



EPS

Escola Politècnica
Superior

Projecte/Treball Fi de Carrera

Estudi: Enginyeria Industrial. Pla 2002

Títol: GENERACIÓ D'UN ENTORN VIRTUAL DE TREBALL PER UN ROBOT STÄUBLI TX60

Document: Memòria i Annexos

Alumne: Ignasi Juny Canals

Director/Tutor: Xavier Cufí Solé

Departament: Arquitectura i Tecnologia de Computadors

Àrea: Arquitectura i Tecnologia de Computadors

Convocatòria (mes/any): 06/2014

ÍNDEX

Índex.....	I
1 Introducció.....	1
1.1 Antecedents.....	1
1.2 Objecte general del projecte.....	1
1.3 Abast i especificacions.....	1
1.4 Objectiu concret a assolir.....	2
2 Descripció de la Solució.....	3
2.1 Programació off-line de robots.....	3
2.1.1 Què és la programació off-line.....	3
2.1.2 Què requereix la programació off-line.....	3
2.1.3 Bondats de la programació off-line.....	3
2.1.4 Tipus de softwares de programació off-line.....	4
2.2 Disseny 3D.....	4
2.2.1 Elecció del software de disseny.....	4
2.2.2 Problemàtica de disseny.....	5
2.2.3 Estructura del model 3D generat.....	5
2.2.4 Assemblatge del banc de treball.....	5
2.2.5 Parts mòbils.....	10
2.2.6 Pinça i sensor de força.....	11
2.3 Ús del software.....	13
2.3.1 Introducció a Staübli Robotics Suite 2013.....	13
2.3.2 Development Studio.....	14
2.3.3 Maintenance Studio.....	14
2.3.4 Emuladors CS8.....	14
2.3.5 Manual d'ús del software SRS 2013.....	15
2.4 Implementació del disseny 3D.....	16
3 Pla d'implementació del software.....	18
3.1 Introducció.....	18
3.2 Requeriments de hardware recomanats.....	18
3.3 Instal·lació del software al laboratori.....	18
3.4 Versió 2.2.....	19
3.5 Xarxa local.....	19
4 Resum econòmic.....	20
5 Índex dels documents del Projecte.....	21
5.1 DOCUMENT NÚM.1 MEMÒRIA I ANNEXOS.....	21
5.2 DOCUMENT NÚM.2 CD AMB ELS MODELS 3D.....	21
6 Propostes.....	22
7 Conclusions.....	23
8 BIBLIOGRAFIA.....	24

1 INTRODUCCIÓ

1.1 Antecedents

A les darreres dècades, els entorns de treball en tres dimensions han esdevingut un element clau en el disseny i simulació de peces i mecanismes. En els darrers anys però, amb l'augment de la potència gràfica dels ordinadors personals, la simulació total de línies industrials ha adquirit una importància cabdal.

És per això que les principals corporacions de robòtica del mercat mundial han llançat nous softwares que, amb el hardware adequat, permeten programar i simular off-line les seves estacions de treball.

La UdG ha cregut convenient enfocar la seva docència en aquesta direcció i, darrerament, ha adquirit el programa Staübli Robotics Suite 2013. Aquest permet la gestió d'un entorn virtual de treball així com la directe programació i simulació de robots Staübli dins l'entorn generat.

1.2 Objecte general del projecte

L'objectiu del treball és emular virtualment l'entorn de treball del robot Staübli Tx60 que hi ha al laboratori de robòtica (dins les possibilitats que ofereix el software adquirit). Aquest laboratori intenta reproduir un entorn industrial de treball en el qual es realitza l'assemblatge d'un conjunt de manera cent per cent automatitzada.

1.3 Abast i especificacions

En una primera fase, es dissenyarà en tres dimensions tot l'entorn de treball que hi ha disponible al laboratori a través del software CAD SolidWorks. Cada un dels conjunts que conformen l'estació de treball es dissenyarà de manera independent.

Finalment s'introduiran tots els elements dins el software Staübli Robotics Suite 2013 i es generarà l'entorn de treball virtual que permetrà simular off-line el robot Staübli Tx60 (veure Figura 1) del qual es disposa al laboratori.



Figura 1: Robot Staübli TX60 de 6 graus de llibertat

1.4 Objectiu concret a assolir

L'objectiu del present treball es divideix en dues etapes:

1. Generar els arxius CAD necessaris per poder simular l'estació de treball.
2. Realitzar un manual d'ús del nou software Staübli Robotics suite 2013.

2 DESCRIPCIÓ DE LA SOLUCIÓ

2.1 Programació off-line de robots

2.1.1 Què és la programació off-line

La programació off-line de robots és la creació i simulació d'aplicacions informàtiques a través de l'ordinador sense necessitat de tenir físicament un robot. Aquest tipus de programació comporta que, durant el procés de creació d'una aplicació informàtica, sigui necessari poder simular el comportament exacte del robot i la seva interacció amb l'entorn.

2.1.2 Què requereix la programació off-line

La programació off-line de robots requereix d'un entorn de treball virtual generat per ordinador dins el qual pot ser simulat tant el robot com les seves interaccions amb els elements circumdants. Per aquest motiu és necessària una potència gràfica molt elevada, normalment en funció del nivell de complexitat de la simulació generada.

2.1.3 Bondats de la programació off-line

La programació off-line comporta un gran estalvi de temps. Aquesta permet testejar els programes generats per l'usuari des del propi ordinador en comptes de fer-ho des d'un robot real. El fet de poder simular el comportament del robot dins d'un entorn virtual fa que, durant la major part del temps de planificació d'una nova línia de producció o la modificació d'una ja existent, aquesta no s'hagi d'aturar en cap moment. Aquest fet es tradueix amb una disminució del temps no productiu de línies industrials dotant a aquestes d'una capacitat productiva molt major i, per tant, augmentant la seva competitivitat.

Per tant, la programació off-line de robots permet accelerar els temps de programació i els temps d'implementació de les aplicacions generades, així com disminuir el temps d'inactivitat del robot.

2.1.4 Tipus de softwares de programació off-line

Actualment existeixen dos tipus de softwares de programació off-line de robots: per una banda hi ha els softwares específics de simulació de robots, els quals permeten incorporar i gestionar geometries generades per altres softwares. Normalment aquests tipus de softwares són els que faciliten les grans empreses de robòtica mundial. Per altra banda, existeixen els softwares que són una extensió d'un programa CAD ja existent. Aquests aprofiten la potencialitat del mateix tot incorporant-li un toolbox de robòtica.

El cas que ens ocupa és el software Stäubli Robotics Suite 2013, el qual és del primer tipus. Per tant, serà necessari generar un entorn virtual 3D amb un programa CAD compatible i, posteriorment, incorporar-lo dins el software de robòtica de l'empresa Stäubli.

2.2 Disseny 3D

2.2.1 Elecció del software de disseny

El disseny 3D de l'estació de treball està generat a través del software CAD SolidWorks. S'ha optat per aquest software de modelització per diversos motius.

D'una banda el fet que la UdG disposi de llicències d'aquest programa ha permès fer un ús eficient dels recursos disponibles i, per tant, abaratir costos. També ha sigut determinant la compatibilitat de formats entre aquest software CAD i el software de simulació Stäubli Robotics Suite.

Per altra banda, aquest programa de disseny actualment és un dels més versàtils i està especialment dissenyat per ser utilitzat a tot tipus d'indústries. A més a més de permetre el disseny de peces i el seu posterior assemblatge, també permet la simulació i optimització per elements finits. Finalment cal esmentar que el software SolidWorks està àmpliament implementat a la indústria del nostre entorn.

Per tots aquests motius s'ha decidit optar per aquest programa.

2.2.2 Problemàtica de disseny

Donat que, per la posterior simulació del robot, les discrepàncies entre el model generat i la realitat han de minimitzar-se, totes les mides han estat preses amb el mateix peu de rei, el qual té una precisió de $\pm 0.05\text{mm}$.

Cal tenir en compte que sempre existirà una certa discrepància o error entre la simulació del robot i la realitat. Per aquest motiu una vegada es té dissenyada l'estació de treball i s'ha programat off-line un procés determinat, quan aquest es testeja a la realitat per primera vegada el més normal és que no funcioni a la perfecció. Habitualment els punts definits en el programa s'han d'acabar d'ajustar als punts reals.

2.2.3 Estructura del model 3D generat

Ara ja referint-nos al model generat, aquest està organitzat en dues parts principals: un assemblatge de totes les parts estàtiques que conformen l'estació de treball, altrament anomenat banc de treball, i un seguit de peces mòbils. Totes les parts mòbils són independents les unes amb les altres. Per altra banda, l'assemblatge del banc de treball està compost de sub-assemblatges els quals, com a elements últims, estan formats per peces independents.

Aquesta part de la memòria mostra cada un d'aquest assemblatges i sub-assemblatges així com cada una de les peces independents, tot adjuntant-hi una breu descripció.

2.2.4 Assemblatge del banc de treball

El banc de treball està compost d'elements que permeten realitzar l'assemblatge de tres components: un cilindre, un pistó i una tapa. Aquestes elements són les següents:

1. Dispensador de cilindres vermells (Figura 2)
2. Dispensador de tapes blaves (Figura 3)
3. Palet d'èmbols (Figura 4)
4. Aparell assemblador (Figura 5)
5. Suport de l'assemblatge finalitzat (Figura 6)
6. Base (Figura 7)
7. Bancada (Figura 8)

Cada una d'aquestes parts té assignada una tasca concreta que tot seguit comentarem:

1. Dispensador de cilindres vermells: és l'element encarregat de subministrar els cilindres a través d'actuadors pneumàtics.



Figura 2: vista isomètrica del dispensador de cilindres vermells

2. Dispensador de tapes blaves: és l'element encarregat de subministrar les tapes a través d'actuadors pneumàtics.



Figura 3: vista isomètrica del dispensador de tapes blaves

3. Palet d'èmbols: és l'element encarregat de subministrar els èmbols que s'han d'introduir dins el cilindre.



Figura 4: vista isomètrica del palet d'èmbols

4. Aparell assemblador: és una base que fa de suport durant la realització de l'assemblatge entre el cilindre, el pistó i la tapa.

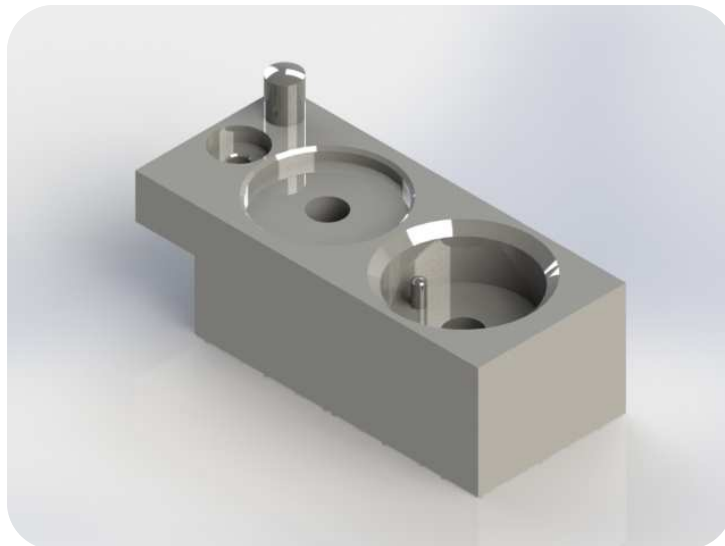


Figura 5: vista isomètrica de l'aparell assemblador

5. Suport assemblatge finalitzat: és l'element on es diposita l'assemblatge del cilindre, pistó i tapa ja finalitzat. Té una rampa suficientment inclinada com perquè l'assemblatge dipositat a la part superior, llisqui cap a la part inferior.



Figura 6: vista isomètrica del suport de l'assemblatge finalitzat

6. Base: és un perfil d'alumini que va unit per un costat a la bancada i per l'altre conté tots els elements necessaris per poder realitzar l'assemblatge. Per tant, és la part que fa pròpiament de banc de treball.

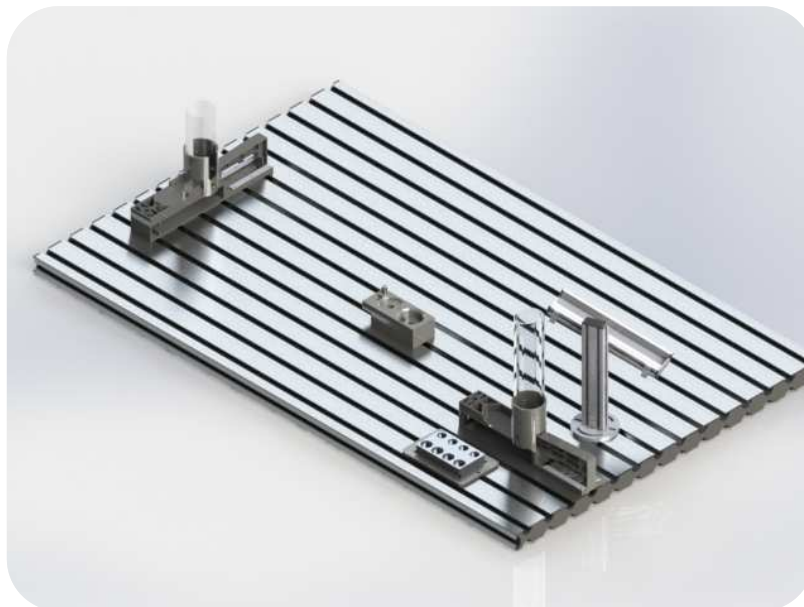


Figura 7: vista isomètrica de la base amb els components ja assemblats

7. Bancada: és l'estructura de l'estació de treball del robot i la seva missió és la de suportar totes les càrregues mecàniques. La seva característica principal és la integritat física, ja que permet que la taula de treball sigui fixa i els punts simulats es corresponguin sempre als reals.

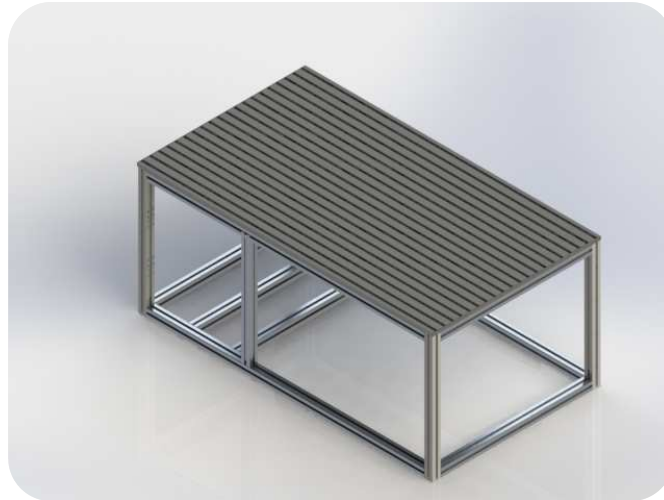


Figura 8: vista isomètrica de la bancada

A l'assemblatge de totes les parts també s'hi ha introduït la base del robot (veure Figura 9). Aquesta ha de suportar tots els esforços mecànics que aquest li transmeti. A més a més està enclavada a terra i, per tant, és l'element de referència.



Figura 9: vista isomètrica del banc de treball del robot Tx60 amb tots els elements

Finalment, a la Figura 10 podem veure com queda l'estació de treball amb tots els components ajuntats.

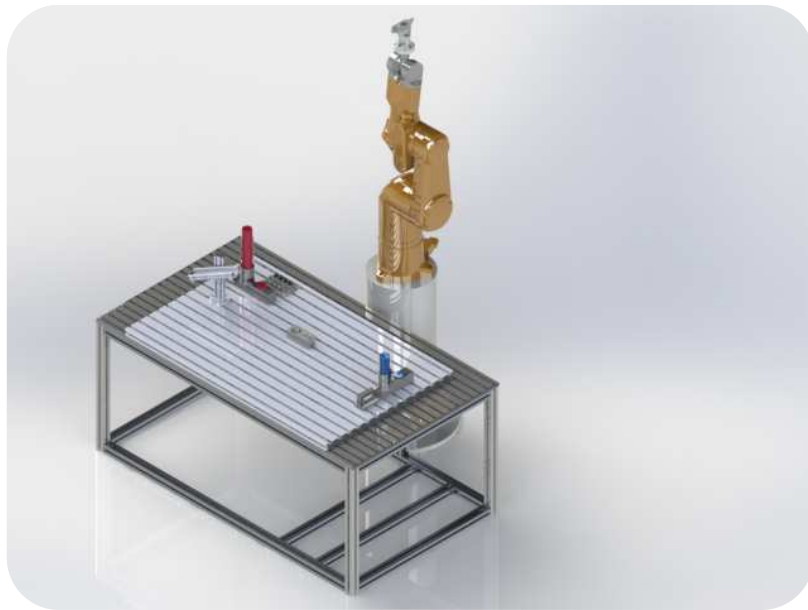


Figura 10: vista isomètrica de l'estació de treball amb el robot Tx60 i les peces a assemblar

2.2.5 Parts mòbils

Tal com ja s'ha esmentat amb anterioritat, les parts mòbils són totes aquelles que es poden moure amb el manipulador dins de l'estació de treball. Concretament es tracta de tots els components que conformen l'assemblatge a automatitzar. Aquestes parts són:

1. Cilindre vermell (Figura 11a)
2. Èmbol (Figura 11b)
3. Tapa blava (Figura 11c)

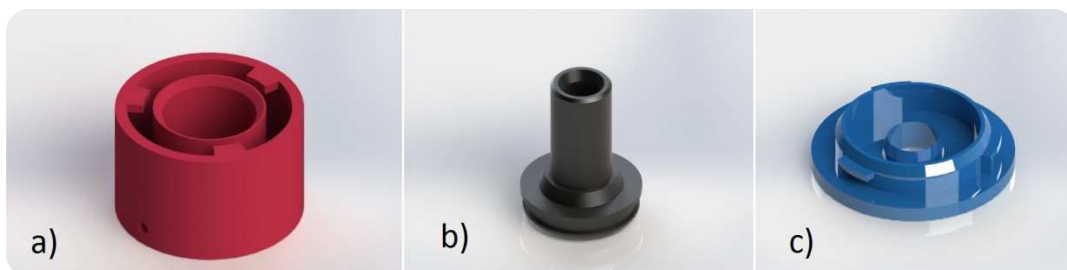


Figura 11: Elements que conformen l'assemblatge a automatitzar

L'assemblatge consisteix en dipositar l'èmbol dins el cilindre i posteriorment col·locar la tapa tot rosquant-la al cilindre. A la Figura 12 podem veure la vista isomètrica de la secció de l'assemblatge finalitzat.

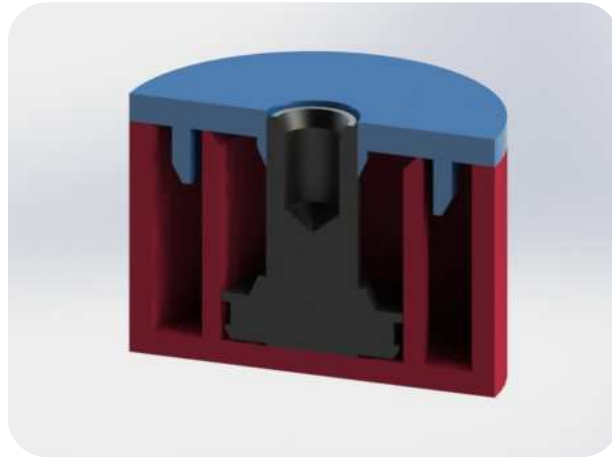


Figura 12: vista isomètrica del conjunt assemblat

2.2.6 Pinça i sensor de força

Per tal de poder simular amb exactitud el moviment del robot, és necessari fer el disseny de la pinça (Figura 14). Aquesta està unida a un sensor de força (Figura 13) el qual, a la vegada, està unit al robot. Aquesta cadena de components fa que hi hagi un offset determinat entre el punt on el robot té la mà i el punt real on hi ha col·locada la pinça. Tot seguit es fa una breu descripció de cada un dels dos components.

1. **Sensor de força:** tal com diu el seu nom, el sensor de força permet determinar quin és l'esforç que està transmetent la pinça del robot. És un element que sempre està al robot del laboratori i, per fer l'assemblatge, no es fa servir. Per tant només interessa conèixer les seves dimensions. Tal com es pot veure a la Figura 13, aquest sensor està compost per una base de subjecció, des de la qual s'adjunta al robot, i el sensor pròpiament dit, el qual té les ranures necessàries per poder-hi acoblar la pinça

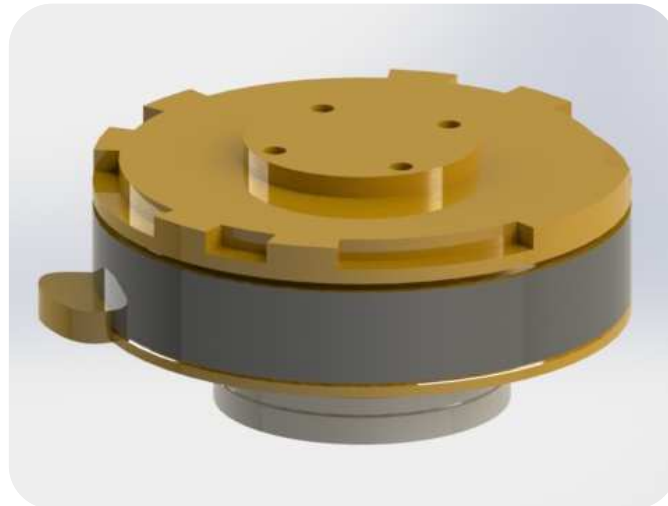


Figura 13: vista isomètrica del sensor de força

2. Pinça: la pinça és l'element manipulador del robot. Aquest és l'encarregat d'actuar sobre el seu entorn que se li demana. La pinça que hi ha al laboratori és una pinça pneumàtica de FESTO a la qual se li ha unit una peça d'una geometria que permeti tenir dos extrems manipuladors de la pinça. Un per peces de diàmetre petit i un altre per peces de diàmetres més grans.

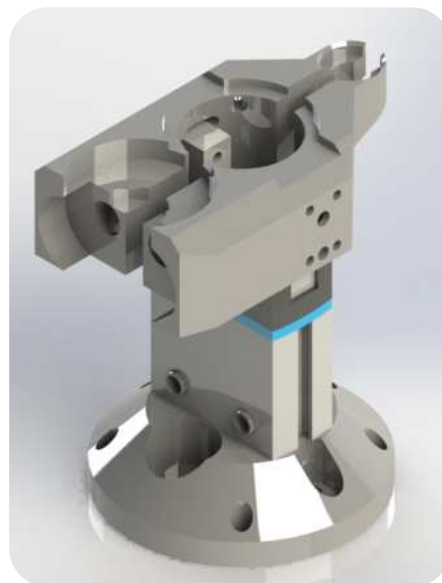


Figura 14: element terminal del robot anomenat pinça el qual s'acobla al sensor de força

La disposició d'aquests dos elements assemblats es pot veure a la Figura 15.

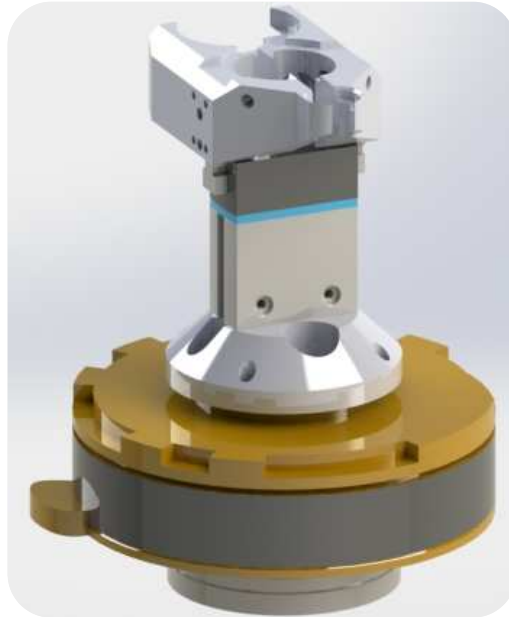


Figura 15: vista isomètrica de l'assemblatge del sensor de força i la pinça

2.3 Ús del software

2.3.1 Introducció a Stäubli Robotics Suite 2013

El software Stäubli Robotics Suite 2013 altrament anomenat SRS 2013, és un entorn de programació per ordinador pensat per poder desenvolupar i fer el manteniment d'aplicacions de robòtica. Aquest està basat amb el llenguatge de programació d'alt nivell VAL3 i utilitza els emuladors CS8.

El software integra les següents funcions bàsiques de comunicació i control:

- Transferència de fitxers entre el controlador del robot i el PC de desenvolupament.
- Realització de còpies de seguretat completes de la cel·la de treball del robot.
- Visualització del braç robòtic dins l'escenari de treball en 3D.
- Execució de programes VAL3 amb l'emulador CS8.

