

Educational Package

Technical Documentation: ER-4iA Robot with R-30iB Mate Plus Controller

V5.3 (S/W Version 9.3)



Education Cell

One common servo and control platform -
infinite opportunities
THAT'S FANUC!



FA
CNCs, Servo Motors
and Lasers



ROBOTS
Industrial Robots,
Accessories and
Software



ROBOCUT
CNC Wire-Cut Electric
Discharge Machines



ROBODRILL
Compact CNC Machining
Centres



ROBOSHOT
Electric CNC Injection
Moulding Machines

1 Contents

1	Installation / Quick Start Guide	5
2	Safety.....	6
	2.1 General Instructions	6
	2.2 AUTO/T1/T2 Switch.....	7
	2.3 DCS Password	7
3	Selecting and running Example program	8
	3.1 Start Up Screen	8
	3.2 Example Program “AAA_DEMO”	9
	3.3 AAA_DEMO Program Sequence with / without <i>iR</i> Vision	13
	3.4 Operation Menu Display	14
	3.5 4D Graphics (Option)	15
4	Demonstration Program Setup / Teaching	16
	4.1 Gripper Setup.....	16
	4.2 UFrame 2 Setup.....	17
	4.3 Orient Origin Point.....	19
	4.4 X-Direction Point	20
	4.5 Y-Direction Point	21
5	Robot Setup	22
	5.1 UTOOL / TCP Setup	22
	5.2 Payload Setup	22
	5.3 DCS Setup	23
	5.3.1 DCS Tool Model	23
	5.3.2 DCS Safe Zone	24
	5.4 I/O Setup	25
	5.5 Macro Setup.....	26
	5.6 TP Function Key Hint Screen	27
	5.7 Home / Reference Position	28
	5.8 Power Up Program.....	29
6	<i>iR</i> Vision Setup (Option)	30
	6.1 Camera Adjustment	30
	6.2 Application Frame	32
	6.3 Camera Calibration	33
	6.4 Example <i>iR</i> Vision Application	35
	6.5 <i>iR</i> Vision Tips	36
	6.6 TP Program.....	38
7	IoT / Industrie 4.0 / PC Connection.....	39
	7.1 Physical Ethernet connection	39
	7.2 Robot Software Setup	40
	7.3 PC Setup	41
	7.4 Robot Homepage	42
	7.5 File Transfer	44
	7.6 Roboguide Simulator / File Transfer.....	45

8	List of Registers	48
	8.1 Numeric registers	48
	8.2 Position Registers	50
9	Program Details	51
	9.1 List of programs	51
	9.2 Sample Program Listings	54
10	User Interface Setup	56
	10.1 Menu Utility Setup	56
	10.2 HTML Screens	57
	10.3 HTML Listing Extract.....	58
11	Electrical Drawings	62
	11.1 Power and Fence	62
	11.2 Gripper	63
12	Bill of Materials	64
13	Technical & Transport Data	65
14	Version	66

Disclaimer

This documentation has been provided by FANUC Europe Corporation S.A. ("FANUC") for convenience and general information purposes only, with no intention of providing comprehensive, complete, accurate or up to date information. Therefore, FANUC does not warrant or assume any legal liability or responsibility whatsoever for the accuracy, completeness, or usefulness of any information, application, apparatus, product, use, processes or services disclosed herein.

Should you be interested in obtaining specific information or advice on FANUC applications, apparatus, products, use, processes or services, please contact FANUC at:

FANUC Europe Corporation

Société Anonyme

Zone Industrielle

L-6468 Echternach

Grand Duchy of Luxembourg

Tel.: +352-727777-1

INTENDED USE OF EDUCATION CELL

The Education Cell is intended for Education and Training Purposes

It is designed for lightness and portability,
not for high speed robot motion or full acceleration.

If robot is programmed to move aggressively
then this may result in undesirable shaking of the frame,
even resulting in the activation of the door safety switch.

Please avoid such unintended use.

1 Installation / Quick Start Guide

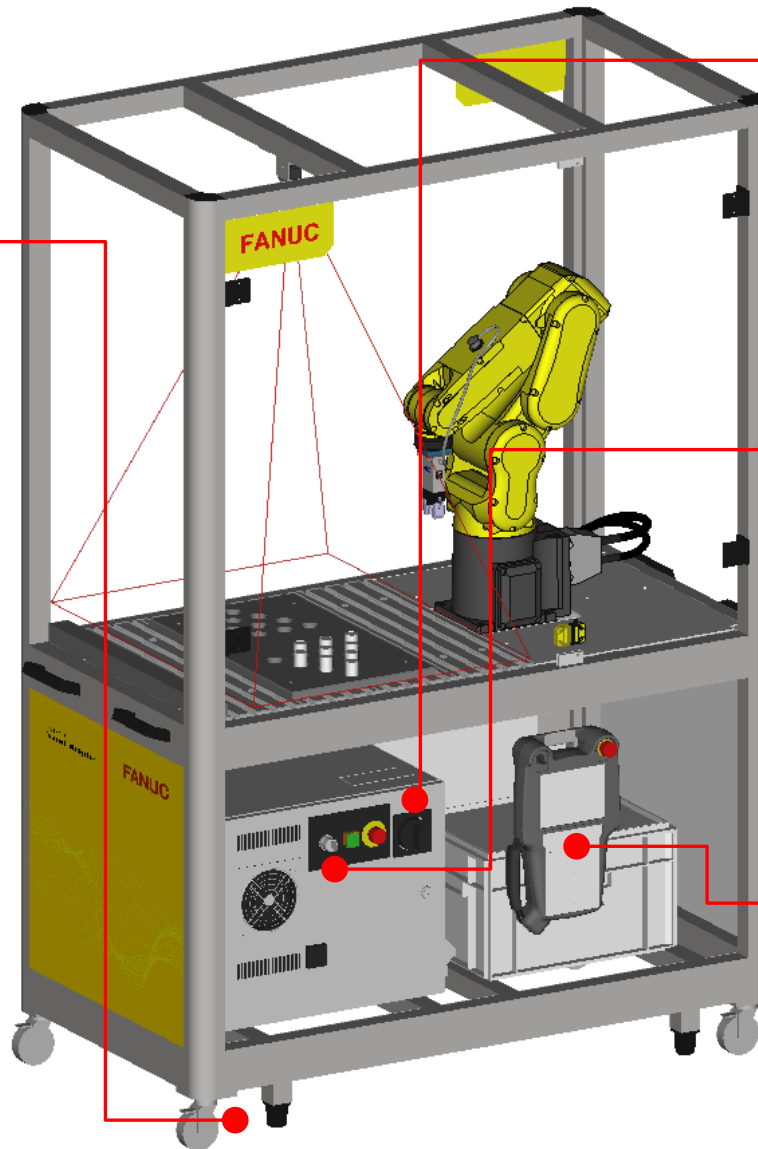
Step 1

Unpack the Cell, wheel it into position and adjust the frame feet

Step 2

Download and read the Technical Documentation:

www.fanuc.eu/educational-package



Step 5

Turn on the Main Power Switch and follow the instructions on the Teach Pendant to run the Demo program

Step 4

Remove the Auto-T1-T2 switch key from inside the controller and insert into the switch on the front of the controller in the Auto position

Step 3

Remove the Teach Pendant from the accessory box and connect the Teach Pendant and Power cables

2 Safety

2.1 General Instructions

Students / new users should be supervised by competent persons who are responsible for their safety.

Users' bodies should not enter the cell while moving/programming the robot

OK

NOK

Working in Cell **without**
Teach Pendant/ Servo OFF

Programming
Outside Cell

Programming with TP
Inside Cell

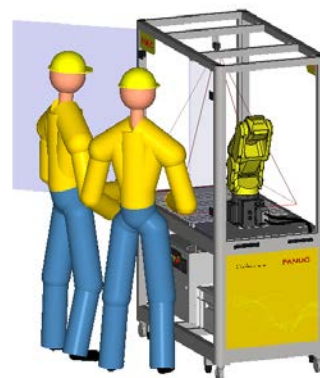


When installing, connecting peripherals etc, Servo power must be cut by E-Stop on Teach Pendant or Controller.

Users should not crowd around one another, especially in front of the cell door

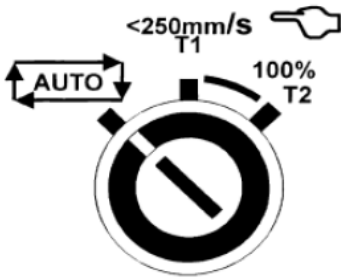
OK

NOK



For more information please refer to FANUC Safety Handbook B-80687EN/15

2.2 AUTO/T1/T2 Switch



Users should do setup / programming etc in T1 mode (<250mm/s) whenever possible.

T2 Mode should be reserved for Test Run and used with caution.

Auto Mode can be used with the door closed after setup and programming are complete.

Use of the Auto / T1 / T2 key should be restricted to authorised persons.

2.3 DCS Password

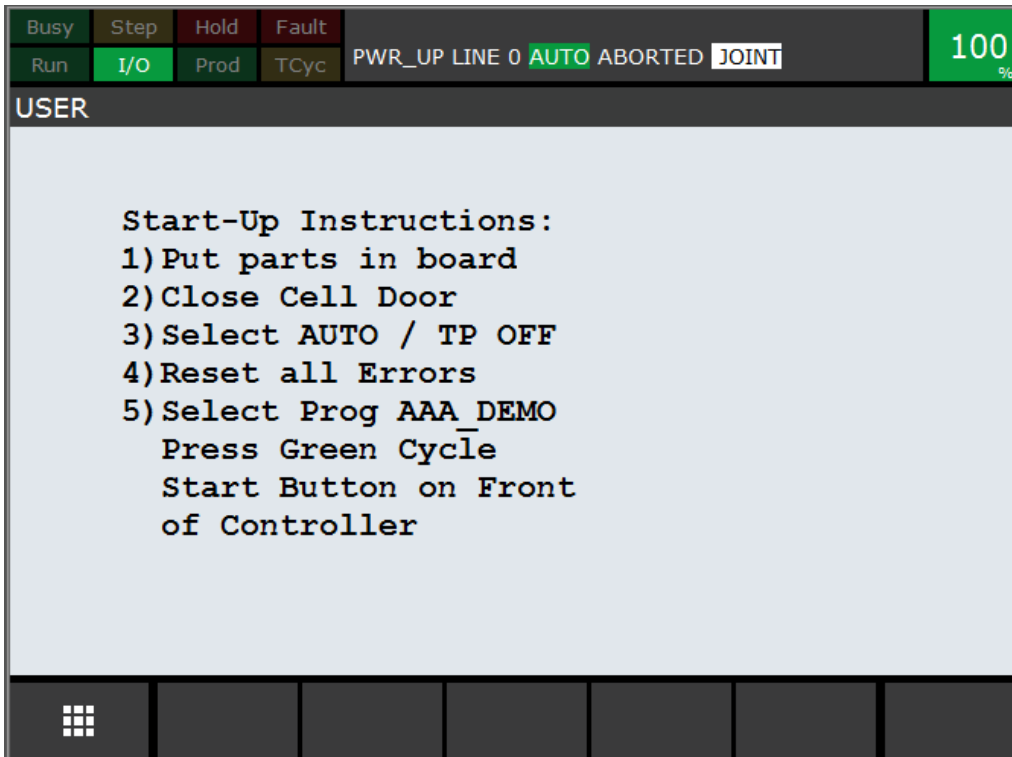
Please note that the DCS password has been left at the default value of '1111'. It is the User's responsibility to decide if it is necessary to change this – for instance if unauthorised access to the cell may result in DCS settings being unexpectedly changed or disabled. Correct DCS settings are necessary for safe operation of the cell – especially in Auto or T2 Mode.

If the DCS Password is changed please make sure to keep a note of the new password in a safe place – if DCS password is lost it is necessary to contact your FANUC representative to recover.

3 Selecting and running Example program

3.1 Start Up Screen

When the robot controller is turned on, it displays the following Start-Up Instructions:



3.2 Example Program “AAA_DEMO”

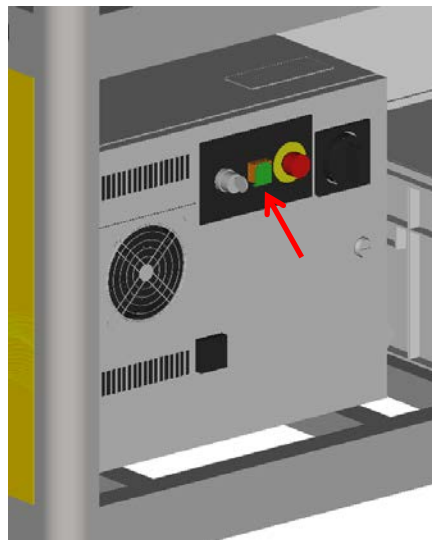
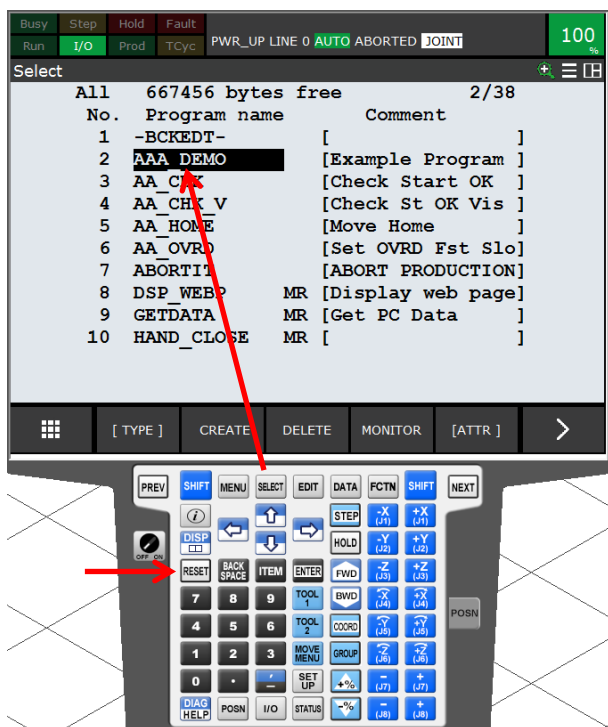
The program “AAA_DEMO” has been provided as an example program.

It transfers the cylindrical parts supplied from one triangular ‘solitaire’ board to another.

To start the program follow these steps:

- Robot must be in ‘AUTO’ mode, all E-Stops released, and cell door (‘Fence’) closed
- Robot should not be in error condition. If robot shows error condition press the “RESET” key on the Teach pendant to try to reset the errors.
- Select the program “AAA_DEMO” and Press the ‘Cycle Start’ button on the front of the controller.

(Teach pendant enable switch must be OFF and Auto/T1/T2 switch must be in ‘Auto’ position)



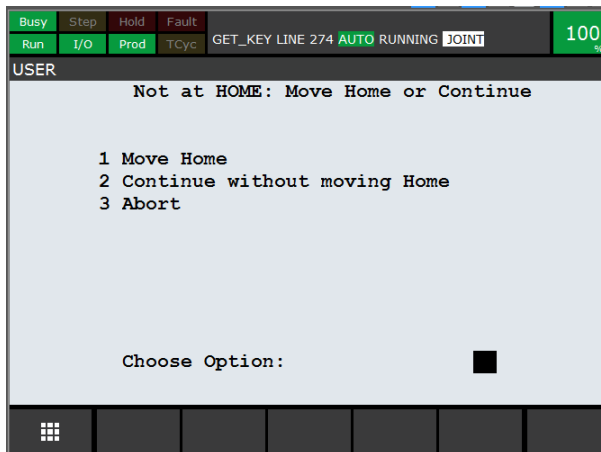
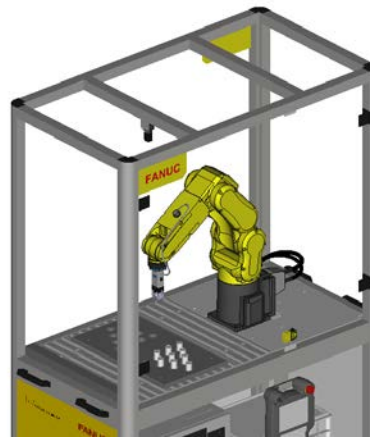
Please note – program “AAA_DEMO” must be selected before pressing ‘Cycle Start’

d. The program AAA_DEMO expects to be started from the HOME position:

Robot at HOME



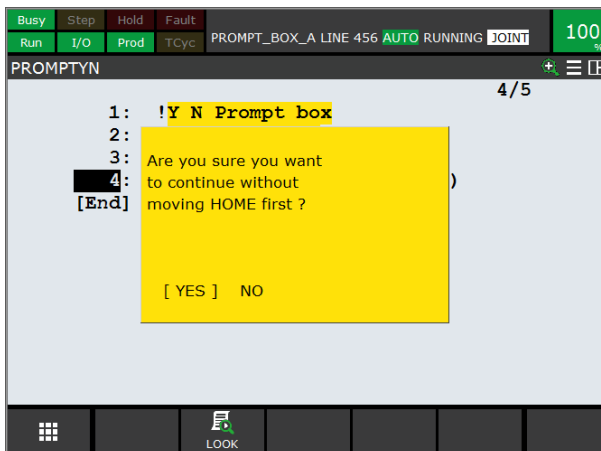
Robot not at HOME



If the Robot is not at the HOME position, the following screen will appear:

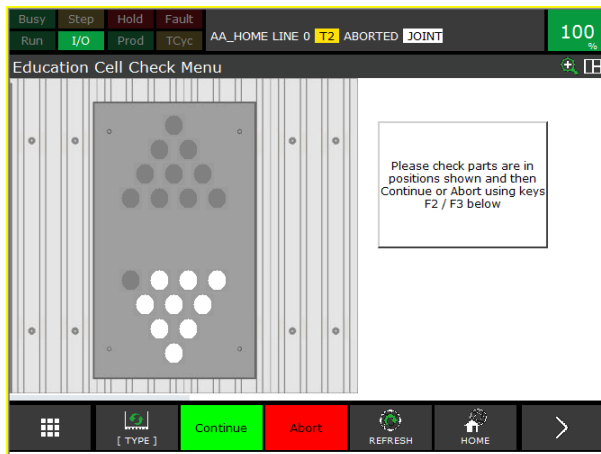
Choose the appropriate action and press 'ENTER'

If Option 2 "Continue" is chosen, then a confirmation screen will be displayed:



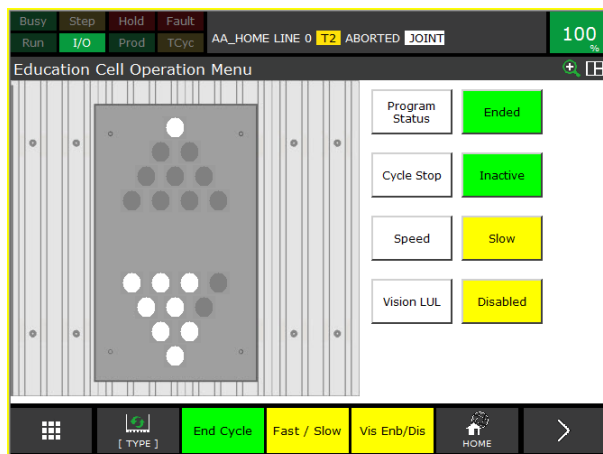
Note that these two functions have been implemented using the 'Menu Utility' function – see later section for details.

