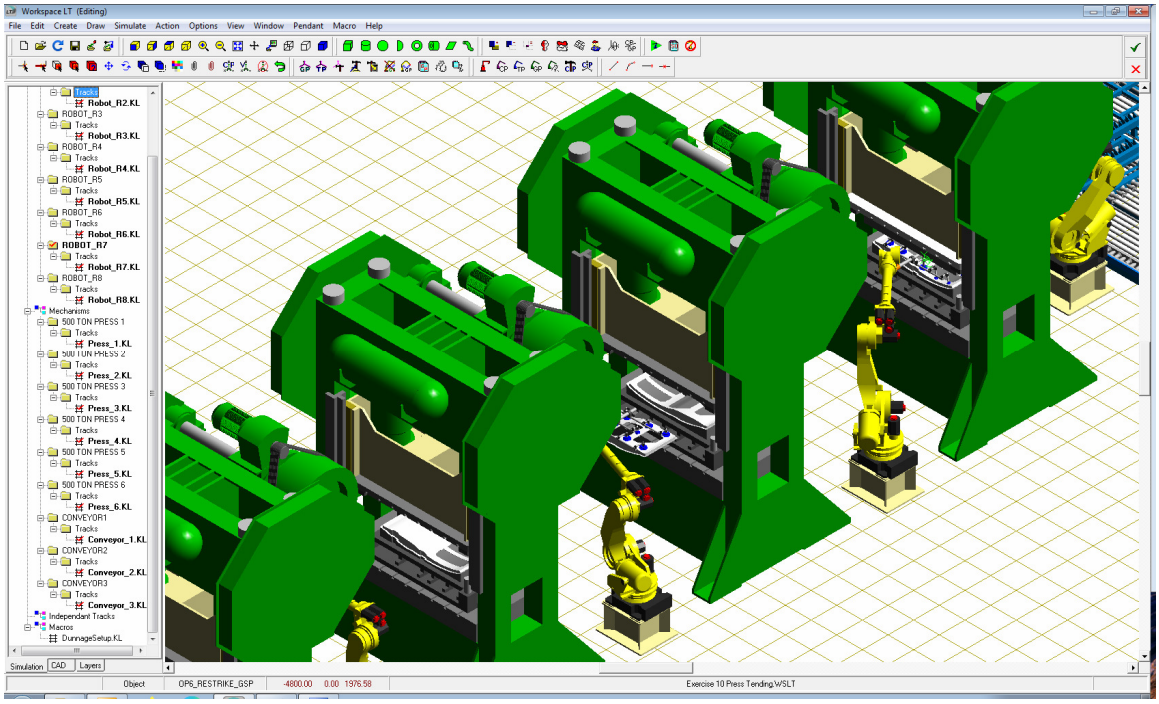


Exercise 10 - Press Tending

Objectives:

- Create a complex "press tending" simulation (loading and unloading parts for a press).
- Understanding Logic statements.
- Using vacuum style end of arm robot tooling.
- Using assigned variables and a Constants.
- Using routines to assign various robot tasks.
- Use of Inputs and outputs to control the simulation flow.
- Robot optimizing using throughput analysis.
- Pack out techniques, loading completed parts into bins for transport.
- Creating a Robotic simulation, avi.

Materials;

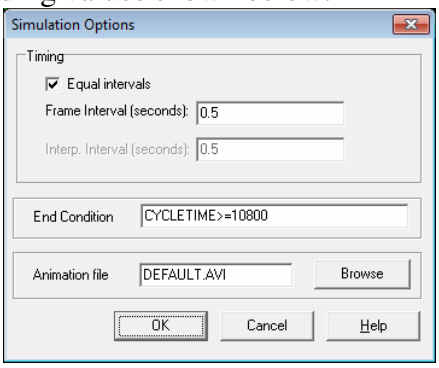
- Workspace LT© simulation software.
- Workspace LT© project file "Exercise 10 Press Tending.wslt".
- Manual "Workspace LT© User Guide.pdf".

