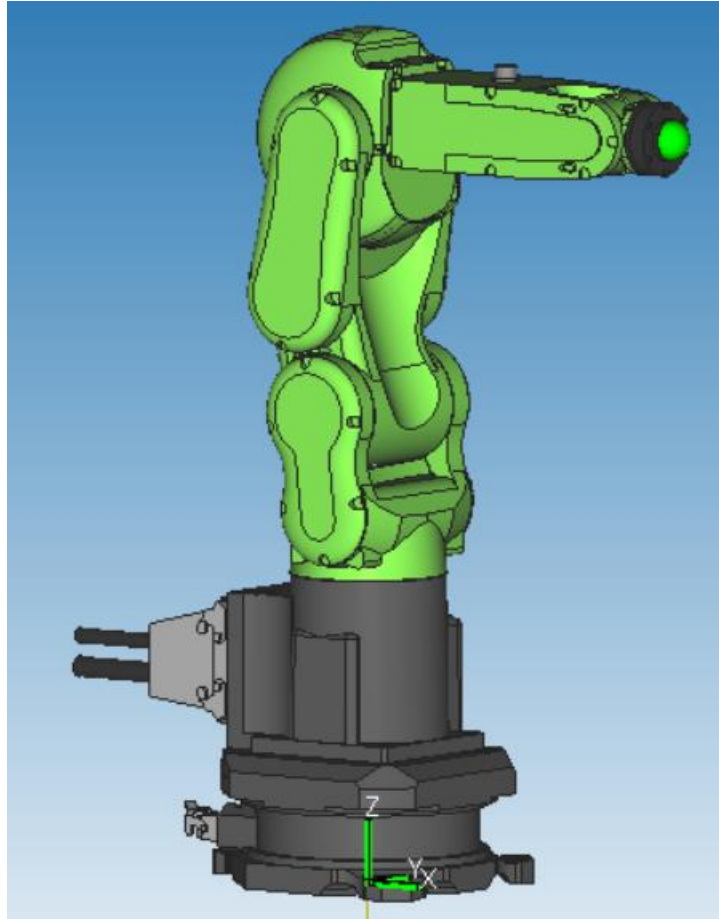
Az elemek mozgathatók, és így képesek vagyunk kamerák vagy más szenzorok működésének szimulációjára és tesztelésére is.

4.1 A BESZERZETT ROBOTOK MODELLJEI

A beszerzett robotok közül az M-1*i*A/0.5A típusú delta robot és a CR-4*i*A típusú kollaboratív robotok modelljei megtalálhatóak a programcsomagban és így alkalmazhatóak programozásra, viszont az SR-3*i*A SCARA robot annyira új terméke még a FANUC-nak, hogy ennek modellje még nincs az alkalmazható robotok listájában a szoftverben.

Az M-1*i*A/0.5A és CR-4*i*A robotok modelljei az alábbi ábrákon láthatóak.





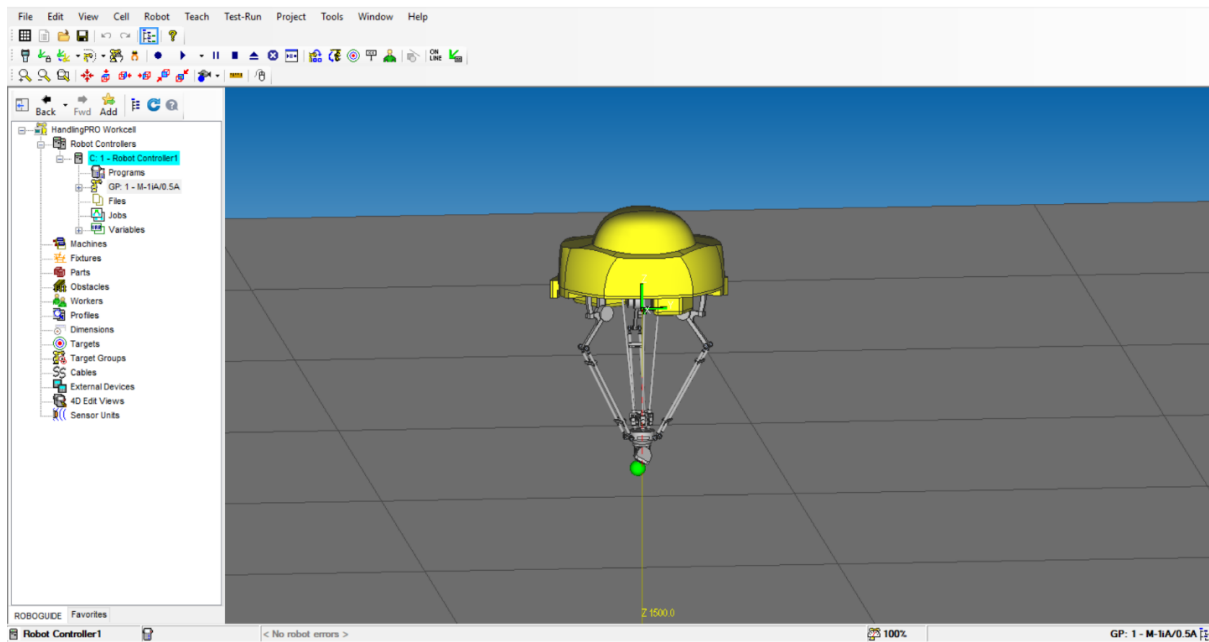
A FANUC SCARA robotjainak programozása el is tér a többi modellnél alkalmazott módtól. A SCARA robotok manuális kezelése és programozása tabletokon keresztül történik, míg a többi típusnál a megszokott Teach Pendant-et kell használni.

4.2 A BESZERZETT ROBOTOK ALKALMAZÁSI TERÜLETEI

A SCARA és delta típusú robotok tipikusan ún. pick and place alkalmazásokban használtak, azaz valamilyen tárgyak áthelyezésére. Ennek oka, hogy a konfigurációjuk alapján a munkaterükben rendkívül gyorsan tudnak mozogni. Például ha futószalagról valamilyen kisebb tárgyat kell valamilyen tárolóra pakolni, akkor ezeknek a konfigurációknak az alkalmazása a legcélszerűbb.

A CR-4iA típusú robot munkatere sokkal nagyobb, mivel ezek hat rotációs csuklóval rendelkeznek. Ezek a konfigurációk különböző feladatokra használhatóak megfelelő effektorok alkalmazásával, mint például festés, hegesztés, stb. Viszont anyagmozgatásra is széles körben használtak, mivel ilyen konfigurációkból nagy teherbírású modellek is léteznek.

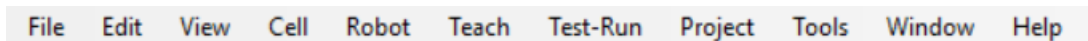
4.3 A ROBOGUIDE FELHASZNÁLÓFELÜLETÉNEK ELEMEI



A szoftvercsomag számos opciót biztosít a felhasználófelületen amely megkönnyíti a programozás folyamatát.

4.4 MENÜPONTOK

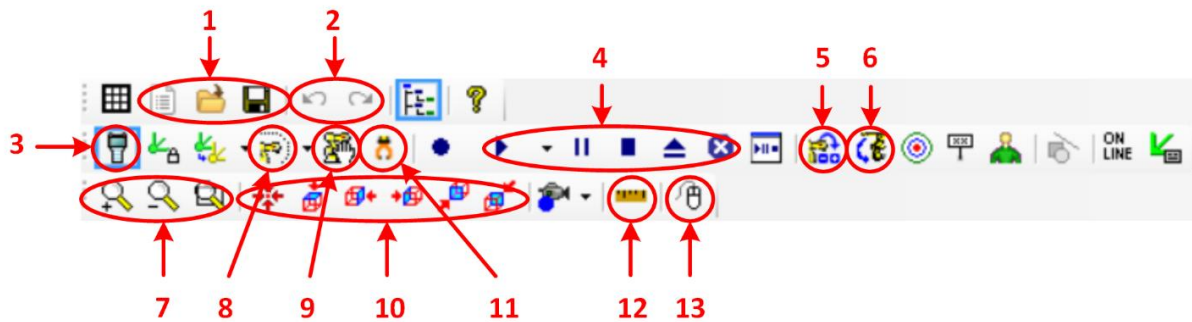
A főmenüben található opciók a következő ábrán láthatóak.