Aquest software està conformat pels anomenats "Development Studio" i "Maintenance Studio". Les seves llicències es poden comprar per separat.

2.3.2 Development Studio

Incorpora totes aquelles eines que permeten el desenvolupament de la cel·la de treball d'un robot:

- Editor de programes VAL3. Inclou funcions d'alt nivell com la depuració de codi de línia, auto completar, etc.
- Capacitat d'importar fitxers CAD. Els formats suportats són molts. Els mes importants: STEP, IGES, STL i VRML.
- Permet construir models 3D amb geometries bàsiques (cubs, esferes, cons, etc.).
- Permet generar un vincle directe entre l'escenari 3D i el programa editor de les geometries (per poder editar dades geomètriques).
- Capacitat de simular diversos robots alhora.
- Cada robot simulat pot ser governat per un controlador real o virtual.
- Permet la detecció de col·lisions: indica els objectes de la col·lisió i permet introduir distàncies mínimes de seguretat.

2.3.3 Maintenance Studio

Ofereix un complet accés remot al sistema del robot. Des d'una ubicació qualsevol es pot accedir al robot i obtenir un retorn visual de qualsevol operació realitzada per un operari en mode manual.

2.3.4 Emuladors CS8

Els emuladors CS8 són controladors virtuals que permeten simular la interfície de control del robot. Concretament permeten emular la MCP¹ del robot.

Aquests controladors virtuals (veure Figura 16) són emuladors complets dels armaris de control CS8 corresponents a la seva versió. Aquests permeten carregar i executar els programes que s'han desenvolupat, visualitzar les entrades i sorties, visualitzar les accions que han anat succeint i, finalment, executar moviments.

¹ MCP: Consola de programació manual (Manual Console Programming)

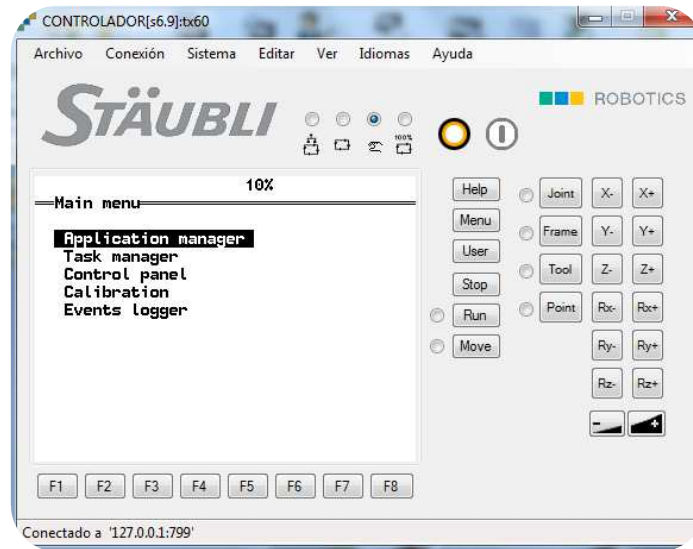


Figura 16: emulador CS8 de l'armari de control del robot Stäubli Tx60

L'emulador es comporta com la MCP real del controlador. Com es pot veure a la Figura 16, les tecles que trobem a l'emulador tenen les mateixes funcionalitats que les tecles de la MCP real. Aquestes funcionalitats són:

1. Selecció del mode de funcionament del braç
2. Activació de la potència del braç
3. Comandes d'execució d'una aplicació: Run, Stop i Move/Hold
4. Ajust de velocitat
5. Elecció del mode de desplaçament
6. Tecles de moviment

Una vegada realitzada la connexió entre l'aplicació i l'emulador, s'hi pot sincronitzar l'entorn 3D de tal manera que, si executem una aplicació, es visualitzi el moviment del robot.

2.3.5 Manual d'ús del software SRS 2013

L'annex A conté un manual complet de com utilitzar el software Stäubli Robotics Suite 2013. Aquest conté els següents apartats:

1. Iniciar SRS 2013
2. Idioma
3. Pestanyes d'eines

4. Barra d'opcions
5. Creació d'una cèl·lula
6. Afegir un robot
7. Creació d'aplicacions VAL3
8. Editor de perfils
9. Edició de l'entorn 3D
10. Emulador
11. Simulació 3D
12. Transfer Manager

Per una informació detallada de cada un dels apartats, veure l'annex A.

2.4 Implementació del disseny 3D

Una vegada generat el disseny 3D, aquest s'ha d'introduir dins el software Staübli Robotics Suite 2013. Tal com ja s'ha esmentat, per poder-lo importar cal que aquest estigui en un format adequat. S'ha optat per treballar amb el format .STEP ja que és un dels més utilitzats i el software SolidWorks permet treballar en aquest format.

Tot seguit es fa una breu descripció de quina ha sigut la metodologia per implementar el disseny 3D dins el software de robòtica. Per una explicació més detallada, veure l'annex A.

Una vegada guardat el model 3D de la taula de treball en el format .STEP, aquest s'ha introduït dins una aplicació generada amb el software SRS 2013. Tal com es pot veure a la Figura 17, el model introduït ha quedat col·locat en una posició no desitjada.

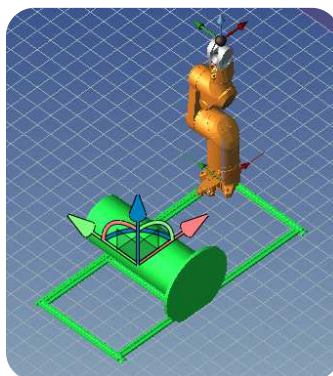


Figura 17: introducció del model 3D del banc de treball dins el software SRS 2013

Tot seguit s'ha editat la posició tant del model 3D del banc de treball com la del robot Staübli Tx60. El resultat obtingut és el que es pot apreciar a la Figura 18.

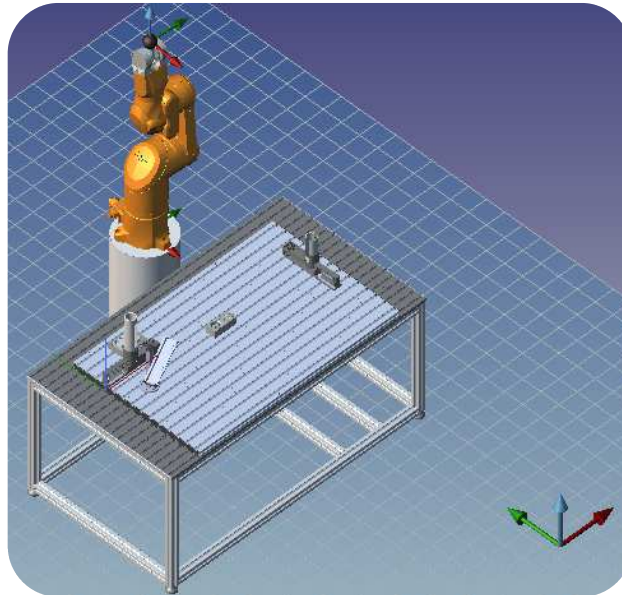


Figura 18: resultat final després d'editar la posició del banc de treball i del robot

Una vegada tenim col·locats el model 3D i el robot a una posició que s'adequa el màxim possible a la realitat del laboratori de robòtica, ja s'ha complert amb l'objectiu del treball i es pot començar a utilitzar el software amb finalitats docents.

3 PLA D'IMPLEMENTACIÓ DEL SOFTWARE

3.1 Introducció

Tot seguit es defineix el pla d'implementació del nou software adquirit perquè pugui ser utilitzat pels alumnes de la UdG amb garanties d'èxit.

Aquest consta dels requeriments essencials que haurien de tenir els ordinadors del laboratori per tal de poder simular els robots i generar entorns de treball 3D. També es descriu com es pot utilitzar el software per implementar-lo dins el pla docent.

3.2 Requeriments de hardware recomanats

El nou software de robòtica té la capacitat de treballar amb entorns virtuals generats en 3D. Com tots els programaris que treballen en tres dimensions, aquest requereix d'una capacitat computacional alta.

Els requeriments mínims recomanats de hardware que s'han de complir per poder treballar correctament amb aquest software són:

- Processador: 2 GHz o superior
- Memòria RAM: 4 Gb
- Targeta gràfica: amb 256 Mb de RAM dedicada i suport per OpenGL 2.1

Els ordinadors del laboratori de robòtica compleixen aquests requisits. Per aquest motiu no hi ha d'haver cap problema a l'hora d'utilitzar el nou software.

3.3 Instal·lació del software al laboratori

El software que es va adquirir és l'Stäubli Robotics Suite 2013 1.0. Aquest ja ha quedat antiquat. Darrerament Staübli a llançat la versió SRS 2013 2.1. Amb la llicència que la UdG ha comprat, la nova versió es pot descarregar i instal·lar de manera gratuïta. El propi software comprova que la versió instal·lada sigui la última i, en cas que no sigui així, es descarrega i instal·la la darrera versió disponible.

3.4 Versió 2.2

És molt recomanable que s'instal·li la versió 2.2 en els ordinadors del laboratori ja que el manual d'ús de l'annex A està fet amb aquesta versió. També cal remarcar que incorpora millores molt importants respecte la 1.0.

Aquestes millores permeten una major agilitat a l'hora de treballar amb el software. A més a més, les traduccions al castellà estan molt més ben fetes. Aquest fet es molt important per la documentació d'ajuda del llenguatge VAL3.

3.5 Xarxa local

Per tal de poder connectar l'armari de control del robot amb les estacions de programació (els ordinadors) del laboratori, fa falta una xarxa local. Aquesta ja està implementada dins al laboratori i, per tant, no fa falta cap modificació en aquest sentit.

4 RESUM ECONÒMIC

El cost total del present treball, incloent el cost de les llicències i el cost de la mà d'obra qualificada necessària per dur-lo a terme, és de TRES MIL SIS-CENTS SET EUROS AMB SEIXANTA-VUIT CÈNTIMS (3.607,68 €) .

CONCEPTE	COST UNITÀRI (€)	UNITATS	COST (€)
Llicències	1.938,42	0,2	387,68
Amortització d'equipaments	820,00	-	820,00
Hores de treball	40,00	50	2.000,00
Hores d'estudi	0,00	100	0,00
Hores de redacció	20,00	20	400,00
COST TOTAL =			3.607,68 €

El cost de les llicències s'ha considerat d'un 20% ja que no s'han comprat exclusivament pel present treball.

5 ÍNDEX DELS DOCUMENTS DEL PROJECTE

El present treball consta dels següents documents:

5.1 DOCUMENT NÚM.1 MEMÒRIA I ANNEXOS

5.2 DOCUMENT NÚM.2 CD AMB ELS MODELS 3D

6 PROPOSTES

El present document ha sigut de vital importància per la millora de la pràctica de la docència relacionada amb la robòtica. S'ha dotat el laboratori de robòtica d'un software d'última generació i s'ha creat un extens i complet manual d'ús del mateix. A més a més, s'ha generat tot l'entorn virtual de treball que hi ha al laboratori.

De moment s'ha dotat al departament d'Arquitectura i Tecnologia de Computadors de les eines necessàries per poder realitzar l'activitat docent habitual utilitzant el nou software. A més a més, s'ha desenvolupat el manual d'ús amb l'objectiu que la incorporació del nou programari no comporti cap modificació a la manera de realitzar les pràctiques.

Donat que el software encara està en desenvolupament, hi ha certes parts de la simulació que no s'han pogut realitzar. Una d'elles és la definició de l'eina com a un mecanisme el qual ha de permetre poder simular el seu moviment. Per altra banda, tampoc s'ha pogut fer la simulació dels sensors que hi ha al banc de treball. Finalment, la simulació del moviment d'un objecte per part del robot, tampoc s'ha pogut realitzar amb èxit.

Donat que cada actualització del software té una quantitat important de noves funcionalitats, es proposa millorar el present projecte continuant la investigació del programari durant els propers anys. Concretament, crec que tenen especial rellevància els punts esmentats anteriorment i, per tant, es proposa donar continuïtat a la investigació en un futur no molt llunyà.

Finalment, donat que el software permet la simulació total de tot tipus de robots Staübli, també es proposa la creació de nous entorns de treball 3D encara que aquests finalment no es puguin portar a la realitat. El fet que el laboratori de robòtica no gaudeixi del material suficient, no ha de limitar la possibilitat d'expressar les capacitats del nou programari adquirit.

Per tant, encoratjo al departament d'Arquitectura i Tecnologia de Computadors a seguir investigant i transferint totes les potencialitats del software a la tasca docent que es realitza a la Universitat de Girona.

7 CONCLUSIONS

Cal remarcar que l'objectiu del present document s'ha acomplert amb èxit. El manual d'ús del nou programari permet adaptar-se al nou entorn de programació de robots amb molta facilitat. A més a més, es donen les idees i eines necessàries per poder seguir desenvolupant la investigació iniciada amb aquest treball.

Finalment, es considera important esmentar que el programari adquirit ha de servir per adequar la docència que es realitza a la UdG als avenços que són inherents al món de la robòtica. Per tant, amb tot l'exposat en la present memòria i en els seus annexos, així com en la resta dels documents que integren el treball, es pot afirmar que queden suficientment definits tots i cadascun dels elements que són precisos per a la utilització del nou software de robòtica Staübli Robotics Suite 2013.

8 BIBLIOGRAFIA

STÄUBLI. Manual de Instrucciones: Brazo – Familia TX Serie 60. Faverges 2005.

STÄUBLI. Manual de Instrucciones: Armario de Control CS8C. Faverges 2006.

STÄUBLI. Manual de Utilización: Stäubli Robotics Suite 2013. Faverges 2013.

STÄUBLI. Manual de Referencia VAL3: Versión 7. Faverges 20013.

STÄUBLI. TX60 Series Industrial Robots. Faverges 2005.

A ANNEX A: MANUAL D'ÚS STÄUBLI ROBOTICS SUITE 2013**Índex**

Índex.....	1
A.1 Introducció.....	2
A.2 Iniciar SRS 2013.....	2
A.3 Idioma.....	3
A.4 Pestanyes d'eines.....	4
A.4.1 Explorador de cèl·lula.....	5
A.4.2 Datos.....	6
A.4.3 Geometria.....	6
A.4.4 Propiedades.....	6
A.4.5 Lista de errores.....	6
A.4.6 Salida.....	6
A.5 Barra d'opcions.....	6
A.5.1 Principal.....	7
A.5.2 VAL3.....	8
A.5.3 Modelat.....	8
A.5.4 Simulació.....	9
A.6 Creació d'una cèl·lula, elecció del controlador i robot.....	9
A.7 Afegir un robot a una cèl·lula ja existent.....	13
A.8 Creació d'aplicacions VAL3.....	15
A.8.1 Introducció.....	15
A.8.2 Assignació de la nova aplicació a un controlador.....	15
A.8.3 Contingut de la nova aplicació a l'explorador de cèl·lula.....	17
A.8.4 Contingut de la nova aplicació a Datos.....	17
A.8.5 Contingut de la nova aplicació a Geometria.....	20
A.8.6 Esborrar dades.....	22
A.8.7 Esborrar un controlador.....	23
A.8.8 Edició de l'aplicació generada (Start i Stop).....	23
A.9 Editor de perfils.....	28
A.10 Edició de l'entorn 3D.....	30
A.10.1 Introducció.....	30
A.10.2 Model 3D.....	30
A.10.3 Inserció de la taula de treball i edició de la seva posició.....	30
A.10.4 Posicionament del robot.....	33
A.10.5 Inserció de l'eina.....	34
A.10.6 Edició geomètrica de l'eina.....	35
A.10.7 Adjuntar l'eina al robot.....	40
A.10.8 Edició de punts i de l'eina.....	41
A.11 Emulador.....	49
A.12 Simulació 3D.....	51
A.13 Detecció de col·lisions.....	53
A.14 Transfer Manager.....	54

A.1 Introducció

El present document té per objectiu explicar el funcionament bàsic del software Stäubli Robotics Suite (2013). Aquest és un entorn de programació i simulació 3D de robots Stäubli. A més a més, té la capacitat d'emular els corresponents armaris de control. Aquesta simulació es realitza a través del software CS8 el qual està integrat dins el software Stäubli Robotics Suite 2013.

L'escassetat de material docent relacionat amb l'ús del software ha fet que la informació presentada en aquest document, sigui fruit tant de reunions personals amb el servei tècnic de Stäubli com d'una acurada investigació realitzada sobre el programari.

A.2 Iniciar SRS 2013

Per tal d'iniciar el software Stäubli Robotics Suite 2013 (SRS 2013) s'ha de fer doble clic sobre l'accés directe de l'escriptori. En el cas que no hi hagi l'accés directe, es pot accedir al programari anant a (visualitzar Figura 1): inici, tots els programes, staubli, SRS, 2013, Stäubli Robotics Suite 2013.

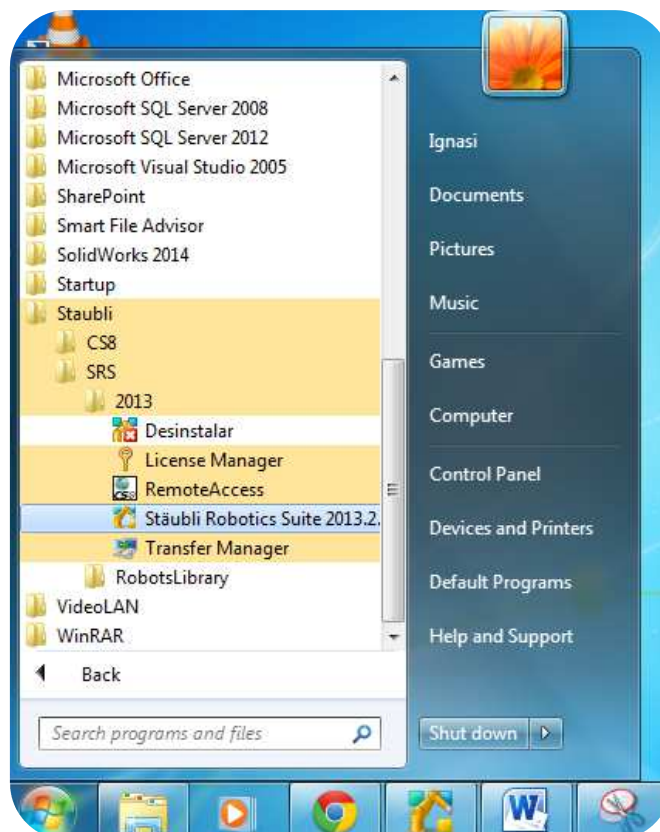


Figura 1: ruta per iniciar el software Stäubli Robotics Studio 2013

A.3 Idioma

Una vegada inicialitzat el programari, ens apareixerà la següent finestra (veure Figura 2):

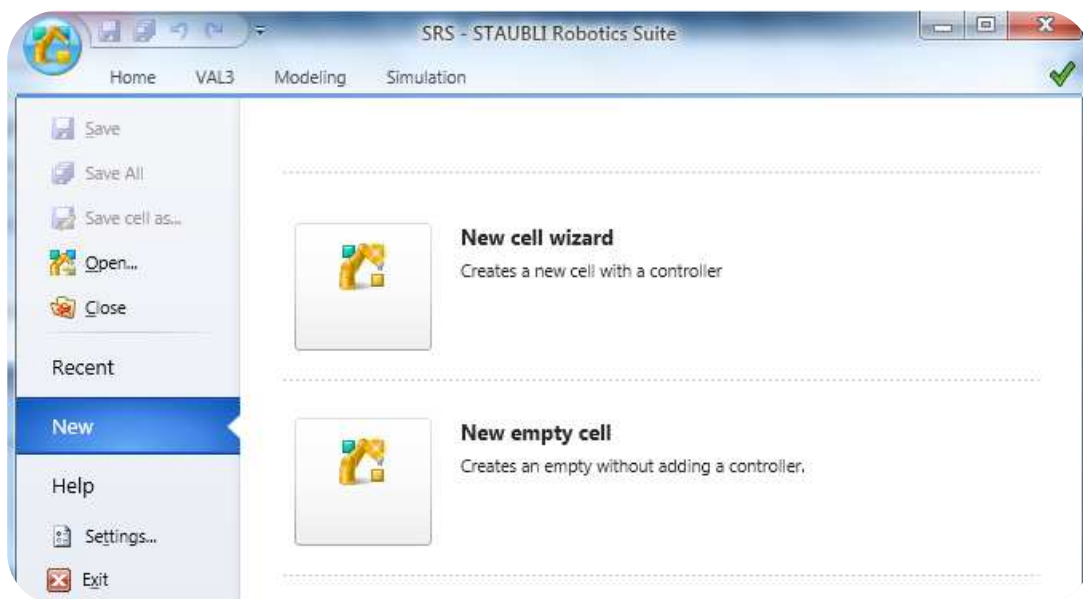


Figura 2: finestra que emergeix al iniciar el software SRS 2013

És aconsellable configurar-lo amb l'idioma disponible el qual hom domina amb més facilitat. Amb aquest objectiu, cliquem la pestanya Settings (veure Figura 2). Tot seguit ens apareixerà una finestra emergent (veure Figura 3).

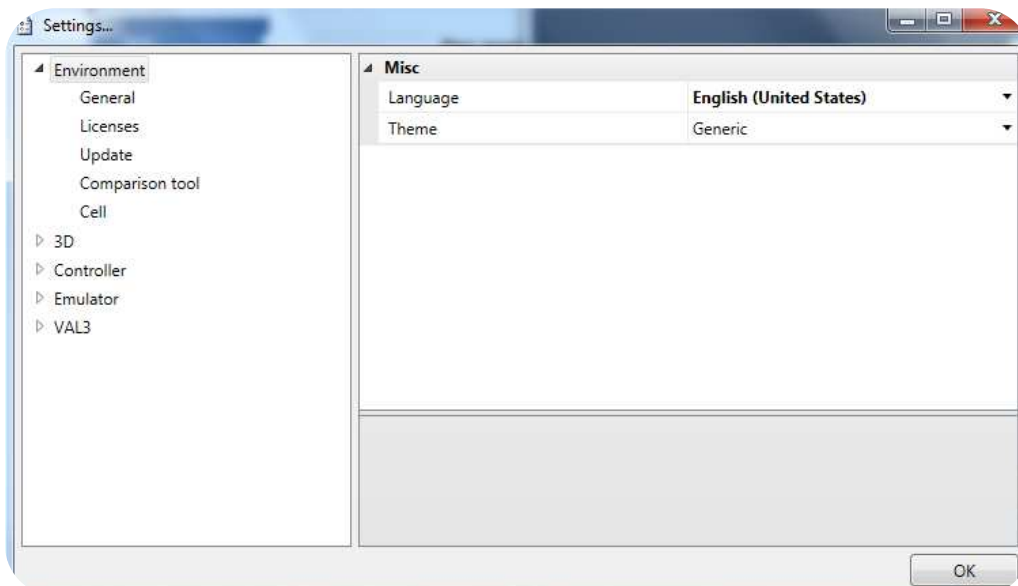


Figura 3: finestra de configuració (settings)

Una vegada seleccionat l'idioma més adequat per cada usuari, es clica OK i l'idioma queda modificat.

En aquest annex es fa servir el castellà amb l'objectiu de simplificar la comprensió per part de tots els usuaris.

A.4 Pestanyes d'eines

Per poder visualitzar les pestanyes d'eines cal clicar la icona superior esquerra la qual conté el dibuix d'un robot (veure Figura 4). Tornant a clicar la mateixa icona es pot tornar a la finestra anterior.



Figura 4: icona per accedir a la finestra de treball

La finestra resultant de clicar la icona (veure Figura 5), conté totes les eines necessàries per poder simular una aplicació sencera.