Helpful Hint;

Before starting this exercise, review previous exercises and the User Guide sections;

- 3.6 Simulation/CAD/Layers View
- 4.20 Using Layers In Your Project
- 5.5 Auxiliary Axes
- 5.17 Testing For Collisions
- 6. The Karel Language
  - 6.3 Comments and Workspace commands
  - 6.4.5 Variable Types And Constants
  - 6.7.1 INTEGER and REAL functions
- 6.5 System Variables
- 7. Karel Robot Movement Commands
- 8. Karel Communications And Signals
  - 8.2.1 DIN..DOUT
- 9.1.1 IF ... ENDIF
- 9.2.2 Repeat..Until
- 9.2.3 WHILE ... ENDWHILE
- 9.3 Execution Control
- 9.3.1 Delay
- 15 Throughput Statistics

1) Procedure: Create tracks for the mechanical presses, robots and conveyors.
   a) Open Workspace LT simulation software.
   b) Open the project file "Exercise 10 Press Tending.WSLT".
   c) Add tracks for "500 TON PRESS 1" thru to "500 TON PRESS 6" (Programs Appendix A - Robot Tracks).
   d) Add tracks for "ROBOT_1" thru to "ROBOT_2" (Programs Appendix B - Press Tracks).
   e) Add tracks for "CONVEYOR1" thru to "CONVEYOR3" (Programs Appendix C - Conveyor Tracks).
   f) Add the macro track "DonnageSetup.KL.
   g) Add Teach points to the press and conveyor #.KL files (Programs Appendix B - Press Tracks and Programs Appendix C - Conveyor Tracks). Using the mouse , hover over the track name Press_1.KL, mouse click the button on the right side of the mouse. From the popup menu, using the mouse select "View TP's. The text editor will open for the teach point file Press_1#.KL. Add the syntax for each teach point. When complete, close the editor. Complete this for each press and conveyor track.
   h) Comment track programs for ease of understanding by others.
   i) Save the project model .
   j) Run the simulation and debug tracks if required. From the upper tool bar use the mouse to select "Simulate". Using the mouse again select "Simulation Options...". In the "Simulation Options" menu below change each menu field value to the corresponding values shown below.



Important Note:

1. Before running the simulation a second time, reload the model.

or

   a. Before running the simulation a second time, Run the Macro DunnageSetup.KL to correctly arrange the bins on the pack out conveyors at the end of the press line. To run this macro, use the mouse to select "Macro" located on the upper tool bar. A drop down menu with the list of available macros will appear, select the DunnageSetup macro. the "Confirm menu will appear, select "No".

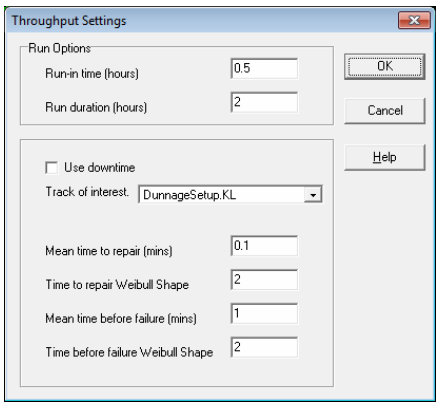
b. Before running the simulation a second time, Delete all parts with the name "DOOR". These parts will be attached to either the EOAT of R8 or be attached to any of the bins.

2. BEFORE doing any changes to the model or tracks, reload the model.

k) Save the project model .

2) Procedure: Throughput optimization.

a) Set throughput options. Using the mouse select Options found on the upper tool bar. From the drop down menu use the mouse to select Throughput Options then select Throughput Settings... . The Throughput Settings menu box will appear. Change "Run time (hours)" to 0.5. Change "Run duration (hours)" to 2. Using the mouse select OK .



b) From the upper tool bar use the mouse to select Simulate. A drop down menu will appear, using the mouse select Run Simulation Blind. Wait for the simulation to end. From the previous procedure, the "End condition was set to " "CYCLETIME>=10800". After 10800 seconds the simulation will stop. Using "Run Simulation Blind" will calculate the simulation run time without updating the modeler, thus the simulation will run faster than actual time (warp speed).

c) The Enter name of file for job data pop-up menu will appear. Using the mouse select OK to accept the default values. The "Job_Summary (Read Only).TXT" text editor box will appear. Review the contents and then using the mouse select to close the text editor. As experienced in "Exercise 8" other files will open one at a time for each track file which is active. Review the contents of each file and then using the mouse select to close each file. When each of the files have been reviewed, the Gant chart of the process will appear. Review the contents and then using the mouse select to close the Gant chart or print the chart. All files with exception to the Gant chart will be saved under the project directory/stats.

d) Determine if there are any bottlenecks and improve the throughput by altering tracks. The required cycle time for each part is 20 seconds or as close to this time as possible.

e) Once all avenues are explored to reduce cycle time of each part apply a staging table between R2 and R3 robots;
   i) Insert model "Staging Table". The table should appear between R2 and R3.
   ii) Place the stage table in a convenient position between R2 and R3 if required. Moving the part located on the table will locate the table.
   iii) Edit both track programs for R2 and R3 and compare the data from previous simulation runs. Review "Exercise 8" if required.
f) Save the project model  when complete.