Az egyes menüpontok funkciói:

- File – cellák létrehozására és mentésére, illetve létező cellák megnyitására találhatóak ebben a csoportban opciók.
- Edit – lépések visszavonása, illetve másolási, vágási és beillesztési opciók.
- View – különböző ablakok láthatóságának beállítása és zoomolási lehetőségek.
- Cell – különböző cellaelemek hozzáadása.
- Robot – Teach Pendant megnyitása, a robot mozgatásához használható eszközök, a kamerák beállításához használható böngésző megnyitása, vezérlő újraindítása, stb.
- Teach – Teach Pendant-es és szimulációs programok létrehozása, illetve a programokban használható egyes műveletek.
- Test-Run – futtatási opciók beállítása.
- Project – a projekt fájljainak kezelése.
- Tools – plug-in-ek, I/O-k és külső eszközök beállításai.
- Window – vizualizációs opciók.
- Help – programozást segítő dokumentumok megnyitása.

A menüben található programozás során gyakran használt opciók a menüsor alatt ikonként is megtalálhatóak.

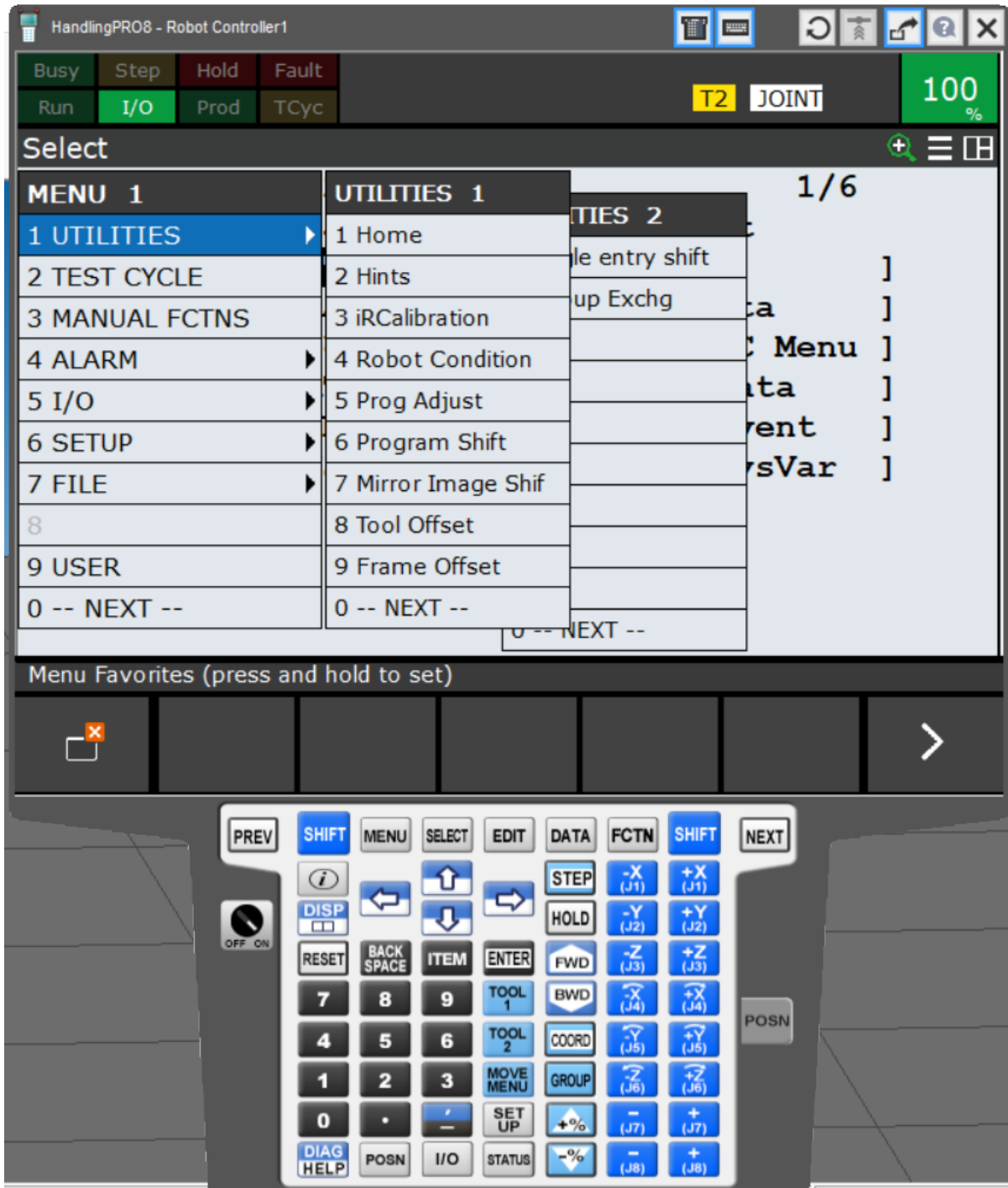


Az ábrán is megjelölt fontosabb pontok a következők:

1. Új cella létrehozása (New Cell), létező cella megnyitása (Open Cell) és cella mentése (Save Cell).
2. Undo és Redo.
3. Teach Pendant megnyitása.
4. Ciklus indítása (Cycle Start), futtatás megszakítása (Hold), futtatás megállítása (Abort), hiba mód megszüntetése (Fault Reset), azonnali megállítás (Immediate Stop).
5. Jogginghoz használható koordinátarendszerek váltására alkalmazható gyorsmenü (Jog Coordinates Quick Bar) megnyitása.
6. Pontok meghatározásához használható gyorsmenü (Move To Quick Bar) megnyitása.
7. Zoomolási opciók.
8. Robot munkaterének (Work Envelope) megjelenítése.
9. Csuklók mozgására szolgáló eszköz (Joint Jog Tool) megjelenítése.
10. Különböző nézetek beállítása.
11. Megfogó becsukása és kinyitása (Open/Close Hand).
12. Távolság mérésére alkalmas eszköz megnyitása.
13. Egérfunkciókat kijelző ablak megnyitása.

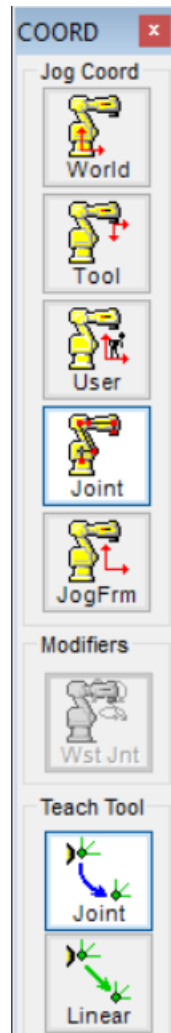
TEACH PENDANT

A robot kezelése teljes mértékben szimulálható a szoftverben található Teach Pendant segítségével, melynek kinézete és gombjainak funkciója is megegyezik a valódival.



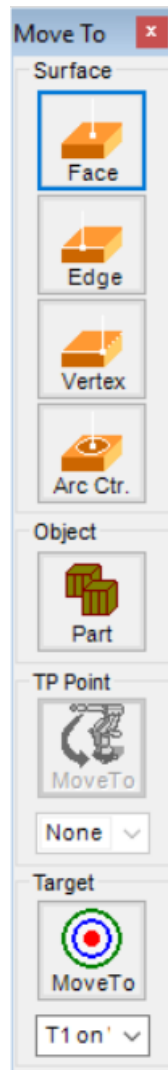
KOORDINÁTARENDSZEREK VÁLTÁSÁRA ALKALMAZHATÓ GYORSMENÜ

A gyorsmenü tartalmaz ikonokat a különböző koordinátarendszerek váltására és a mozgások típusának váltására.