- e. Before the program starts, a screen will ask for confirmation that the parts are in the correct starting position:



Make sure that the parts are in the positions shown, then press "Continue".

- f. When the program is running, a Status / Menu screen is shown:.

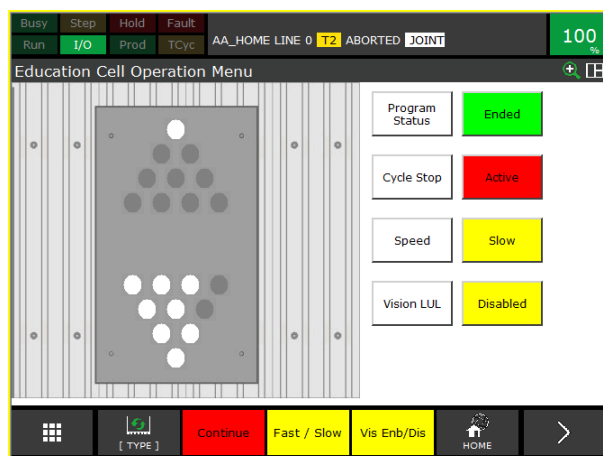


The red/white circles show the current position of the pieces

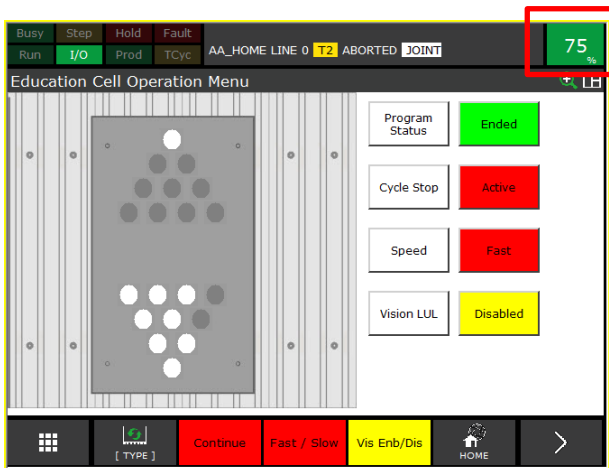
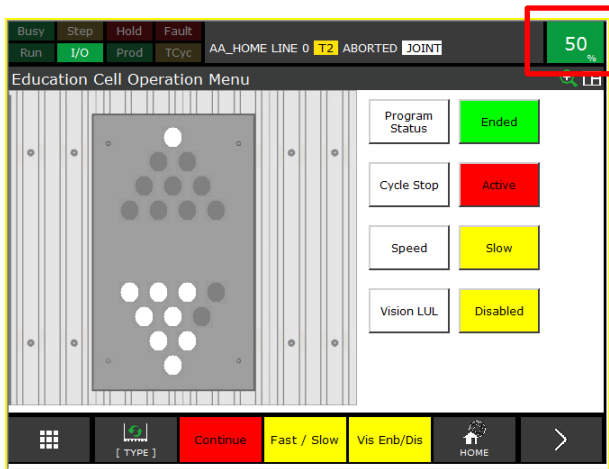
There are two function Key Selections: End Cycle / Continue and Fast/Slow.

If 'End Cycle' is pressed, this changes Register R[1] from 1 to 0, and the Cycle Stop status changes from Inactive to Active:

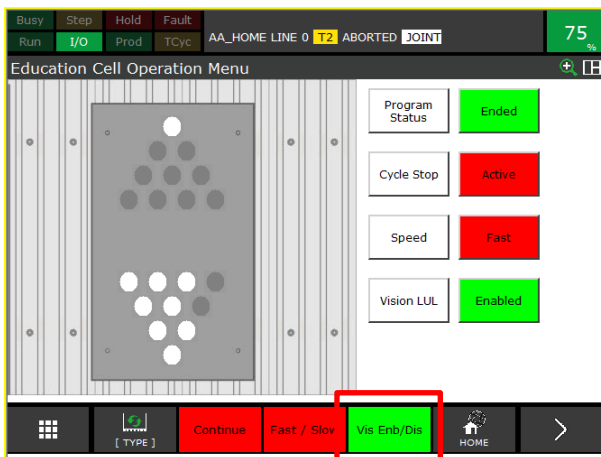
This will mean that the program will stop at the end of the next complete cycle and the Program Status will then become 'Ended'



g. Likewise, the Fast/Slow Function key changes the Override:



Note: the Fast / Slow Override speeds are set according to the values in R[4] and R[5] – see later section for details

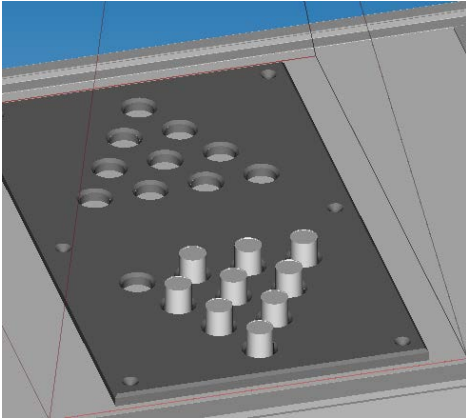


And the F4 function Key enables / disables the Vision Load / Unload section of the program – see next section for details.

Note: *iR*Vision must be installed and set up for this function to be used.

3.3 AAA_DEMO Program Sequence with / without *iR*Vision

There are 3 basic positions for the parts in/on the Solitaire board:



In the Left Hand side of the board:

This is the start and end position of the example program

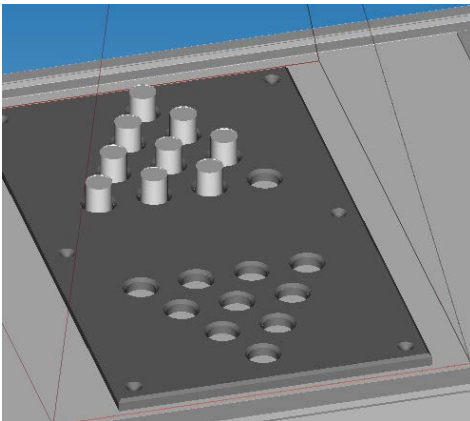
If Vision L/UL is Disabled then sequence is:

- Move Parts from Left Hand side to Right Hand side
- Move Parts from Right Hand side to Left Hand side
- Repeat

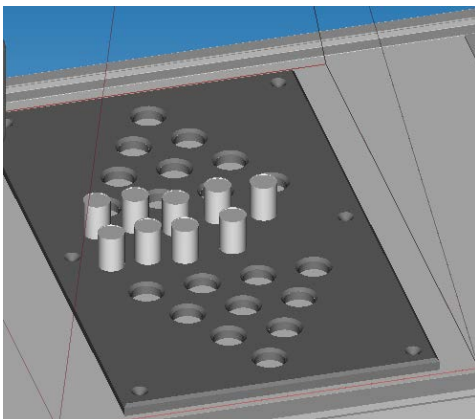
If Vision L/UL is Enabled then sequence is:

- Move Parts from Left Hand side to Right Hand side
- Move Parts into middle of board
- Move Parts from middle of board to Left Hand side using *iR*Vision.
- Repeat

In the Right Hand side of the board:



On the middle of the board:



This position is only used if *iR*Vision is enabled for the example program using "F4" function key

3.4 Operation Menu Display

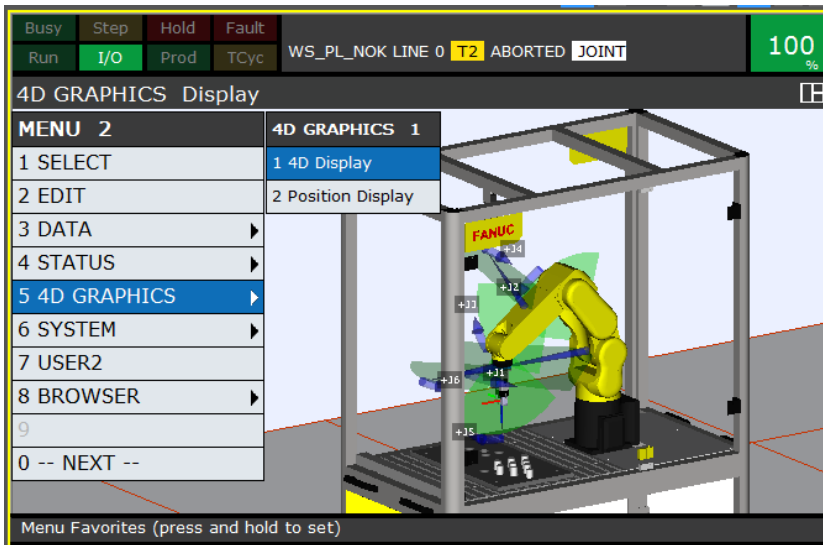
The Operation Menu can be displayed at any time by pressing the MENU key and then selecting the “Operate” shortcut that pops up:



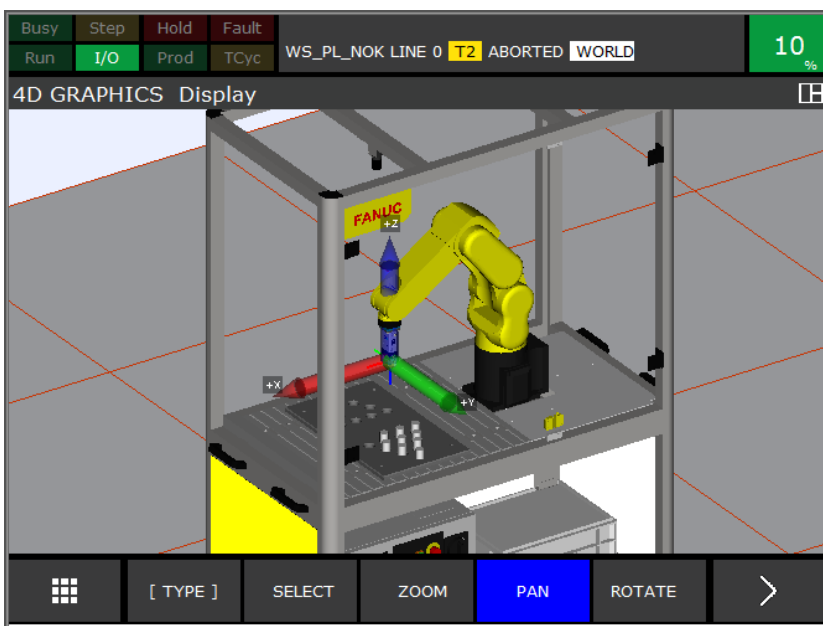
3.5 4D Graphics (Option)

The “4D Graphics” function is available as an option, with a model of the cell loaded into the robot controller.

This model can be displayed by pressing the MENU button and selecting “4D Graphics”:



The 4D graphics function has many features, for example as shown below the display of the Jog Coordinates – in this case the WORLD Jog Coordinate System:



4 Demonstration Program Setup / Teaching

The positions of the holes in the tray are calculated, not taught – so there should be no need to touchup / reteach the individual hole positions. However, it may be necessary to touchup / reteach the User Frame (UFrame) which defines the position of the tray relative to the robot.

4.1 Gripper Setup

To teach the positions it is necessary to use a part held in the gripper.

Gripper Force should be set at 50% using the small rotary switch on the gripper.

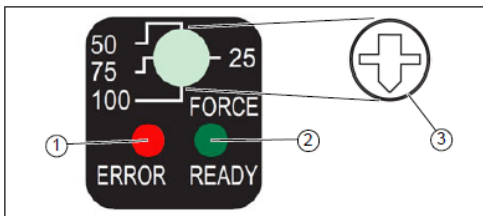


Fig. 4 Function label with rotary switch of EGP 40

1	Error-LED
2	Power-On-LED
3	Rotary switch (not available for EGP-S)

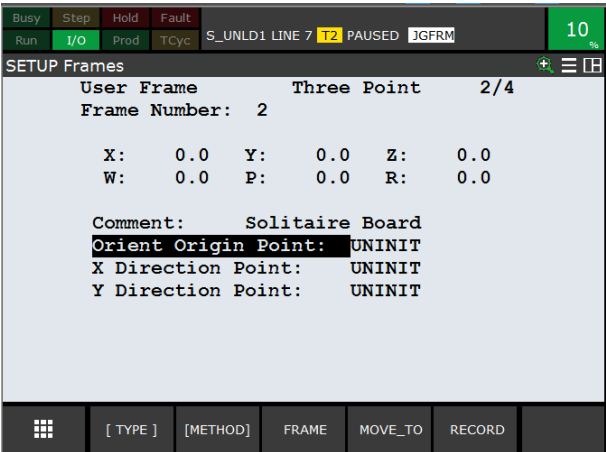


4.2 UFrame 2 Setup

The User Frame touchup is done using the Menu>Setup>Frames function:



This requires 3 Points to be touched-up / re-taught:

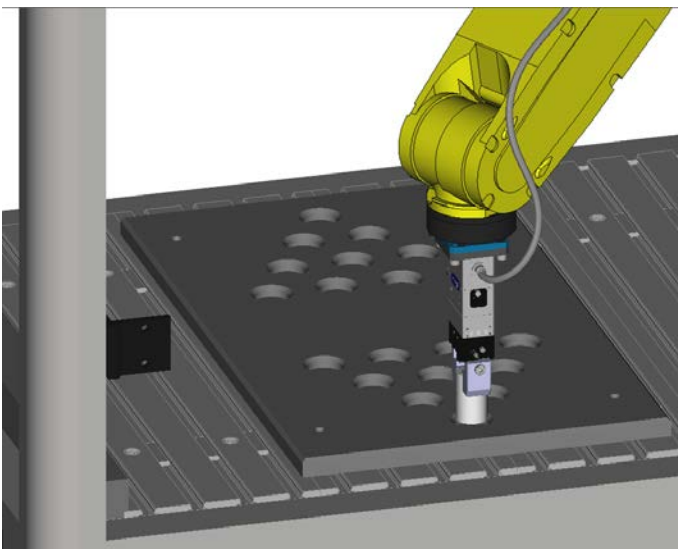


Note that the MOVE_TO function key provides a method to easily check the current taught positions

First, insert a piece into the gripper:

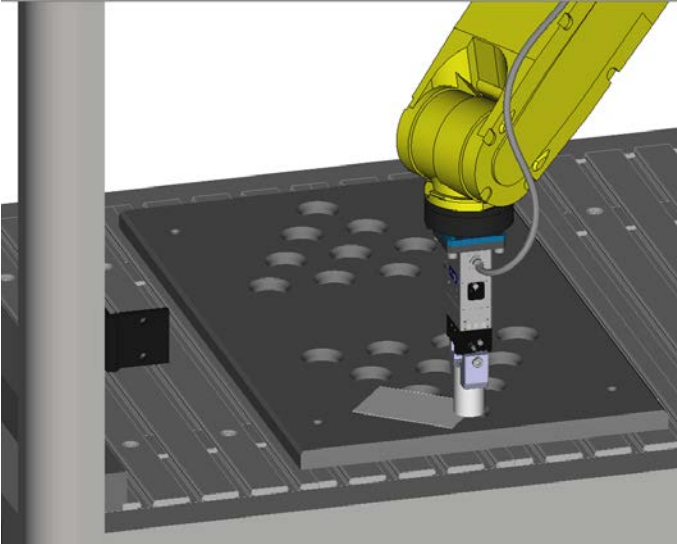


Make sure gripper is vertical, and jog the robot so that the piece is central in the Left-most hole on the tray shown below:



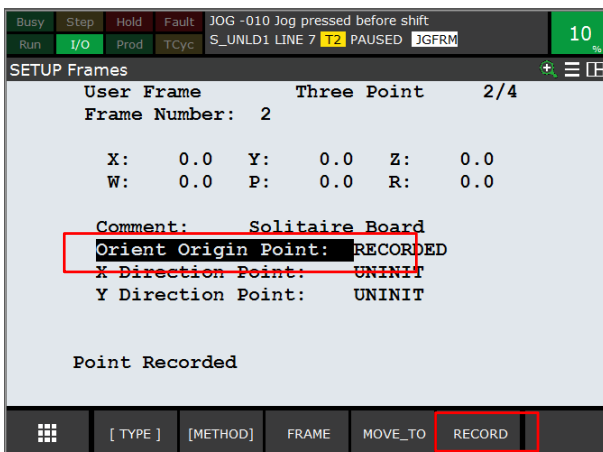
It is difficult to know when the piece is touching the table at the bottom of the hole – so this position is not used for the reference point.