3) Procedure: Create an AVI.
   a)

Appendix A - Press Tracks

PROGRAM Press_1

```
--  Workspace LT KAREL 2 Program for 500 TON PRESS 1

VAR
  -- TEACHPOINT DECLARATIONS
  RAM_ADJUST : AUXPOS
  RAM_DWN : AUXPOS
  RAM_UP : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Press 1 requests R1 load Blank
  --! SIGNALDEF DOUT[2],TRACK     -- Press 1 requests R2 unload part Draw

  --! SIGNALDEF DIN[1],Robot_R1.KL,1  -- R1 loaded part Blank and clear
  --! SIGNALDEF DIN[2],Robot_R2.KL,1  -- R2 unloaded part Draw and clear

  --  Throughput Declarations
  -- ! LABELUSAGE ‘ROBOT_MOTION’,2
  -- ! LABELUSAGE ‘PRESS_MOTION’,3
  -- ! LABELUSAGE ‘CONVEYOR_MOTION’,4
  -- ! LABELUSAGE 'VACUUM',5

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Press_1#
  -- ! MakeInvisible,2,'OP1_ BLANK','OP1_DRAW'
  WITH $MOTYPE=JOINT
    MOVE AUX TO RAM_UP
Repeat
  -- Request R1 load blank
  DOUT[1]=ON
  --! LABELEVENT 'Wait for R1 loaded and clear',0
  WAIT FOR DIN[1]=ON
  DOUT[1]=OFF
  --! LABELEVENT 'Press 1 Cycle',3
  MOVE AXIS 1 BY 180.00
  -- ! MakeInvisible,1,'OP1_ BLANK'
  -- ! MakeVisible,1,'OP1_DRAW'
  MOVE AXIS 1 BY 180.00
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R2  unloaded and clear',0
  WAIT FOR DIN[2]=ON
  DOUT[2]=OFF
```

```
  --! ENDOFJOB
UNTIL FALSE
END Press_1
```

-- Press_1#.KL, Teach points

```
RAM_ADJUST[2] = -112
-- END RAM_ADJUST
RAM_DWN[1] = 180
RAM_DWN[2] = -112
-- END RAM_DWN
RAM_UP[1] = 0
RAM_UP[2] = -112
-- END RAM_UP
```

PROGRAM Press_2

```
--  Workspace LT KAREL 2 Program for 500 TON PRESS 2

VAR
  -- TEACHPOINT DECLARATIONS
  RAM_ADJUST : AUXPOS
  RAM_DWN : AUXPOS
  RAM_UP : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Press 1 requests R3 load Draw
  --! SIGNALDEF DOUT[2],TRACK     -- Press 1 requests R4 unload part Trim

  --! SIGNALDEF DIN[1],Robot_R3.KL,1  -- R3 loaded part Draw and clear
  --! SIGNALDEF DIN[2],Robot_R4.KL,1  -- R4 unloaded part Trim and clear

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Press_2#
  -- ! MakeInvisible,2,'OP1_DRAW2','OP2_TRIM'
  WITH $MOTYPE=JOINT
    MOVE AUX TO RAM_UP
Repeat
  -- Request R3 load blank
  DOUT[1]=ON
  --! LABELEVENT  'wait for R3 loaded and clear',0
  WAIT FOR DIN[1]=ON
  DOUT[1]=OFF
  --! LABELEVENT 'Press 2 Cycle',3
  MOVE AXIS 1 BY 180.00
  -- ! MakeVisible,1,'OP2_TRIM'
  -- ! MakeInvisible,1,'OP1_DRAW2'
```

```
  MOVE AXIS 1 BY 180.00
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R4  unloaded and clear',0
  WAIT FOR DIN[2]=ON
  DOUT[2]=OFF
UNTIL FALSE
END Press_2
```