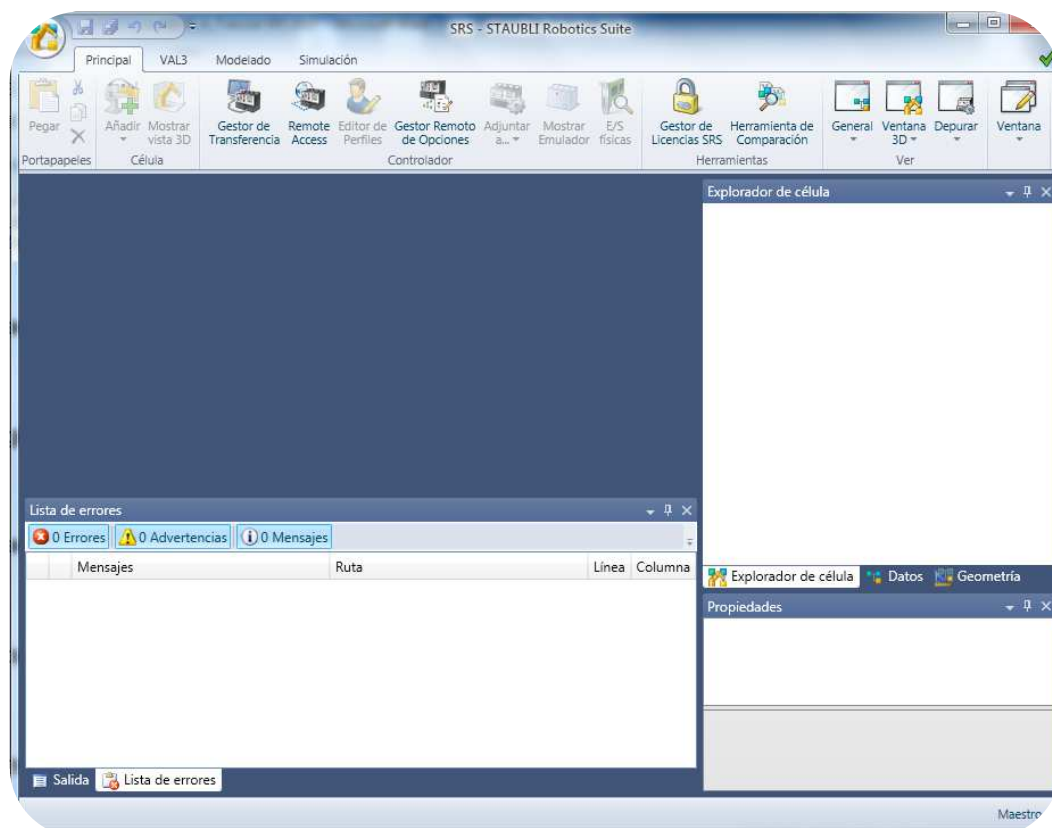


Figura 5: finestra de treball principal del software SRS 2013 sense cap cèl·lula creada

Tot seguit es comentaran les pestanyes d'eines més importants a les quals es pot accedir a través de la pestanya "General" (veure Figura 6). També s'hi pot accedir clicant la finestreta corresponent a cada pestanya (veure Figura 5).

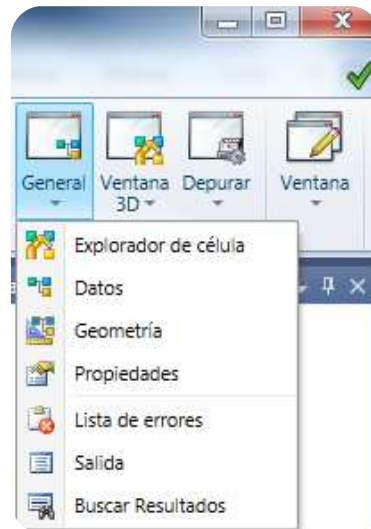


Figura 6: menú de pestanyes d'eines

A.4.1 Explorador de cèl·lula

És un explorador on s'organitzen tots els programes i biblioteques de cada una de les aplicacions obertes. Quan es crea una nova cèl·lula de treball, aquesta es mostra dins l'explorador de cèl·lula (veure Figura 7).

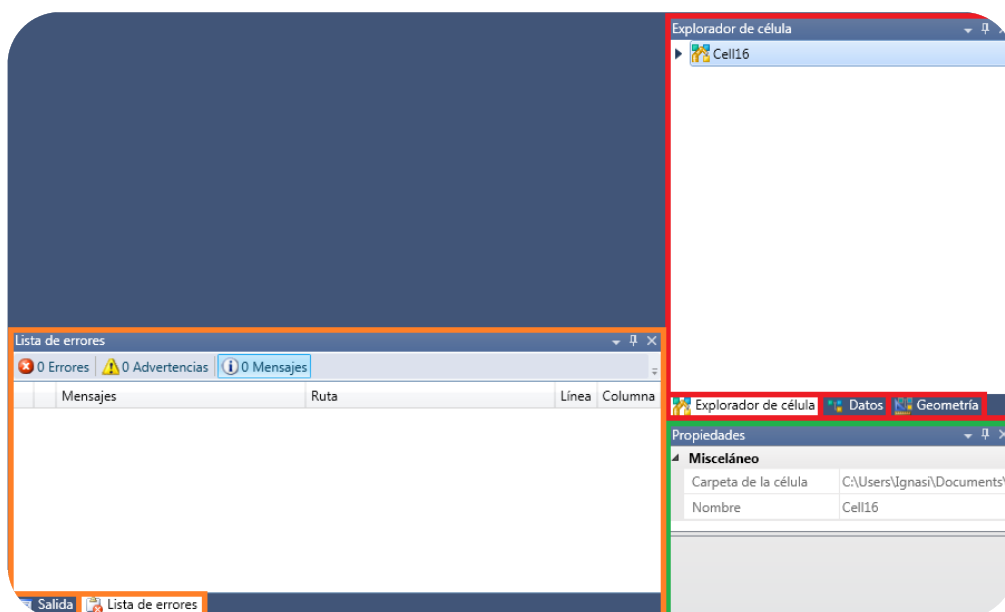


Figura 7: localització de les pestanyes d'eines més importants

A.4.2 Datos

Aquesta pestanya (veure Figura 7) conté totes les dades de cada aplicació. Aquests estan classificats segons el tipus de dada. Des d'aquesta pestanya es poden modificar els seus valors fent doble clic a sobre d'una d'elles. També s'hi poden crear noves dades.

A.4.3 Geometría

Aquesta pestanya (veure Figura 7) conté tots els punts, marques i eines de la cèl·lula. És on podem trobar i editar les propietats geomètriques de totes les figures, eines, punts i sistemes de referència.

A.4.4 Propiedades

Aquesta pestanya (veure Figura 7) mostra les propietats de l'objecte que es té seleccionat.

A.4.5 Lista de errores

Aquesta pestanya (veure Figura 7) mostra una llista amb els missatges, errors i advertències que es generen durant la comprovació de la sintaxis, la depuració i l'execució d'una aplicació.

A.4.6 Salida

Aquesta pestanya (veure Figura 7) mostra els missatges emesos. Aquests estan organitzats en tres categories: general, vista 3D i emulador.

A.5 Barra d'opcions

La barra d'opcions és una barra que esta organitzada per defecte en 4 grups principals. Aquests són: Simulació, Modelat, VAL3 i Principal. (veure Figura 8).



Figura 8: barra d'opcions amb el menú "Principal" seleccionat

En funció de la tasca específica que s'estigui realitzant en cada moment, pot ser que temporalment aparegui algun grup més.

Cada un d'aquests grups està subdividit en conjunts més petits. Aquests agrupen pestanyes que estan relacionades en funció de la seva utilitat. A la majoria de les pestanyes que apareixen en aquests grups, també s'hi pot accedir a través dels menús desplegable que apareixen al fer clic amb el botó de la dreta sobre de determinades icones (es veurà durant el desenvolupament del manual).

El software SRS 2013 permet amagar aquesta barra d'opcions. D'aquesta manera tant sols apareix quan es selecciona algun dels menús principals. Per fer-ho hem d'entrar al menú desplegable anomenat "Personalizar Barra de Herramientas de Acceso Rápido..." i clicar "Minimizar Barra de Opciones" (veure Figura 9).



Figura 9: ruta per minimitzar la barra d'opcions

En el nostre cas, deixarem la barra d'opcions sense minimitzar, ja que tenim suficient espai a la pantalla i no ens molesta la seva presència.

Tot seguit anem a comentar, de manera general, cada un dels grups esmentats.

A.5.1 Principal

Aquest grup (veure Figura 8), tal com diu el seu nom, conté les eines principals. Aquestes permeten realitzar les accions necessàries per afegir nous robots a la cèl·lula i visualitzar-la en 3D. També gestionar el controlador i l'emulador així com totes les seves respectives connexions. A més a més, permet obrir el gestor de llicències. En cas que haguem tancat alguna de les finestres de treball de manera accidental, també ens permet desplegar les finestres que es desitgi. Finalment a través del menú "ventana", "Distribuciones de ventana", es permet gestionar el layout de pestanyes i guardar configuracions de layout específiques (veure Figura 10).

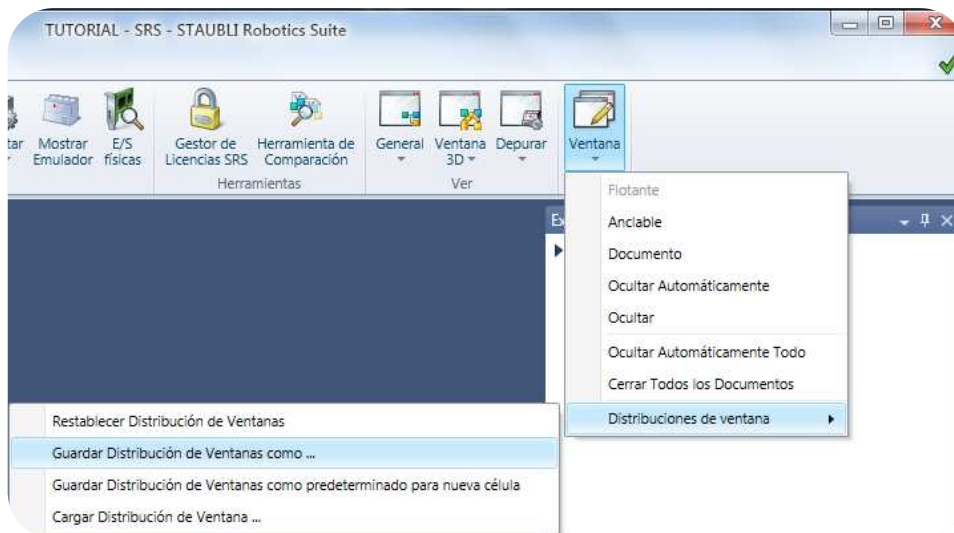


Figura 10: ruta per guardar o obrir un layout específic

A.5.2 VAL3

Aquest grup (Figura 11) permet crear i gestionar aplicacions VAL3. Conté les opcions de comprovació de la sintaxis, depuració i execució de les aplicacions, etc. Des d'aquest grup també es poden afegir llibreries.

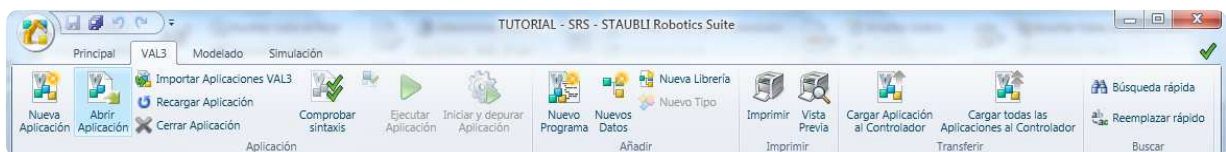


Figura 11: barra d'opcions amb el menú "VAL3" seleccionat

A.5.3 Modelat

Aquest grup (Figura 12) permet gestionar les geometries 3D. Es poden afegir geometries creades a través d'altres softwares CAD o generar geometries bàsiques. També permet definir les posicions d'aquestes geometries. En cas que es vulgui, el submenú "Herramientas" permet afegir una geometria com a eina, tot editant-ne les seves característiques.



Figura 12: barra d'opcions amb el menú "Modelado" seleccionat

A.5.4 Simulació

Aquest grup (Figura 13) conté totes les pestanyes relacionades amb la simulació 3D. Des d'aquí es pot seleccionar el tipus de moviment al qual es vol sotmetre el robot i editar punts a través de la posició del robot. També permet generar un conjunt de punts per mostrar la trajectòria realitzada pel robot i fer la detecció de col·lisions. Finalment, conté el menú de sincronització que permet sincronitzar amb un mateix tren de polsos diversos controladors.



Figura 13: barra d'opcions amb el menú "Simulación" seleccionat

A.6 Creació d'una cèl·lula, elecció del controlador i robot

Una vegada iniciat el programari, per crear una cèl·lula de treball cal que tornem a la finestra inicial (veure Figura 2) clicant l'ícona que té un robot dibuixat (veure Figura 4).

Una vegada aquí, ja podem començar el procés de creació d'una nova cèl·lula. Primer cal clicar: Nuevo, Asistente nueva cèl·lula (veure Figura 14).

Com es pot veure, la cèl·lula de treball també es podria crear buida, és a dir, sense cap controlador. En cas que es fes així, posteriorment des de dins la cèl·lula buida, s'hauria d'afegir un controlador amb un robot. Per aquest motiu ho fem ja directament.

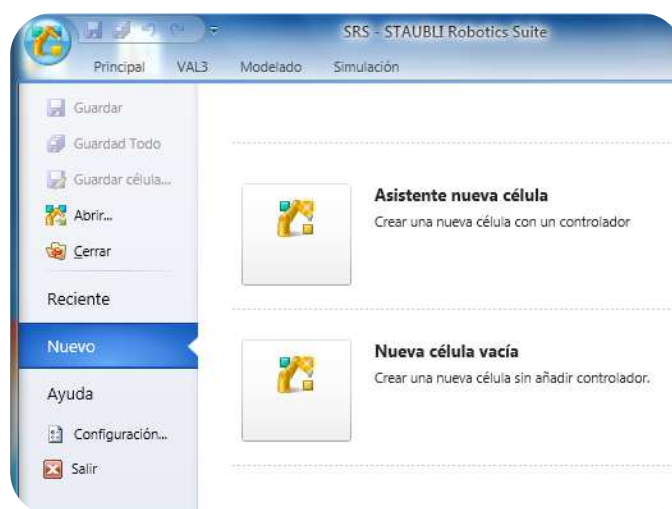


Figura 14: per crear una nova cèl·lula amb robot incorporat, cliquem Asistente nueva cèl·lula

Una vegada clicat, ens apareixerà una finestra emergent (veure Figura 15) la qual permet editar el nom de la cèl·lula i la seva ruta. Es recomana deixar la ruta ja especificada i editar el nom de la cèl·lula. En aquest cas posarem el nom: TUTORIAL.

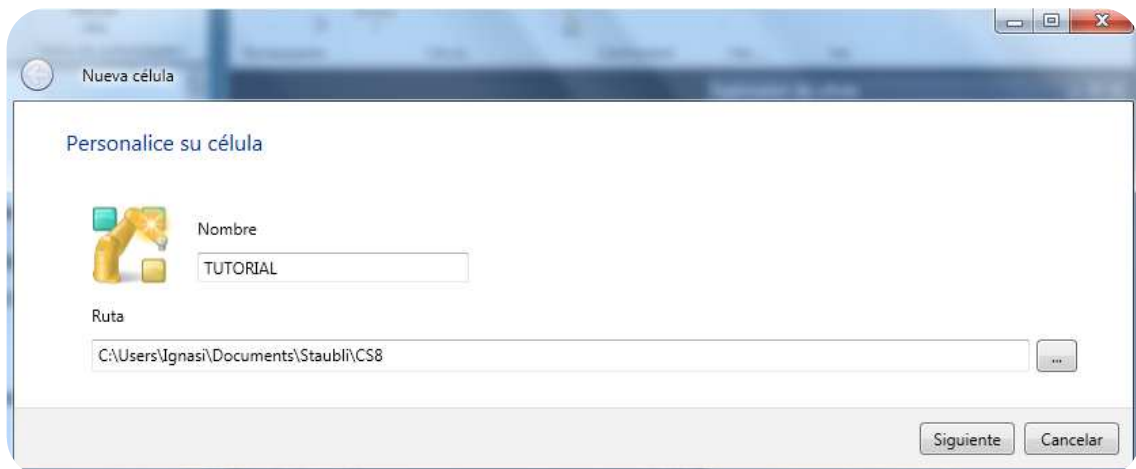


Figura 15: finestra emergent que permet editar el nom i la ruta de la nova cèl·lula de treball

Tot seguit haurem de seleccionar el tipus de controlador amb el que desitgem treballar (veure Figura 16). En el nostre cas, elegirem el controlador local ja que el que anem a fer és una simulació des del propi ordinador.



Figura 16: finestra per seleccionar el tipus de controlador amb el que es treballarà (local)

Una vegada seleccionat el controlador local, emergeix una nova finestra (veure Figura 17) la qual permet seleccionar la versió del controlador, tot editant-li el nom i el tipus de robot amb el que es vol treballar. Aquest es selecciona buscant-lo directament a la llista o editant-ne les seves característiques i, finalment, seleccionant el robot desitjat. En el nostre cas, editarem tots aquests paràmetres tal com es mostra a la Figura 17.

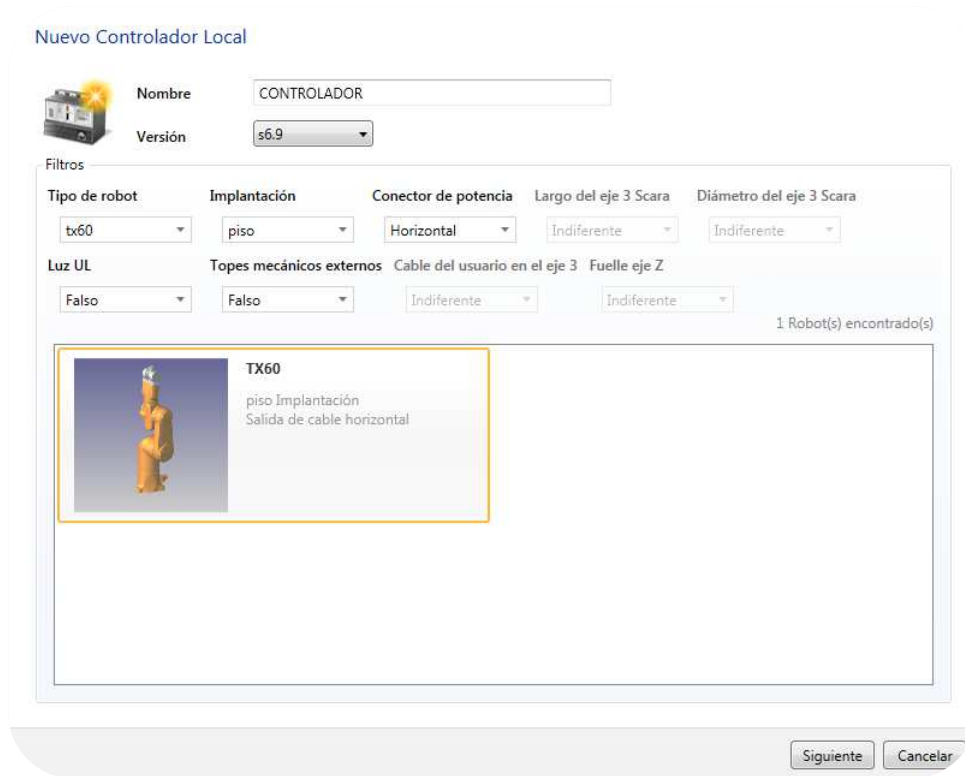


Figura 17: elecció del controlador i configuració del tipus de robot a seleccionar (TX60)

Tot seguit, haurem de seleccionar les opcions que volem que incorpori el nostre controlador. En el present tutorial es seleccionaran les opcions que es poden veure a la Figura 18.

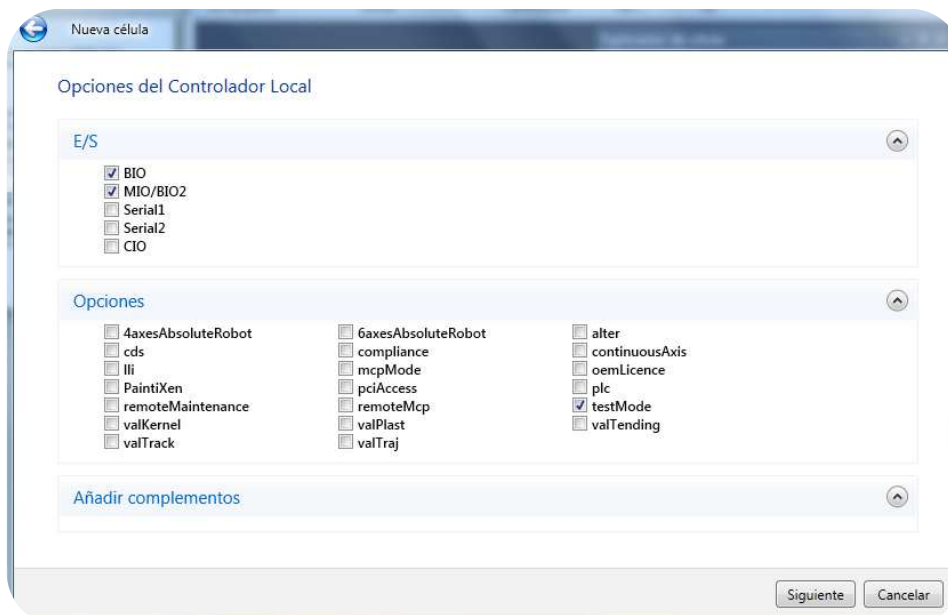


Figura 18: finestra per seleccionar les opcions del controlador local

Finalment ens apareix una finestra purament informativa (Figura 19) la qual ens indica els resultats del procés el qual acabem de realitzar.

Página final

La cèlula 'TUTORIAL' será creada con:
- un controlador local 'CONTROLADOR' con un brazo tipo 'tx60'.

Figura 19: finestra informativa que indica que la creació de la cèl·lula està a punt d'acabar

Cliquem finalitzar i hauré acabat el procés de creació de la cèl·lula. El resultat final hauria de ser el que es pot veure a la Figura 20. Com podem veure, hem entrat automàticament a la finestra de pestanyes d'eines. Com que ara ja tenim la cèl·lula creada i per tant, el controlador i el robot editats; el software ens porta automàticament a la finestra de pestanyes d'eines perquè puguem començar a treballar amb la nova cèl·lula. A la imatge es pot observar que a dins de les pestanyes d'eines comentades amb anterioritat, ha aparegut una nova cèl·lula de treball anomenada TUTORIAL. Aquesta pestanya es pot desplegar i editar tant a l'explorador de cèl·lula com a datos i a geometría.

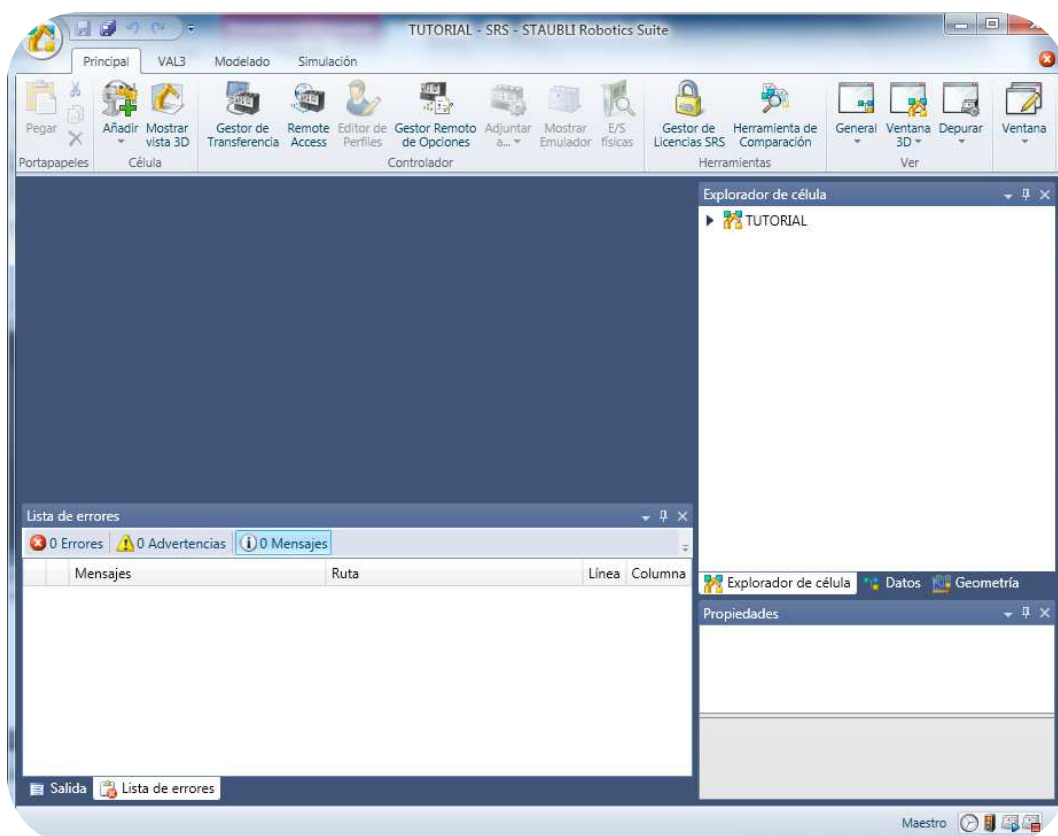


Figura 20: finestra resultant després d'haver creat una nova cèl·lula de treball

A.7 Afegir un robot a una cèl·lula ja existent

Una vegada creada la cèl·lula de treball seguint el procediment anterior, tenim un controlador amb un sol robot. Ara però, anem a suposar que estem programant un procés en el qual hi ha més d'un robot implicat. En aquest cas, seria útil poder introduir més d'un robot a una mateixa cèl·lula de treball. Així, podríem programar-los i sincronitzar-los. D'aquesta manera es poden evitar col·lisions o comportaments no desitjats. Amb aquest objectiu, el software SRS 2013 permet afegir nous robots a una cèl·lula de treball. Cada un d'aquests robots té el seu propi controlador la versió del qual pot ser diferent dels altres controladors. Per introduir un nou robot només cal anar a l'explorador de cèl·lula, clicar amb el botó dret sobre la cèl·lula ja creada (TUTORIAL) i clicar añadir, Nuevo robot (veure Figura 21).