Instead, jog the robot upwards using WORLD+Z until the bottom of the piece is just above the top of the tray. Use a thin piece of card to help judge this correctly:



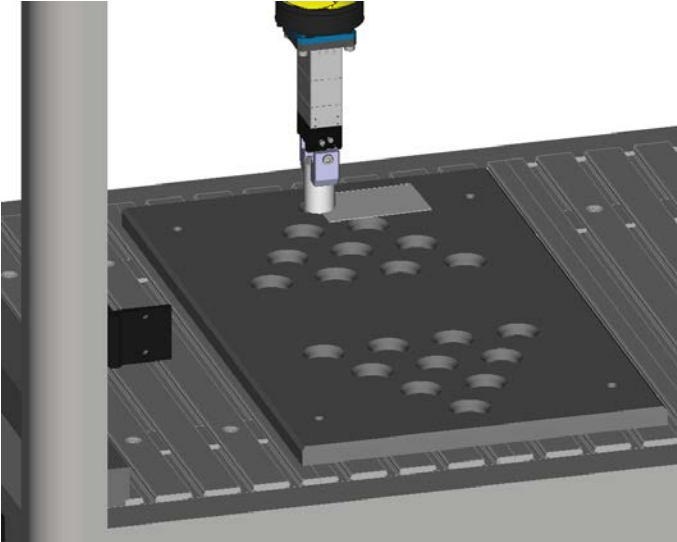
4.3 Orient Origin Point

Then RECORD this position as Orient Origin Point – see below:

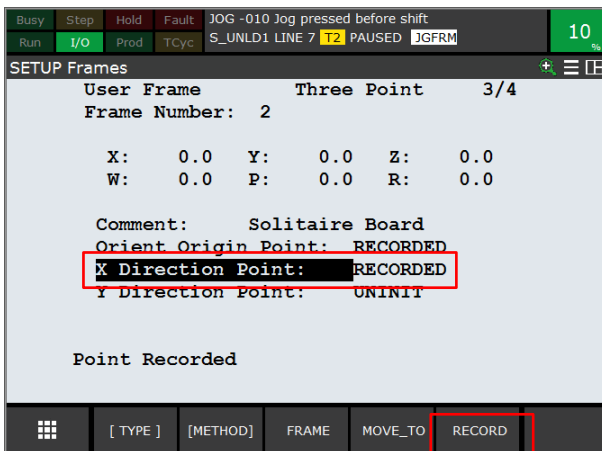


4.4 X-Direction Point

Now do the same thing for the Right-most hole in the tray:



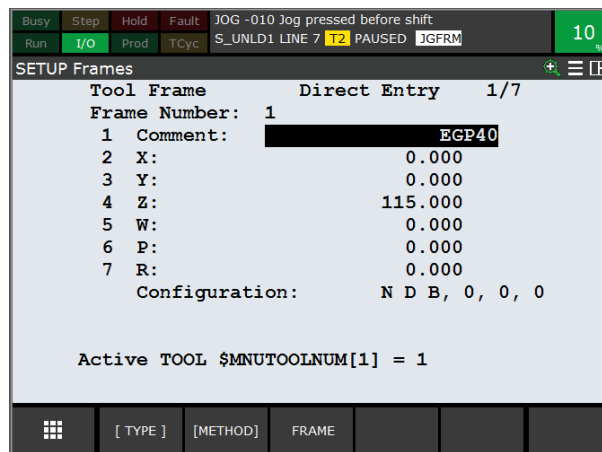
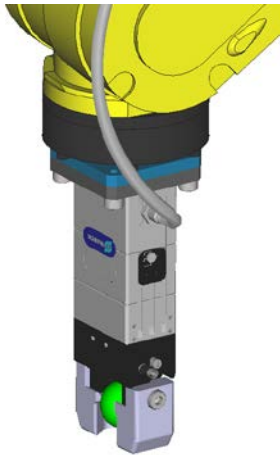
And RECORD this position as X-Direction Point – see below:



5 Robot Setup

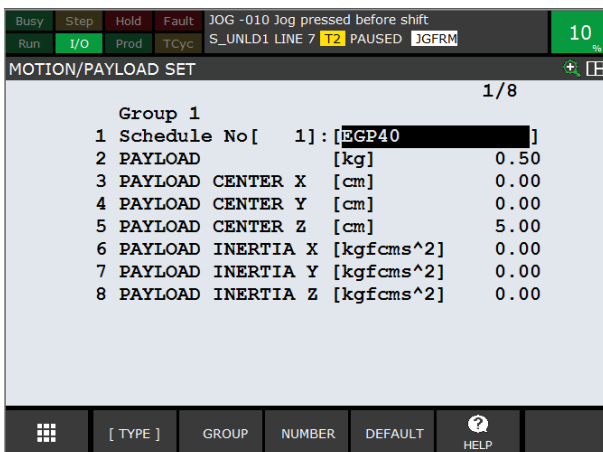
5.1 UTOOL / TCP Setup

Because of the simple shape and mounting of the Schunk gripper, a simple TCP with an offset of 115mm in Z is sufficient.



5.2 Payload Setup

The payload is quite low for this application – so only one payload of 0.5 kg has been set:



5.3 DCS Setup

DCS (Dual Check Safety) has been used to ensure that the robot cannot accidentally hit the walls of the cell. This requires setting up a Tool Model and a Safe Zone.

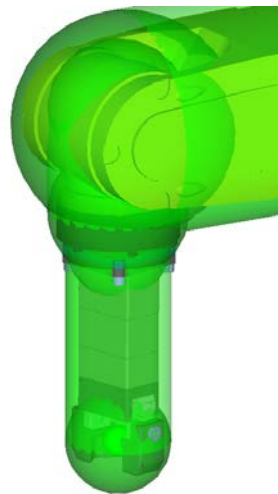
5.3.1 DCS Tool Model.

A simple DCS model using one "Line_seg" type model has been used.

For more details please refer to DCS manuals.

```
DCS
User element detail
No. 1 [Schunk Gripper ]
Element: 1 Status: CHGD

1 Enable/Disable:          ENABLE
2 Link No. (99:FacePlate): 99
3 Link type:              NORMAL
4 Tool frame:             0
5 Shape:                  Line_seg
6 Size (mm):              30.0
Pos1
7 X: 0.000 Y: 0.000 Z: 0.000
Pos2
8 X: 0.000 Y: 0.000 Z: 115.000
```



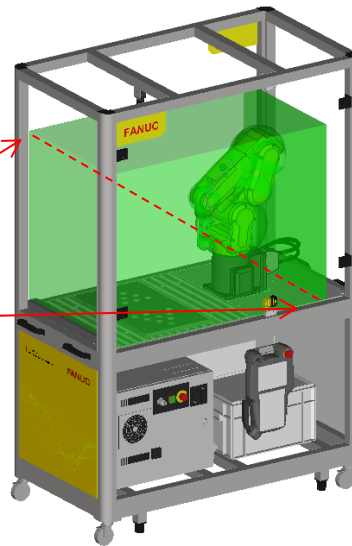
5.3.2 DCS Safe Zone

A simple DCS Cartesian Position Check Zone has been set up using a Diagonal line to define a cuboid zone, running from a point on the “ top left “ of the cell to the “bottom right “ – see screenshot below - where the inside of the cuboid is safe.

If the robot or the tool comes close to the edge of this zone, the robot will stop.

For more details please refer to DCS manuals.

```
DCS
Cartesian position check
No. 1          Status: CHGT
1 Comment: [Ed Cell Frame ]
2 Enable/Disable: ENABLE
3 Method: Working zone(Diagonal)
4 Group: 1
5 Target model 1: Robot model
6 Target model 2: User model 1
7 Target model 3: DISABLE
8 Base frame: User Frame : 0
Position(mm):
  Current   Point 1   Point 2
9 X      487.9   750.0   -250.0
10 Y      27.8   -290.0   290.0
11 Z      65.0   400.0   -330.0
12 Stop type: Stop Category 1
13 Speed check: <DETAIL>
14 Disabling input: ---[ 0: ]
15 Use Stop Position Prediction: Yes
```



5.4 I/O Setup

Only 2 I/Os are used, for gripper Open / Close

These outputs are configured as Complementary, so when, for example RO[7] is set ON, RO[8] is automatically forced to OFF and vice-versa.

The screenshot displays the FANUC I/O setup interface. At the top, there are status indicators: Busy, Step, Hold, Fault, Run, I/O, Prod, and TCyc. The main display area is split into two panels. The left panel, titled 'I/O Robot Out', shows a table of 8 robot outputs (RO[1] to RO[8]). The right panel, titled 'I/O Robot Out Port Detail', shows the configuration for the selected output (RO[7]).

#	SIM	STATUS	
RO[1]	U	OFF	[]
RO[2]	U	OFF	[]
RO[3]	U	OFF	[]
RO[4]	U	OFF	[]
RO[5]	U	OFF	[]
RO[6]	U	OFF	[]
RO[7]	U	ON	[Open Gripper]
RO[8]	U	OFF	[Close Gripper]

Port Detail 1/3

Robot Dig. Output [7]

1 Comment: [Open Gripper]

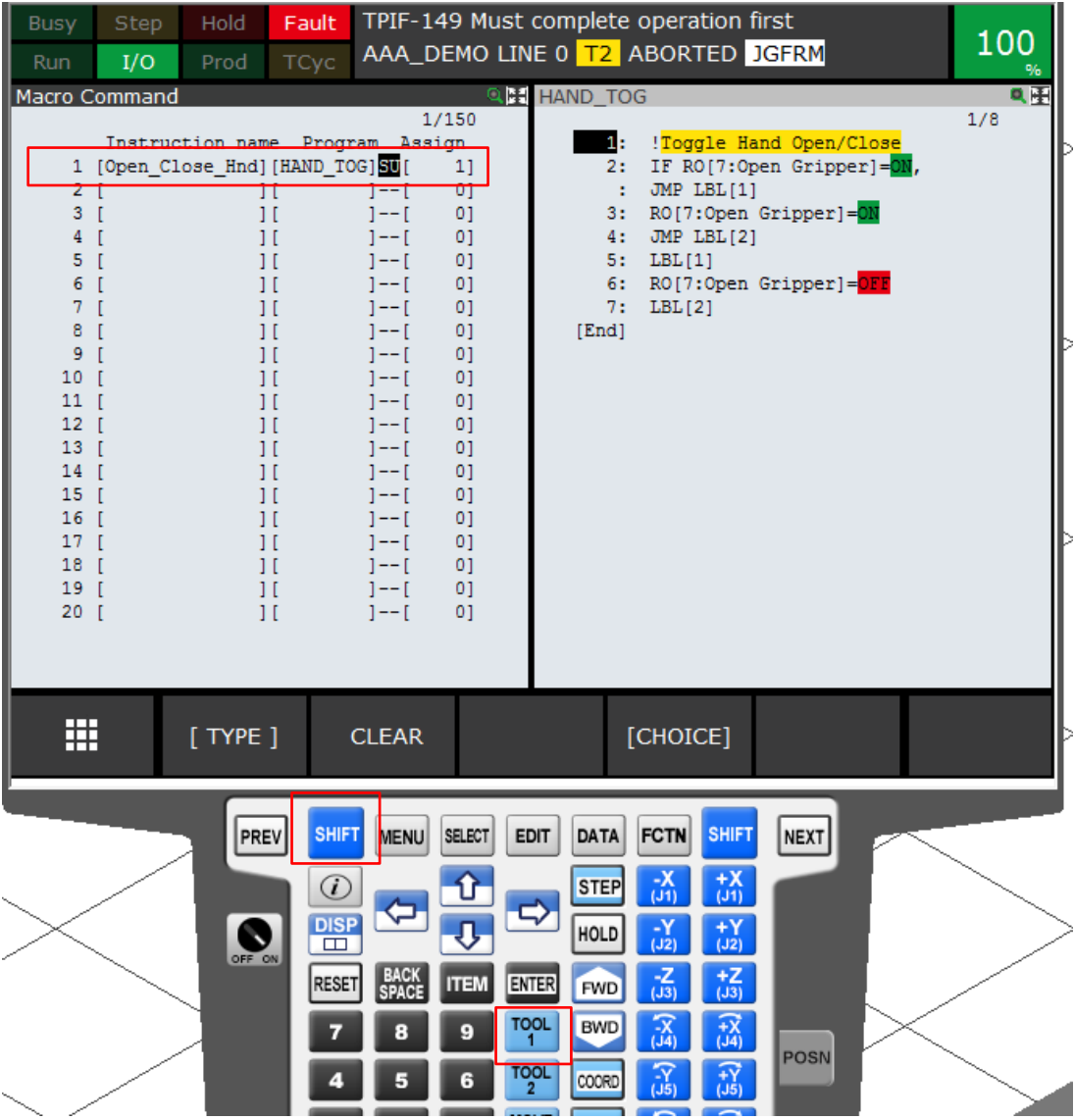
2 Polarity: NORMAL

3 Complementary: TRUE [7 - 8]

At the bottom of the interface, there are navigation buttons: a grid icon, [TYPE], IN/OUT, ON (green), OFF (red), and a right arrow.

5.5 Macro Setup

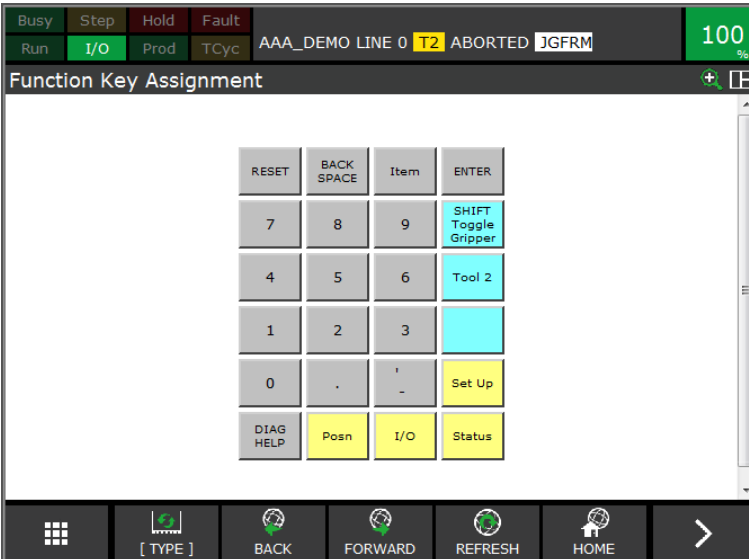
One macro has been set up, to allow the user to easily toggle the gripper open / close using SHIFT + User Key 1 on the Teach Pendant:



5.6 TP Function Key Hint Screen

To allow the user to easily remember the setting of the Function Key, a Hint Screen has been set up using an HTML Page:

When the "Menu" key is pressed, the Shortcut key F2 "FKey" pops up:



Pressing F2 will display the FKey Hint screen:

This screen shows the current assignment of the function keys.

The text which is displayed is the text stored in String Registers SR[21] – SR[25]

5.7 Home / Reference Position

One reference position has been set up, which is used for the HOME position check. Home position is this:



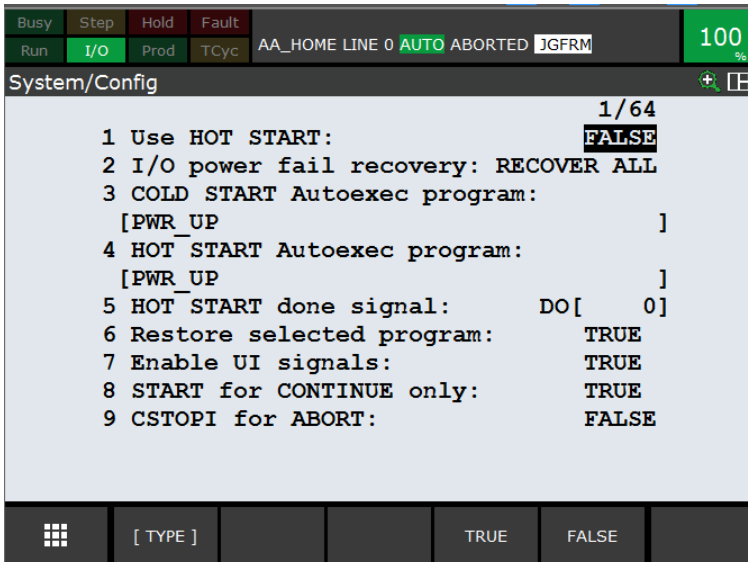
And setup is this:

Busy	Step	Hold	Fault	AA_HOME LINE 0 AUTO ABORTED JGFRM		100 %
Run	I/O	Prod	TCyc			
REF POSN			I/O Digital Out			98/512
Reference Position 1/13						
Ref.Position Number: 1						
1	Comment:	[Home Position]				
2	Enable/Disable:	ENABLE				
3	Is a valid HOME:	TRUE				
4	Signal definition:	DO [101]				
5	J1:	0.000 +/- 2.000	DO[91]	*	*	[]
6	J2:	-28.000 +/- 2.000	DO[92]	*	*	[]
7	J3:	-35.000 +/- 2.000	DO[93]	*	*	[]
8	J4:	0.000 +/- 2.000	DO[94]	*	*	[]
9	J5:	-55.000 +/- 2.000	DO[95]	*	*	[]
10	J6:	0.000 +/- 2.000	DO[96]	*	*	[]
11	J7:	0.000 +/- 0.000	DO[97]	*	*	[]
12	J8:	0.000 +/- 0.000	DO[98]	*	*	[]
13	J9:	0.000 +/- 0.000	DO[99]	*	*	[]
			DO[100]	*	*	[]
			DO[101]	U	ON	[HOME Signal]
			DO[102]	U	OFF	[]
			DO[103]	U	OFF	[]
			DO[104]	U	OFF	[]
			DO[105]	U	OFF	[]
			DO[106]	U	OFF	[]
			DO[107]	U	OFF	[]
			DO[108]	U	OFF	[]
			DO[109]	U	OFF	[]
			DO[110]	U	OFF	[]
			DO[111]	U	OFF	[]

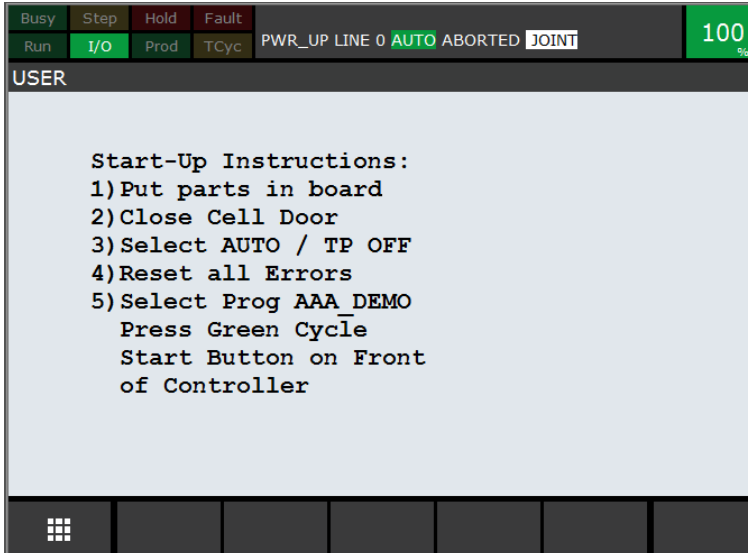
When the robot is in this position, the output DO [101] will be ON – and this can be checked by the TP program AA_CHK

5.8 Power Up Program

To provide instructions to the user, a Power-Up program "PWR_UP.TP" has been assigned to both Hot Start and Cold Start.