-- Press_2#.KL, Teach points

```
RAM_ADJUST[2] = -112
- - END RAM_ADJUST
RAM_DWN[1] = 180
RAM_DWN[2] = -112
-- END RAM_DWN
RAM_UP[1] = 0
RAM_UP[2] = -112
-- END RAM_UP
```

PROGRAM Press_3

```
--  Workspace LT KAREL 2 Program for 500 TON PRESS 3

VAR
  -- TEACHPOINT DECLARATIONS
  RAM_ADJUST : AUXPOS
  RAM_DWN : AUXPOS
  RAM_UP : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Press request part load from R4
  --! SIGNALDEF DOUT[2],TRACK     -- Press request part unload from R5

  --! SIGNALDEF DIN[1],Robot_R4.KL,2  -- R4 loaded part and clear
  --! SIGNALDEF DIN[2],Robot_R5.KL,1  -- R5 unloaded part and clear

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Press_3#
  -- ! MakeInvisible,2,'OP1_DRAW','OP3_PIERCE'
  WITH $MOTYPE=JOINT
    MOVE AUX TO RAM_UP
Repeat
  -- Request R4 load Trim
  DOUT[1]=ON
  --! LABELEVENT  'Wait for R4 loaded and clear',0
```

```
  WAIT FOR DIN[1]=ON
  DOUT[1]=OFF
  --! LABELEVENT 'Press 3 Cycle',3
  MOVE AXIS 1 BY 180.00
  -- ! MakeVisible,1,'OP3_PIERCE'
  -- ! MakeInvisible,1,'OP1_DRAW'
  MOVE AXIS 1 BY 180.00
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R5  unloaded and clear',0
  WAIT FOR DIN[2]=ON
  DOUT[2]=OFF
UNTIL FALSE
END Press_3
```

-- Press_3#.KL, Teach points

```
RAM_ADJUST[2] = -112
-- END RAM_ADJUST
RAM_DWN[1] = 180
RAM_DWN[2] = -112
-- END RAM_DWN
RAM_UP[1] = 0
RAM_UP[2] = -112
-- END RAM_UP
```

PROGRAM Press_4

```
--  Workspace LT KAREL 2 Program for 500 TON PRESS 4

VAR
  -- TEACHPOINT DECLARATIONS
  RAM_ADJUST : AUXPOS
  RAM_DWN : AUXPOS
  RAM_UP : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Press request part load from R5
  --! SIGNALDEF DOUT[2],TRACK     -- Press request part unload from R6

  --! SIGNALDEF DIN[1],Robot_R5.KL,2  -- R5 loaded part and clear
  --! SIGNALDEF DIN[2],Robot_R6.KL,1  -- R6 unloaded part and clear

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Press_4#
  -- ! MakeInvisible,2,'OP3_PIERCE1','OP4_PIERCE'
```

```
  WITH $MOTYPE=JOINT
    MOVE AUX TO RAM_UP
Repeat
  -- Request R1 load blank
  DOUT[1]=ON
  --! LABELEVENT  'Wait for R5 loaded and clear',0
  WAIT FOR DIN[1]=ON
  DOUT[1]=OFF
  --! LABELEVENT 'Press 4 Cycle',3
  MOVE AXIS 1 BY 180.00
  -- ! MakeVisible,1,'OP4_PIERCE'
  -- ! MakeInvisible,1,'OP3_PIERCE1'
  MOVE AXIS 1 BY 180.00
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R6  unloaded and clear',0
  WAIT FOR DIN[2]=ON
  DOUT[2]=OFF
UNTIL FALSE
END Press_4
```

-- Press_4#.KL, Teach points

```
RAM_ADJUST[2] = -83.8
-- END RAM_ADJUST
RAM_DWN[1] = 180
RAM_DWN[2] = -83.8
-- END RAM_DWN
RAM_UP[1] = 0
RAM_UP[2] = -83.8
-- END RAM_UP
```