PONTOK MEGHATÁROZÁSÁHOZ HASZNÁLHATÓ GYORSMENÜ

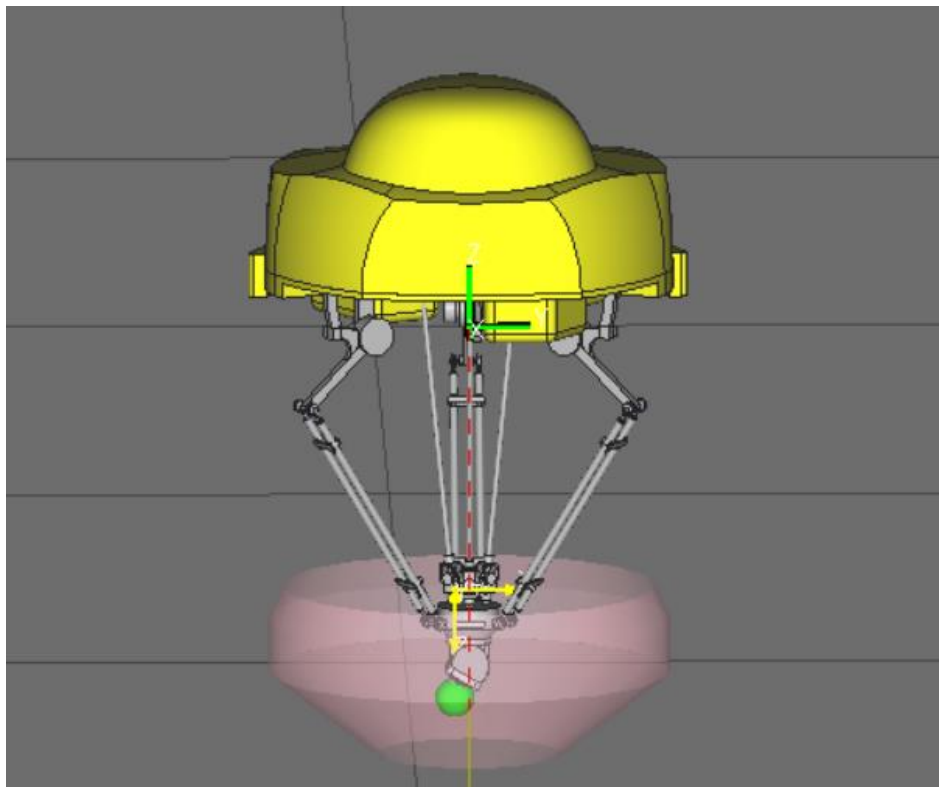
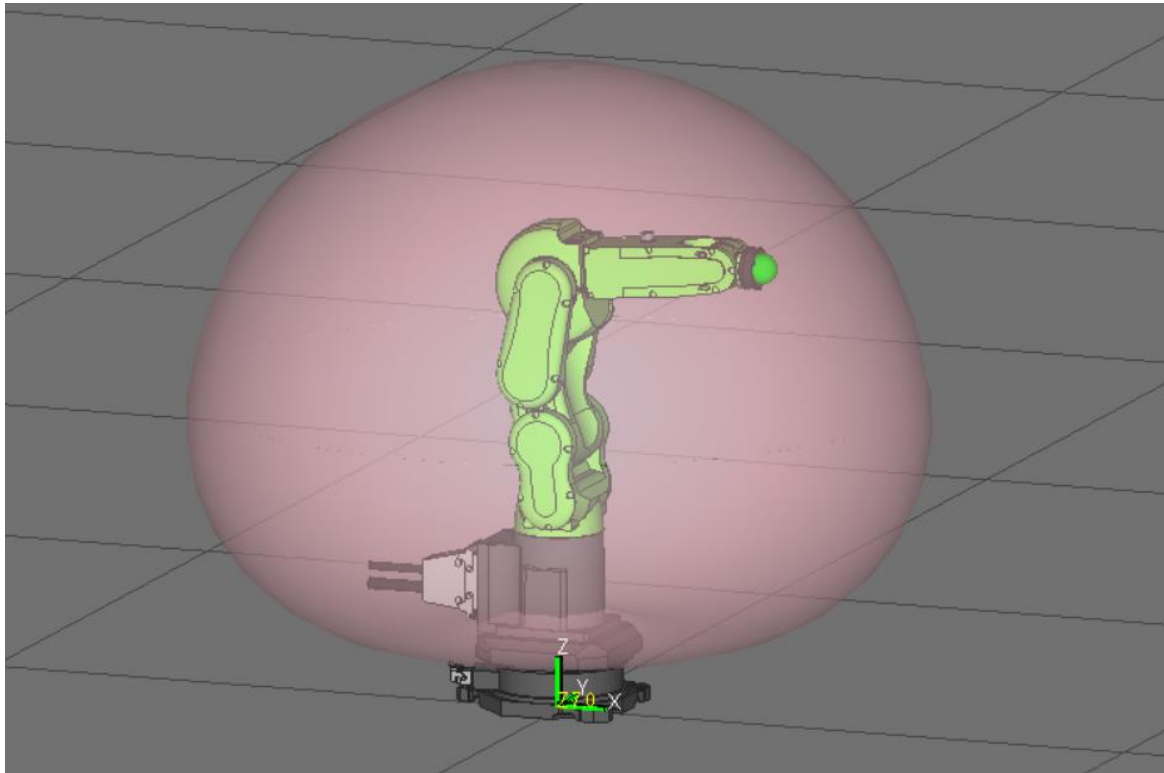
Ebben a csoportban találhatóak olyan opciók amelyek valamilyen felületen segítenek a pontok felvételében (lap közepe – Face, él – Edge, sarokpont – Vertex, stb.), a Move To Target pedig a robotot egy előzőleg felvett ponthoz (Target) mozgatja.



A ROBOT MUNKATERE

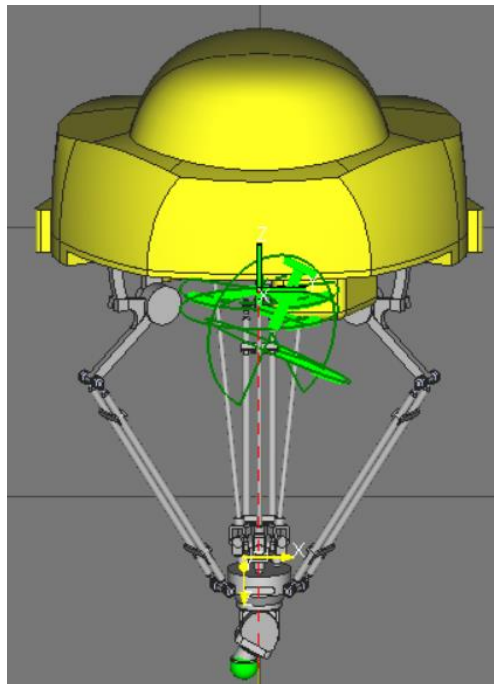
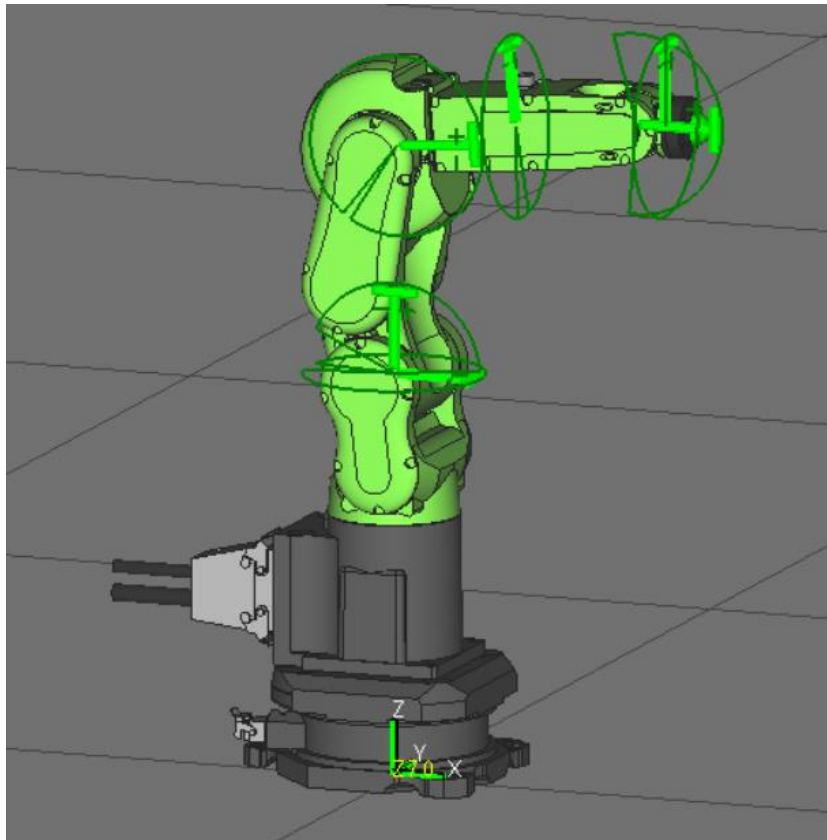
A robot munkaterének kijelzésével könnyebben meghatározhatjuk, hogy hova kell helyezni a térben egy adott eszközt.