The program simply writes instructions to the Teach Pendant:



6 iRVision Setup (Option)

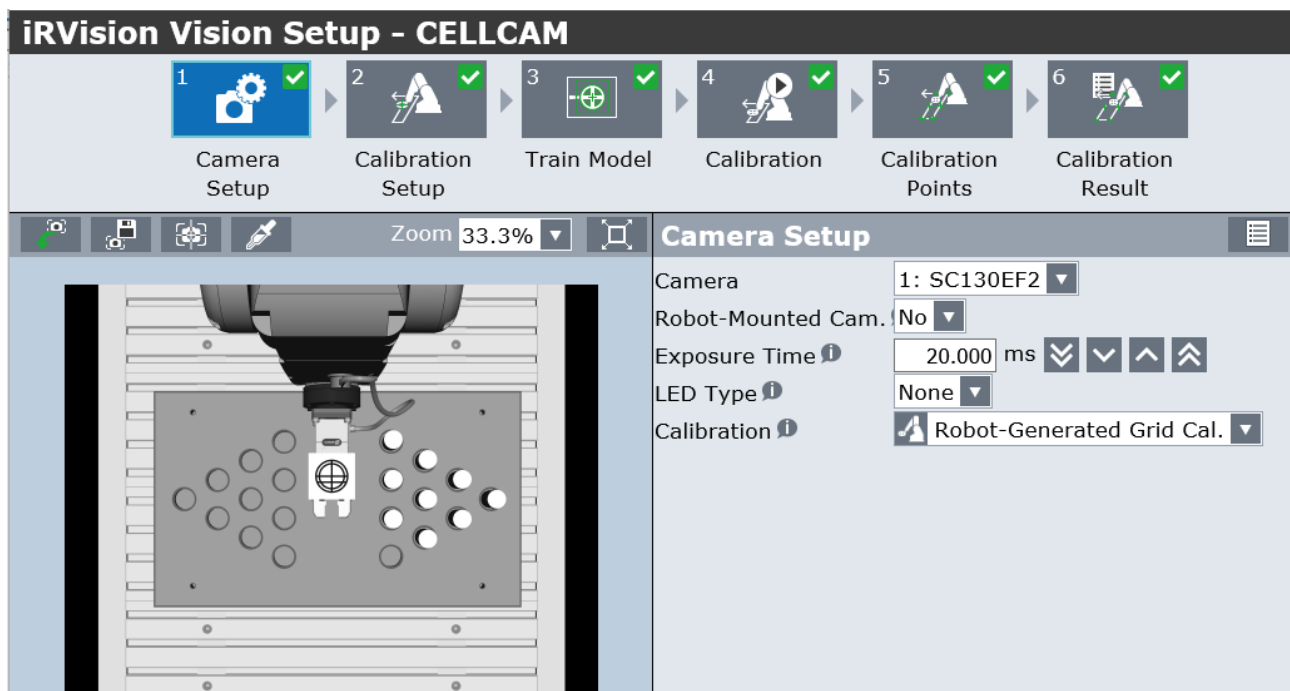
The Education Cell is available with and without integrated iRVision using KOWA video camera.

In either case, the iRVision Setup has not been done on the Education Cell controller. Please contact your local FANUC representative to arrange training on iRVision

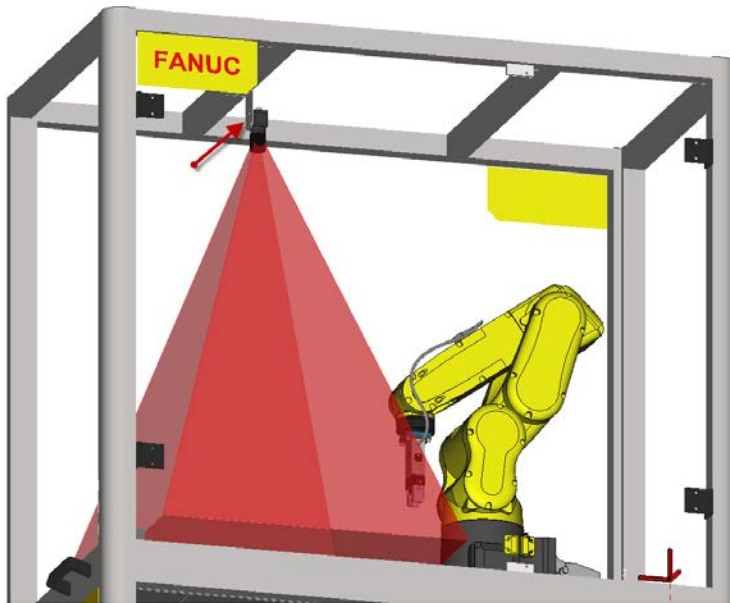
The following section is just to give a short overview of a sample iRVision setup, and is not intended to take the place of a proper iRVision training.

6.1 Camera Adjustment

If this option has been ordered, the basic camera setup should already have been done by FANUC Europe, so the camera should display an image something like this:

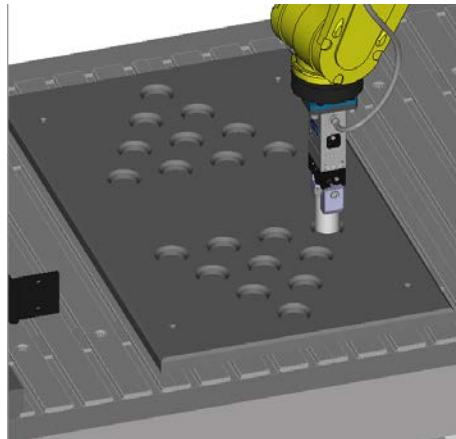


If the Camera View is not like this, then please adjust using the screws on the Camera Mounting bracket:

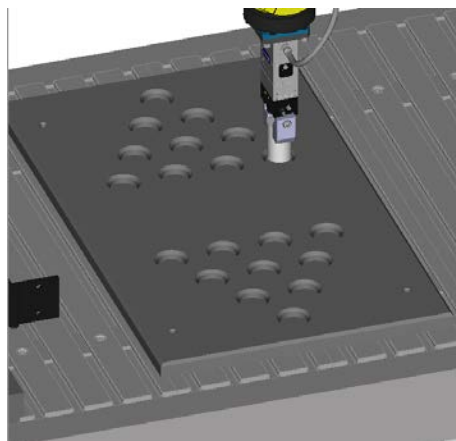


6.2 Application Frame

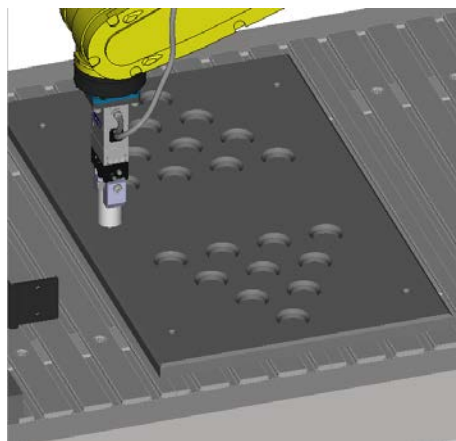
It is good practice to set up an Application Frame to use with *iR*Vision, for example Uframe 9. This can be done in the same way as for the UFrame 2 Setup



Orient Origin Point:



X-Direction Point:

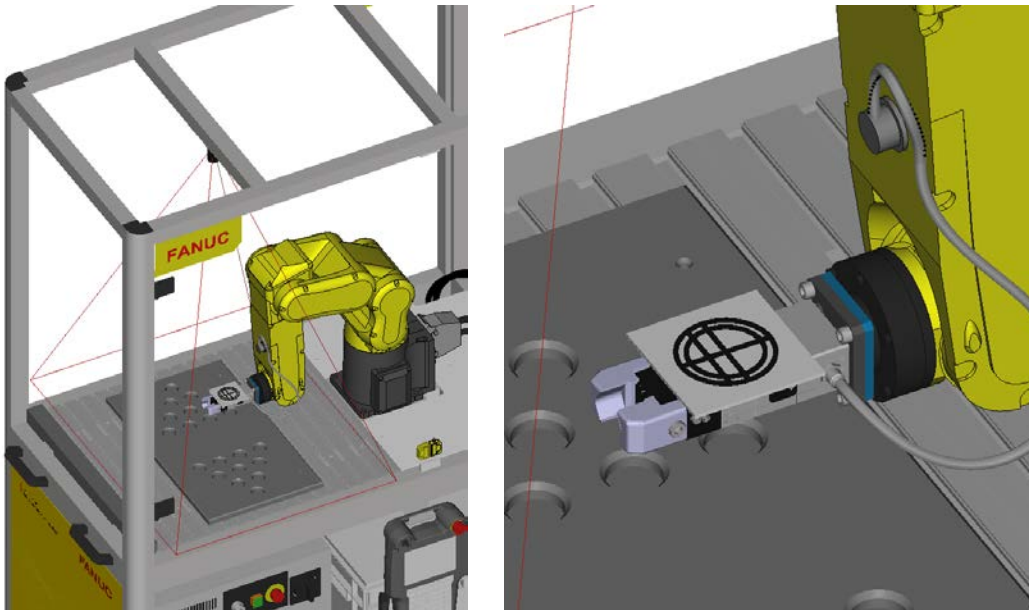


Y- Direction Point:

6.3 Camera Calibration

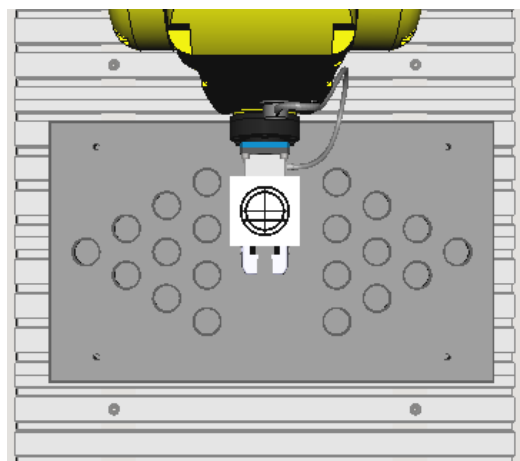
The *iR*Vision Calibration Grids are not included in the Education Cell. Instead the “Robot Generated Grid Calibration” Method can be used – please refer to the *iR*Vision Operation Manual for details.

Basically a target should be temporarily attached to the Gripper as shown:



(The design of the target is explained in the *iR*Vision manual – it should be approx. 50mm diameter)

It will appear something like this in the camera field-of-view:

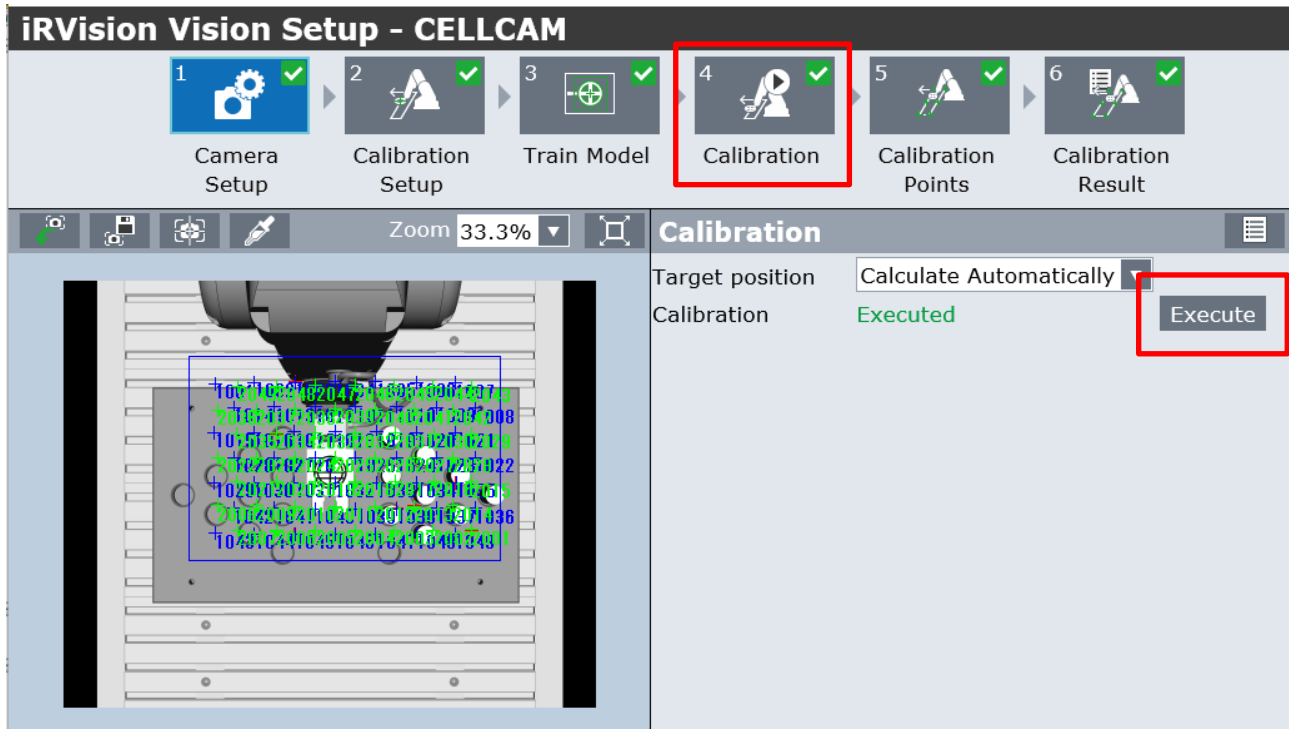


The Camera Calibration takes place in 2 steps:

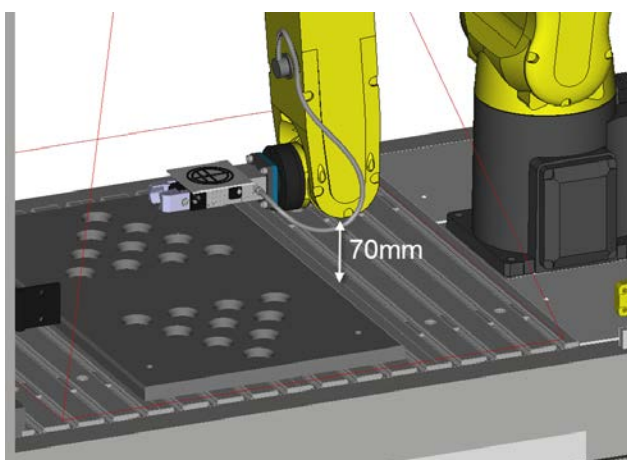
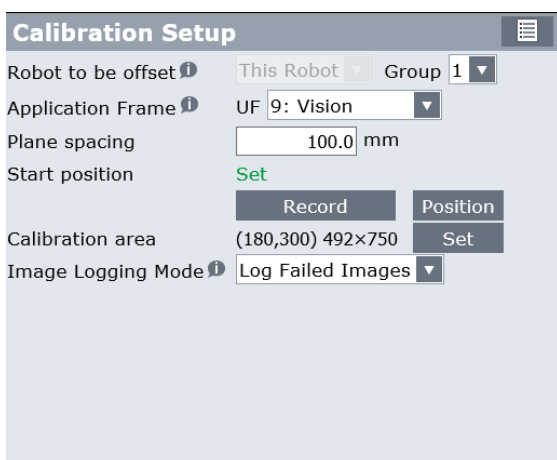
- First find the relationship of the target to the robot
- Second move the target around the field of view to calibrate the camera

Please note that it may be necessary to disable DCS while calibrating the camera field of view since the robot moves the target right to the edges of the field of view, close to the side walls.