PROGRAM Press_5

```
--  Workspace LT KAREL 2 Program for 500 TON PRESS 5

VAR
  -- TEACHPOINT DECLARATIONS
  RAM_ADJUST : AUXPOS
  RAM_DWN : AUXPOS
  RAM_UP : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Press request part load from R6
  --! SIGNALDEF DOUT[2],TRACK     -- Press request part unload from R7
```

```
  --! SIGNALDEF DIN[1],Robot_R6.KL,2  -- R6 loaded part and clear
  --! SIGNALDEF DIN[2],Robot_R7.KL,1  -- R7 unloaded part and clear

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Press_5#
  -- ! MakeInvisible,2,'OP4_PIERCE1','OP5_PIERCE'
  WITH $MOTYPE=JOINT
    MOVE AUX TO RAM_ADJUST
Repeat
  -- Request R1 load blank
  DOUT[1]=ON
  --! LABELEVENT  'Wait for R6 loaded and clear',0
  WAIT FOR DIN[1]=ON
  DOUT[1]=OFF
  --! LABELEVENT 'Press 5 Cycle',3
  MOVE AXIS 1 BY 180.00
  -- ! MakeVisible,1,'OP5_PIERCE'
  -- ! MakeInvisible,1,'OP4_PIERCE1'
  MOVE AXIS 1 BY 180.00
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R7 unloaded and clear',0
  WAIT FOR DIN[2]=ON
  DOUT[2]=OFF
UNTIL FALSE
END Press_5
```

-- Press_5#.KL Teach points

```
RAM_ADJUST[2] = -83.8
-- END RAM_ADJUST
RAM_DWN[1] = 180
RAM_DWN[2] = -83.8
-- END RAM_DWN
RAM_UP[1] = 0
RAM_UP[2] = -83.8
-- END RAM_UP
```

PROGRAM Press_6

```
--  Workspace LT KAREL 2 Program for 500 TON PRESS 6

VAR
  -- TEACHPOINT DECLARATIONS
  RAM_ADJUST : AUXPOS
  RAM_DWN : AUXPOS
```

```
  RAM_UP : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Press request part load from R7
  --! SIGNALDEF DOUT[2],TRACK     -- Press request part unload from R8

  --! SIGNALDEF DIN[1],Robot_R7.KL,2  -- R7 loaded part and clear
  --! SIGNALDEF DIN[2],Robot_R8.KL,1  -- R8 unloaded part and clear

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Press_6#
  -- ! MakeInvisible,1,'OP6_RESTRIKE'
  WITH $MOTYPE=JOINT
    MOVE AUX TO RAM_ADJUST
Repeat
  -- Request R1 load blank
  DOUT[1]=ON
  --! LABELEVENT 'Wait for R7 loaded and clear',0
  WAIT FOR DIN[1]=ON
  DOUT[1]=OFF
  --! LABELEVENT 'Press 6 Cycle',3
  MOVE AXIS 1 BY 180.00
  -- ! MakeVisible,1,'OP6_RESTRIKE'
  MOVE AXIS 1 BY 180.00
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R8 unloaded and clear',0
  WAIT FOR DIN[2]=ON
  DOUT[2]=OFF
UNTIL FALSE
END Press_6
```

Press_6#.KL Teach points

```
RAM_ADJUST[2] = -113.2227
-- END RAM_ADJUST
RAM_DWN[1] = 180
RAM_DWN[2] = -113.2227
-- END RAM_DWN
RAM_UP[1] = 0
RAM_UP[2] = -113.2227
-- END RAM_UP
```