A munkatér kijelezhető a robothoz hozzátartozó szerszámot figyelembe véve, de nélküle is.



CSUKLÓK MOZGATÁSÁRA SZOLGÁLÓ ESZKÖZ

A Joint Jog Tool aktiválásával könnyen tudjuk az egér segítségével változtatni az egyes csuklók koordinátáit.



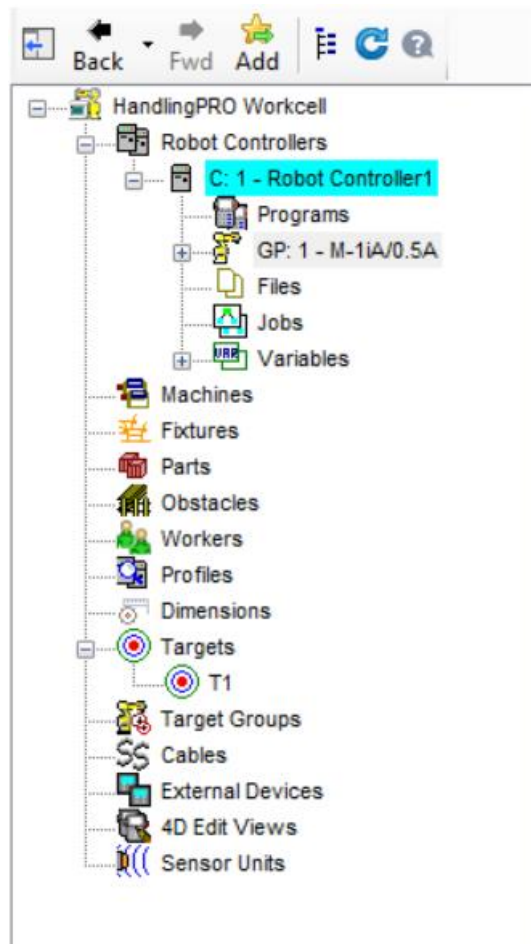
EGÉRFUNKCIÓKAT KIJELEZŐ ABLAK

Különböző billentyűkombinációkkal az egér számos funkciót elláthat. Ezek megtanulásáig nagy segítséget nyújthat ennek az ablaknak az aktiválása.

3D World Mouse Commands			
View Functions	Object Functions		MoveTo Functions
Rotate view:	RIGHT Drag	Move object, one axis: LEFT Drag triad axis	Move robot to surface: [CTRL] + [SHIFT] + LEFT-Click
Pan view:	[CTRL] + RIGHT Drag	Move object, multiple axes: [CTRL] + LEFT Drag triad	Move robot to edge: [CTRL] + [ALT] + LEFT-Click
Zoom in/out:	BOTH Drag (mouse Y axis)	Rotate object: [SHIFT] + LEFT Drag triad axis	Move robot to vertex: [CTRL] + [ALT] + [SHIFT] + LEFT-Click
Select object:	LEFT-Click	Object property page: DOUBLE-LEFT Click	Move robot to center: [SHIFT] + [ALT] + LEFT-Click

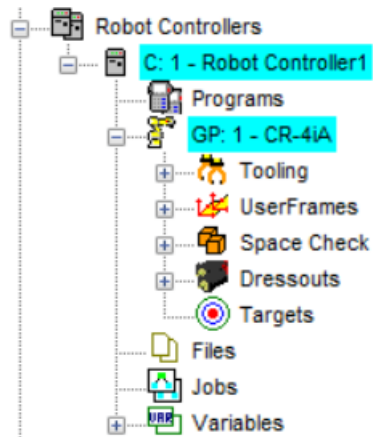
4.5 A CELLÁK ELEMEI

A cella elemeinek egyszerű áttekintésére a Cell Browser segítségével van lehetőségünk, mely alapesetben az ablak bal oldalán található.



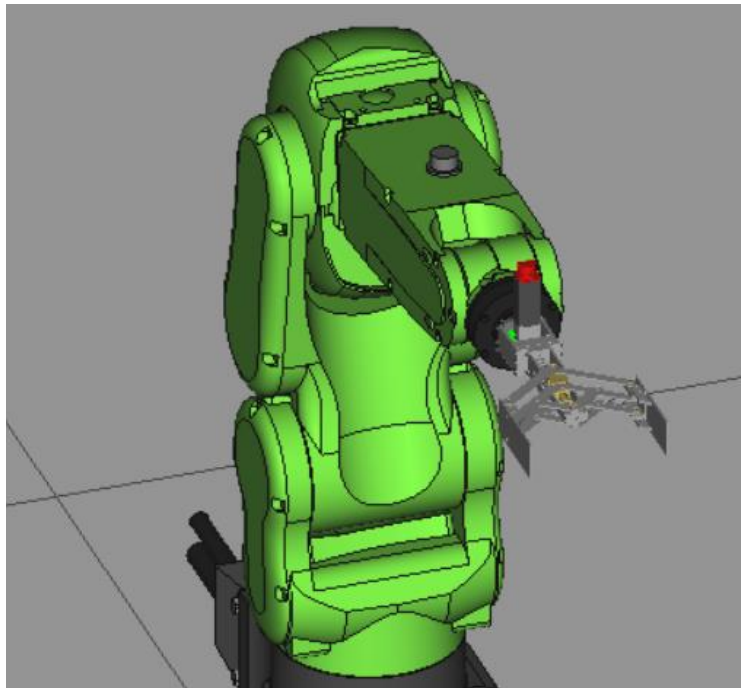
4.5.1 ROBOT VEZÉRLŐK

A Robot Controllers csoportban találhatóak a cellához hozzáadott vezérlők és a hozzájuk tartozó robotok és programok. Egy adott robotnál megtekinthetők még a felvett Tool és User koordinátarendszerek, illetve a robozhoz csatolt elemek.



SZERSZÁM HOZZÁADÁSA A ROBOZHOZ

Szerszám hozzáadásához a robot csoportján belül a Tooling csoportot kell lenyitni, majd az UT:1-nél az Add Link opciót kell választani. Beépített modell kiválasztása esetén a CAD Library ponton belül keresünk megfelelő elemet. A modellek között találunk megfogókat, hegesztőpisztolyokat, vákuumos megfogókat, stb.