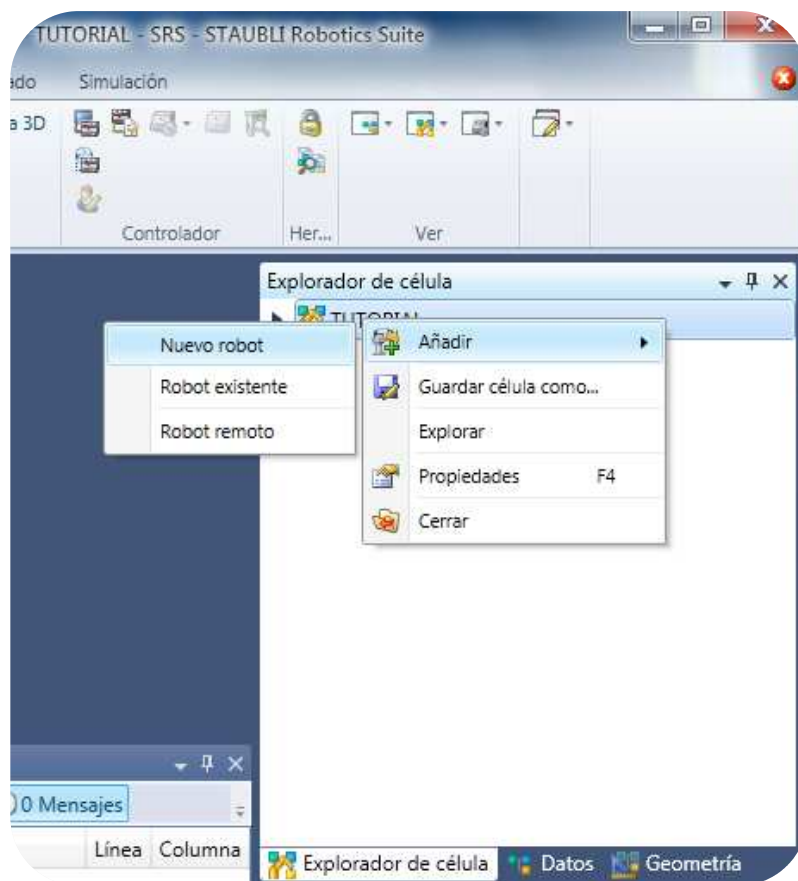


Figura 21: ruta per afegir un nou robot amb el seu controlador

Una vegada hem clicat Nuevo robot, ens apareixerà una finestreta com la de la Figura 17. En aquest cas configurarem el nou robot una mica diferent. Per no repetir el tipus de robot,

seleccionarem un robot RS60 i anomenarem el seu controlador com CONTROLADOR ROB2. La seva configuració serà la que es pot veure a la Figura 22.

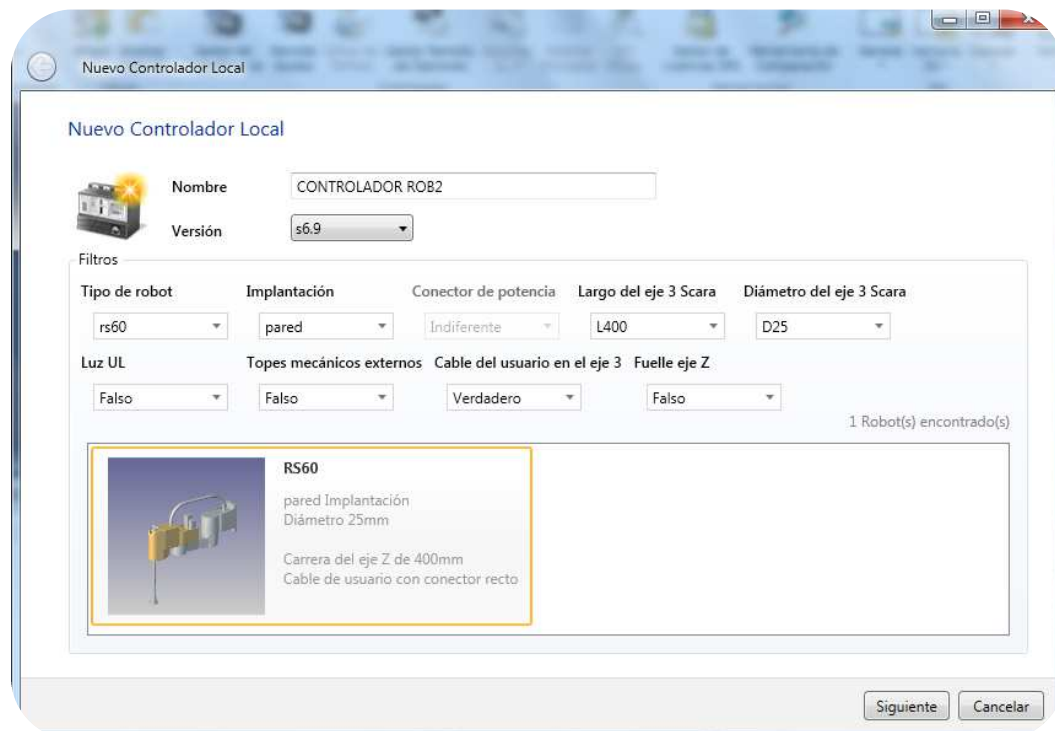


Figura 22: elecció del controlador i configuració del tipus de robot a seleccionar (RS60)

Posteriorment, configurem les opcions del controlador local igual que les que havíem configurat pel robot TX60 (veure Figura 18).

Tot seguit, anem a veure com ha quedat l'explorador de cèl·lula. Com es pot observar a la Figura 23, a l'explorador de cèl·lula ara apareixen dos controladors anomenats CONTROLADOR i CONTROLADOR ROB2. Aquests contenen un robot Tx60 i Rs60 respectivament.

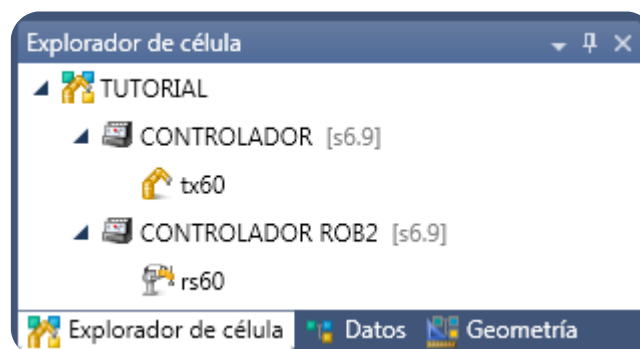


Figura 23: explorador de cèl·lula amb dos robots

Com es pot veure a la Figura 23 el nom dels controladors va seguit, entre claudàtors, de la versió del controlador.

Ara ja tenim creada una cèl·lula de treball a la que hi tenim dos robots. Aquests però, encara no tenen implementada cap aplicació. Tot seguit caldrà explicar com editar una aplicació i com la podem relacionar amb un controlador en concret.

A.8 Creació d'aplicacions VAL3

A.8.1 Introducció

Per crear una aplicació VAL3 es necessita una cèl·lula de treball i un controlador amb un robot associat. Tal com es pot veure a la Figura 23, ja tenim la cèl·lula creada i hi tenim dos controladors. Ara, per tant, ja podem crear una nova aplicació VAL3.

A.8.2 Assignació de la nova aplicació a un controlador

Tot seguit crearem una aplicació associada al robot Tx60 el qual esta associat al controlador anomenat CONTROLADOR. Per crear l'aplicació fem doble clic sobre el nom del controlador i, del menú desplegable que s'obre, cliquem Nueva aplicación (veure Figura 24).

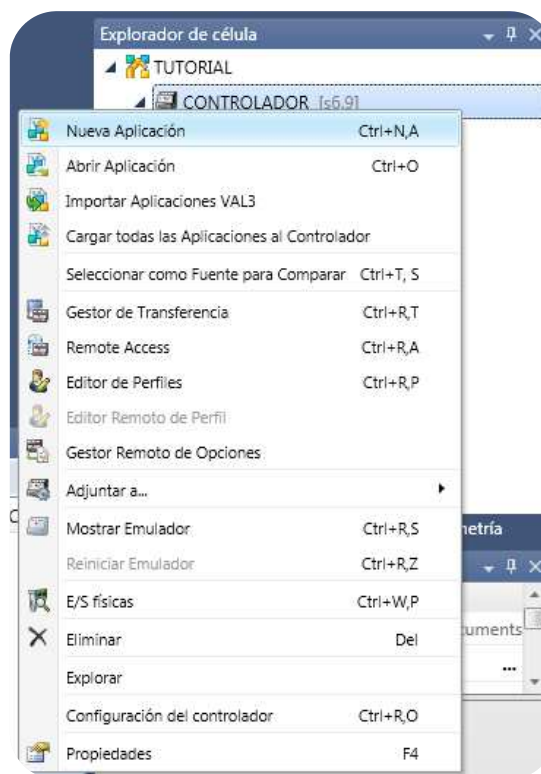


Figura 24: passos a seguir per crear una nova aplicació a un determinat controlador

Després de clicar “Nueva aplicación” ens apareix una finestra emergent (Figura 25) la qual permet editar el nom de l'aplicació, la seva ubicació i, finalment, permet començar-la amb una plantilla determinada (no ho farem servir). En el nostre cas deixarem la ubicació i la plantilla que ja surt per defecte i l'anomenarem AppTuto.

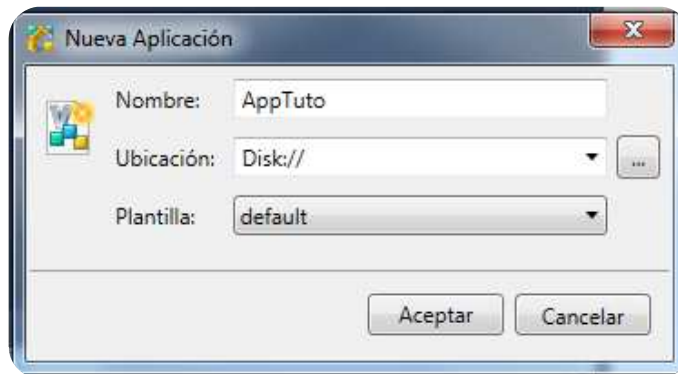


Figura 25: finestra emergent de configuració de la nova aplicació

Com es pot veure a la Figura 26, l'aplicació creada està associada al controlador corresponent al robot Tx60.

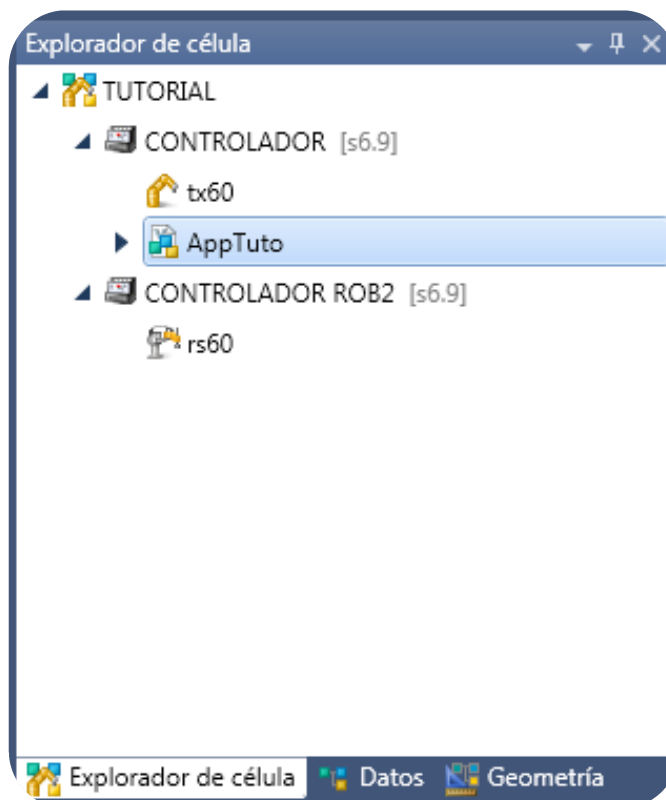


Figura 26: explorador de cèl·lula amb la nova aplicació associada al controlador del robot Tx60

A.8.3 Contingut de la nova aplicació a l'explorador de cèl·lula

Tal com ja succeïa a les versions anteriors del software de Stäubli, la nova aplicació conté dos rutines anomenades “start” i “stop”. La rutina “start” és la que s'executa automàticament al iniciar l'aplicació. La rutina “stop” és la que s'executa al finalitzar-la.

Per altra banda, tal com es pot veure a la Figura 27, l'aplicació creada també conté una llista desplegable de totes les entrades i sortides (io) que es poden utilitzar a l'aplicació.

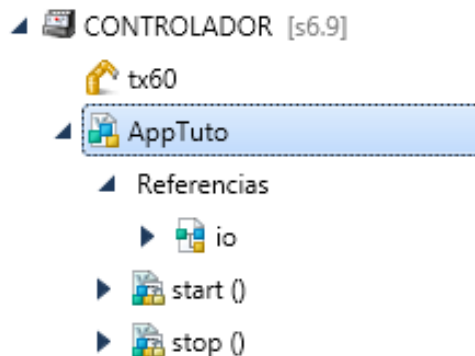


Figura 27: visualització del contingut de l'aplicació generada

A.8.4 Contingut de la nova aplicació a Datos

Tal com es mostra a la Figura 28, dins la pestanya “Datos” també ha aparegut l'aplicació creada. Aquesta conté les dades corresponents a aquesta aplicació. En aquest cas, podem veure com hi ha la variable anomenada mNomSpeed, la qual conté el valor per defecte de la velocitat del braç.

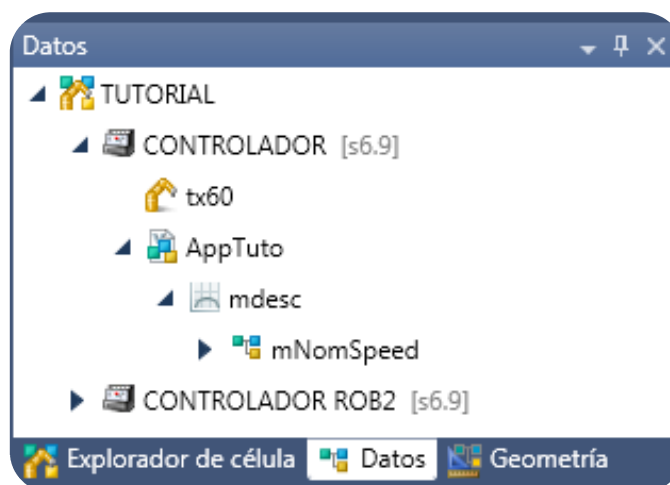


Figura 28: estructura de les dades de l'aplicació generada

Per afegir noves dades, cliquem amb el botó de la dreta sobre el nom de l'aplicació, seleccionem "añadir", Nuevos datos (veure Figura 29).

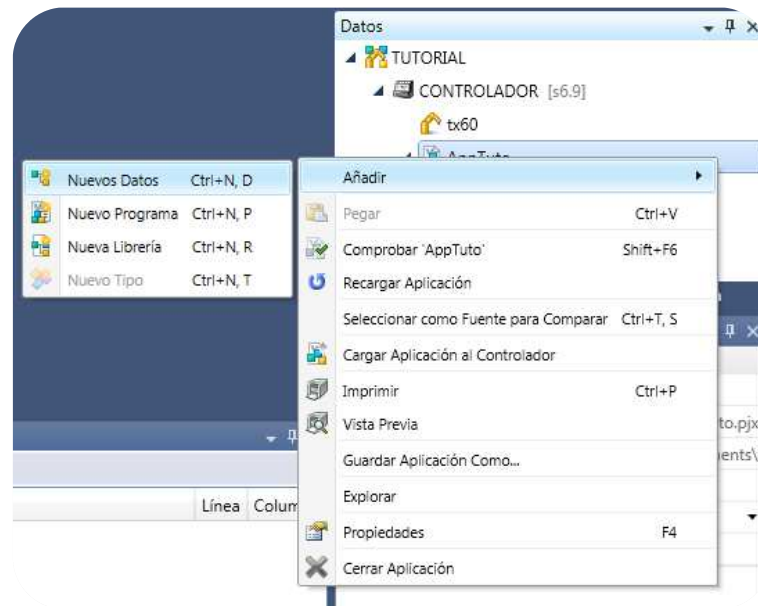


Figura 29: inserció de noves dades des de l'aplicació generada

Seguidament, apareixerà una finestra emergent (Figura 30) a on es podrà seleccionar el tipus de dada que es vol afegir. En funció de si és una dada geomètrica o no, aquesta s'afegirà a "Datos" o a "Geometría" (sovint a tots dos). Anem doncs a afegir una nova dada no geomètrica, per exemple una variable (un número).

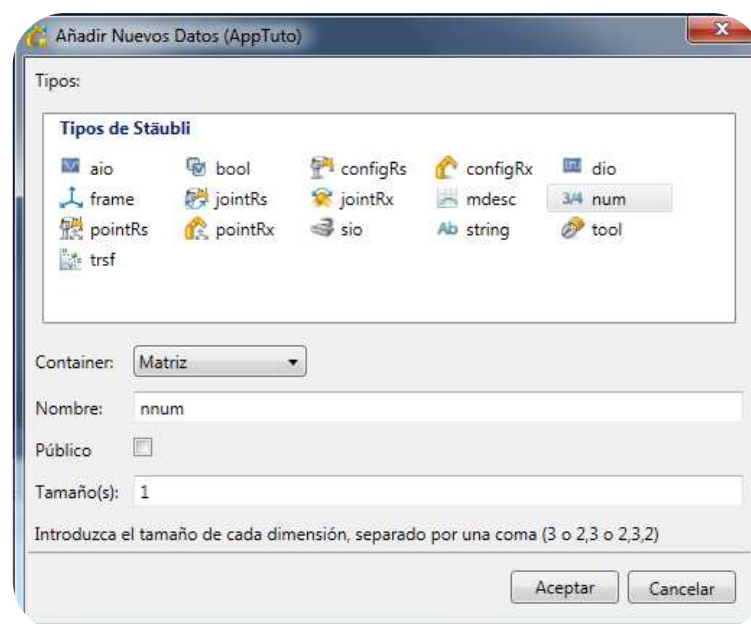


Figura 30: finestra que permet seleccionar el tipus de dada que es vol afegir a una aplicació

Si configurem la nova dada tal com es mostra a la Figura 30 i cliquem acceptar, veiem com aquesta nova dada, que és del tipus num, apareix dins la pestanya de "Datos".

Tal com es pot veure a la Figura 31, les dades no geomètriques queden agrupades en funció del tipus de dada. Així doncs, de moment tenim dos grups: mdesc, el qual conté les dades que guarden els valors de les velocitats del braç; i num, el qual conté dades que guarden valors numèrics.

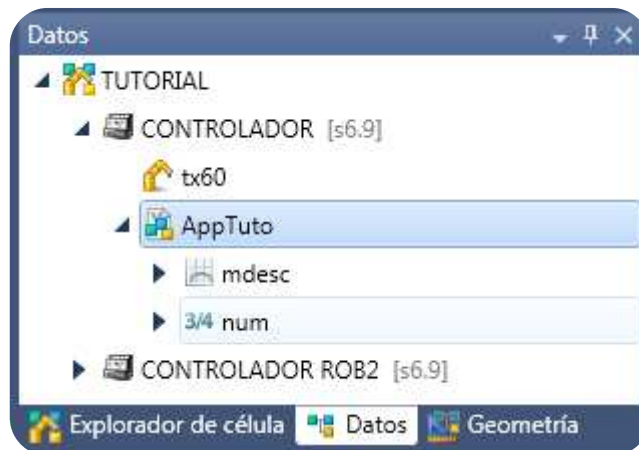


Figura 31: organització de les dades dins la pestanya "Datos" per tipus de dada

Si introduïm una nova dada numèrica i despleguem la pestanya num, obtenim el resultat que es pot observar a la Figura 32.



Figura 32: organització de les dades d'un mateix tipus, en aquest cas, dades numèriques

Per introduir aquesta nova dada podem fer-ho repetint el procés anterior (des de l'aplicació) o, en cas que ja existeixi el tipus de la dada que es vol afegir, clicant directament sobre el tipus de dada i seleccionant sobre "Nuevos Datos" (veure Figura 33). Això farà que la finestra emergent (Figura 30) ja sigui del tipus seleccionat.

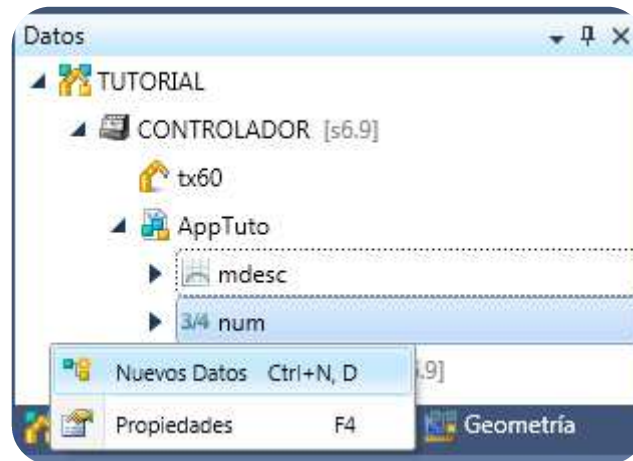


Figura 33: inserció de dades d'un tipus específic

A.8.5 Contingut de la nova aplicació a Geometria

Tal com es pot veure a la Figura 34, dins la pestanya Geometría també ha aparegut l'aplicació creada. Tal com ja s'ha comentat, dins aquesta pestanya hi ha totes les dades geomètriques de l'aplicació.



Figura 34: organització de les dades dins la pestanya "Geometría" per tipus de dada geomètrica

Repetint un procés similar al explicat anteriorment, es poden afegir noves dades geomètriques tant des de l'aplicació com des d'un tipus de dada específic. En ambdós casos s'ha de clicar amb el botó de la dreta i seleccionar "Nuevos Datos" (Figura 35).

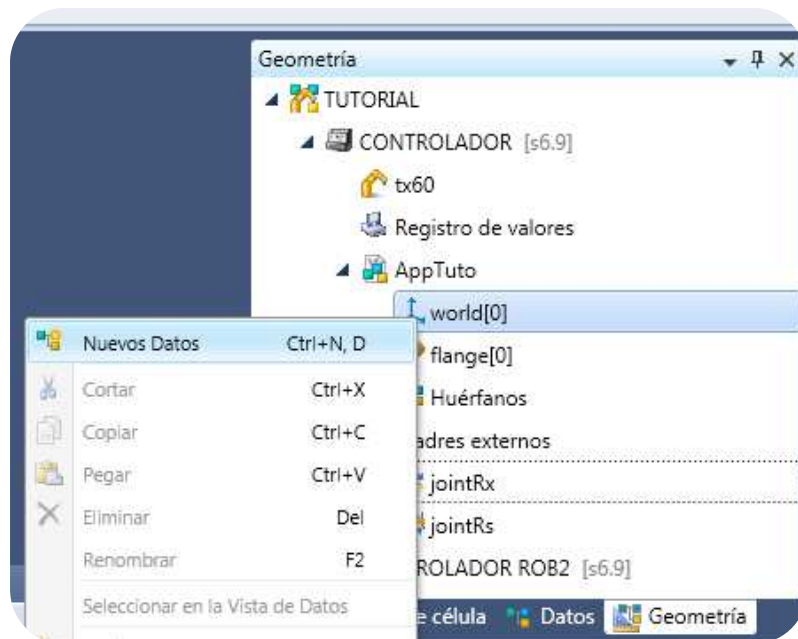


Figura 35: procés de creació de noves dades des de l'aplicació dins la pestanya "Geometría"

Després s'obre una finestra on es pot seleccionar el tipus de dada, editar-ne el seu nom, la seva dimensió, etc. (veure Figura 36).