All of this is done through the Teach Pendant using the *i*RVision Utilities Menu:



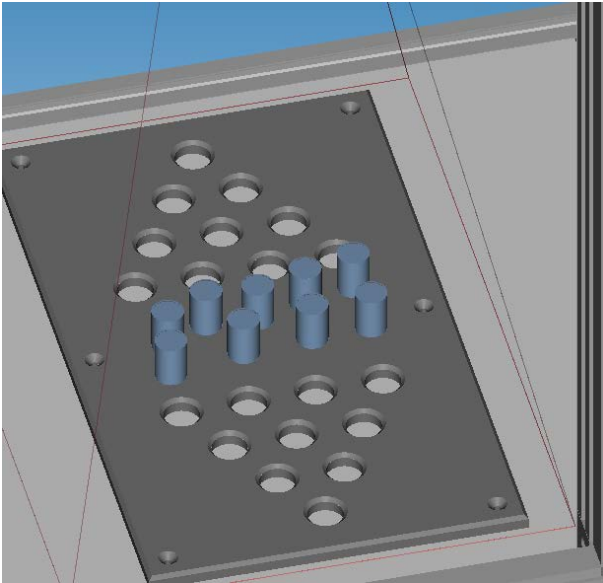
For details please refer to the *i*RVision Operation Manual. Note that for the Start Position, there must be enough space between the robot wrist and the cell – approx. 70mm is good:



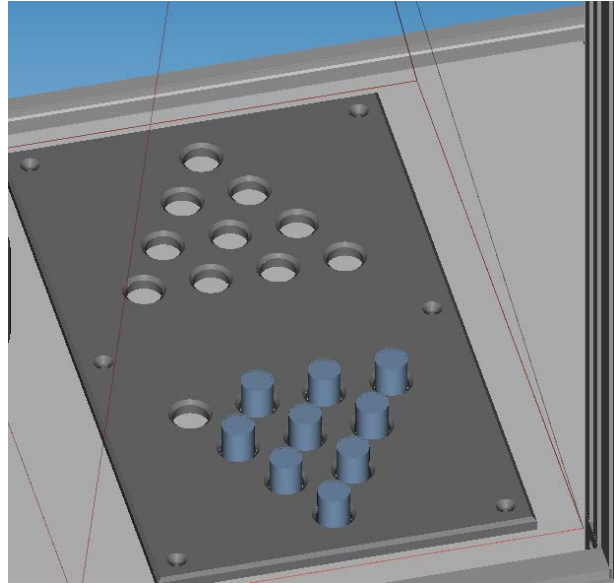
6.4 Example *iR*Vision Application

A simple application could be to load the parts from the centre of the board into the starting position for the Solitaire program

So the parts start like this,
randomly placed in the centre of the board



And end like this,
in the start position for 'Solitaire'



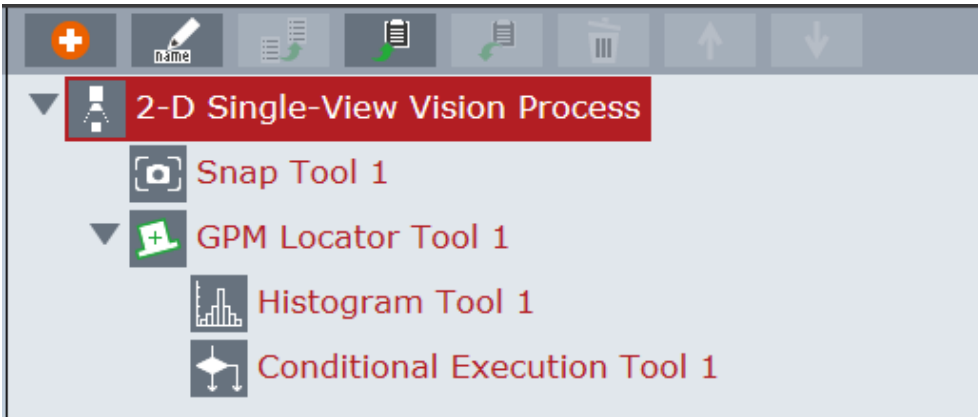
Using FANUC *iR*Vision this is quite straightforward to do:

- Calibrate the Camera (as shown previously)
- Teach a reference pick-up position for one of the parts
- Teach *iR*Vision to recognise and locate one of the parts.
- Create a TP program to use the *iR*Vision information to pick the part and then place into the board.

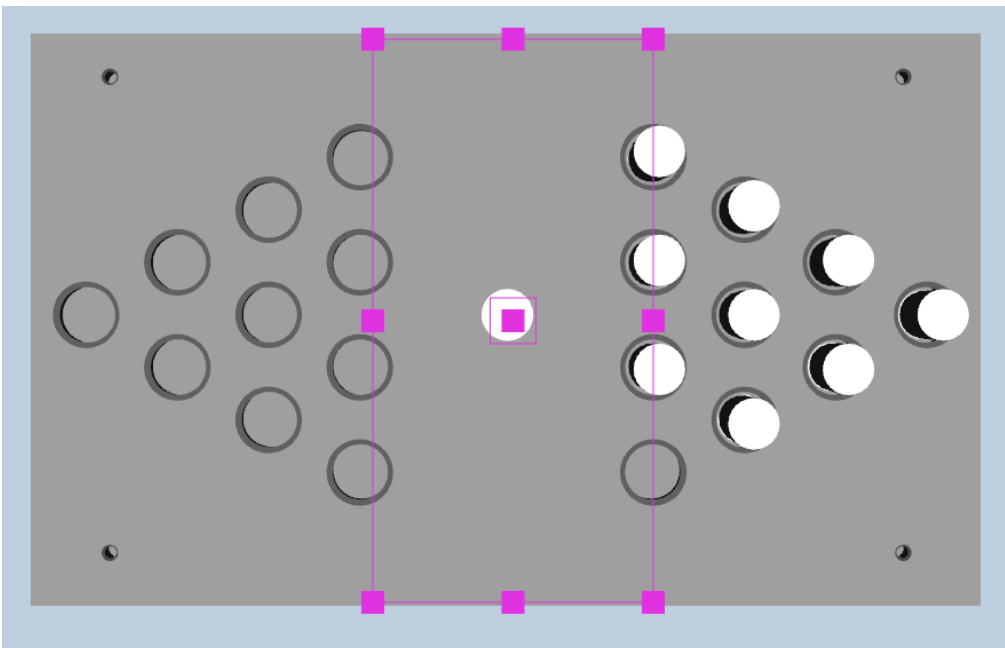
(This was suggestion but has been now added to the demo cell as shipped by FANUC – see section 1.3)

6.5 iRVision Tips

FEC set this application up using the Vision Process below:

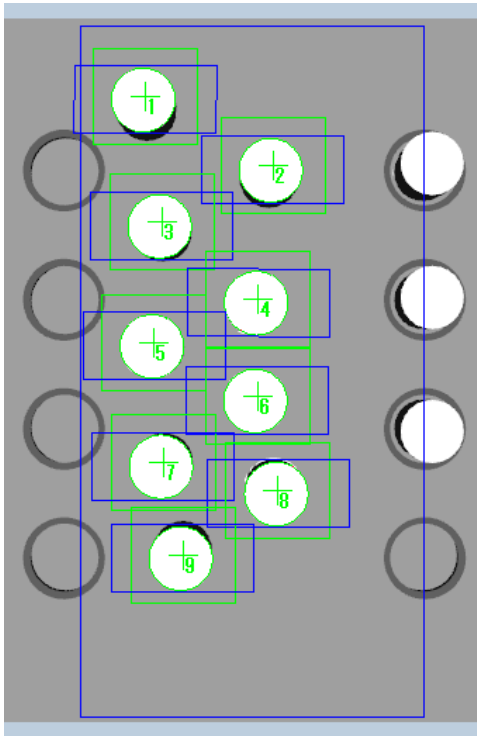


- A simple GPM locator tool to find the cylindrical part
- The GPM Locator has a search window to restrict the vision to the centre of the board to avoid finding the 'holes' instead of the parts:

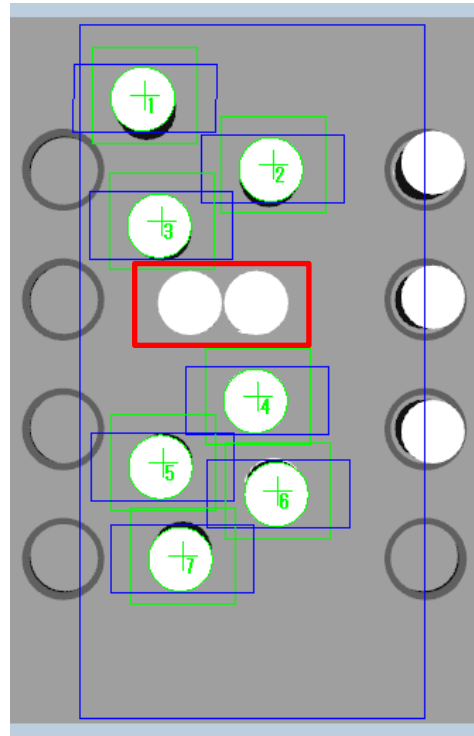


- Then the Histogram Tool and the Conditional Execution Tool are used to make sure that there is enough free space around the part to allow it to be picked up.

So all of these parts can be picked:



But the two central parts here cannot be picked because they are too close together.



6.6 TP Program

Write a simple TP program to Pick and Place the parts:

```

IRV_PICK LINE 0 AUTO ABORTED JGFRM 100%
1: !Pick Part with Vision
2:
3: UTOOL_NUM=1
4: UFRAME_NUM=9
5:
6: LBL[1]
7:
8: VISION RUN_FIND 'FIND_CYL'
9: VISION GET_OFFSET 'FIND_CYL' VR[1] JMP LBL[90]
10: CALL HAND_OPEN
11:J P[1] 25% FINE VOFFSET,VR[1] Tool_Offset,PR[40:ToolOffset Abo]
12:L P[1] 500mm/sec FINE VOFFSET,VR[1] Tool_Offset,PR[41:ToolOff PickPl]
13: CALL HAND_CLOSE
14:L P[1] 500mm/sec FINE VOFFSET,VR[1] Tool_Offset,PR[40:ToolOffset Abo]
15:
16: JMP LBL[99]
17: LBL[90]
18: !ask operator to fix
19:L P[2] 500mm/sec FINE
20: CALL PROMPTOK(2)
    
```

It is advisable also to add in some simple error handling to check that the Vision is working OK. This can be done using the Menu Utility as described in section 7

```

SETUP Menu Utility
Prompt msg menu no: 2
Menu name: VisFail
Line Text
1 Vision Failed to find Part
2 Check
3
4 Press
5
Vision Failed to find Part
Check Parts placed OK

Press OK to Retry

[ OK ]

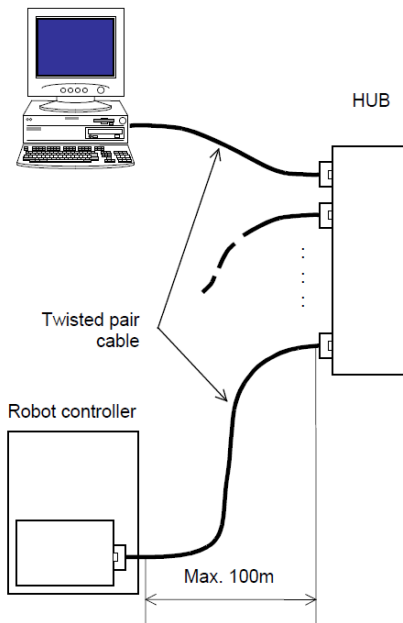
Checking menu variable definitions...
    
```

For more details please refer to the *iR*Vision Operation Manual or arrange training with your local FANUC Europe representative.

7 IoT / Industrie 4.0 / PC Connection

The FANUC Europe Education Cell is delivered ready for connection to a PC or to a Network via Ethernet to allow full connectivity (For full details please refer to the FANUC Ethernet Function OPERATOR'S MANUAL B-82974EN/04).

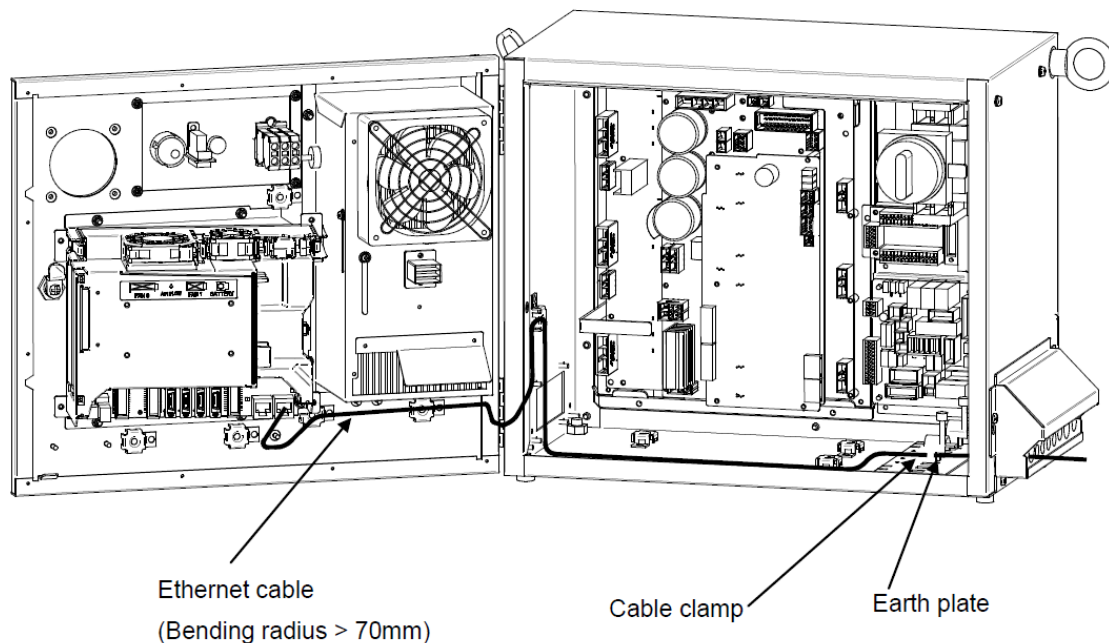
7.1 Physical Ethernet connection



The maximum distance between controller and Hub or PC is 100m.

The Ethernet cable should be fastened by a cable clamp as shown below to prevent tension being applied to the RJ-45 connector.

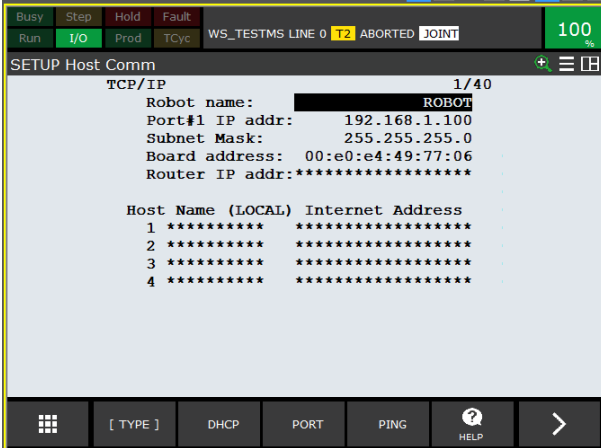
This clamp also grounds the cable shield.



7.2 Robot Software Setup

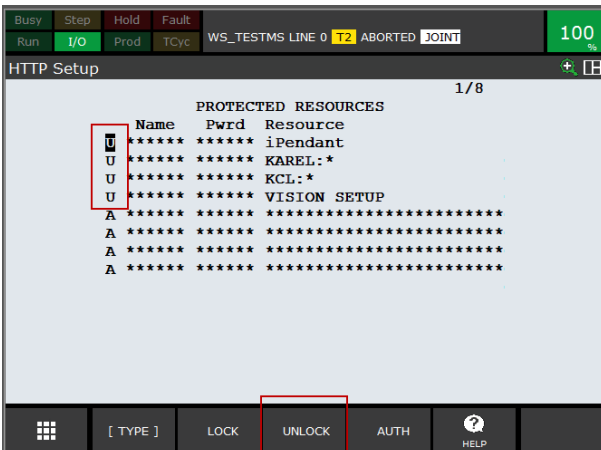
To set up the Ethernet communication between PC and robot, select:

MENU>SETUP>Host Comm> TCP/IP > DETAIL:



Please set a suitable IP Address and Subnet Mask
Other settings may be required depending on your specific network connections.