4.5.2 EGYÉB ELEMÉK

A cellákban a robotokon kívül található elemek különböző kategóriákba sorolhatók a funkciójuk alapján:

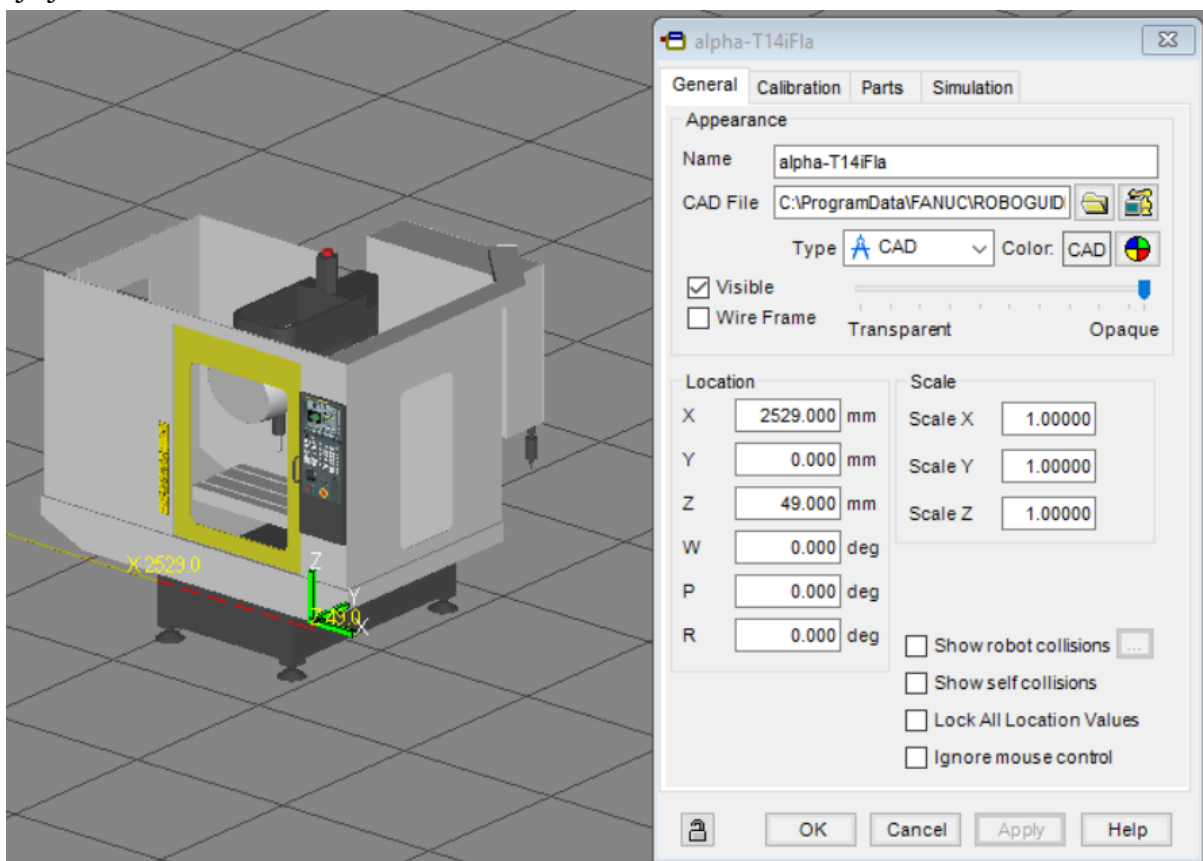
- Gépek (Machine)
- Alkatrészek (Fixtures)
- Munkadarabok (Parts)
- Akadályok (Obstacles)
- Munkások (Workers)

Az alkatrészek beillesztésénél nagyon fontos figyelni, hogy a megfelelő típusba soroljuk be.

GÉPEK (MACHINE)

Sok alkalmazásban szükséges, hogy a robot valamilyen alkatrészt egy gépbe helyezzen, majd a művelet befejezése után a robot azt kivegye a gépből, és adott helyre helyezze.

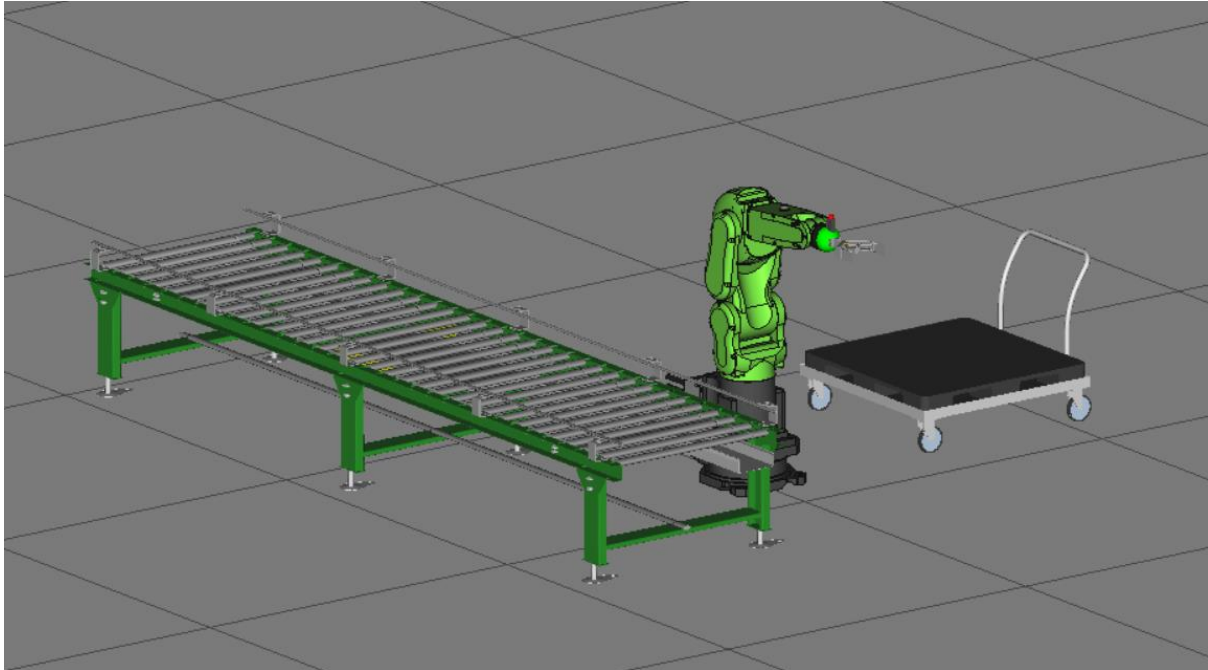
A ROBOGUIDE tartalmaz ilyen gépekről modelleket CAD fájlok formájában, de saját modelleket is beilleszthetünk. Ezeknél az eszközöknél általában animáció is használható a gép ajtajának becsukására.



ALKATRÉSZEK (FIXTURES)

Az alkatrészek csoportba olyan eszközök tartoznak, amelyekre szeretnénk munkadarabokat helyezni. Ennek programozás során van fontos szerepe, mivel a munkadarabokat

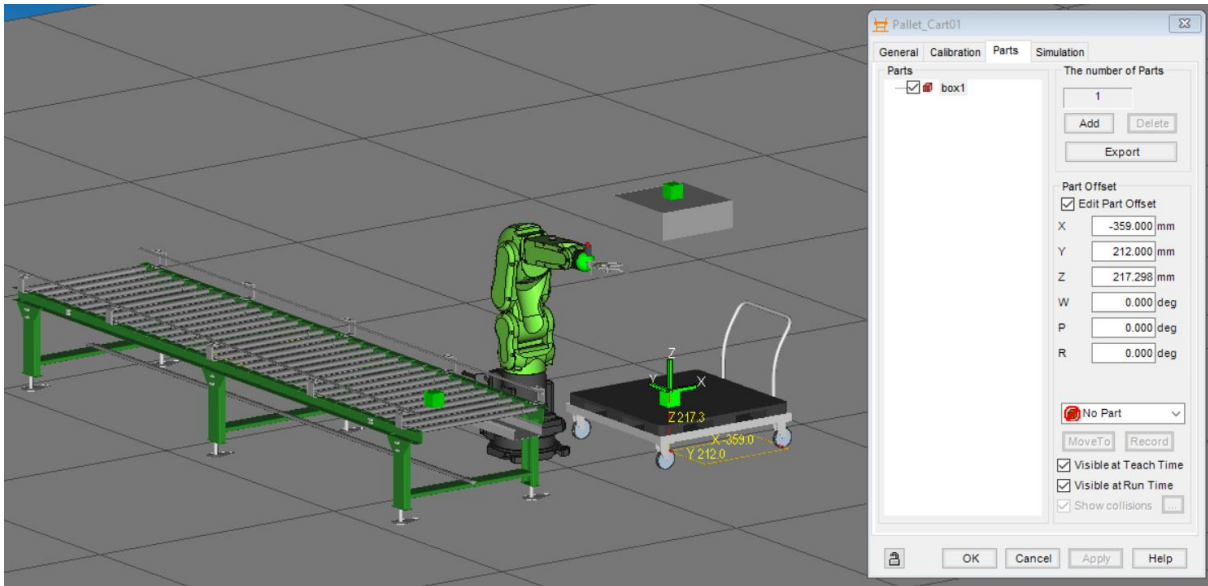
automatikusan hozzárendelhetjük ilyen alkatrészekhez, majd szimuláció készítése során egyszerűen megadhatjuk, hogy vegyen fel róla vagy helyezzen rá egy munkadarabot. Ilyen eszközök a futószalagok, aszalok, paletták, polcok, stb. A beépített CAD fájlokon kívül itt is alkalmazhatunk saját modellt.