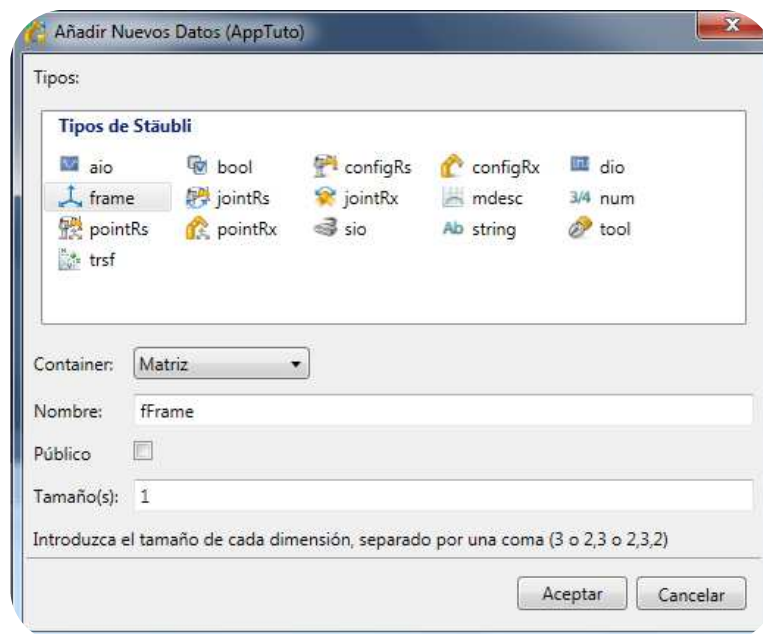


Figura 36: finestra de selecció del tipus de dada amb edició del nom i la dimensió

De manera similar a la pestanya “Datos”, la pestanya “Geometría” organitza les dades per grups desplegable en funció del tipus (veure Figura 37).

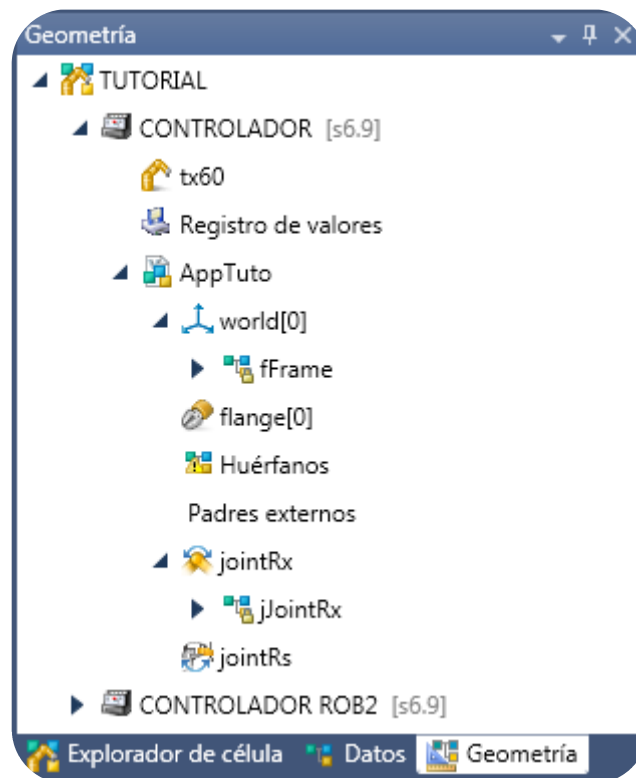


Figura 37: organització de les dades geomètriques per grups de tipus de dada

A.8.6 Esborrar dades

Per esborrar dades ja introduïdes, cliquem amb el botó dret a la dada que desitgem esborrar i, del menú desplegable que s'obre, seleccionem “Eliminar” (veure Figura 38).

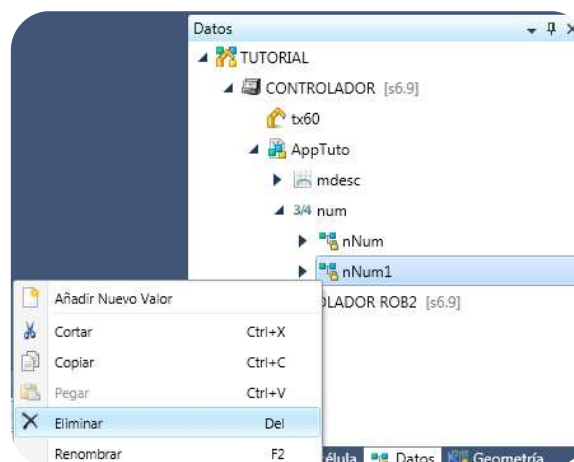


Figura 38: ruta per eliminar dades des del menú “Datos”

Repetim aquest procés per les dades numèriques introduïdes com a exemple ja que no les utilitzarem.

A.8.7 Esborrar un controlador

Per esborrar controladors ja introduïts a la cèl·lula, cliquem amb el botó dret el controlador que desitgem esborrar i, del menú desplegable que s'obre, seleccionem "Eliminar" (veure Figura 39).

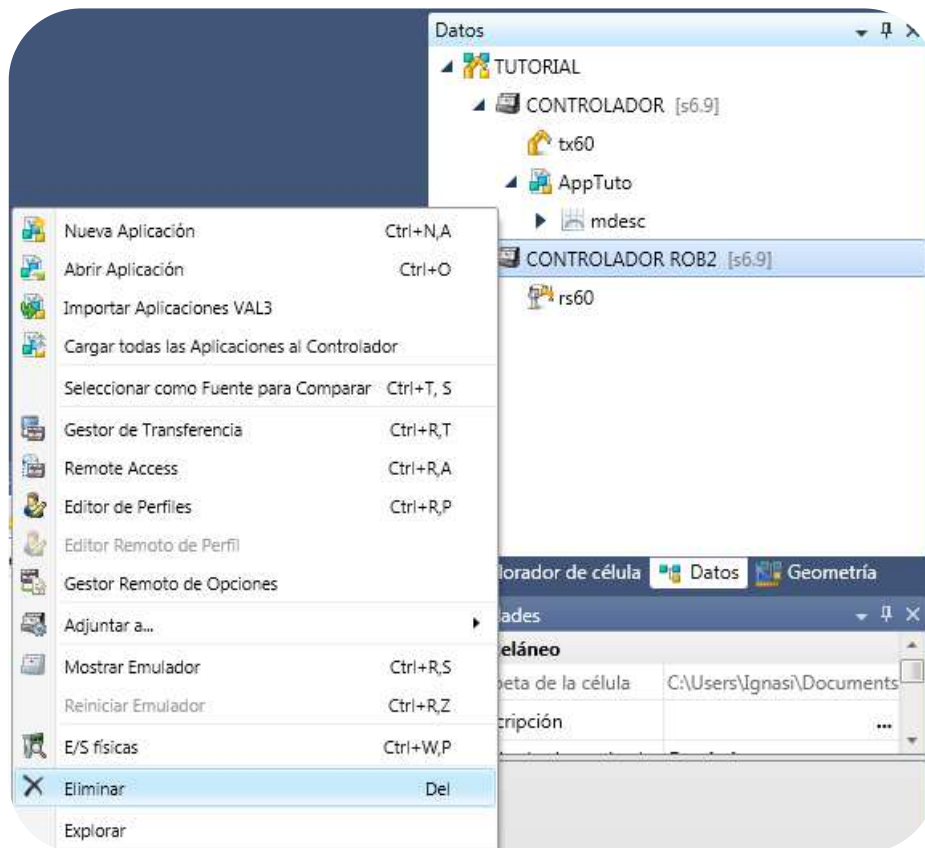


Figura 39: ruta per eliminar un controlador des de la pestanya "Datos"

Eliminem el CONTROLADOR ROB2 ja que està associat a un robot del tipus Rs60 i, de moment, no disposem d'ell al laboratori. Per tant no l'utilitzarem.

A.8.8 Edició de l'aplicació generada (Start i Stop)

Per tal d'editar l'aplicació AppTuto, anem a editar les rutines Start i Stop. Tal com ja s'ha comentat anteriorment, la rutina Start és la que s'executa al iniciar l'aplicació i la rutina Stop s'executa al finalitzar-la.

Si anem a l'explorador de c elula i fem doble clic sobre les rutines Start() i Stop() (Figura 27), veiem que s'obre una nova pestanya dins la qual podem editar directament cada una d'aquestes rutines (Figura 40).

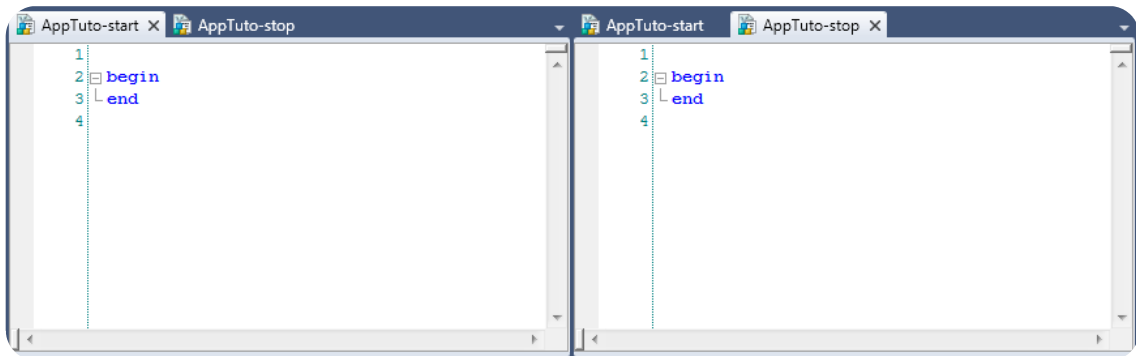


Figura 40: pestanyes per editar les rutines Start() i Stop()

Tamb e observem que a la barra d'opcions apareix una nova pestanya anomenada "Herramientas de texto" (veure Figura 41).



Figura 41: barra d'opcions amb el nou men  "Herramientas de texto" seleccionat

Utilitzarem el seg ent codi el qual s'ha de copiar dins de les rutines Start() i Stop():

Start():

```

Begin
userPage()
cls()
putln("Inici del programa")
enablePower()
mNomSpeed.vel=30
movej(posicio2[0],eina0[0],mNomSpeed)
close(eina0[0])
putln("Posicio 2a assolida. Element terminal tancat.")
open(eina0)
movej(posicio2[1],eina0[1],mNomSpeed)
close(eina0[1])
putln("Posicio 2b assolida. Element terminal tancat.")
end

```

Stop():

begin

```
movej(posicio0,eina0,mNomSpeed)
```

```
open(eina0)
```

```
putln("Posicio 0 assolida. Element terminal obert. Programa finalitzat")
```

```
popUpMsg("Pending movement commands have been canceled")
```

```
resetMotion()
```

end

Aquestes comandes corresponen a un codi molt senzill el qual s'utilitza per mostrar com emprar el software. Aquest annex no està dedicat a l'explicació del codi.

Els elements que s'hauran de definir perquè l'aplicació pugui compilar-se són: `posicio0`, `posicio2`, `eina0`, `mNomSpeed`. Tot seguit els editarem un per un:

1. `posicio0`: anem a datos, clic amb el botó dret, "añadir", "Nuevos datos" (Figura 29). Tot seguit configurem el nou punt tal com mostra la Figura 42.

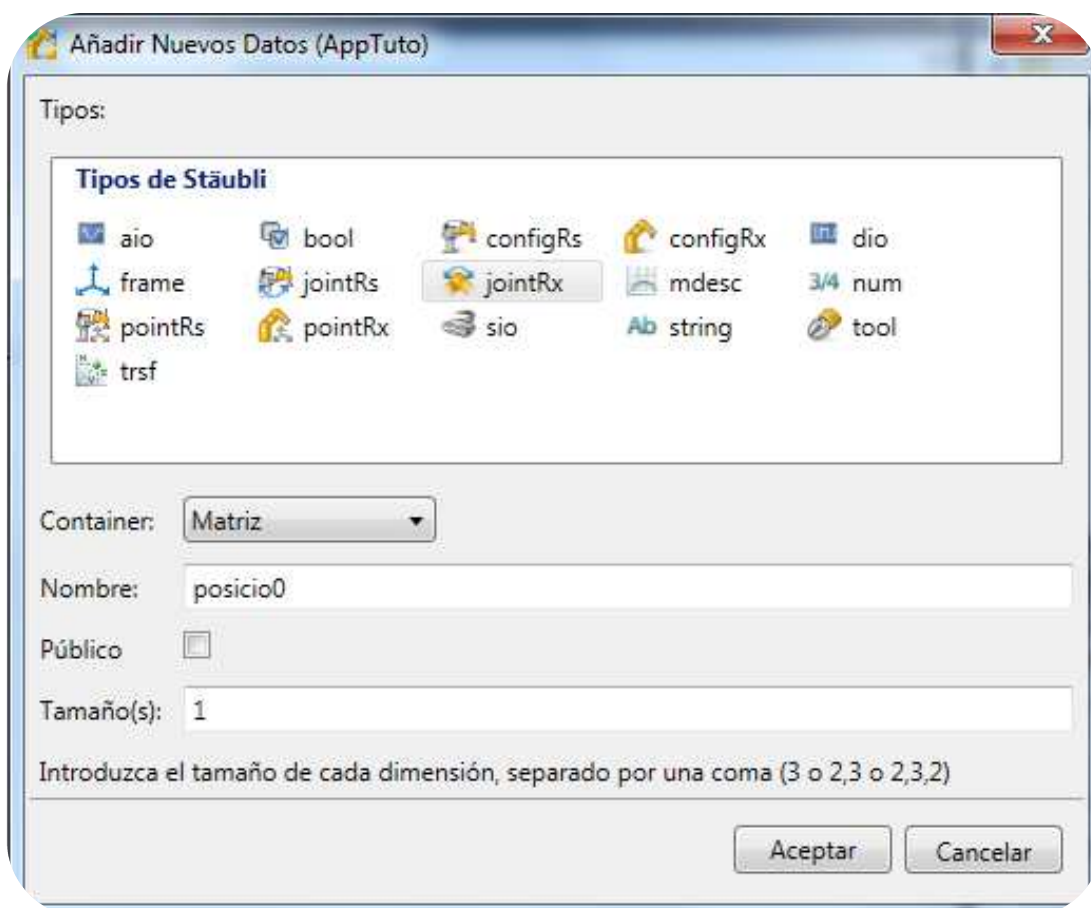


Figura 42: edició el punt "posicio0", del tipus jointRx (posicions articulars)

2. posicio2: repetim el procés anterior i configurem la nova dada (punt tipus pointRx) com es mostra a la Figura 43.

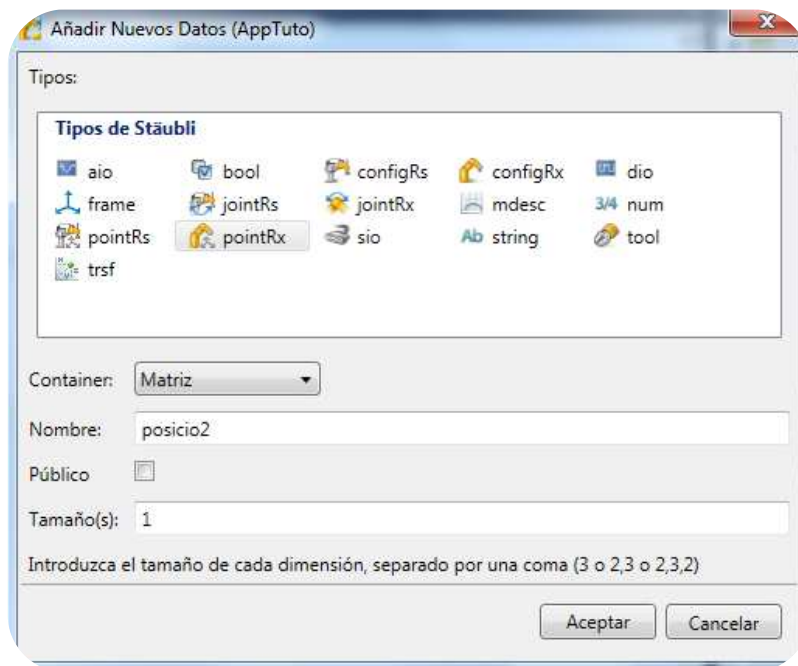


Figura 43: edició el punt “posicio2”, del tipus pointRx (posició cartesiana)

3. eina0: repetim el procés anterior i configurem la nova dada (eina tipus tool) com es mostra a la Figura 44.

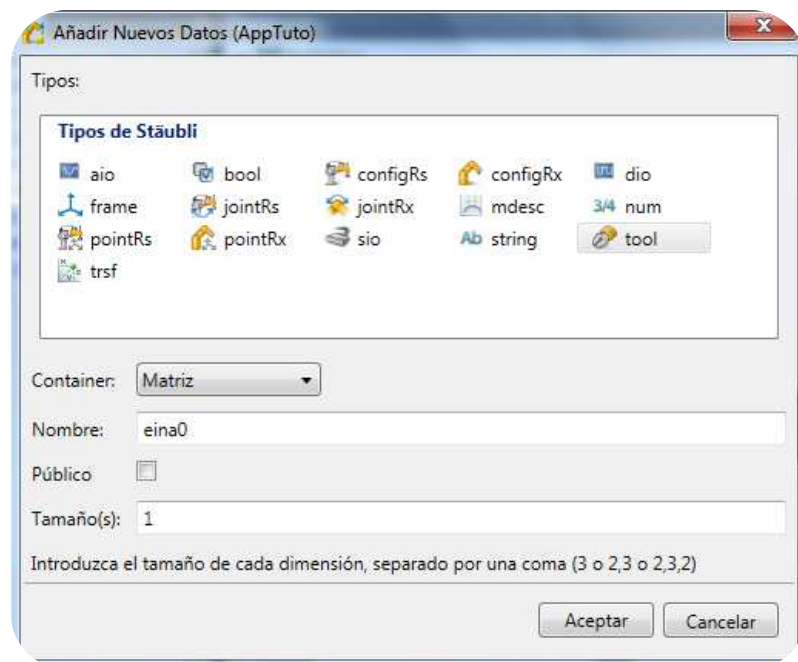


Figura 44: edició de l'eina “eina0”, del tipus tool

4. mNomSpeed: per defecte ja està definida al crear l'aplicació.

A la Figura 45 veiem com queda la pestanya “Datos” després de declarar totes les dades necessàries per poder executar l'aplicació “AppTuto” editada.

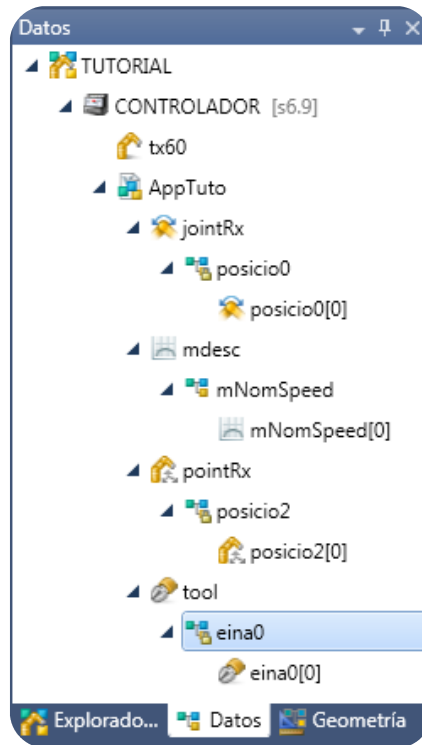


Figura 45: estructura de les dades declarades dins la pestanya “Datos”

Tot seguit, anem a comprovar el codi. Per fer-ho, tanquem les pestanyes de l'editor de codi. Quan preguntem si el volem guardar i li diem que sí (podríem haver-lo guardat prèviament amb la icona del disquet o fent Ctrl+S). Després anem a l'explorador de cèl·lula, cliquem sobre l'aplicació AppTuto i, a la barra d'opcions dins la pestanya VAL3, cliquem comprovar sintaxis (veure Figura 46).



Figura 46: comprovació de la sintaxis

Podem observar (Figura 47) que no hi ha cap error ni cap advertència que ens alerti d'algun error en l'edició del codi.

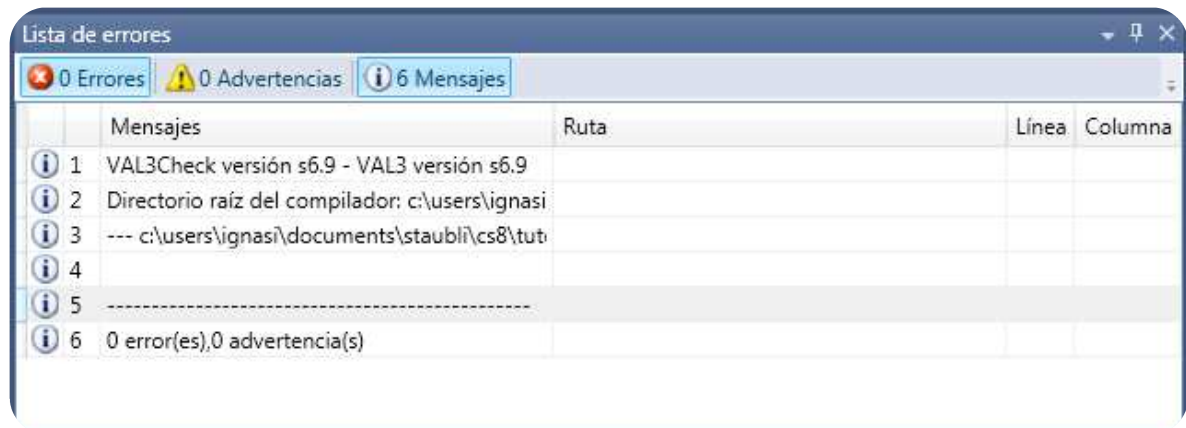


Figura 47: llista d'errors que es mostra al comprovar la sintaxis d'un codi

Finalment doncs, amb les dades necessàries creades i el codi escrit, ja tenim l'aplicació generada. Tant sols quedarà simular-la a través de l'emulador i vincular-la a l'entorn de treball 3D.

Abans però, donat que serà necessari per explicar l'emulador, es comenta breument com funciona l'editor de perfils.

A.9 Editor de perfils

Tal com les versions anteriors del software, aquest conté un editor de perfils d'usuari. Nosaltres treballarem amb el perfil anomenat "default" el qual no té contrasenya. El software però, permet la creació de perfils específics als quals els hi podem concedir unes capacitats concretes. A més a més, podem posar una contrasenya a cada un dels perfils. Si anem a la pestanya principal de la barra d'opcions, dins el subgrup "controladors" trobarem una casella anomenada "Editor de perfils" (veure Figura 48). La cliquem.



Figura 48: casella per accedir a l'editor de perfils

Una vegada clicada la casella, s'obre la finestra que es pot veure a la Figura 49.

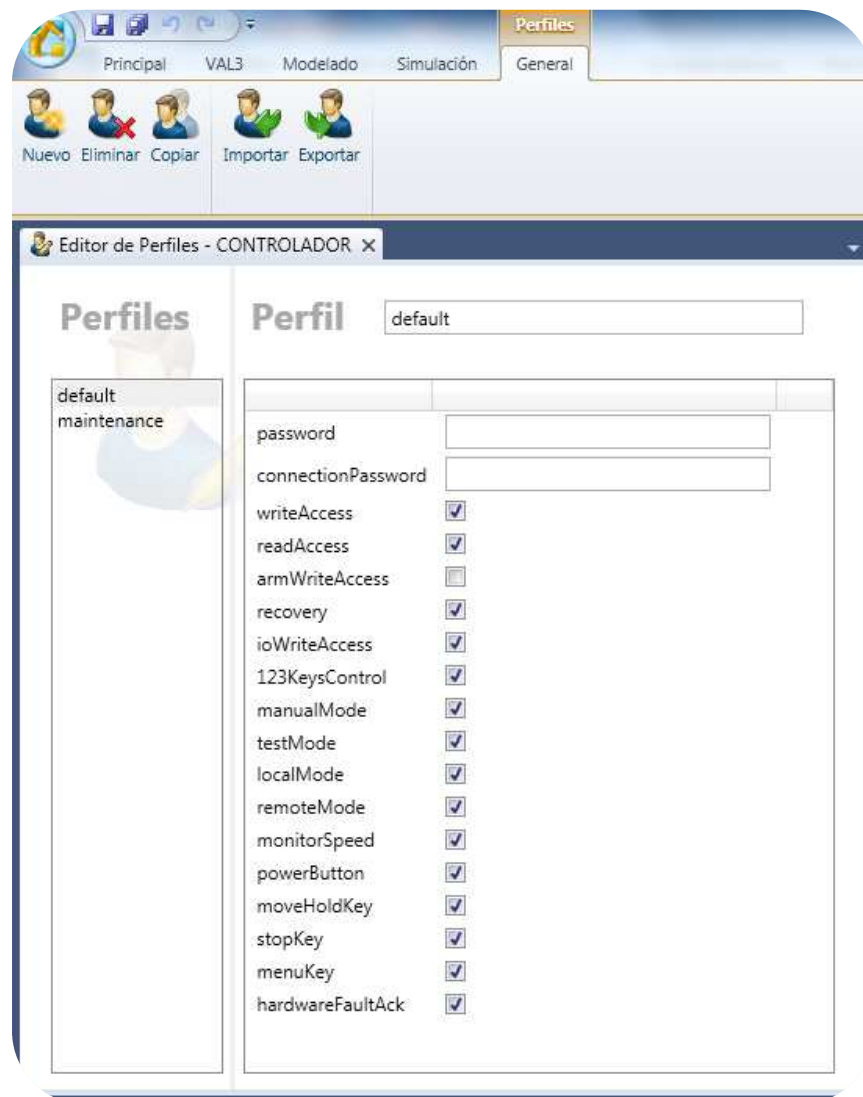


Figura 49: editor de perfils

Observem que a la barra d'opcions apareix una nova pestanya anomenada "Perfiles". Aquesta és l'encarregada de gestionar els perfils. Com es pot veure, es poden generar perfils nous, copiar-ne de ja existents i eliminar-los. Per altra banda, el software també permet importar i exportar perfils ja creats amb anterioritat.