Appendix B - Robot Tracks

PROGRAM Robot_R1

```
--  Workspace LT KAREL 2 Program for ROBOT_R1 Robot

VAR
  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Blank sheet loaded, R1 cleared press 1

  --! SIGNALDEF DIN[1],Press_1.KL,1  -- Press 1 requests R1 load Blank

ROUTINE RetrieveBlank
Begin
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot move to blank grasp',2
  MOVE TO BLANKSTACK_CLR
  $MOTYPE=Linear
  $SPEED=250
  MOVE TO BLANK_GRSP
  --! LABELEVENT 'Robot grasp blank',5
  -- ! MakeVisible,1,'BLANK_R1'
  DELAY 1500
  -- ! MakeVisible,1,'BLANK'
  --! LABELEVENT 'Robot Move to stack clear',2
  MOVE TO BLANKSTACK_CLR
END RetrieveBlank

ROUTINE Press1Load
BEGIN
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot Move to press 1 clear',2
  MOVE TO PRESS1_CLR
  --! LABELEVENT 'Wait for Press 1 load request',0
  WAIT FOR DIN[1]=ON
  $MOTYPE=Linear
  $SPEED=500
  --! LABELEVENT 'Robot Move to blank release',2
  MOVE TO OP1_BLANK_CLR
  $SPEED=250
  MOVE TO OP1_BLANK_RLS
  --! LABELEVENT 'Robot part blank release',5
  DELAY 1500
```

```
  -- ! MakeInvisible,1,'BLANK_R1'
  -- ! MakeVisible,1,'OP1_ BLANK'
  --! LABELEVENT 'Robot Move to press 1 clear',2
  $SPEED=500
  MOVE TO OP1_BLANK_CLR
  MOVE TO PRESS1_CLR
  $MOTYPE=Joint
  $SPEED=2000
  DOUT[1]=ON
  --! LABELEVENT 'Robot move home',2
  MOVE TO R1_Home
  DOUT[1]=OFF
END Press1Load

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R1#
  $UTOOL=POS(803,0,233.8,180,0,0,'')
  $SPEED=2000
  -- ! MakeInvisible,1,'BLANK_R1'
  -- ! MakeVisible,1,'BLANK'
  MOVE TO R1_Home
Repeat
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot move home',2
  MOVE TO R1_Home
  RetrieveBlank
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot move home',2
  MOVE TO R1_Home
  Press1Load
UNTIL FALSE
END Robot_R1
```

PROGRAM Robot_R2

```
--  Workspace LT KAREL 2 Program for ROBOT_R2 Robot

VAR
  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Part Draw unloaded, R2 cleared press 1
  --! SIGNALDEF DOUT[2],TRACK     -- Hand off Part Draw from R2 to R3

  --! SIGNALDEF DIN[1],Press_1.KL,2  -- Press requests R2 unload part Draw
  --! SIGNALDEF DIN[2],Robot_R3.KL,2  -- R3 clear R2
```

```
  --! SIGNALDEF DIN[3],Robot_R3.KL,2  -- R3 has Part Draw from R2

ROUTINE Press1Unload
Begin
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=1000
  --! LABELEVENT 'Wait for Press 1 unload request',0
  WAIT FOR DIN[1]=ON
  --! LABELEVENT 'Robot Move to draw grasp',2
  MOVE TO OP1_DRAW_CLR
  $MOTYPE=Linear
  $SPEED=250
  MOVE TO OP1_DRAW_GRSP
  --! LABELEVENT 'Robot part draw grasp',5
  -- ! MakeVisible,1,'OP1_DRAW_R2'
  DELAY 1500
  -- ! MakeInvisible,1,'OP1_DRAW'
  --! LABELEVENT 'Robot Move to press 1 clear',2
  $SPEED=500
  MOVE TO OP1_DRAW_CLR
  $SPEED=1000
  MOVE TO PRESS1_CLR1
  $MOTYPE=Joint
  $SPEED=2000
  DOUT[1]=ON
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R2_Home
  DOUT[1]=OFF
END Press1Unload

ROUTINE R3Trasfer
BEGIN
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  --! LABELEVENT 'Robot Move to draw release',2
  MOVE TO R2_RLS
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R3 draw grasp',0
  WAIT FOR DIN[3]=ON
  Delay 1500
  DOUT[2]=OFF
  --! LABELEVENT 'Wait for R3 clear',0
  WAIT FOR DIN[2]=ON
```

```
  --! LABELEVENT 'Robot Move to press 1 clear',2
  $SPEED=2000
  MOVE TO PRESS1_CLR1
END R3Trasfer

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R2#
  $UTOOL=POS(865.4,0,253.8,180,0,0,'')
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  -- ! MakeInvisible,1,'OP1_DRAW_R2'
  MOVE TO R2_Home
Repeat
  --! LABELEVENT 'Robot Move to press 1 clear',2
  MOVE TO PRESS1_CLR1
  Press1Unload
  R3Trasfer
UNTIL FALSE
END Robot_R2
```