MUNKADARABOK (PARTS)

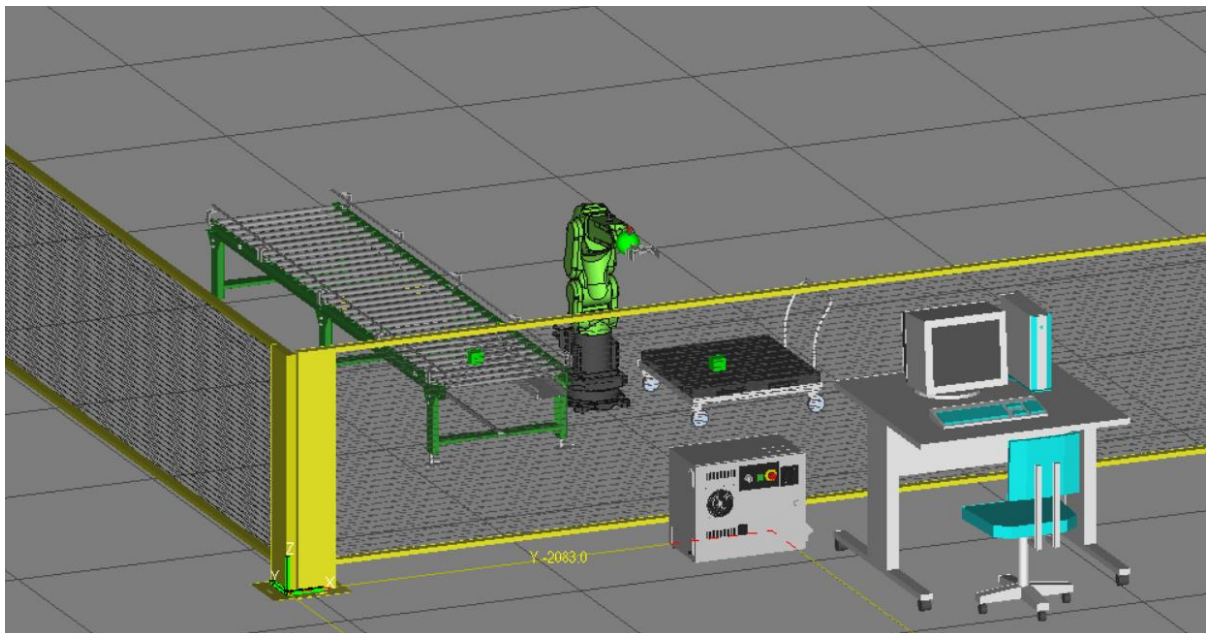
A munkadarabok azok az eszközök amelyeket a robot segítségével mozgatni szeretnénk. A beépített modellek között találhatóak például gépkocsialkatrészek is, de egyszerű testeket is létrehozhatunk, mint például egy kockát vagy egy hengert. A munkadarabnak a dimenziója mellett megadható a tömege is, amelyet a szimuláció majd figyelembe fog venni.

A munkadarabok alkatrészekhez való hozzárendelésével több másolat fog létrejönni a munkadarabból. Fontos, hogy a munkadarabot a robothoz is hozzá kell adni ha szeretnénk mozgatni.



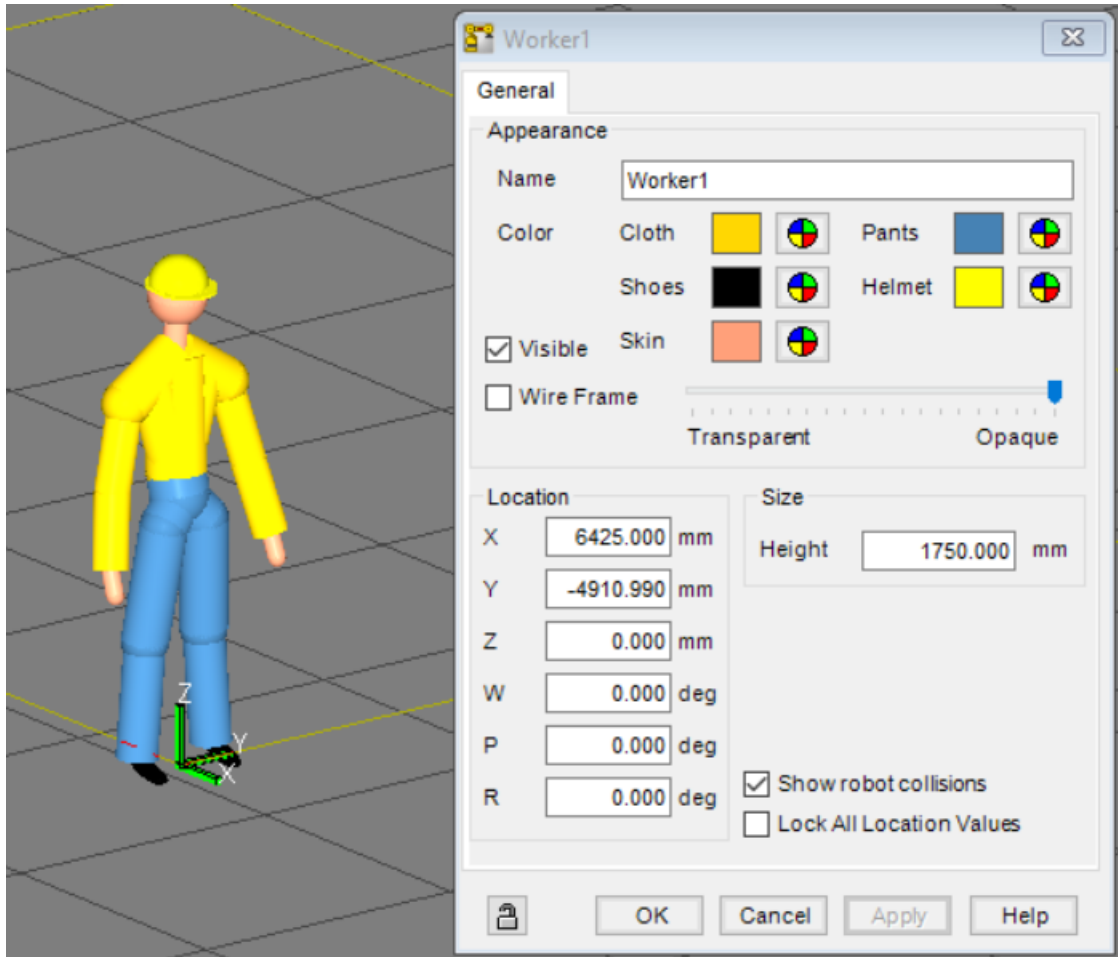
AKADÁLYOK (OBSTACLES)

Az akadályok olyan eszközök amelyekkel a robot mozgás során nem fog kapcsolatba lépni. Ilyen eszközök a kerítések, kameraállványok, vezérlődobozok, stb. Ezeknek szimuláció során nincs szerepük, viszont élethűbbé teszik a cellát.