Si es volgués crear un perfil nou, s'hauria de clicar "Nuevo" i després editar el nom del perfil, la contrasenya i les opcions que desitgem que tingui disponibles.

A.10 Edició de l'entorn 3D

A.10.1 Introducció

Amb l'objectiu final de poder simular el comportament del robot, s'han d'introduir les geometries 3D generades a través del software CAD SolidWorks a dins la cèl·lula de treball del robot. Aquesta inserció es pot realitzar important l'arxiu CAD directament o important-lo com un enllaç (veure Figura 50). Aquesta última opció permet que si la geometria CAD es veu modificada, la cèl·lula de treball importarà actualitzades aquestes modificacions.



Figura 50: opcions disponibles per importar un fitxer CAD

A.10.2 Model 3D

Per poder visualitzar el model 3D, hem de clicar la pestanya anomenada "Mostrar vista 3D" (Figura 51) que es troba dins la pestanya principal de la barra d'opcions.

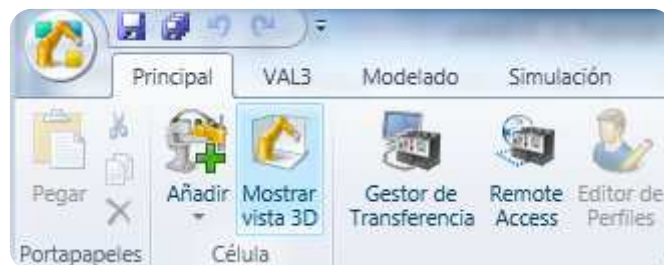


Figura 51: icona que permet visualitzar l'entorn 3D

A.10.3 Inserció de la taula de treball i edició de la seva posició

Tot seguit anem a importar l'estació de treball generada amb SolidWorks. Per fer-ho, la importarem com a enllaç. Una vegada clicada aquesta opció (Figura 50), seleccionem la geometria a inserir (Figura 52) i l'obrim.



Figura 52: selecció de la geometria CAD a importar corresponent a l'estació de treball

Després d'un cert interval de temps el qual es destina a processar la geometria importada, aquesta apareix dins el model 3D (Figura 53).

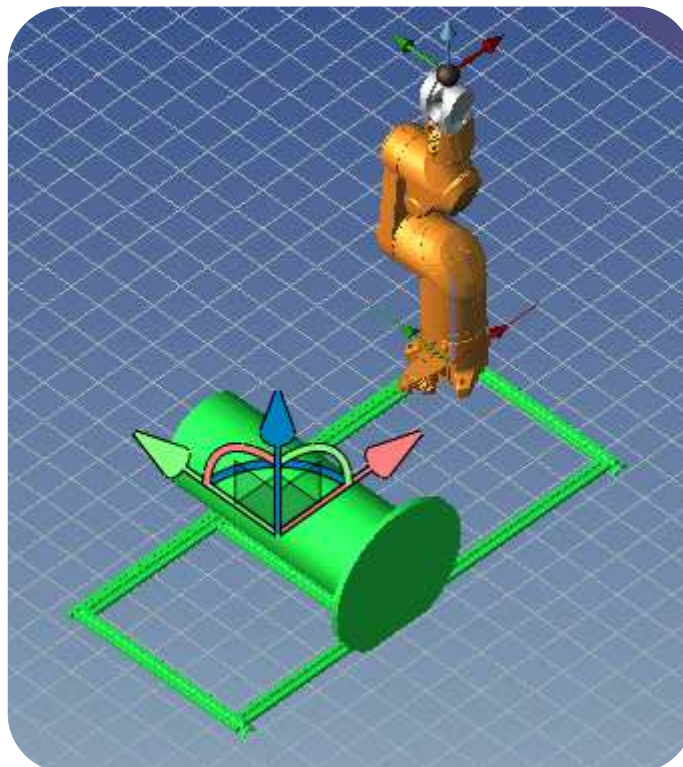


Figura 53: vista del model 3D de la taula de treball inserida

El sistema de coordenades adjunt al model permet moure'l dins l'entorn 3D. Amb l'objectiu d'editar una posició escaient per a la taula de treball, cliquem amb el botó dret sobre l'origen del sistema de coordenades de la taula i seleccionem "Editar posición" ().

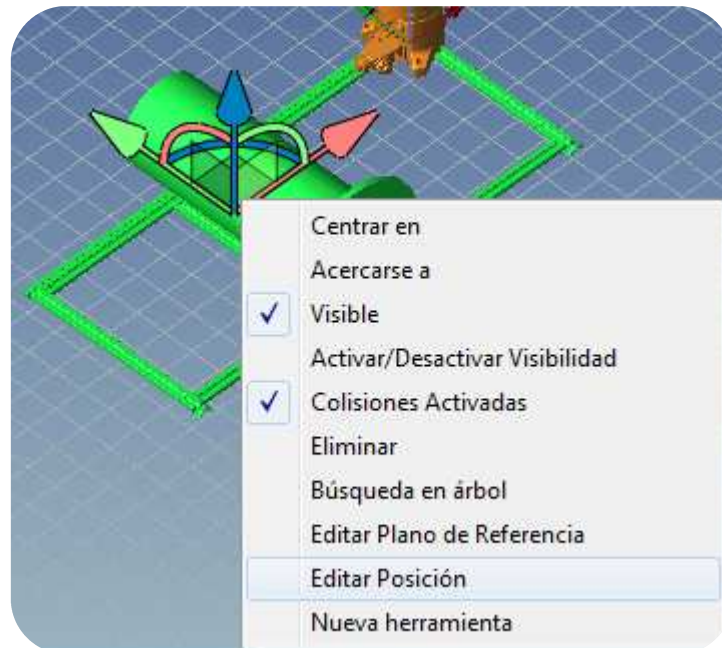


Figura 54: ruta per editar la posició de la taula de treball

Tot seguit apareixerà una nova pestanya que permet editar la posició de l'objecte 3D. Editem les seves coordenades tal com es mostra a la Figura 55. En cas que no es permeti editar la posició, clicar sobre "editar posició" a la part superior de la nova pestanya.

Posición Absoluta ⌵

X: 0,00 ⌵

Y: -470,00 ⌵

Z: 668,00 ⌵

Rx: -90,00 ⌵

Ry: 0,00 ⌵


Rz: -180,00 ⌵

Figura 55: posició absoluta de la taula de treball


A.10.4 Posicionament del robot


La posició del robot s'ha d'editar de tal manera que s'ajusti al màxim possible amb la realitat. Seguint un procediment similar a l'anterior, cliquem sobre el robot i, fent clic amb el botó de la dreta sobre el sistema de referència que apareix a la seva base, seleccionem editar posició.

Tot seguit editem la seva posició amb les coordenades que s'especifiquen a la Figura 56.


Posición Absoluta 

X:  

Y:  

Z:  

Rx:  

Ry:  

Rz:  

Figura 56: posició absoluta del robot

A la Figura 57 podem veure la vista 3D obtinguda.

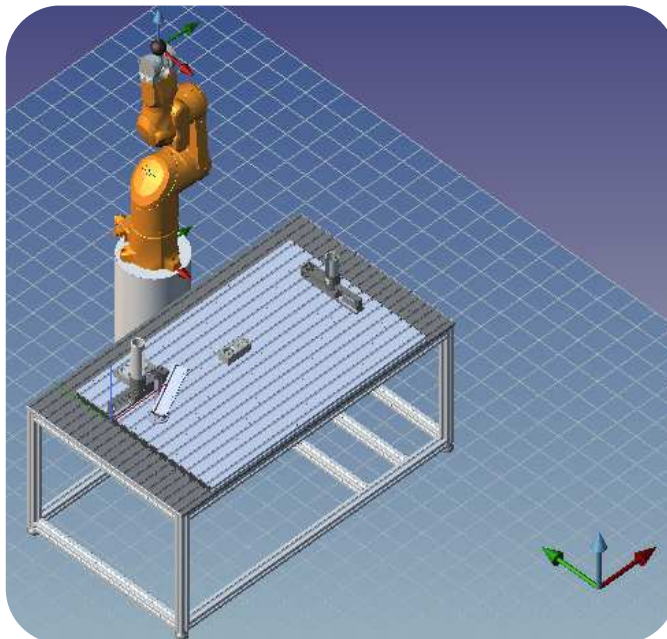


Figura 57: vista 3D isomètrica la posició absoluta de la taula i el robot editades

A.10.5 Inserció de l'eina

Una vegada s'ha col·locat el robot, ha arribat el moment de realitzar la inserció de l'eina. Per fer-ho repetim el procés utilitzat anteriorment per la taula de treball, és a dir, afegint l'arxiu CAD com un enllaç (veure Figura 58).

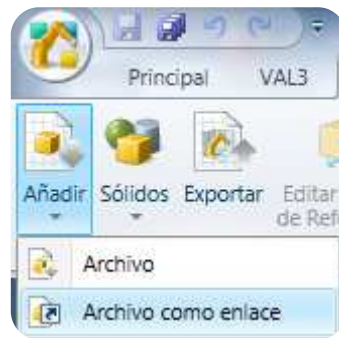


Figura 58: ruta per inserta la pinça com un enllaç

Tot seguit seleccionem la geometria de l'eina generada amb SolidWorks (Figura 59) i la importem seleccionant Open.

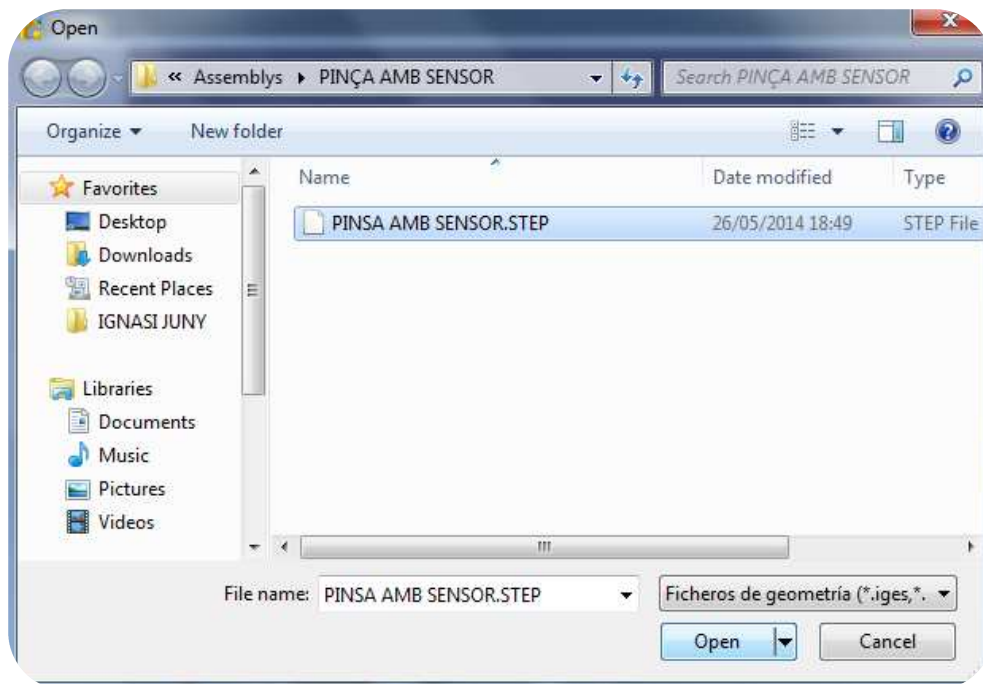


Figura 59: fitxer en format .step a seleccionar per afegir la pinça com un enllaç

És possible que es trigui uns segons o fins i tot minuts a processar la geometria importada. Una vegada enllaçada la pinça amb el model 3D, s'haurà d'editar la seva posició.

A.10.6 Edició geomètrica de l'eina

Quan s'introdueix l'eina, aquesta queda col·locada a l'origen de coordenades de l'entorn 3D. Tal com es pot veure a la Figura 60, la seva posició no és la desitjada. Abans d'editar-la però, caldrà definir els punts per on l'eina pot ser agafada i per on pot agafar objectes.

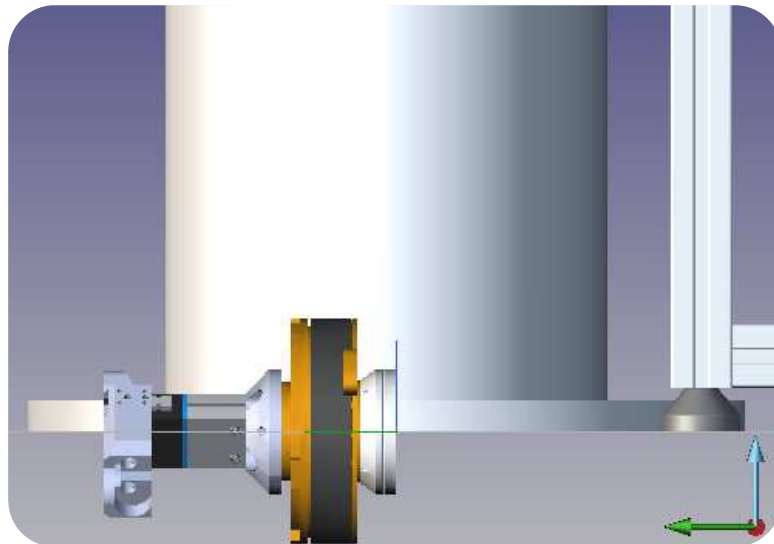


Figura 60: posició de l'eina després de ser inserida en el model 3D

Tot seguit s'ha d'editar l'eina. Aquesta té una part per on és agafada per l'extrem del robot i dues altres per on pot agafar elements (la pinça).

Per poder editar aquests punts, prèviament s'ha de designar la nova geometria 3D com a un element del tipus "tool" que quedarà definit com un mecanisme. Per fer-ho cal clicar la geometria de la pinça per tal de seleccionar-la (Figura 61).

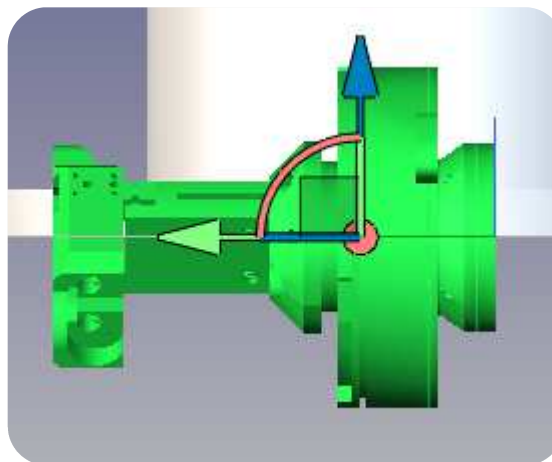


Figura 61: selecció de la geometria pinça per poder-la definir com un element tipus "tool"

Tot seguit, amb la geometria seleccionada cliquem “Nuevo” dins la barra d’opcions “Modelado” dins el subgrup “Herramientas” (veure Figura 62).



Figura 62: ruta per definir la geometria com a eina

Una vegada clicat “Nuevo”, veurem que dins de la “vista de arbol” de l’escenari 3D, apareix un nou mecanisme anomenat Tool1 (veure Figura 63). Aquest mecanisme és el corresponent a la nova eina. Per altra banda, el mecanisme anomenat “Controlador: tx60” és el robot.

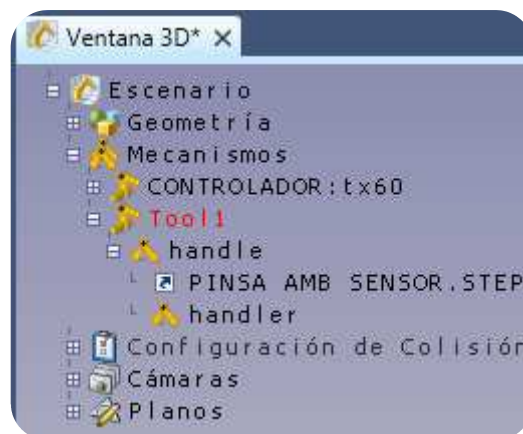


Figura 63: vista d’arbre de l’escenari 3D amb el nou mecanisme Tool1

Tal com es pot veure a la Figura 63, l’eina conté un punt anomenat “Handle” i un altre anomenat “Handler”. El Handle és el punt per on l’eina pot ser agafada. El Handler és el punt per on l’eina pot agafar objectes. Un mateix mecanisme pot tenir més d’un handler. Una vegada definits aquests dos conceptes, cal dir que el robot també té el seus propis handle i handler. El Handle del robot està situat a la seva base, mentre que el Handler està situat a la seva punta. Per tal que l’eina quedi adherida al braç, caldrà adjuntar el handle de l’eina amb el handler del robot.

L’eina del laboratori té un sol punt per on és agafada (Handle) i dos punts per on pot agafar objectes (Handlers). Aquests s’han de definir per tal de poder-los fer servir. Tot seguit es descriu el procediment a seguir per editar les seves posicions i afegir un segon Handler.

Per poder modificar les posicions dels handle i handler, primer de tot s'ha d'editar l'eina. Per fer-ho, amb l'eina seleccionada cliquem la pestanya "Editar" de dins la barra d'opcions "Modelado" dins el subgrup "Herramientas" (veure Figura 64).



Figura 64: per editar l'eina cliquem Editar dins el subgrup "herramientas" de la barra d'opcions

Una vegada clicat editar, l'eina quedarà de color groc i li apareixeran dos nous sistemes de coordenades. Aquests són els corresponents al Handle i el Handler. Com que a la geometria 3D no es diferencien, per seleccionar-ne un d'ells anem a la vista d'arbre i editem la posició del handle (veure Figura 65).



Figura 65: selecció del handle de l'eina a través de la vista d'arbre

Tot seguit es desplegarà una pestanya que permet editar la posició absoluta i relativa del sistema de coordenades seleccionat. Com que la geometria de l'eina és coneguda, és fàcil poder determinar com s'ha de definir cada sistema de coordenades. Després d'analitzar com està col·locada la pinça en el robot del laboratori, s'ha n trobat les posicions i orientacions de cada sistema de coordenades, tant el del handle com els dels handlers. Pel sistema en qüestió, entrem els valors que es poden veure a la Figura 66.

Posición Absoluta ▼

X: 0,00 ▲▼

Y: 0,00 ▲▼

Z: 0,00 ▲▼

Rx: -90,00 ▲▼

Ry: 0,00 ▲▼

Rz: 0,00 ▲▼

Figura 66: valors de la posició absoluta del handle de l'eina

Seguim un procediment similar per editar la posició del handler. L'editem a través de la vista d'arbre (veure Figura 67). Finalment introduïm els valors que es poden veure a la .

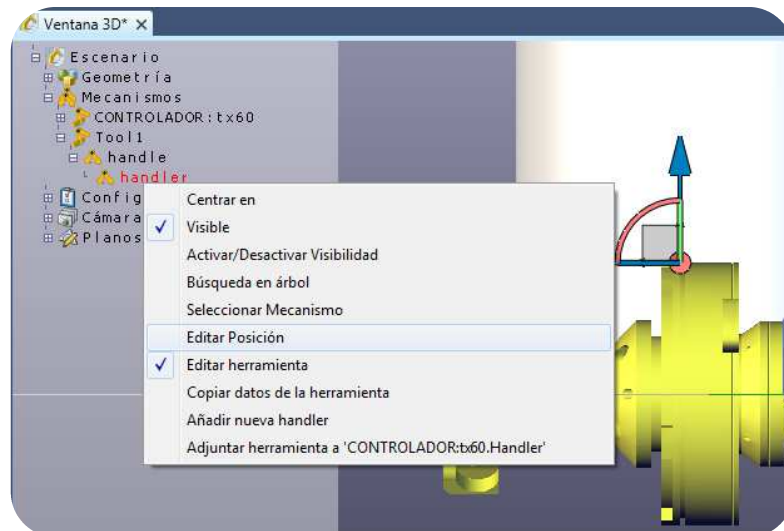


Figura 67: edició de la posició del handler de l'eina

Posición Absoluta ▼

X: 28,50 ▲▼

Y: 180,00 ▲▼

Z: 28,50 ▲▼

Rx: -90,00 ▲▼

Ry: 0,00 ▲▼

Rz: 0,00 ▲▼

Figura 68: valors de la posició absoluta del primer handler de l'eina

Una vegada col·locats els dos sistemes de coordenades que es generen per defecte quan es defineix l'eina com una geometria, com que l'eina té dos elements terminals, s'ha d'introduir un nou handler. Per fer-ho, cliquem "Añadir Nueva Handler" dins la barra d'opcions (veure Figura 69).



Figura 69: pestanya per afegir un nou handler a l'eina

Tot seguit editem la posició del nou handler afegit repetint el procés anterior. Editem la seva posició (veure Figura 70) i introduïm els valors que es poden veure a la Figura 71.

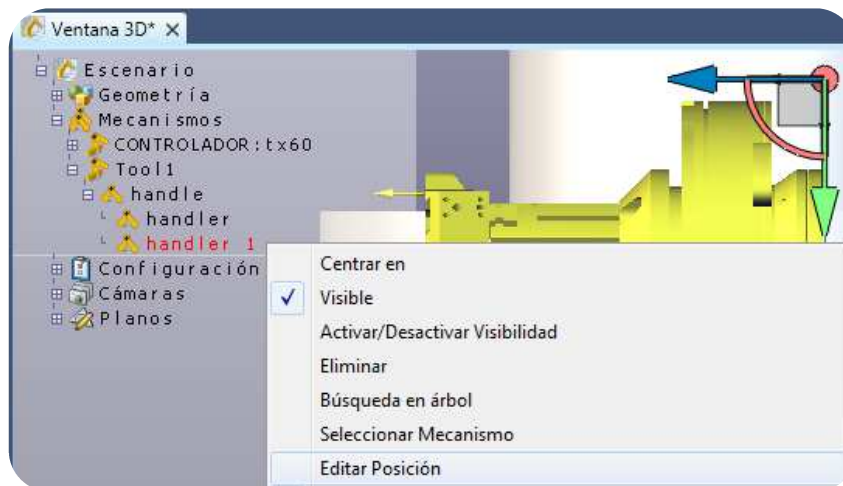


Figura 70: edició de la posició del nou handler

Posición Absoluta ▼

X:	-30,00	▲▼
Y:	167,00	▲▼
Z:	-29,00	▲▼
Rx:	-180,00	▲▼
Ry:	-45,00	▲▼
Rz:	-180,00	▲▼

Figura 71: posició i orientació absolutes del nou handler introduït

Finalment des-cliquem la pestanya "Editar" (veure Figura 64). D'aquesta manera l'eina queda completament definida.

A.10.7 Adjuntar l'eina al robot

Per tal que l'eina es mogui solidària al moviment del robot, se l'ha d'adjuntar al handler del mateix. Per fer-ho, amb l'eina seleccionada, despleguem la pestanya "Adjuntar a". Aquesta pestanya mostra tots els handlers disponibles als quals es podria adjuntar l'eina. En aquest cas, tant sols apareix el handler del robot, ja que és l'únic que està definit. Tal com es pot veure a la , seleccionem el handler del robot.