To allow full access from remote device, select: MENU>SETUP>Host Comm> TCP/IP > DETAIL:



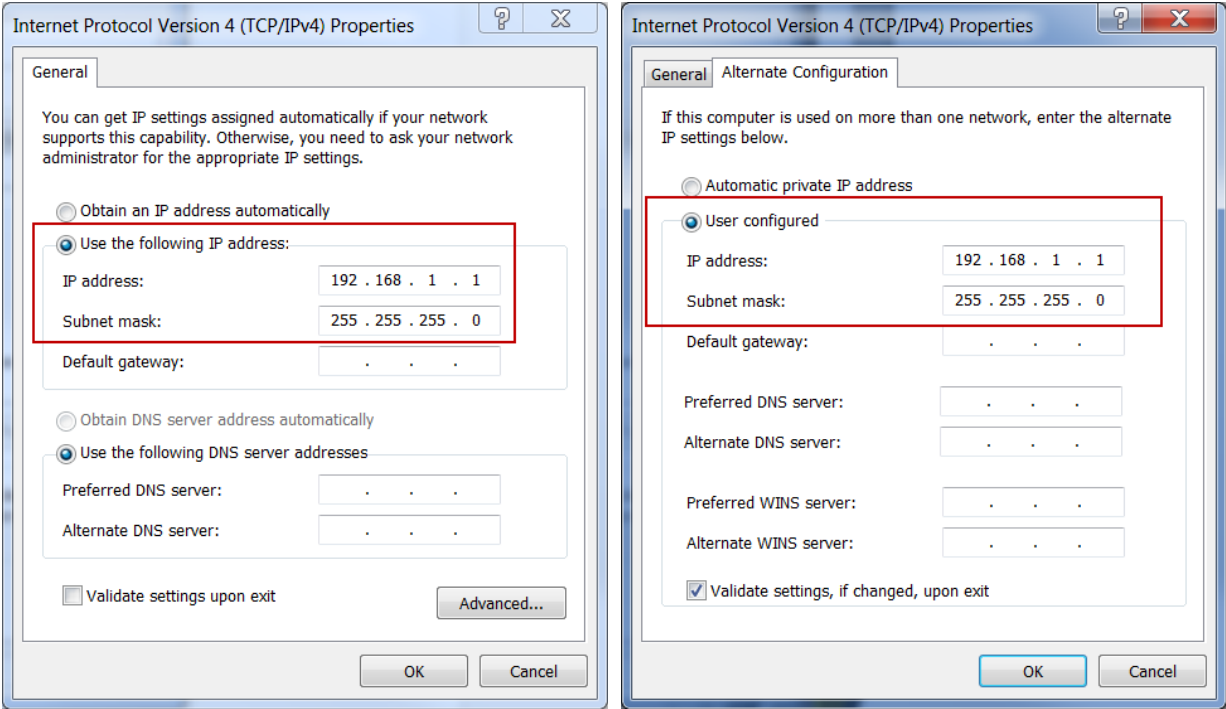
Please ensure that these resources are set to 'U' – Unlocked – to allow access.



(If you do not do this then you may see a message such as the one shown here)

7.3 PC Setup

There are different ways to set up the PC to communicate to the robot, depending on whether the PC will be permanently or temporarily connected. In either case the IP Address and Subnet mask must be suitably set:



7.4 Robot Homepage

Once the setup has been done, the Robot Homepage can be accessed from the PC by typing in the Robot IP address:

The screenshot shows the 'iRVision' section of the Robot Homepage. On the left is a navigation sidebar with the following items: **ROBOT Homepage** (Hostname: ROBOT, Robot No: E118087), **iRVision** (highlighted), PC iPendant, Current Robot Status, Active Program/Variables/Diagnostics, Robot Tools, Contact Information, and Note.

The main content area is titled 'iRVision' and is divided into three sections:

- Teaching:** Contains a button for 'iRVision Vision Setup' (with a pencil icon) and a description: 'Perform all the work necessary for teaching iRVision, such as initializing the camera settings and creating a vision process. You can also run tests of the vision process you created to confirm its operation.'
- Production Status:** Contains a button for 'iRVision Vision Runtime' (with a play icon) and a description: 'Display vision results and images during production.'
- Information and Settings:** Contains four buttons: 'Vision Devices' (with a camera icon), 'Vision Log' (with a document icon), 'Image Registers' (with a camera icon), and 'Vision Config' (with a gear icon). Each button has a corresponding description below it.
 - Vision Devices:** Display connection information about the iRVision devices connected to the robot controller.
 - Vision Log:** Display vision results and images recorded in the vision log.
 - Image Registers:** Display the images stored in the image register. The image register is an area for temporarily storing captured images.
 - Vision Config:** Configure the general iRVision system settings. In most cases, you don't have to change these settings, and you can use iRVision with the default configuration.

On the right side of the main content area, there are two additional boxes:

- Information for Beginners:** Explanations of iRVision basic concepts, setup procedures, and fundamental terminology. Please read this before using iRVision for the first time.
- Tutorials:** Materials for learning the basics of using iRVision. By following the tutorials, you can learn how to set up a typical application.

From this Homepage, sub-pages can be accessed giving access to the internal robot controller data:

The screenshot shows the 'Robot Tools' sub-page. The left sidebar is identical to the previous screenshot, but the 'Robot Tools' item is highlighted in blue.

The main content area is titled 'Robot Tools' and contains three large buttons:

- Comment Tool:** Represented by a speech bubble icon with two exclamation marks.
- Browser Tool:** Represented by a globe icon.
- Search Tool:** Represented by a magnifying glass icon.

For instance the Comment Tool which allows quick access to comments and data – the example below shows Numeric Registers:

Register	Comment	Value
R[1]	Continue Cycle	0
R[2]	In Cycle	0
R[3]	Speed Mode	0
R[4]	Fast OVRD	75
R[5]	Slow OVRD	50
R[6]	User Input	-1
R[7]	Vision Installed	1
R[8]	Vision Enabled	0
R[9]		0
R[10]	SPEED1	750
R[11]	SPEED2	750

Or the user can access the TP programs in the controller:

ROBOT Homepage

Hostname: ROBOT
Robot No: E118087

IRVision

PC iPendant

Current Robot Status

Active Program/
Variables/Diagnostics

Robot Tools

Contact Information

Note

Active Program Variables / Diagnostics

\$ —
\$ —

Variable files available on MD:

TP Program files available on MD:

Vision Data Files

✕ —
✕ —

Error/Diagnostic files (text) available on MD:

Other files available on MD:

Select a program:

WEB SERVER

Hostname: ROBOT
Robot No: E118087
File Name: /MD/INDEX_TP.HTM
Date: 19/09/23 Time: 14:52:28

TP Program files files available on MD:

[Home Page](#)

Binary format	ASCII format	Comment
_BCKED8-.TP	_BCKED8-.LS	
_BCKED9-.TP	_BCKED9-.LS	
_BCKEDT-.TP	_BCKEDT-.LS	
AAA_WSEX_V.TP	AAA_WSEX_V.LS	WS Ex TP Vision
AAA_WSTPK.TP	AAA_WSTPK.LS	WS TP Kazan 2019
AA_HOME.TP	AA_HOME.LS	Move Home
AA_OVRD.TP	AA_OVRD.LS	Set OVRD Fst Sio
ABORTIT.TP	ABORTIT.LS	ABORT PRODUCTION
CELLCAM.TP	CELLCAM.LS	CCRG grp:1
CHK_TOG.TP	CHK_TOG.LS	Toggle Chuck
COPY_PR.TP	COPY_PR.LS	
DSP_WEBP.TP	DSP_WEBP.LS	Display web page
GETDATA.TP	GETDATA.LS	Get PC Data
HAND_CLOSE.TP	HAND_CLOSE.LS	Close Gripper
HAND_OPEN.TP	HAND_OPEN.LS	Open Gripper
HAND_TOG.TP	HAND_TOG.LS	Toggle Gripper
IRV_RINGCHK.TP	IRV_RINGCHK.LS	

And see the program contents:

FANUC

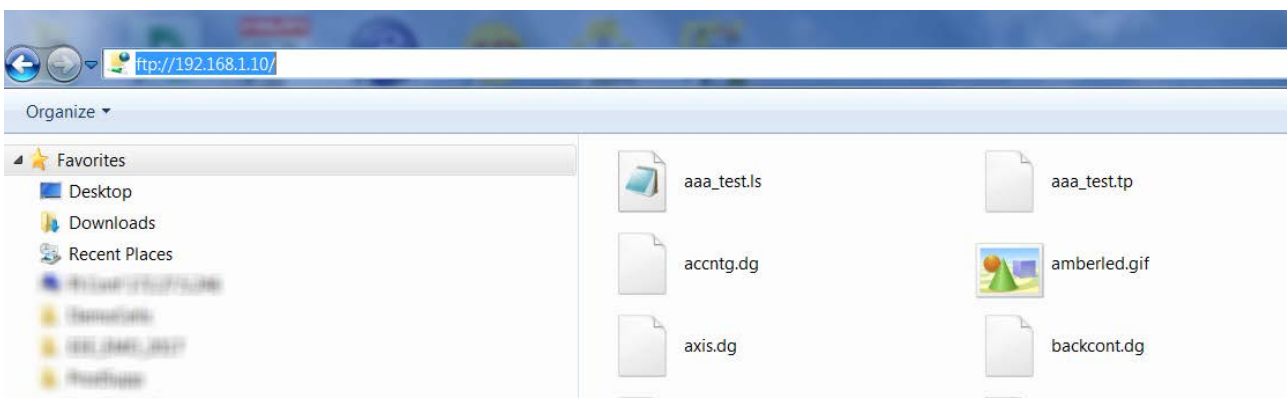
WEB SERVER
Hostname: ROBOT
Robot No: E118087
File Name: /MD/HAND_OPEN.LS
Date: 19/09/23 Time: 14:54:02

```
/PROG HAND_OPEN      Macro
/ATTR
OWNER                = MNEDITOR;
COMMENT              = "Open Gripper";
PROG_SIZE            = 180;
CREATE               = DATE 15-09-03  TIME 22:34:10;
MODIFIED             = DATE 15-09-03  TIME 22:34:10;
FILE_NAME            = ;
VERSION              = 0;
LINE_COUNT           = 3;
MEMORY_SIZE         = 672;
PROTECT              = READ_WRITE;
TCD:  STACK_SIZE     = 0,
      TASK_PRIORITY  = 50,
      TIME_SLICE     = 0,
      BUSY_LAMP_OFF  = 0,
      ABORT_REQUEST  = 0,
      PAUSE_REQUEST  = 0;
DEFAULT_GROUP       = *,*,*,*,*;
CONTROL_CODE        = 00000000 00000000;
/APPL
/MN
1:  LBL[1] ;
2:  RO[7:Open Gripper]=ON ;
3:  IF RO[8:Close Gripper]<>OFF,JMP LBL[1] ;
/POS
/END
```

7.5 File Transfer

After the connection between PC and Robot has been established, files can be transferred between them.

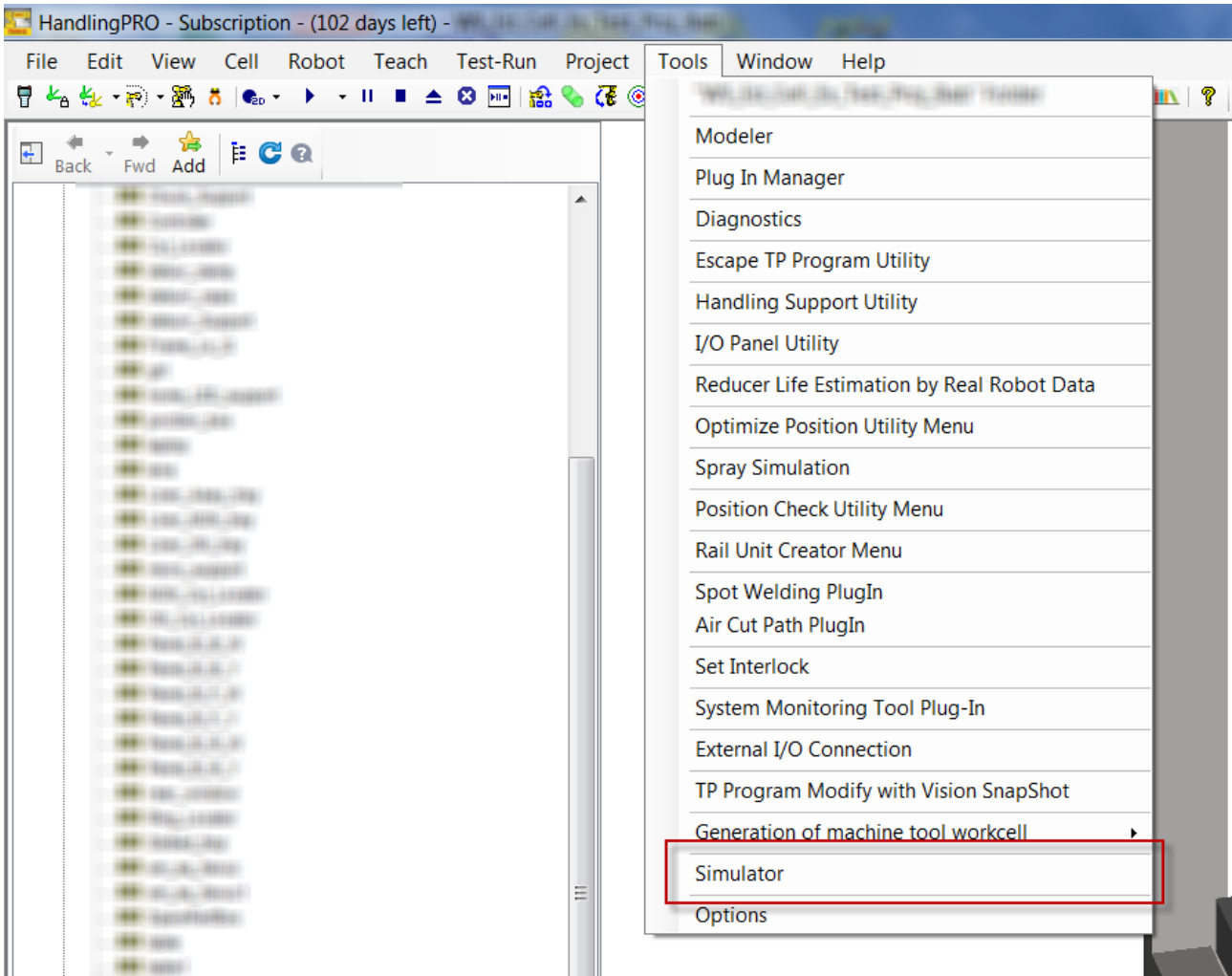
See example below – copy and paste of program ‘aaa_test.tp’ using simple ‘ftp://[robot_ip_address] – other software such as Filezilla can also be used.



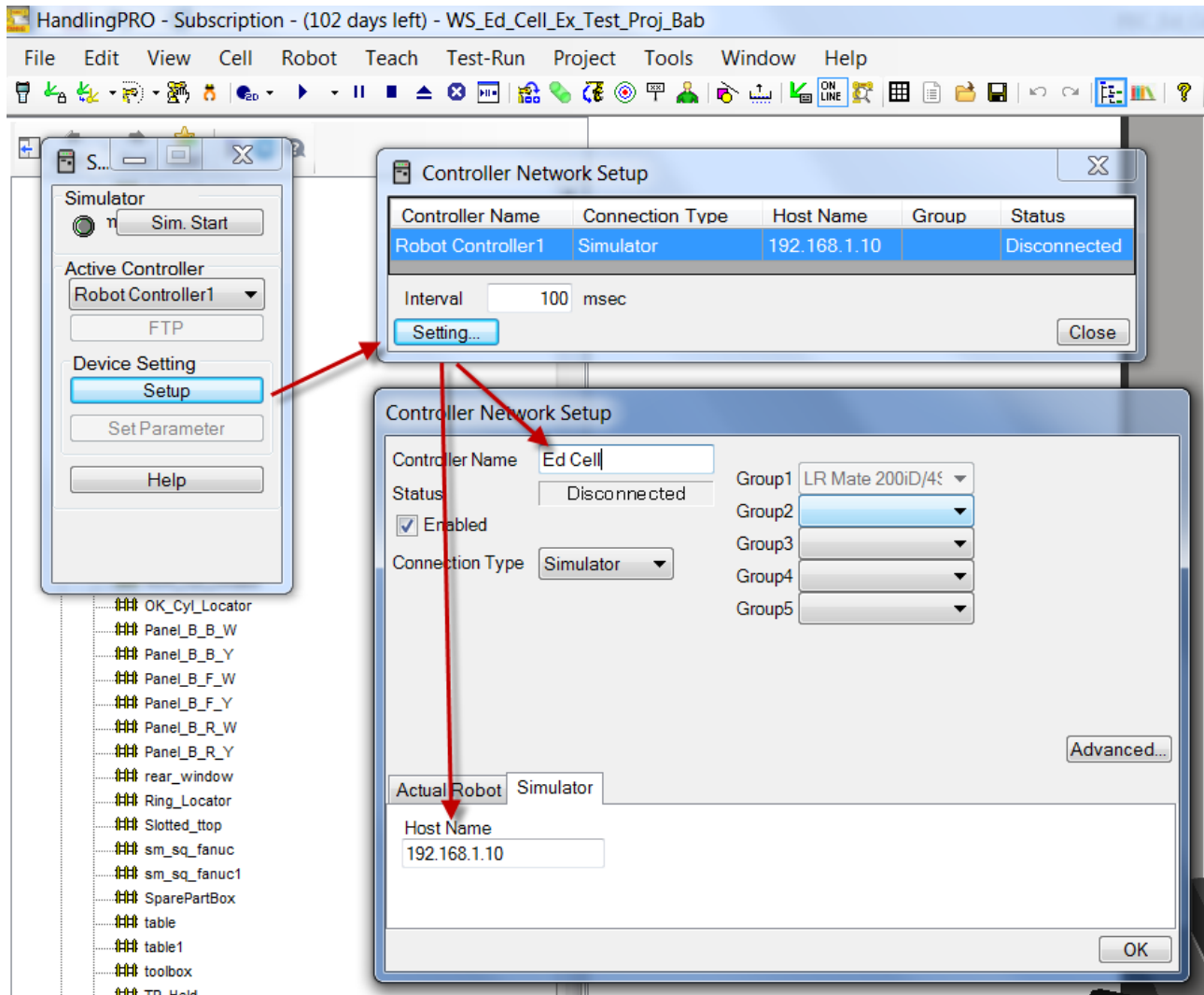
7.6 Roboguide Simulator / File Transfer

File transfer can also be set up to work between Roboguide and the real robot:

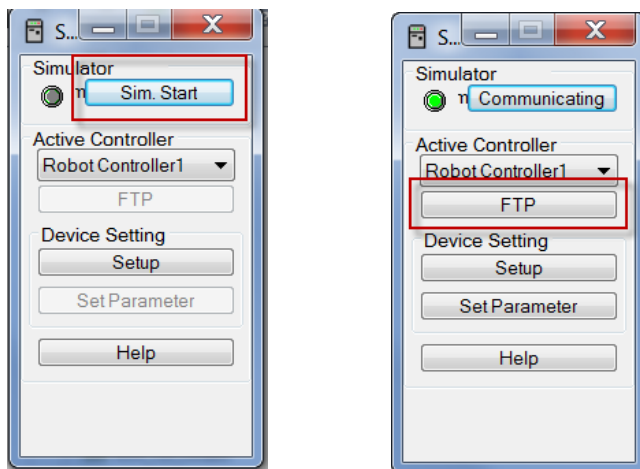
First select Tools > Simulator:



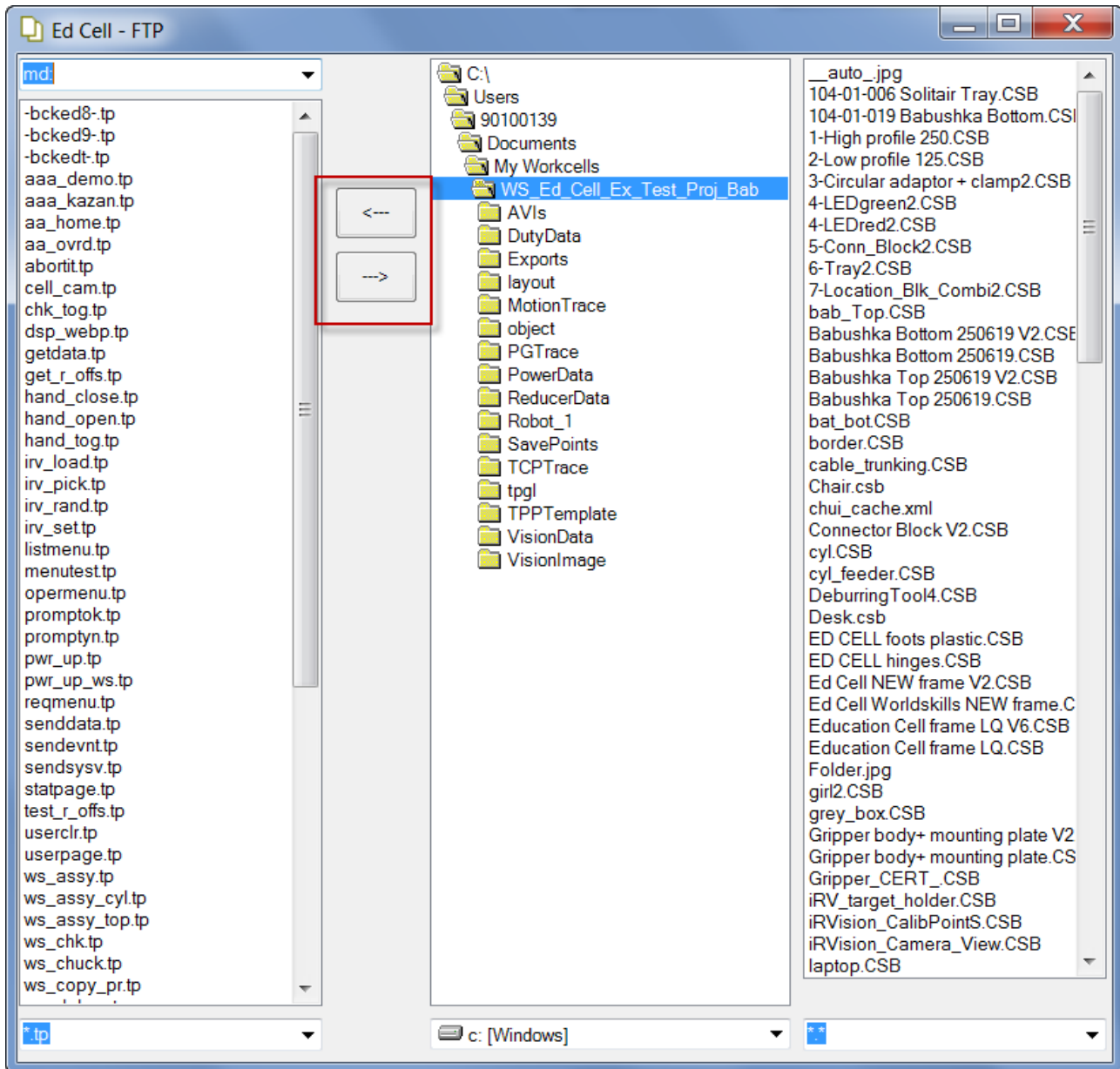
Then set up the Simulator to communicate with the real robot connected via Ethernet:



Once setup has been done, the Simulator function can be started and FTP selected:



Now files can easily be transferred between Roboguide and real robot:



Please note that this is not the only function of the Simulator function.

For example - when the Simulator is active, the robot in the Roboguide workcell will mimic the movement of the real robot. For more details please refer to the Roboguide help files.

8 List of Registers

8.1 Numeric registers

Registers are used to store settings to control the cell operation, and used by the program internally.

Register	Comment	Description	Default value
1	Continue Cycle	This register is set to 1 at start of "AAA_DEMO". If value is 1 then the program will loop continuously. If value is 0, program will stop at end of next full cycle. Value can be set manually or by the Menu screen described in earlier section.	1 to loop 0 to end
2	In Cycle	This register is set to 1 at start of "AAA_DEMO", and is set to 0 at end of program	1 when in cycle 0 at end of cycle
3	Speed Mode	This register is set via the Menu screen described earlier.	1 is fast mode, anything else slow.
4	Fast OVRD	This is value that will be used for Override in Fast Mode	75%
5	Slow OVRD	This is value that will be used for Override in Slow Mode	50%
6	User Input	Used by the Menu Utility to return the User's Choice of actions	1
7	Vision Installed	This should be set during software installation / setup.	1 is installed, anything else not installed
8	Vision Enabled	This is set by the Operation Menu to enable / disable Vision L/UL. Both R[7] and R[8] must be 1 for the vision section of the example program to run	1 is enabled, anything else not enabled
10 11 12	SPEED1 SPEED2 ACC	These are speeds and acceleration used for moves. They are set low to avoid frame shaking when on wheels. (If frame is fixed more securely, speed and acceleration could be increased)	750mm/sec 750mm/sec 75%

FANUC

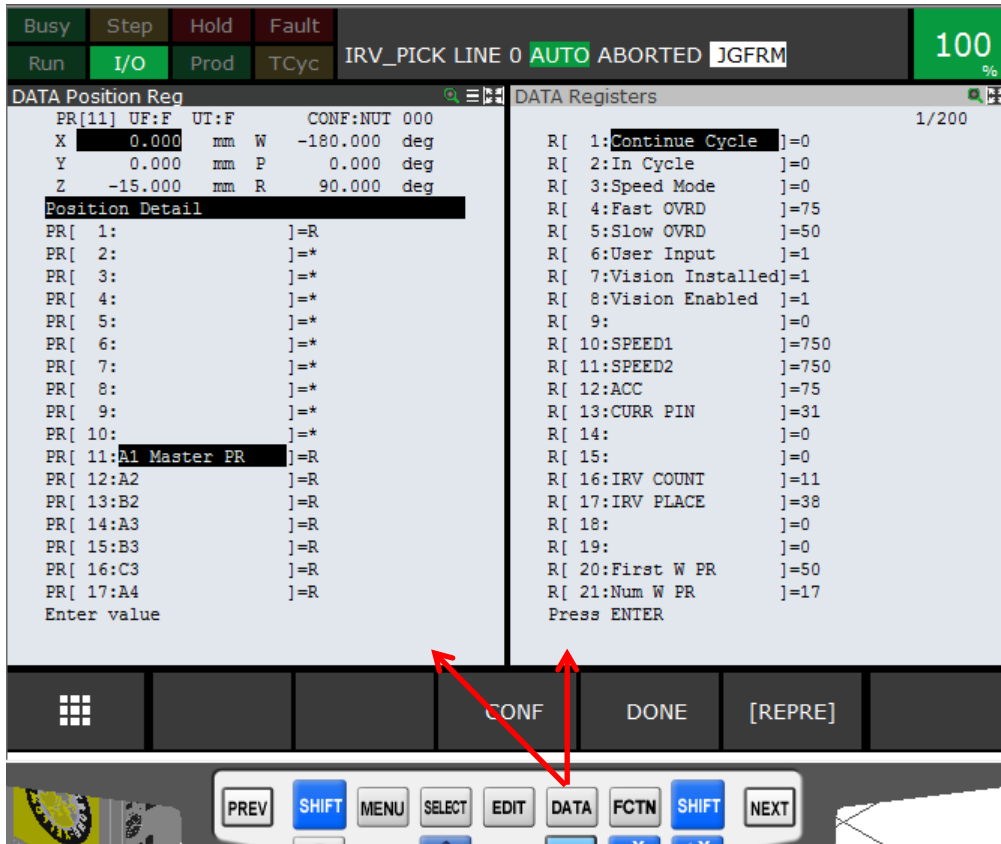
Register	Comment	Description	Default value
13	CURR PIN	This value is used internally by the program to keep track of the part number	n/a
16	IRV COUNT	Used to load 9 parts with Vision	n/a
100 101 102 103	ID_PICK_CYLINDER ID_PLACE_HOLE ID_REMOVE_CYLIND ID_EMPTY_PLATE	These values are used internally by the program to control the sequence of moves	n/a
105	Tmp_cyl_reg	This value is used internally by the program to set the status registers below	n/a
111 to 130	A1 in to -D4 in	These registers are used to store the status of the parts and holes. The values in these registers are linked to the Menu Displays using the <i>i</i> Pendant Controls – see later section	1 = occupied 0 = unoccupied

8.2 Position Registers

Position Registers are used to store positions

Position Register	Comment	Description
11 to 30	A1 to -D4	Locations of the holes in the tray. Note that these are calculated values, not taught values. PR[11] was taught, then the other PR[]s were calculated relative to it.
31 to 39	Mid 1 to Mid 9	'Random' positions in middle of board – not in holes – robot will find actual position using iRVision
40 41	ToolOff ToolOff1	These are Tool Offsets, used to create the motion above / to the pick and place locations.

The Registers can be displayed using the 'Data' button on the Teach Pendant:



9 Program Details

9.1 List of programs

The following programs are installed in the Education Cell:

Program	Comment	Description
AAA_DEMO	Example program	This is Main Example program – it must be selected before pressing 'Cycle Start'
AA_CHK	Check Start OK	Program to check robot at home and parts in correct position. Uses Menu Utility to display choices and confirmation to user – see later section for details. Also uses .STM file to display graphic of correct part position – see later section for details.
AA_HOME	Move Home	This program moves the robot to the home position. Note that it uses Joint definition position – so independent of any UFrame or UTool settings
AA-OVRD	Set OVRD Fst Slo	Set override to value specified in R[4], R[5] depending on value of mode R[3]
ABORTIT	ABORT PRODUCTION	Predefined system program – not used by this application
DSP_WEBP	Display Web Page	Macro installed by the 'Menu Utility' Option. This macro is used to display the Menu Screens.
GETDATA	Get PC Data	Predefined system program – not used by this application
HAND_CLOSE	Close Gripper	Program to Close Schunk Gripper (Note 1)
HAND_OPEN	Open Gripper	Program to Open Schunk Gripper (Note 1)
HAND_TOG	Toggle Gripper	Program to toggle gripper between Open and Close. This program is linked to the User Key 1 on the Teach Pendant – see details in later section
IRV_LOAD	Get Parts Vision	Pick parts from centre of board using vision and place in left hand side of board
IRV_PICK	Get 1 Part	Pick 1 part from centre of board using vision and place in left hand side of board – called from IRV_LOAD

FANUC

Program	Comment	Description
IRV_RAND	Put Parts Vision	Pick parts from Right hand side of board and place in semi-random positions on middle of board for robot to find using <i>iR</i> Vision
IRV_SET	IRV Setup Posn	Program containing suitable start position for <i>iR</i> Vision Camera Calibration
LISTMENU	LIST MENU MACRO	Macro installed by the 'Menu Utility' Option. This macro is used to display 3 choices to the user when the robot is not at HOME
OPERMENU	Entry Menu Macro	Macro installed by the 'Menu Utility' Option. This macro is not used in this application
PROMPTOK	Prompt Box OK	Macro installed by the 'Menu Utility' Option. This macro is not used in this application
PROMPTYN	Prompt Box Y N	Macro installed by the 'Menu Utility' Option. This macro is used to confirm the choice by the user when the robot is not at HOME
REQMENU SENDDATA SENDEVNT SENDSYSV STATPAGE	Request PC Menu Send PC Data Send PC Event Send PC Sysvar StatusMenu Macro	Predefined system programs – not used by this application
S_SET1_PR	Solit Setup 1PR	Set coordinates of one Position Register for one hole
S_SETUP_PRS	Solit Setup PRs	Set coordinates of all Position Register for all holes
S_SOLIT_RESET	Reset Status Rs	Reset registers showing part status to start conditions
S_UNLD1	Solit Unload 1	Make one move, jumping over a piece and then moving the jumped-over piece to the other half of the board.
S_UNLDALL_L	Solit Unl All L	Sequence for the left side of the board, moving pieces to the right side
S_UNLDALL_R	Solit Unl All R	Sequence for the right side of the board, moving pieces to the right side
S_UNLDLST	Solit Unld Last	Make one move – for first or last pieces
USERCLEAR USERPAGE	Clear User Page Show User Page	Macros installed by the 'Menu Utility' Option. These macros are not used in this application
ZERO	Move to Zero	Move all axes to zero



Note 1) In order for the Roboguide Simulation to work correctly picking and placing the virtual parts, the HAND_CLOSE and HAND_OPEN .TP programs must be replaced with Simulation programs. Please refer to actual Roboguide cell for details

9.2 Sample Program Listings

This is listing of main program "AAA_DEMO":

```
/PROG AAA_DEMO
1: !FANUC EUROPE EDUCATION CELL ; << Remark
2: !Example Solitaire Program ; << Remark
3: !with optional vision ;
4: CALL AA_CHK ; << Call program to check Start OK
5: R[1:Continue Cycle]=1 ; << Set Register to run continuously
6: R[2:In Cycle]=1 ; << Set Register to show status
7: CALL AA_OVRD ; << Call program to set Override %
8: CALL HAND_OPEN ; << Make sure gripper is open
9: LBL[1] ; << Label to loop up to if needed
10: CALL S_SOLIT_RESET ; << Reset status registers
11: CALL S_UNLDALL_L ; << Sequence to move pieces from Left to Right side of board
12: WAIT 2.00(sec) ; << Short delay
13: IF R[7:Vision Installed]<>1,JMP LBL[2] ; << Check if use Vision or not
14: IF R[8:Vision Enabled]<>1,JMP LBL[2] ; << Check if use Vision or not
15: CALL IRV_RAND ; << Move parts from Right of board to Middle
16: WAIT 2.00(sec) ; << Short delay
17: CALL IRV_LOAD ; << Pick parts from Middle of board using Vision and put in Left Side
18: WAIT 2.00(sec) ; << Short delay
19: JMP LBL[3] ; << Jump over next section
20: LBL[2] ; << Label for Jump
21: CALL S_UNLDALL_R ; << Sequence to move pieces from Right to Left side of board
22: LBL[3] ; << Label for Jump
23: WAIT 2.00(sec) ; << Short delay
24: IF R[1:Continue Cycle]=1,JMP LBL[1] ; << Repeat if register 1 is equal to 1
25: R[2:In Cycle]=0 ; << Set register to show status at end of program
/POS
/END /END
```