PROGRAM Robot_R3

```
--  Workspace LT KAREL 2 Program for ROBOT_R3 Robot

VAR
  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- Part Draw unloaded, R2 cleared press 2
  --! SIGNALDEF DOUT[2],TRACK     -- Hand off Part Draw from R2 to R3
  --! SIGNALDEF DOUT[3],TRACK     -- Hand off, Part Draw from R2 to R3 grasped

  --! SIGNALDEF DIN[1],Press_2.KL,1  -- Press requests R2 unload part Draw
  --! SIGNALDEF DIN[2],Robot_R2.KL,2  -- R3 has Part Draw from R2

ROUTINE R2TransferR3
Begin
  $MOTYPE=JOINT
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Wait for R2, transfer request',0
  WAIT FOR DIN[2]=ON
  --! LABELEVENT 'Robot Move to draw grasp',2
  MOVE TO R3_PNC
  $MOTYPE=Linear
  $SPEED=250
```

```
  MOVE TO R2_RLS
  --! LABELEVENT 'R3 part draw grasp',5
  -- ! MakeVisible,1,'OP1_DRAW_R3'
  DELAY 1500
  -- ! MakeInvisible,1,'OP1_DRAW_R2'
  --! LABELEVENT 'Wait for R2, part draw release',0
  DOUT[3]=ON
  DELAY 1500  --R2 release part
  DOUT[3]=OFF
  --! LABELEVENT 'Robot Move clear',2
  $SPEED=500
  MOVE TO R3_PNC
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  DOUT[2]=ON
  MOVE TO R3_HOME
  --! LABELEVENT 'Robot Move press 2 clear',2
  MOVE TO PRESS2_CLR
  DOUT[2]=OFF
END R2TransferR3

ROUTINE Press2Load
BEGIN
  $MOTYPE=Linear
  $TERMTYPE=FINE
  $SPEED=1000
  --! LABELEVENT 'Wait for Press 2 load request',0
  WAIT FOR DIN[1]=ON
  --! LABELEVENT 'Robot Move part draw release',2
  MOVE TO DRAW_CLR
  $SPEED=250
  MOVE TO DRAW_RLS
  --! LABELEVENT 'Robot part draw release',5
  -- ! MakeVisible,1,'OP1_DRAW2'
  DELAY 1500
  -- ! MakeInvisible,1,'OP1_DRAW_R3'
  --! LABELEVENT 'Robot Move press 2 clear',2
  $SPEED=500
  MOVE TO DRAW_CLR
  $SPEED=1000
  MOVE TO PRESS2_CLR
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  DOUT[1]=ON
```

```
  MOVE TO R3_HOME
  DOUT[1]=OFF
END Press2Load

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R2#
  $UTOOL=POS(740.4,0,228.8,0,0,180,'')
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  -- ! MakeInvisible,1,'OP1_DRAW_R3'
  MOVE TO R3_HOME
Repeat
  R2TransferR3
  Press2load
UNTIL FALSE
END Robot_R3
```

PROGRAM Robot_R4

```
--  Workspace LT KAREL 2 Program for ROBOT_R4 Robot

VAR
  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- R4 unload Part trim, R4 cleared press 2
  --! SIGNALDEF DOUT[2],TRACK     -- R4 load part trim, Press 3 clear press 3

  --! SIGNALDEF DIN[1],Press_2.KL,2  -- Press 2 requests R3 unload part Trim
  --! SIGNALDEF DIN[2],Press_3.KL,1  -- Press 3 requests R3 load part Trim

ROUTINE Press2Unload
Begin
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot move to press 2 clear',2
  MOVE TO PRESS2_CLR1
  --! LABELEVENT 'Wait for Press 2 unload request',0
  WAIT FOR DIN[1]=ON
  $MOTYPE=Linear
  $SPEED=1000
  --! LABELEVENT 'Robot Move part trim grasp',2
  MOVE TO TRIM_CLR
  $SPEED=250
  MOVE TO OP2_TRIM_GSP
```

```
  --! LABELEVENT 'Robot part trim grasp',5
  -- ! MakeVisible,1,'OP2_TRIM_R4'
  DELAY 1500
  -- ! MakeInvisible,1,'OP2_TRIM'
  --! LABELEVENT 'Robot Move press 2 clear',2
  $SPEED=500
  MOVE TO TRIM_CLR
  $SPEED=1000
  MOVE TO PRESS2_CLR1
  DOUT[1]=ON
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R4_HOME
  DOUT[1]=OFF
END Press2Unload

ROUTINE Press3Load
BEGIN
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot Move press 3 clear',2
  MOVE TO PRESS3_CLR
  --! LABELEVENT 'Wait for Press 2 unload request',0
  WAIT FOR DIN[2]=ON
  $MOTYPE=Linear
  $SPEED=1000
  --! LABELEVENT 'Robot Move part trim release',2
  MOVE TO TRIM_CLR1
  $SPEED=250
  MOVE TO OP2_TRIM_RLS
  --! LABELEVENT 'Robot part trim release',5
  -- ! MakeVisible,1,'OP2_TRIM_2'
  DELAY 1500
  -- ! MakeInvisible,1,'OP2_TRIM_R4'
  --! LABELEVENT 'Robot Move press 3 clear',2
  $SPEED=500
  MOVE TO TRIM_CLR1
  $SPEED=1000
  MOVE TO PRESS3_CLR
  $MOTYPE=Joint
  $SPEED=2000
  DOUT[2]=ON
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R4_HOME
```

```
  DOUT[2]=OFF
END Press3Load

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R4#
  $UTOOL=POS(790.4,0,230.3691,0,0,0,'')
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  -- ! MakeInvisible,1,'OP2_TRIM_R4'
  MOVE TO R4_HOME
Repeat
  Press2Unload
  Press3Load
UNTIL FALSE
END Robot_R4
```