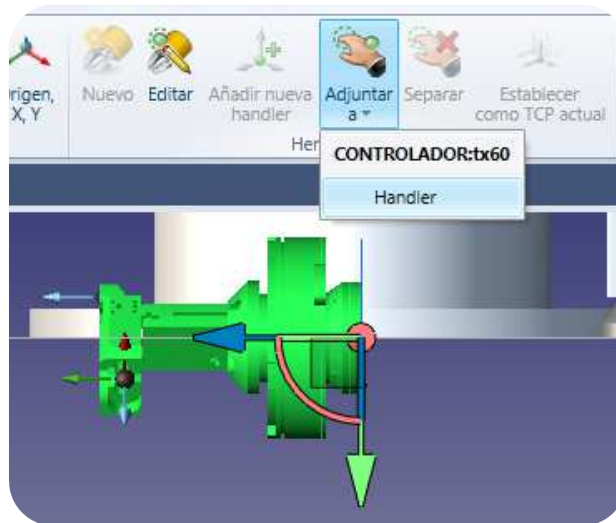


Figura 72: ruta per adjuntar l'eina al robot

Finalment, l'eina queda adjuntada al manipulador del robot (Figura 73).

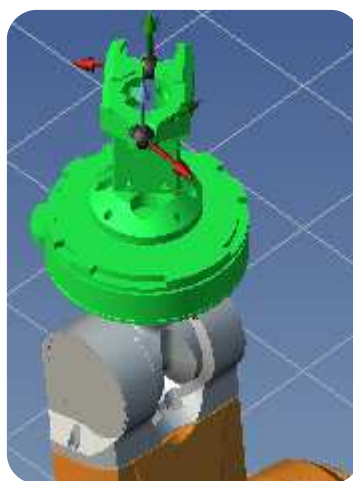


Figura 73: eina adjuntada al robot

A.10.8 Edició de punts i de l'eina

Anteriorment, dins la pestanya datos, s'han declarat dos punts i una eina. Tot seguit anem a editar les seves posicions i característiques. Començarem pel punt anomenat "posicio0" del tipus jointRx, seguirem pel punt anomenat "posicio2" del tipus pointRx i, finalment acabarem amb l'eina anomenada "eina0" que és del tipus tool.

Cada un d'aquests elements s'hauria de definir de manera específica ja que cada un d'ells té característiques diferents. Per exemple, els punts de tipus joint es defineixen a través de les posicions angulars de cada un dels graus de llibertat del robot. Per altra banda, els punts de tipus point es defineixen a través de les posicions cartesianes i la posició angular del sistema de coordenades corresponent al punt. Finalment, l'eina també s'hauria de definir de manera similar als elements de tipus point. El software però, permet editar geomètricament cada un d'aquests punts i, posteriorment, assignar la geometria als elements ja definits, siguin del tipus que siguin.

Dins la barra d'opcions, en el submenú "Planos de construcción" (veure Figura 74), hi podem trobar les pestanyes necessàries per poder editar punts dins l'escenari 3D. Aquests, podran assignar-se a les dades ja definides prèviament a l'aplicació.



Figura 74: pestanyes per editar punts dins la geometria 3D

Tot seguit es realitzarà aquest procés per cada un dels tres elements necessaris per executar el programa de mostra:

1. Punt "posició0": aquest punt és el de repòs del robot. És del tipus joint i li corresponent totes les posicions angulars iguals a zero. Tal com es pot veure a la geometria 3D, el robot ja es troba a aquesta posició. A més a més, els valors per defecte del punt ja són amb tots els angles iguals a zero. Per tant, es recomana no realitzar cap canvi.
2. Punt "posició2": aquest element conté dos punts dintre seu. Quan l'hem definit s'ha fet de manera que fos unidimensional. Tal com està escrit el codi de mostra aquest

element ha de ser bidimensional. Per augmentar la seva dimensió anem a la pestanya “Datos” i el seleccionem.

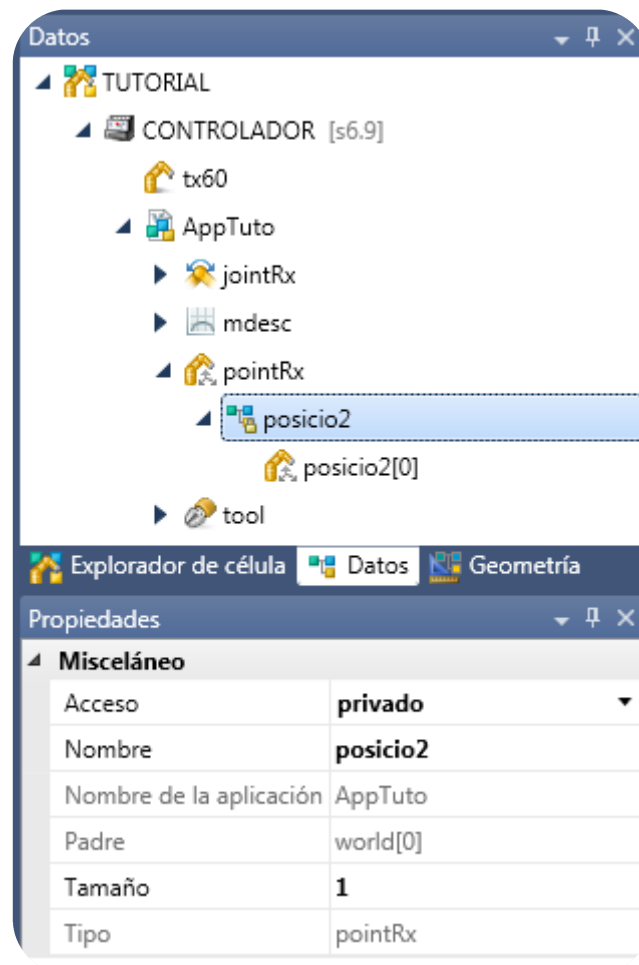


Figura 75: edició de la dimensió del punt tipus pointRx anomenat posicio2

Com es pot veure a la Figura 75, dins de la pestanya de propietats veiem que el “Tamaño” del punt és de 1. Procedim a modificar aquest valor indicant-li que volem que sigui de dos. A la Figura 76 es pot apreciar com el punt anomenat “posicio2” ha deixat de ser unidimensional i ara conté dos elements. Aquests poden editar-se i canviar els seus valors. Si fem doble clic a sobre el punt, s’obrirà una nova pestanya (Figura 77) a la qual podem editar els valors de cada un dels punts. Com que els definirem geomètricament, de moment es deixaran amb els seus valors per defecte.

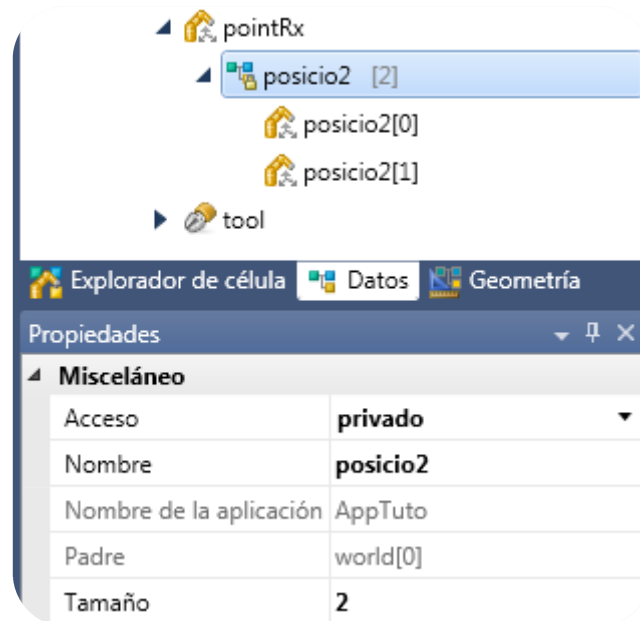


Figura 76: visualització del frame bidimensional el qual conté 2 elements tipus pointRx

The screenshot shows a 3D window titled 'Ventana 3D' with a data table for 'Datos AppTuto-posicio2*'. The table has 11 columns: 'Índice', 'x', 'y', 'z', 'rx', 'ry', 'rz', 'shoulder', 'elbow', and 'wrist'. There are two rows of data, both with '0' in the 'Índice' column.

Índice	x	y	z	rx	ry	rz	shoulder	elbow	wrist
0	0	0	0	0	0	0	ssame	esame	wsame
1	0	0	0	0	0	0	ssame	esame	wsame

Figura 77: llista de variables que conformen un punt tipus pointRx

Tal com ja s'ha esmentat, dins la barra d'opcions "Modelado" (Figura 74) hi trobem les pestanyes per definir punts dins l'entorn 3D. Anem a definir-ne un a través de la pestanya "Centro de 3 puntos". Una vegada clicada aquesta opció, seleccionem 3 punts adients per tal de definir un nou punt dins l'entorn 3D (veure Figura 78).



Figura 78: punts seleccionats per generar un nou punt a la geometria 3D

Una vegada seleccionats els tres punts, contingut al pla i en el centre de la circumferència definits pels punts seleccionats, apareixerà el nou sistema de coordenades que defineix el nou element geomètric (veure Figura 79). Per acabar d'editar la seva posició el cliquem amb el botó de la dreta i seleccionem "Editar posición".

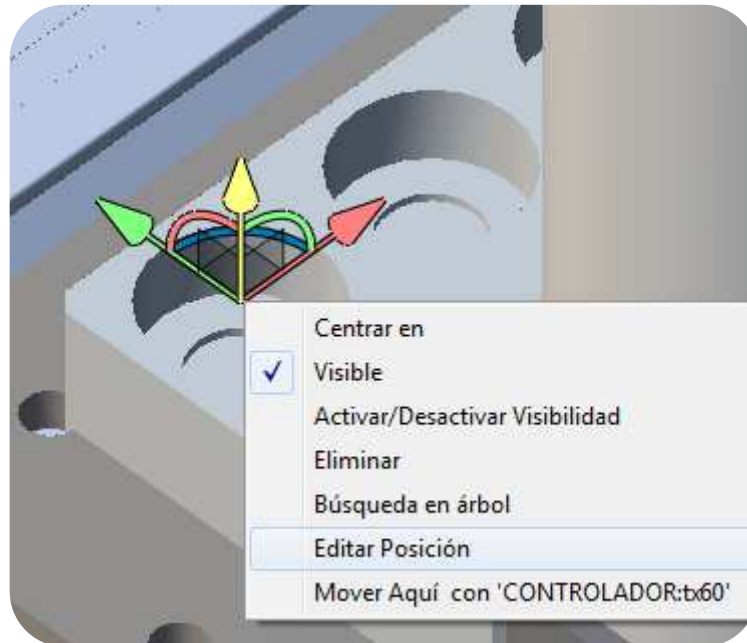


Figura 79: pestanya per editar el sistema de coordenades del nou punt

Tot seguit acabem de definir la posició del punt editant els seus valors absoluts. Introduïm els valors que es poden veure a la Figura 80.

Posición Absoluta ⌵

X:	-266,50	⬆	⬇	⬆
Y:	-329,50	⬆	⬇	⬆
Z:	705,00	⬆	⬇	⬆
Rx:	-180,00	⬆	⬇	⬆
Ry:	0,00	⬆	⬇	⬆
Rz:	180,00	⬆	⬇	⬆

Figura 80: posició absoluta del punt posicio2[0]

Per definir un nou punt a la geometria 3D, seguim un procediment similar a l'anterior. Primer es crea el punt amb la pestanya "Centro de 3 puntos" (Figura 81) i després s'edita la seva posició (Figura 83) per adequar-la a la desitjada.

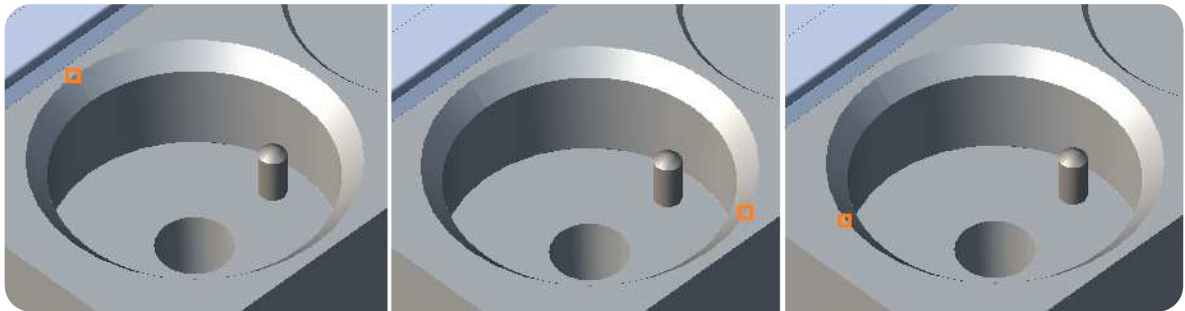


Figura 81: punts seleccionats per generar un nou punt a la geometria 3D

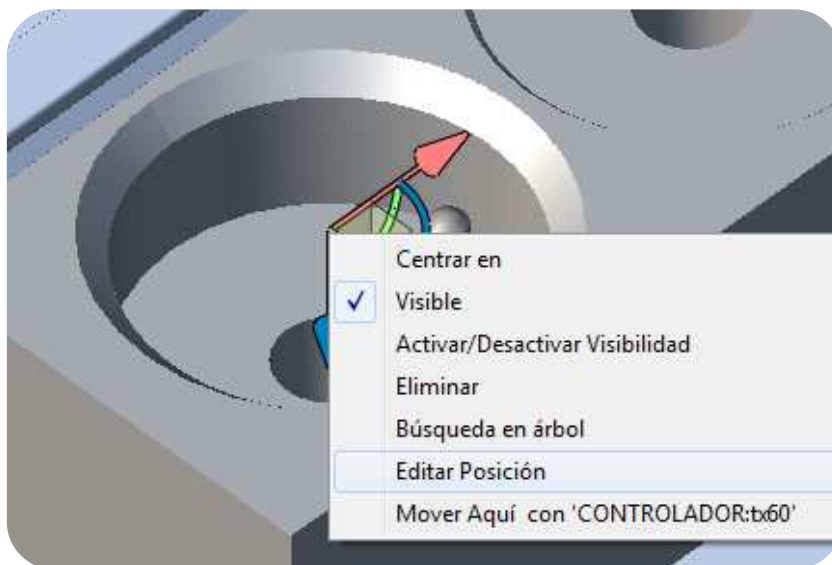


Figura 82: pestanya per editar el sistema de coordenades del nou punt



Figura 83: posició absoluta del punt posicio2[1]

Per tal que aquests punts definits geomètricament puguin ser utilitzats pel codi d'exemple, cal que les seves característiques geomètriques s'adjuntin als punts definits dins la pestanya "Datos". Per fer-ho, col·locarem el braç a la posició desitjada i posteriorment assignarem la posició als punts de la pestanya "Datos".

El procediment a realitzar és el següent (Figura 84):

1. Establir la punta de l'eina desitjada com a eina actual.
2. Moure el robot a la posició geomètrica definida a través del punt.
3. Dins la pestanya "Datos", assignar la posició del robot al punt desitjat.

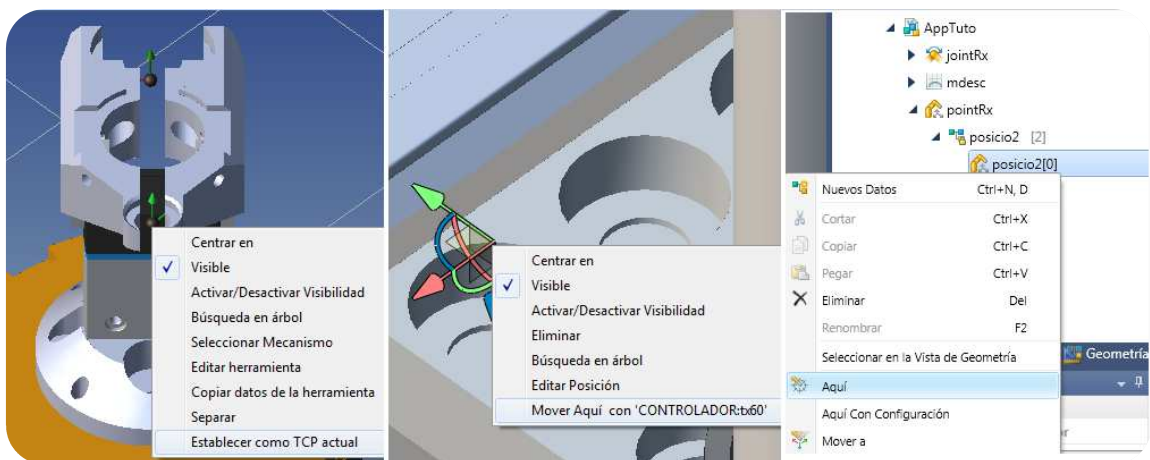


Figura 84: assignació del punt geomètric al punt posicio2[0]

Tot seguit es torna a ubicar el robot a la posició inicial (Figura 85) i es repeteix el mateix procés pel segon punt (Figura 86).

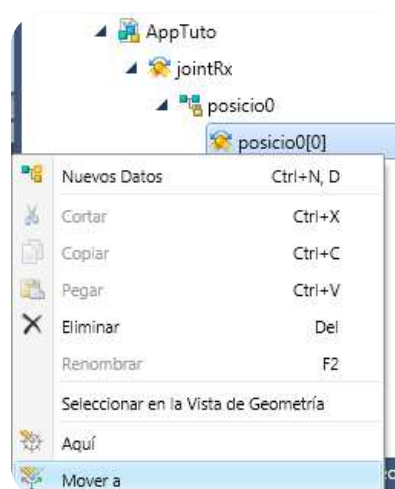


Figura 85: moviment del robot a la posició inicial posicio0[0]



Figura 86: assignació del punt geomètric al punt posicio2[1]

Si dins la pestanya “Datos” fem doble clic sobre un punt, veurem com aquests han quedat modificats (Figura 87).

Índice	x	y	z	rx	ry	rz	shoulder	elbow	wrist
0	329,500005	-489,5	-385,18	-180	0	90	ssame	esame	wsame
1	546,999991	-161,000011	-365,18	180	0	-90	ssame	esame	wsame

Figura 87: característiques de la dada posicio2

Finalment tornem a col·locar el robot a la posició inicial (Figura 85).

3. Eina “eina0”: per tal d’adjuntar les característiques geomètriques de l’eina dins la pestanya “Datos”, tant sols cal que es copiïn les dades (Figura 88) de cada “Handler” i s’enganxin a l’eina declarada a “Datos” (Figura 89).

Realitzarem aquest procés dues vegades, una per cada element terminal.

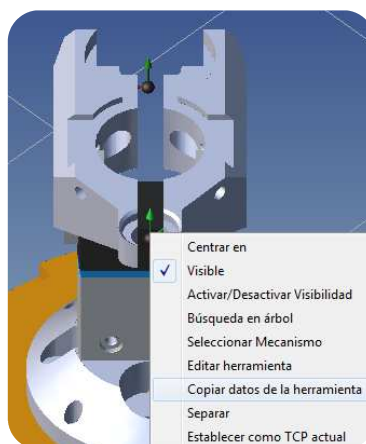


Figura 88: copia de les dades de l’eina

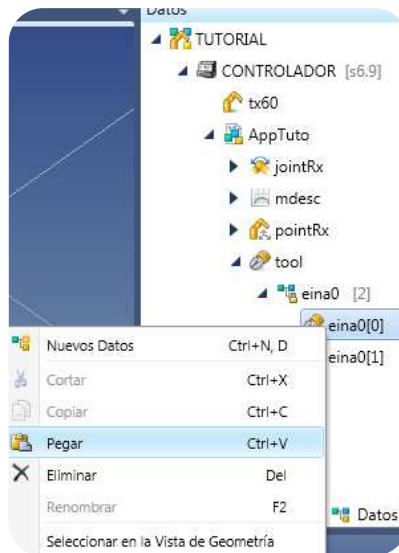


Figura 89: assignació de les dades geomètriques a l'eina eina0[0]

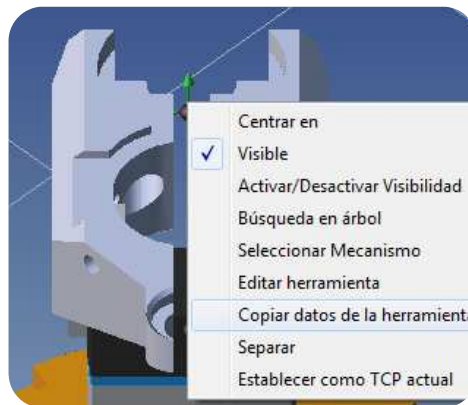


Figura 90: copia de les dades de l'eina

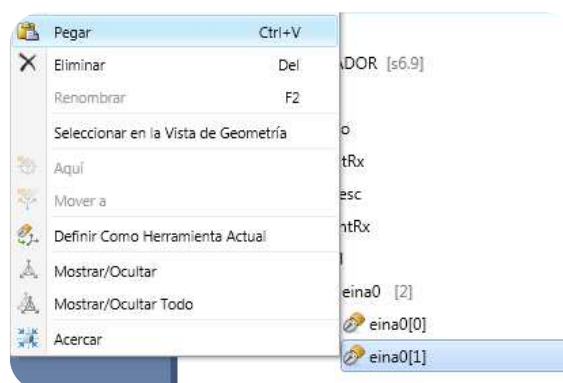


Figura 91: assignació de les dades geomètriques a l'eina eina0[1]

Finalment ja es disposa dels elements completament definits. El codi generat ja podrà executar-se sense que es generi cap senyal d'alarma.

A.11 Emulador

L'emulador és el software encarregat de simular el comportament de l'armari de control del robot, concretament la seva MCP¹. Per poder-lo executar cal clicar amb el botó de la dreta sobre el controlador i seleccionar "Mostrar emulador" (veure Figura 92).

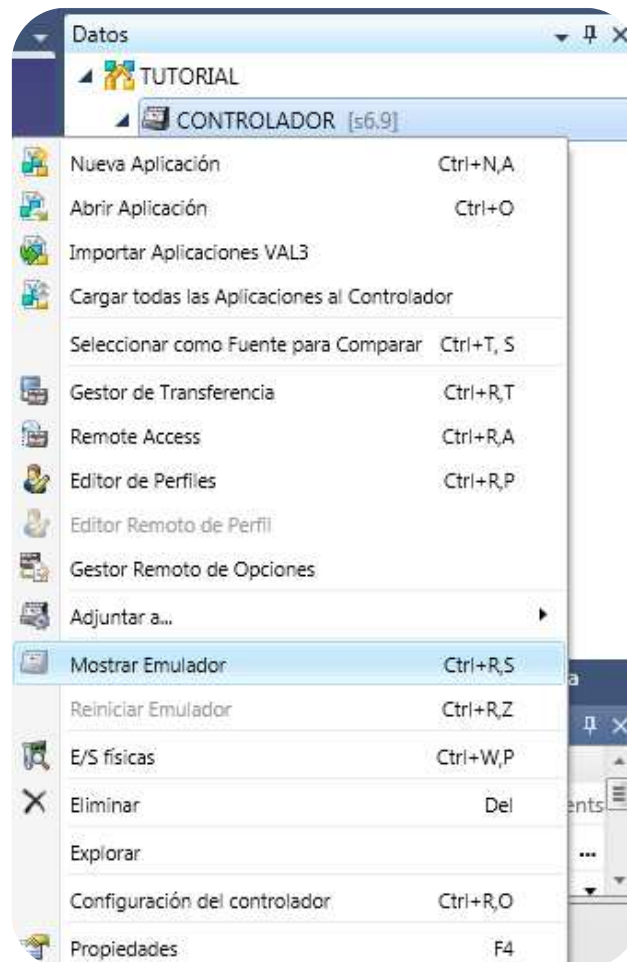


Figura 92: ruta per activar l'emulador de la MCP de l'armari de control

Una vegada hem clicat "Mostrar Emulador", apareixerà una finestra emergent (Figura 93). La qual és l'encarregada de simular el comportament de la MCP de l'armari de control.

¹ MCP: Manual Console Programming (Consola de Programació Manual).