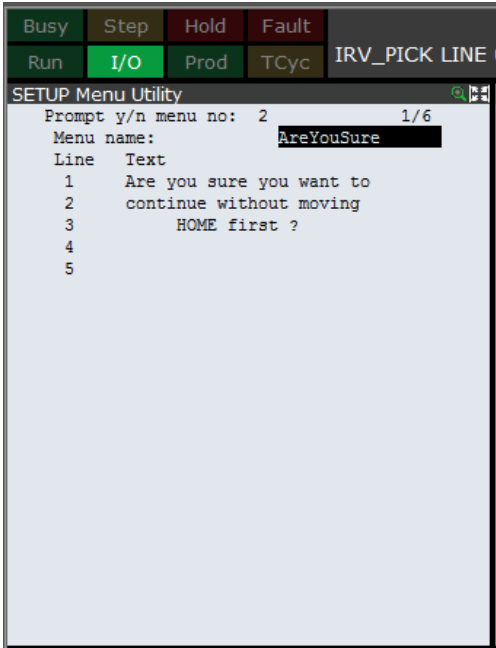
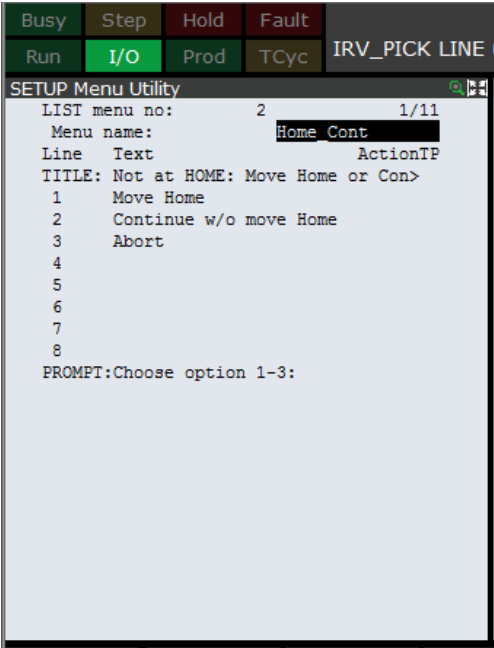
This is listing of program "AA_CHK":

```
/PROG AA_CHK
  1: !Check Start Conditions OK ;          << Remark
  2: ;
  3: !Check at HOME ;                    << Remark
  4: IF DO[101:HOME Signal]=ON,JMP LBL[10] ; << Check HOME signal - see later section for detail
  5: LBL[1] ;
  6: R[6:User Input]=0 ;
  7: CALL LISTMENU(2,6) ;                << Call Menu Utility to display User Menu 2, result in R[6]
  8: SELECT R[6:User Input]=1,JMP LBL[3] ;
  9:     =2,JMP LBL[5] ;
 10:     ELSE,JMP LBL[2] ;
 11: LBL[2] ;
 12: ABORT ;
 13: JMP LBL[10] ;
 14: LBL[3] ;
 15: CALL AA_HOME ;
 16: JMP LBL[10] ;
 17: LBL[5] ;
 18: R[6:User Input]=0 ;
 19: CALL PROMPTYN(2,6) ;                << Call Menu Utility to display Prompt Box 2, result in R[6]
 20: IF R[6:User Input]=1,JMP LBL[10] ;
 21: JMP LBL[1] ;
 22: LBL[10] ;
 23: ;
 24: !Check parts OK ;
 25: CALL S_SOLIT_RESET ;
 26: R[6:User Input]=0 ;
 27: !Display Check Page ;
 28: CALL DSP_WEBP(3) ;                  << Call Menu Utility to display User Status / Confirmation Menu
 29: WAIT R[6:User Input]<>0 ;
 30: IF R[6:User Input]=1,JMP LBL[20] ;
 31: ABORT ;
 32: LBL[20] ;
 33: !Display Run Page ;
 34: CALL DSP_WEBP(4) ; << Call Menu Utility to display User Status / Operation Menu
/END
```

10 User Interface Setup

10.1 Menu Utility Setup

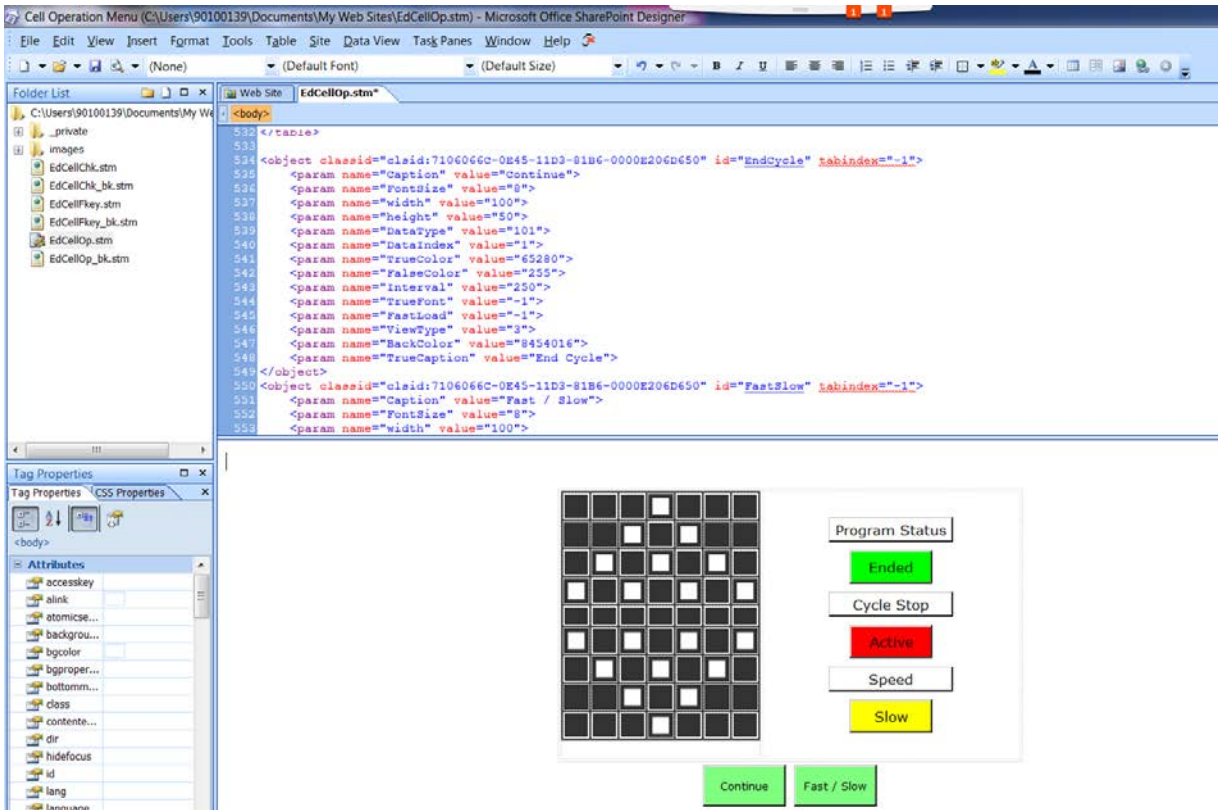
The User Menu and Prompt at the start of the AA_CHK program have been set up using the Menu Utility. See below for screenshots.



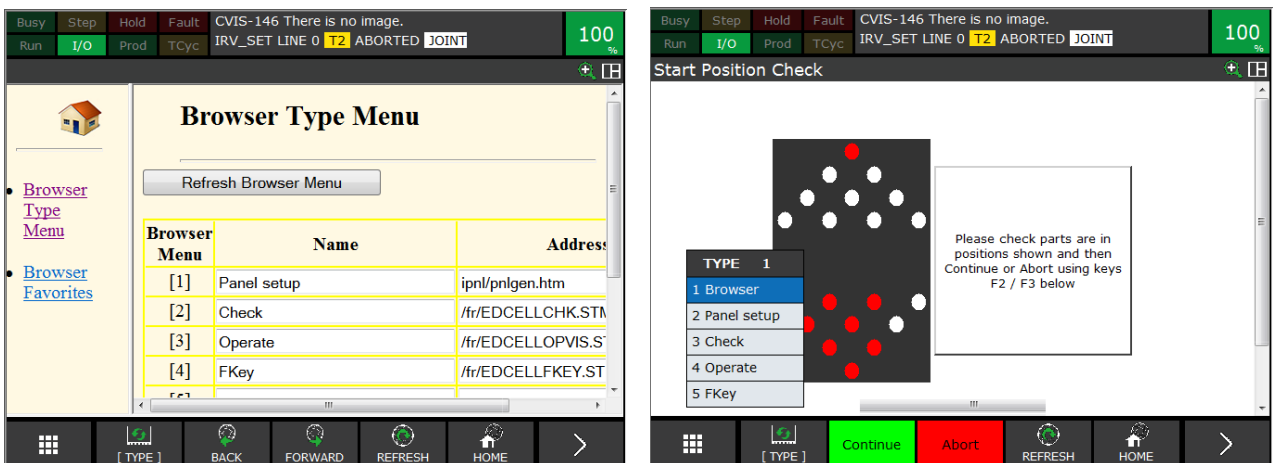
Once these menus have been set up, they can be called using the predefined macros "LISTMENU" and "PROMPTYN"

10.2 HTML Screens

The Status / Menu Screens were created using MS Sharepoint Designer 2007:



Then the files were loaded into the controller and added to the browser favourites menu:



So that they can be displayed manually or using the CALL DSP_WEBP(3) program, which is included in the Menu Utility

10.3 HTML Listing Extract

This is extract of listing for "EdCellOp.stm"

```
<head>           Title of the Page
<meta http-equiv="Content-Language" content="en-gb">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Education Cell Menu</title>
<style type="text/css">
.style1 {
    border-color: #FFFFFFF;
    border-width: 0;
    background-color: #333333;
}
.style2 {
    text-align: center;
}
</style>
</head>
<body>
<div class="style2">

<table style="width: 41%"> Table with two columns, left column for board layout, right column for
status boxes
    <tr>
        <td style="width: 235px">

<table style="width: 140px; height: 180px; float: right;" class="style1">
    <tr> Table with 7 columns and 9 rows to show board positions
        <td></td>
        <td></td>
        <td></td>
```

FANUC

```
<td style="width: 30px">                Definition of one board position
<object classid="clsid:71060668-0E45-11D3-81B6-0000E206D650" id="Sol1"
style="width: 20px; height: 20px">
    <param name="Caption" value="">
    <param name="FontSize" value="14">
    <param name="width" value="20">
    <param name="height" value="20">
    <param name="DataType" value="101"> State linked to Register
    <param name="DataIndex" value="121"> Register number 121
    <param name="TrueColor" value="255"> True Colour = Red
    <param name="FalseColor" value="16777215"> False colour = Light grey
    <param name="Interval" value="250">
    <param name="TrueFont" value="-1">
    <param name="FastLoad" value="-1">
    <param name="Border" value="1">
    <param name="ViewType" value="1">
</object>
</td>
<td></td>
<td></td>
<td style="width: 49px"></td>
</tr>
```

Repeat for all other board positions up to 20

```
</table>                Set up objects in right column
</td> Add label
<td class="style2">
<object classid="clsid:7106065C-0E45-11D3-81B6-0000E206D650" id="FRIPLabel1"
style="height: 30px; width: 150px">
    <div class="style2">
        <param name="Caption" value="Program Status">
```

FANUC

```
<param name="FontSize" value="10">
<param name="width" value="150">
<param name="height" value="30">
<param name="DataType" value="100">
<param name="DataIndex" value="">
<param name="Interval" value="250">
<param name="TrueFont" value="-1">
<param name="FastLoad" value="-1">
```

```
</div>
```

```
</object>
```

```
<br> Add Lamp Object to show status of program
```

```
<object classid="clsid:71060668-0E45-11D3-81B6-0000E206D650"
id="FRIPToggleLamp4" style="height: 40px">
```

```
<div class="style2">
```

```
<param name="Caption" value="Ended"> Set text for False
```

```
<param name="FontSize" value="10">
```

```
<param name="width" value="100">
```

```
<param name="height" value="40">
```

```
<param name="DataType" value="101"> Link to Register
```

```
<param name="DataIndex" value="2"> Register 2
```

```
<param name="TrueColor" value="255">
```

```
<param name="FalseColor" value="65280">
```

```
<param name="Interval" value="250">
```

```
<param name="TrueFont" value="-1">
```

```
<param name="FastLoad" value="-1">
```

```
<param name="TrueCaption" value="In Cycle"> Set Text for True
```

```
</div>
```

```
</object>
```

```
<br>
```

```
Repeat for other lamps
```

```
</table>
```

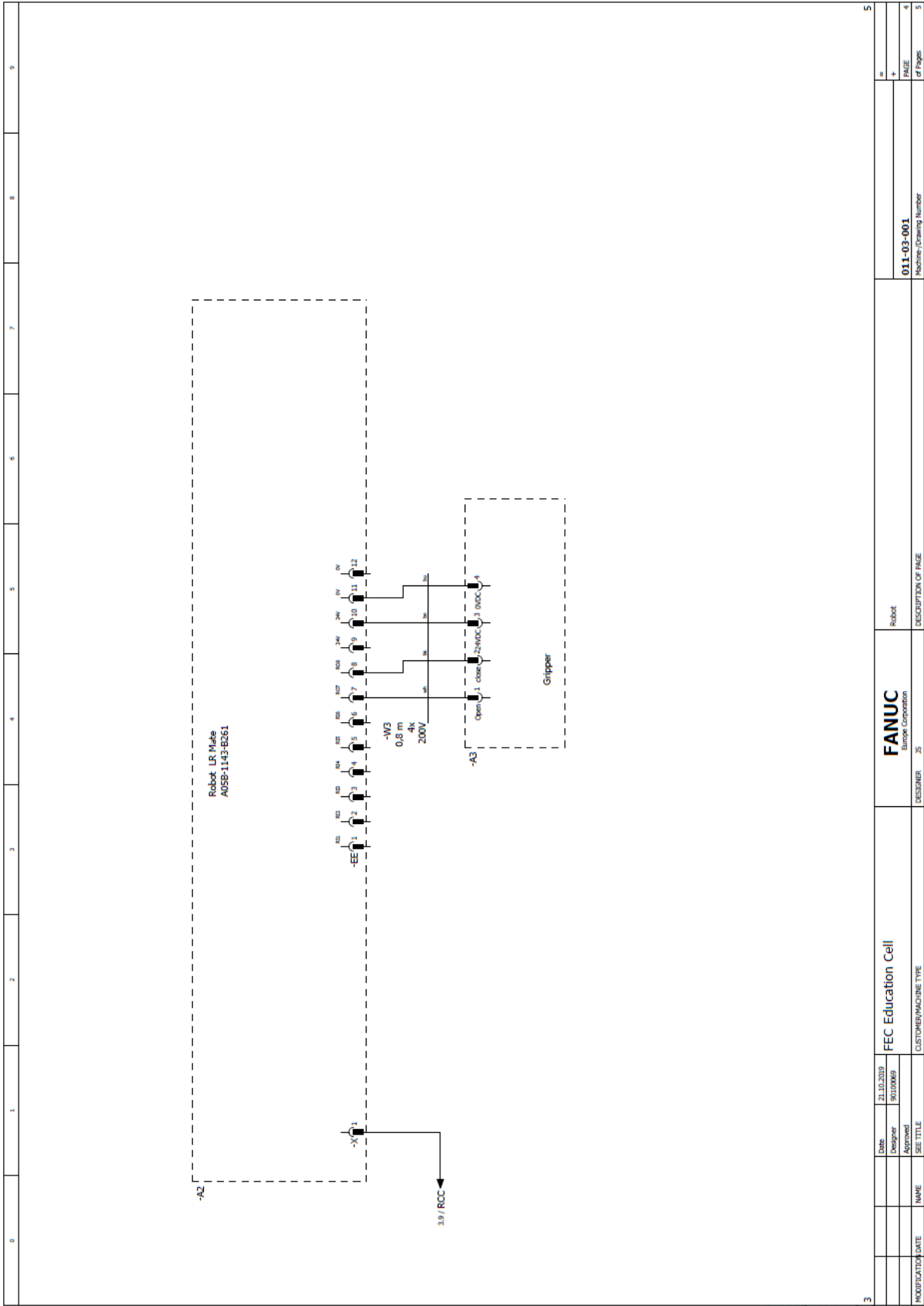
Add TP Key labels / functions objects

```
<object classid="clsid:7106066C-0E45-11D3-81B6-0000E206D650" id="EndCycle" tabindex="-1">
  <param name="Caption" value="Continue">
  <param name="FontSize" value="8">
  <param name="width" value="100">
  <param name="height" value="50">
  <param name="DataType" value="101"> Sets Register
  <param name="DataIndex" value="1"> Register 1
  <param name="TrueColor" value="65280">
  <param name="FalseColor" value="255">
  <param name="Interval" value="250">
  <param name="TrueFont" value="-1">
  <param name="FastLoad" value="-1">
  <param name="ViewType" value="3">
  <param name="BackColor" value="8454016">
  <param name="TrueCaption" value="End Cycle">
</object>
```

Repeat for other Key

```
</p>
<p>&nbsp;</p>
</div>
</body>
</html>
```


11.2 Gripper



12 Bill of Materials

Item	Description	Education Cell BOM (with Vision)	Comments
1	Robot, incl. Controller, Cables, TP etc	ER4IA-30P-M-ER1/0 (ER-4iA)	New ER-4iA Robot for Education Market
2	iRVision option - Camera	A05B-1426-K001	KOWA Camera
	Camera Cable	A05B-2680-J340	
	Lens	LX-1-RO-VI-16-0-0008	
3	Frame, incl brackets, Parts box etc	LX-1-RO-ZZ-17-0-0002 (Silver) LX-1-RO-ZZ-17-0-0003 (Black)	Also includes 'Solitaire' board & parts, and transport pallet
4	Schunk Gripper - EGP40	LX-P-RO-TS-15-0-0001	
5	Gripper Fingers	LX-1-RO-TS-30-1-0001 (x 2)	
6	Gripper Mounting Plates	LX-0-RO-ZZ-09-10021 + LX-0-RO-ZZ-09-10022	
7	EE Gripper Connection Cable	LX660-4060-T901/L800R0	
8	PILZ safety switch	LX-1-RO-ZZ-34-0-0001	PILZ PSEN1.1p-20
9	Cables & fuses safety switch connection kit	LX-1-RO-ZZ-34-0-0004	
10	Power Cable	LX-0-RO-ZZ-08-1-0007	Includes moulded on 220-240V Power Plug

13 Technical & Transport Data

Technical data:	
Power rating	230V 16A single phase
Power consumption	1 KW/h
Connection	Schuko plug (German style)
Air supply	None
Air consumption	None
Installation size	Base 1,3 x 0,7 m Height 1,8m
Weight	170kg

Transport data	
Size	Base 1,3 x 0,7 m Height 1,8m
Weight	170kg
Packing	Bubble foil
Handling	Delivered on Pallet – When removed from pallet on wheels

14 Version

Document title:

Educational Package

Technical Documentation: ER-4iA Robot with R-30iB Mate Plus Controller

V5.1	Conversion from R-30iB Plus LR-Mate 200iD Sliding door type frame to R-30iB Plus ER-4iA with Hinged door type frame. New HMI screens.
V5.2	IoT / Ind 4.0 Section added. Safety Section incl. DCS password advice added.
V5.3	Small changes to iRVision for s/w version 9.3

Version: V5.3 18-12-2019 N. Ramsden