MUNKÁSOK (WORKERS)

A robotok környezetébe munkásokat jelképező modelleket is elhelyezhetünk, hogy a cella még valóságosabb legyen. A szimuláció során az akadályokhoz hasonlóan nincs szerepük.



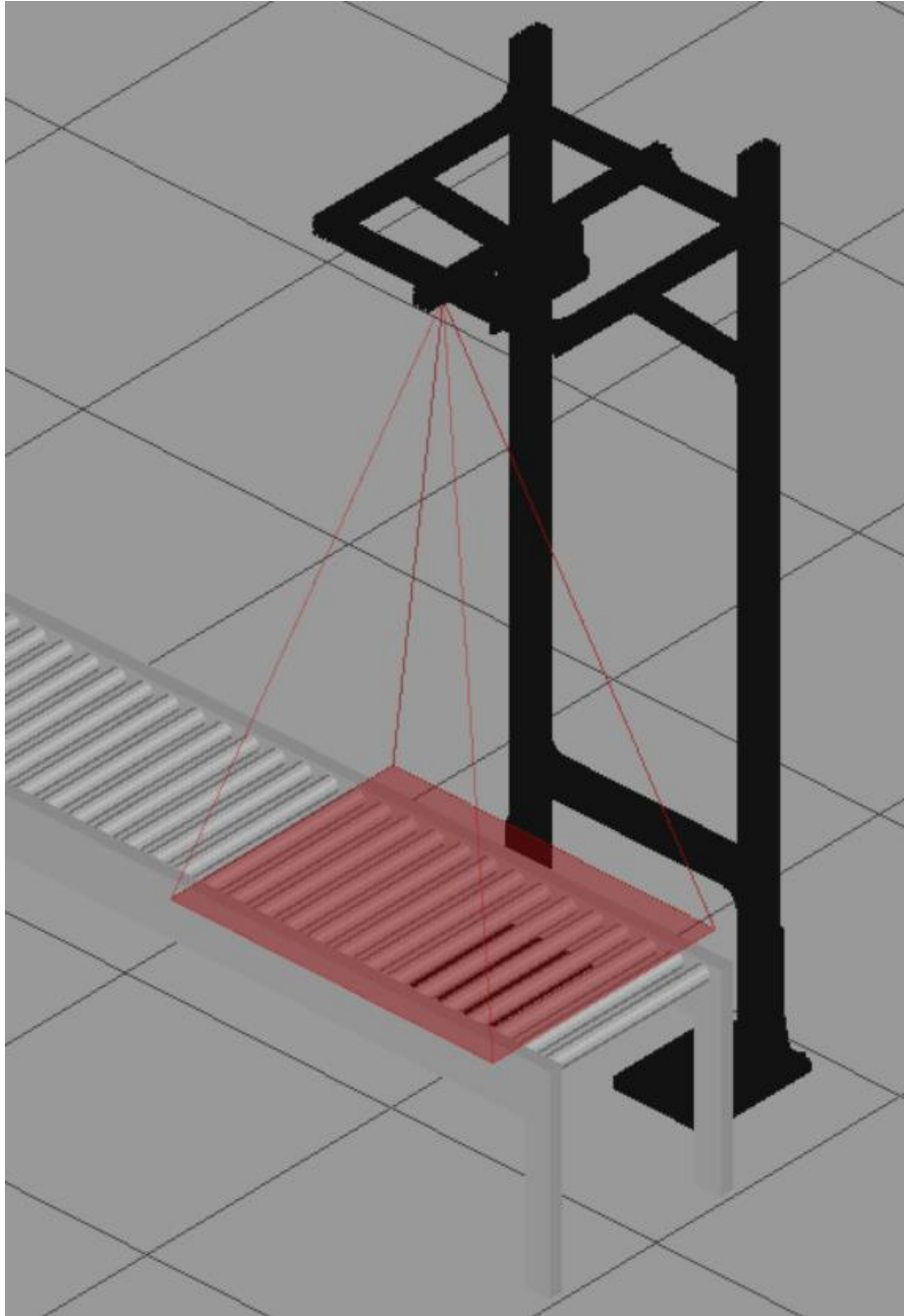
4.5.3 SZENZOR EGYSÉGEK

A ROBOGUIDE több típusú szenzor hozzáadását teszi lehetővé a Cell Browser-ben, a Sensor Units pont alatt:

- 2D kamera (2D Camera)
- 3D lézer szenzor (3D Laser Vision Sensor)
- 3D területérzékelő (3D Area Sensor)

A szenzorok mellett lámpát is hozzáadhatunk a cellánkhoz.

Ezek szimulálásában a FANUC iRVision plug-in segít.



4.6 PROGRAMOK LÉTREHOZÁSA

A ROBOGUIDE szoftvercsomag segítségével két típusú programot hozhatunk létre:

- Teach Pendant programok
- Szimulációs programok

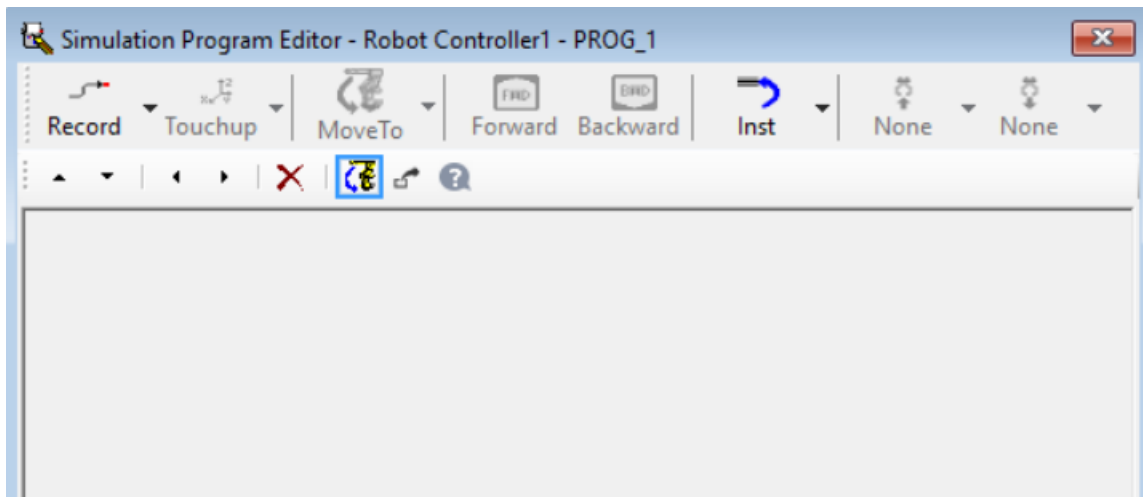
4.6.1 TEACH PENDANT PROGRAMOK

A Teach Pendant alapú programok létrehozása ugyanúgy történik mint valós robotok esetén, mivel a Teach Pendant minden funkcióját tudjuk alkalmazni a szimulációs környezetben.

A robot mozgatása is ugyanúgy megvalósítható ezen keresztül az előző fejezetekben leírtak szerint.

4.6.2 SZIMULÁCIÓS PROGRAMOK

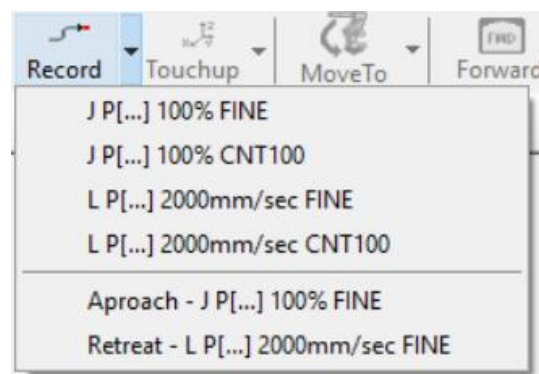
A szimulációs programokat a Simulation Program Editor segítségével hozhatjuk létre, amely a Teach menüpont Add Simulation Program opciójával nyitható meg.