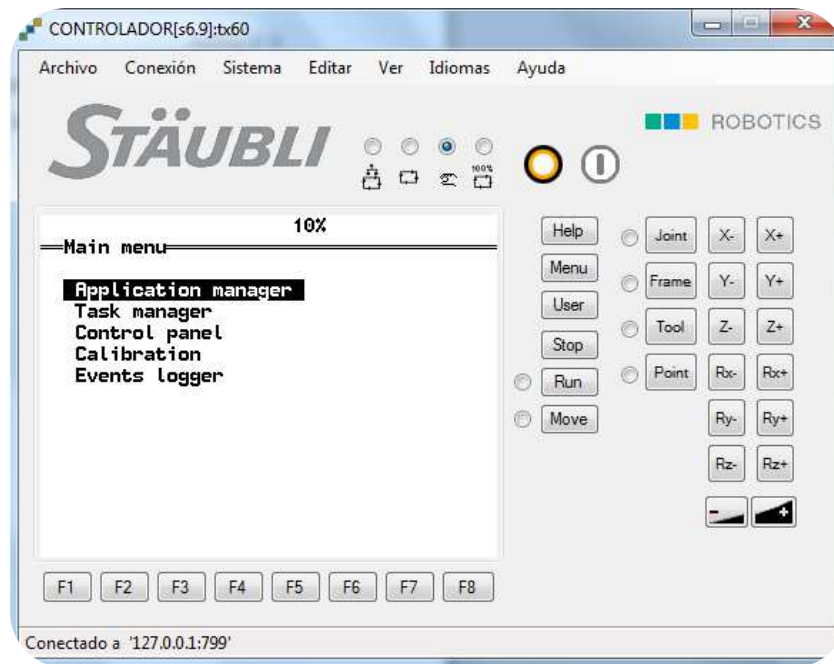


Figura 93: emulador de la MCP de l'armari de control CS8

Les fletxes del teclat ens permeten desplaçar-nos pels menús de la MCP emulada. La fletxa a la dreta ens permet seleccionar el menú en negreta. Per tornar al menú principal, amb el cursor cliquem la tecla menú.

Tot seguit anem a obrir l'aplicació AppTuto per poder simular la seva execució. Per fer-ho, entrem dins el menú "Application manager", "Val3 Applications", "Disk" (Figura 94).

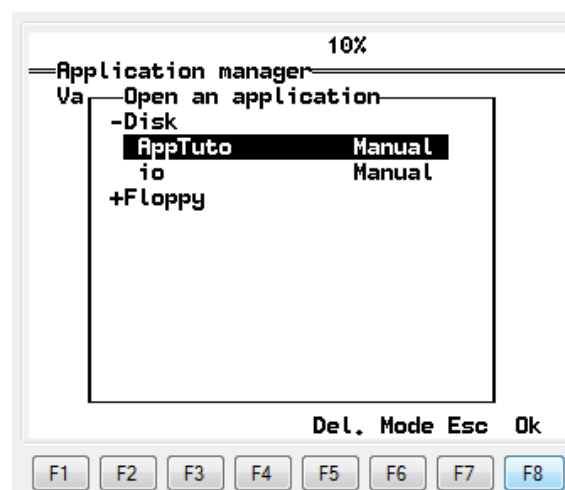


Figura 94: obertura de l'aplicació AppTuto a través de l'emulador

Després de clicar Ok, seleccionem "Menu". Finalment, ja estem en disposició de començar la simulació.

A.12 Simulació 3D

Per realitzar la simulació és necessari que l'emulador estigui activat i sincronitzat amb l'entorn 3D. Amb aquest objectiu anem a la barra d'opcions, simulació dins el subgrup sincronització (Figura 95).

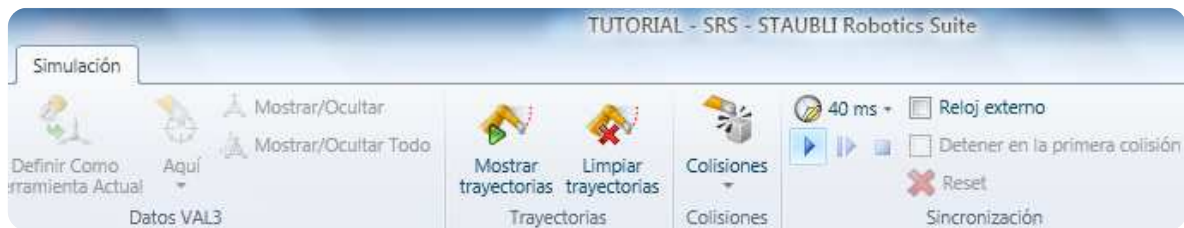


Figura 95: barra d'opcions amb la pestanya simulació seleccionada

Tal com es pot veure a la Figura 1, es recomana seleccionar un temps de sincronització de 40 ms. Aquest és prou ràpid per visualitzar la simulació sense col·lapsar l'ordinador. Per tal que es sincronitzi l'execució d'un programa amb l'emulador i l'entorn 3D, és necessari que el símbol de play estigui activat.

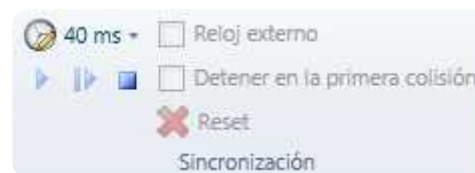


Figura 96: submenú sincronización de la pestanya de simulació

Tal com es pot veure a la Figura 96, el símbol de play és de color blau clar quan està activat. Ara ja estem en disposició de simular el comportament del robot amb el codi generat. Amb l'objectiu que es pugui visualitzar la trajectòria que realitza el robot, seleccionarem la pestanya "mostrar trayectorias" (Figura 95).

Tot seguit anem a l'emulador CS8 i cliquem la tecla Run. Posteriorment s'obrirà una pestanya dins la pantalla de la MCP emulada (Figura 97) la qual ens indica les aplicacions que hi ha carregades a l'emulador. Com que hem carregat l'aplicació AppTuto, aquesta apareix a la llista. La seleccionem a través de les fletxes del teclat i cliquem Ok.

Una vegada realitzat aquest procés, anem a donar potència al braç per al que es pugui començar el moviment. Per fer-ho cliquem el símbol que es pot veure a la Figura 98.

Veurem com aquest adquireix un color verd. Això ens indica que el braç ja disposa de potència. Recordem que el mode de treball ha de ser el que es pot veure a la Figura 98.

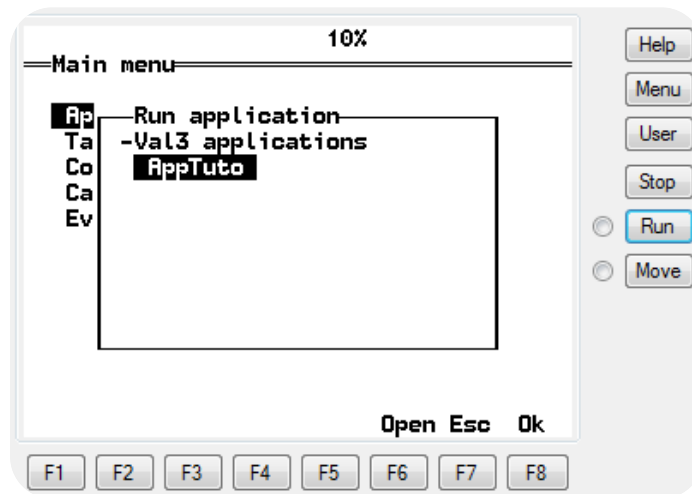


Figura 97: selecció de l'aplicació a simular a través de la tecla Run



Figura 98: tecla per donar potència al braç i mode de treball

Finalment, ja estem en disposició de començar el moviment. Per fer-ho, cliquem la tecla "Move" de l'emulador. El resultat al final de la simulació és el següent:

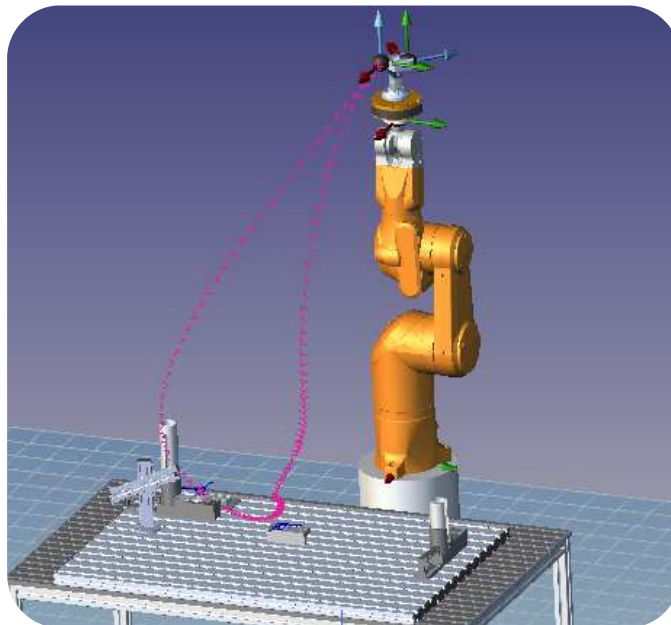


Figura 99: trajectòria realitzada per la simulació.

A.13 Detecció de col·lisions

Durant el moviment del robot, pot haver-hi col·lisions entre les geometries. Dins la simulació no es genera la ruptura de cap de les parts, però a la realitat, una col·lisió pot comportar la ruptura total del robot. Si la col·lisió és petita, és molt probable que el robot es descalibri.

Per aquest motiu, el software incorpora la possibilitat de detectar col·lisions durant la simulació. Per activar aquesta opció, cal clicar la pestanya anomenada "Colisiones" (Figura 100).

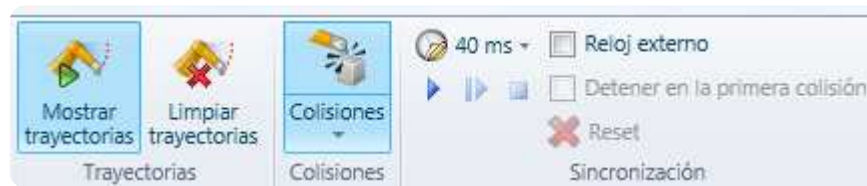


Figura 100: pestanya a activar per realitzar la detecció de col·lisions

Si tornem a realitzar la simulació repetint el procés seguit anteriorment, durant la simulació es visualitzen les zones on hi ha les col·lisions.

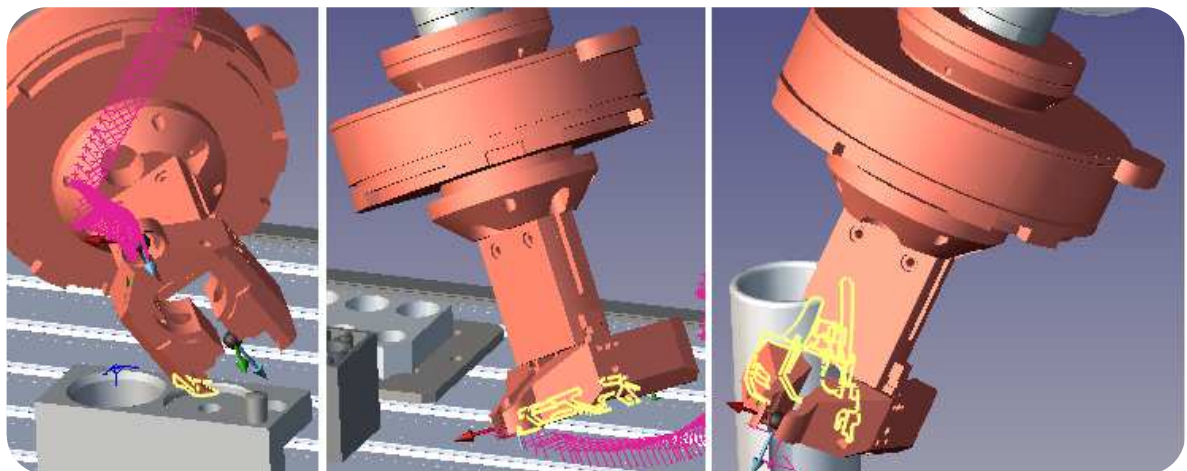


Figura 101: visualització de les col·lisions durant la simulació

Per evitar les col·lisions, tant sols cal editar més punts i redefinir la trajectòria per l'eina. També Seria possible realitzar aproximacions als punts ja definits a través del codi.

A.14 Transfer Manager

El transfer manager és una eina que incorpora el software SRS 2013. Aquesta permet que les aplicacions generades i simulades dins l'entorn virtual es traslladin a l'armari de control CS8 que controla el robot real.

La pestanya corresponent a aquesta eina es troba a la barra d'opcions, en el grup principal i dins el subgrup controlador (veure Figura 102).



Figura 102: pestanya per accedir al gestor de transferència

Aquesta eina requereix d'una connexió de xarxa a la qual hi hagi connectat el controlador real. En el cas del laboratori, ja hi ha instal·lada una xarxa local on hi ha tots els ordinadors connectats amb l'armari de control CS8 del robot.

Per procedir a transferir una aplicació a l'armari de control real, cliquem la pestanya "Gestor de transferència" (Figura 102). Una vegada clicada la pestanya, apareixerà una finestra emergent (veure Figura 103) la qual ens permet seleccionar el directori que volem obrir. En el nostre cas, seleccionem el directori del nostre controlador anomenat "CONTROLADOR" que és a on hi tenim guardada l'aplicació anomenada AppTuto.

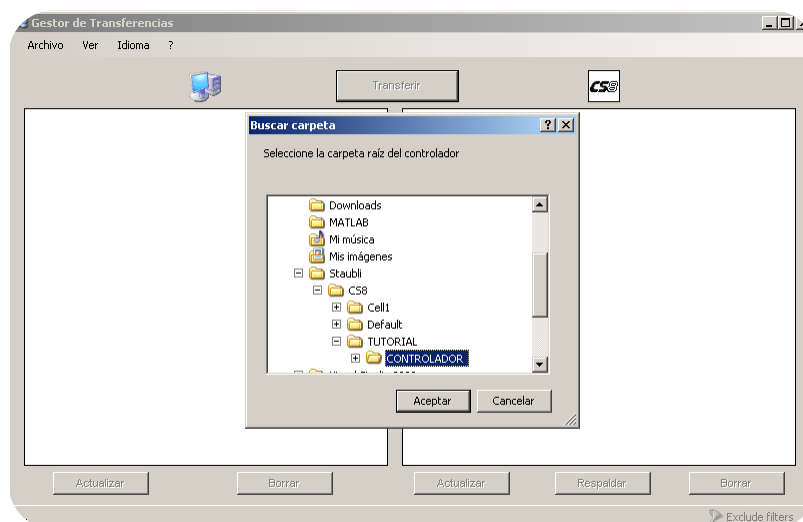


Figura 103: finestra per seleccionar el directori de l'ordinador que volem obrir

Finalment cliquem acceptar. Podem observar que a la part de l'esquerra apareix un menú desplegable que es correspon amb les carpetes que hi ha dins el directori del controlador.

Tot seguit apareixerà una nova finestra emergent anomenada "Parámetros de conexión". Aquesta permet editar la IP de l'armari de control, el port i el perfil d'usuari. Normalment deixarem tots els valors que ja hi ha per defecte, ja que són els que la xarxa detecta de manera automàtica.

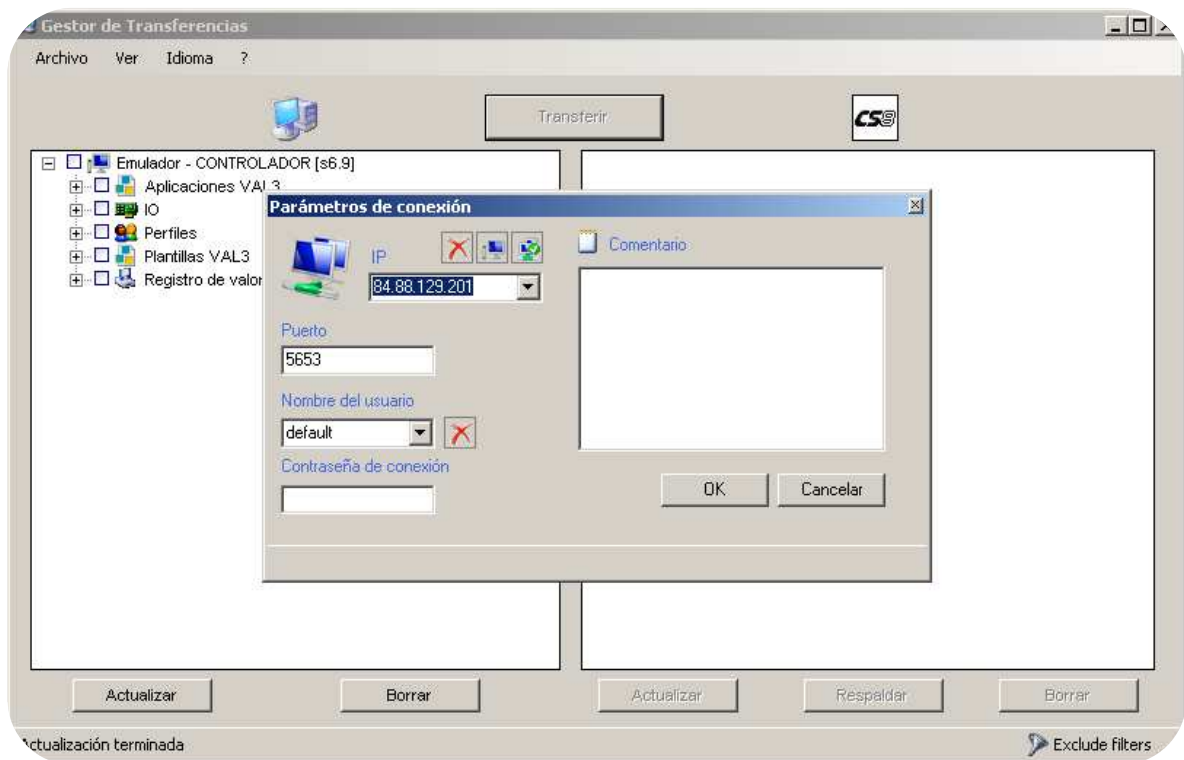


Figura 104 finestra per editar la IP del controlador real

En cas que sorgís alguna complicació amb la IP, se l'ha d'editar perquè coincideixi amb la IP de l'armari de control. Aquesta es pot trobar amb la MCP seguint la ruta: Menú, Panel de control, Configuración del controlador, Red, IP. Escrivim en el transfer manager alguna de les IP que es mostren a la pantalla de la MCP i cliquem OK.

Veurem que ens apareix un missatge d'error (Figura 105). Aquest es degut a que la versió CS8 de la tarja de control de l'armari del robot és molt antiga. Per aquest motiu s'ha optat per treballar amb la versió 6.9 ja que, tot i no ser la mateixa que la de l'armari de control, amb la majoria de les funcionalitats sí que és compatible. Cliquem OK i seguim com si aquest error no hagués aparegut mai.

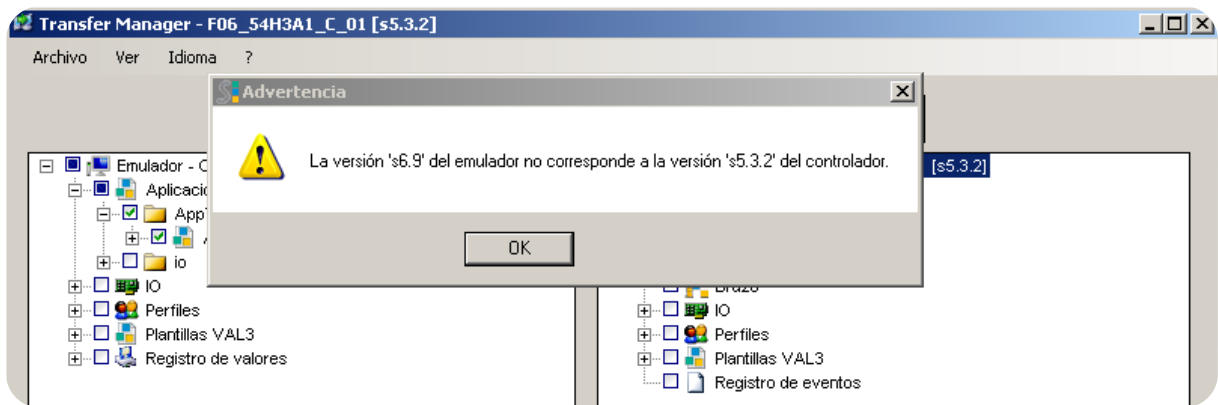


Figura 105: missatge d'error degut a la divergència de versions entre l'emulador i l'armari CS8 real

Una vegada connectat l'ordinador amb l'armari de control, ja estem en disposició de transferir l'aplicació anomenada "AppTuto" cap a l'armari de control. Per fer-ho, seleccionem l'aplicació i cliquem "Transferir" (veure Figura 106) :

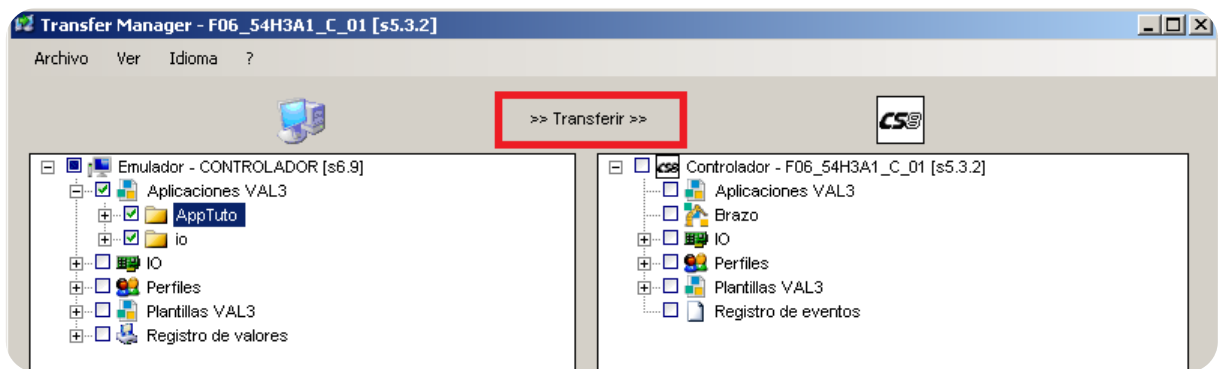


Figura 106: selecció i transferència de l'aplicació VAL3 anomenada "AppTuto"

Una vegada transferida l'aplicació tal com es pot veure a la Figura 107, a la finestra corresponent a l'armari de control CS8 apareix l'aplicació "AppTuto" amb totes les seves subcarpetes. Aquest fet ens indica que la transferència s'ha realitzat correctament.

Si anem a la MCP de l'armari de control real, veurem que la nova aplicació apareix a la pantalla quan entrem al gestor d'aplicacions. Tot seguit ja podem tancar el gestor de transferència. En cas que es realitzi alguna modificació al codi o a les dades, l'aplicació s'ha de tornar a transferir a l'armari de control seguin el mateix procediment anterior. En cas que doni un error al transferir, li diem que volem sobre escriure l'aplicació a l'armari de control ja que l'anterior ja no ens serveix.

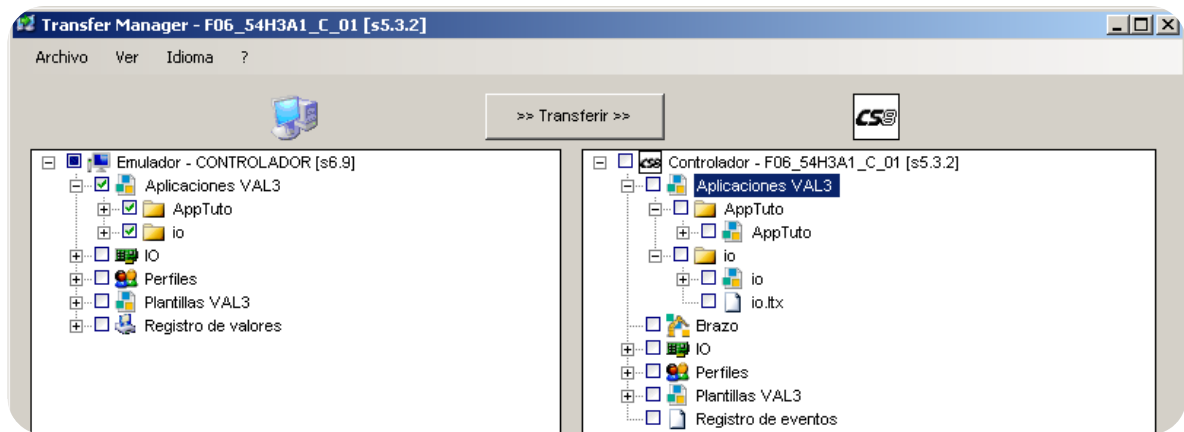


Figura 107: l'aplicació "AppTuto" ja està transferida a l'armari de control CS8

Finalment, si ho desitgem també podem transferir una aplicació des de l'armari de control CS8 fins a l'ordinador. Per fer-ho, fem un procediment similar a l'anterior però, aquesta vegada, seleccionant l'aplicació de l'armari de control.

AUTOR: Ignasi Juny Canals

Girona 19/06/2014