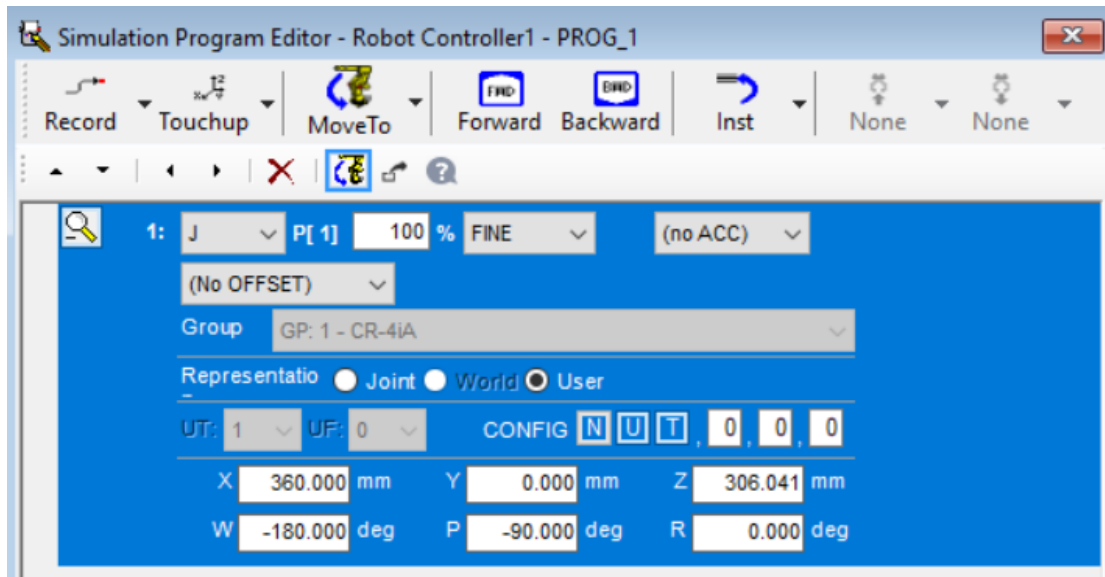
Ezzel a módszerrel nagyon gyorsan készíthetünk szimulációs programot, amellyel meg tudjuk figyelni a cella működését, viszont ez a program nem tölthető rá a robotra.

Számos egyszerűsítést tartalmaz, különösen a munkadarabok mozgatása esetén.

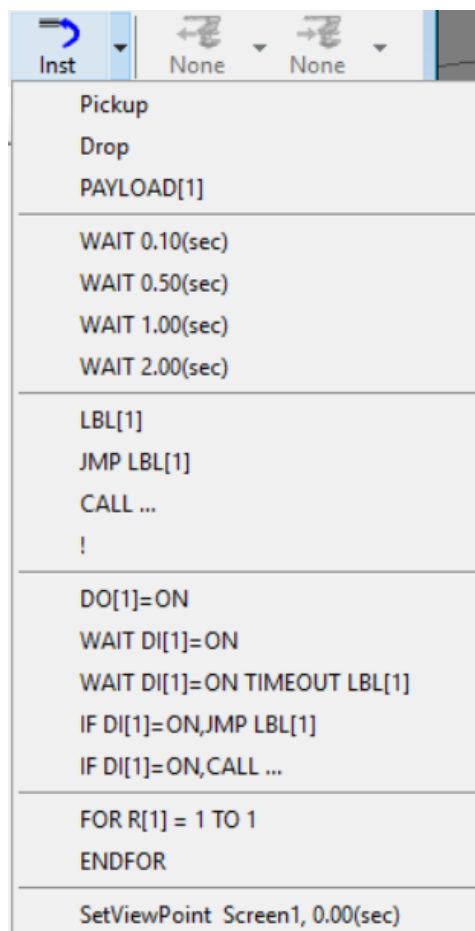
Pontokat a Record opcióval rögzíthetünk, amelynél a következő mozgásokat hozhatjuk létre a gombra kattintva:



A mozgásokat módosíthatjuk is az ablakban.

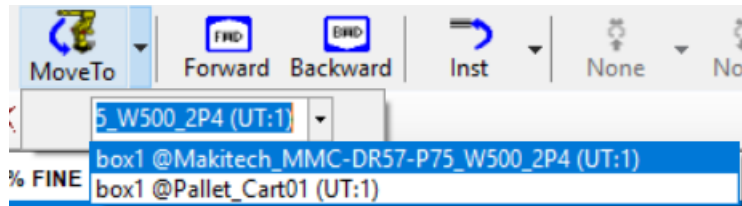


Az Inst opcióval különböző parancsokat adhatunk a programhoz.

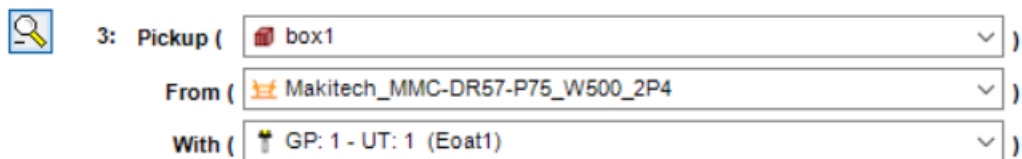


A Pickup paranccsal egy munkadarabot felvehetünk egy alkatrészről, a Drop parancs segítségével pedig letehetjük azt egy másikra.

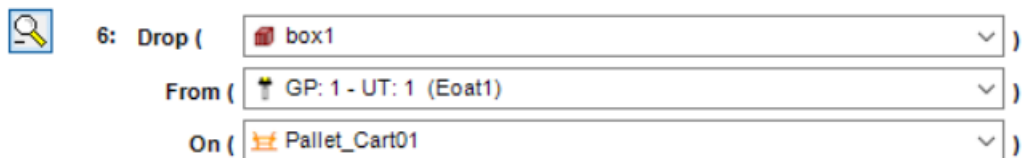
A Move To gomb segítségével nem kell minden alkalommal beállítani az adott pozíciókat, hanem például egyszerően abba a pozícióba tudunk ugrani a robottal amelyben felvehetjük a dobozt a futószalagról.



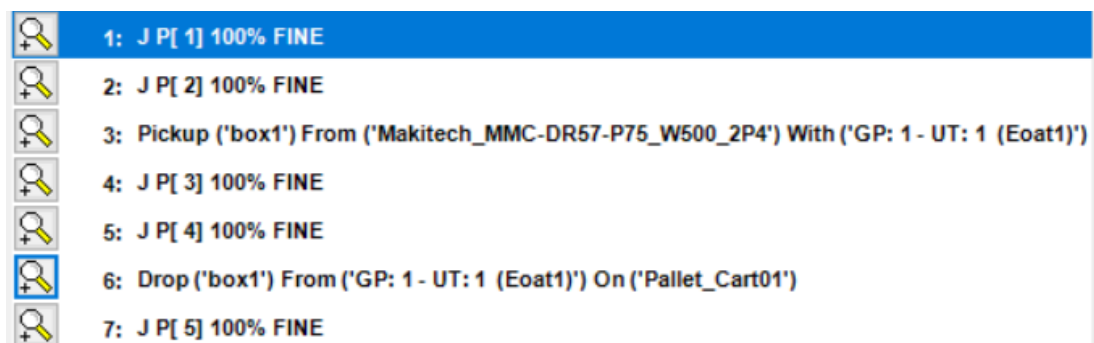
A Pickup parancs beillesztésénél be kell állítani, hogy mely munkadarabot szeretnénk honnan felemelni és mely megfogóval.



A Drop parancs esetén szintén be kell állítani a paramétereiket.



A következő ábrán látható egyszerű program először elmegy alappozícióba, majd odamegy a futószalaghoz és felveszi a dobozt, felemeli kicsit az ütközések elkerüléséért, odamegy a palettához és leteszi a dobozt, majd visszatér az alappozícióba.



JOGI NYILATKOZAT

Ez a dokumentum az Európai Unió pénzügyi támogatásával valósult meg. A dokumentum tartalmáért teljes mértékben a Visoka tehnička škola strukovnih studija - Subotica vállalja a felelősséget, és az semmilyen körülmények között nem tekinthető az Európai Unió és / vagy az Irányító Hatóság állásfoglalását tükröző tartalomnak.