PROGRAM Robot_R5

```
--  Workspace LT KAREL 2 Program for ROBOT_R4 Robot

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- R5 unload part pierce, R5 clear press 3
  --! SIGNALDEF DOUT[2],TRACK     -- R5 load part pierce, R5 clear press 4

  --! SIGNALDEF DIN[1],Press_3.KL,2  -- Press 3 requests R5 unload part pierce
  --! SIGNALDEF DIN[2],Press_4.KL,1  -- Press 4 requests R5 load part pierce


ROUTINE Press3Unload
Begin
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot move to press 3 clear',2
  MOVE TO PRESS3_CLR1
  --! LABELEVENT 'Wait for Press 3 unload request',0
  WAIT FOR DIN[1]=ON
  $MOTYPE=Linear
  $SPEED=1000
  --! LABELEVENT 'Robot Move part pierce grasp',2
  MOVE TO PIERCE1_CLR
  $SPEED=250
  MOVE TO OP3_PIERCE_GSP
  --! LABELEVENT 'Robot part pierce grasp',5
```

```
  -- ! MakeVisible,1,'OP3_PIERCE_R5'
  DELAY 1500
  -- ! MakeInvisible,1,'OP3_PIERCE'
  --! LABELEVENT 'Robot Move press 3 clear',2
  $SPEED=500
  MOVE TO PIERCE1_CLR
  MOVE TO PRESS3_CLR1
  DOUT[1]=ON
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R5_HOME
  DOUT[1]=OFF
END Press3Unload

ROUTINE Press4Load
BEGIN
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot Move press 4 clear',2
  MOVE TO PRESS4_CLR
  --! LABELEVENT 'Wait for Press 2 unload request',0
  WAIT FOR DIN[2]=ON
  $MOTYPE=Linear
  $SPEED=1000
  --! LABELEVENT 'Robot Move part pierce release',2
  MOVE TO PIERCE2_CLR
  $SPEED=250
  MOVE TO OP3_PIERCE_RLS
  --! LABELEVENT 'Robot part pierce release',5
  -- ! MakeVisible,1,'OP3_PIERCE1'
  DELAY 1500
  -- ! MakeInvisible,1,'OP3_PIERCE_R5'
  --! LABELEVENT 'Robot Move press 3 clear',2
  $SPEED=500
  MOVE TO PIERCE2_CLR
  $SPEED=1000
  MOVE TO PRESS4_CLR
  DOUT[2]=ON
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R5_HOME
  DOUT[2]=OFF
END Press4Load
```

```
BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R5#
  $UTOOL=POS(790.4,0,230.4261,0,0,180,'')
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  -- ! MakeInvisible,1,'OP3_PIERCE_R5'
  MOVE TO R5_HOME
Repeat
  Press3Unload
  Press4Load
UNTIL FALSE
END Robot_R5
PROGRAM Robot_R6
--  Workspace LT KAREL 2 Program for ROBOT_R6 Robot

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- R6 unload part pierce, R6 clear press 4
  --! SIGNALDEF DOUT[2],TRACK     -- R6 load part pierce, R5 clear press 5

  --! SIGNALDEF DIN[1],Press_4.KL,2  -- Press 5 requests R6 unload part pierce
  --! SIGNALDEF DIN[2],Press_5.KL,1  -- Press 6 requests R6 load part pierce


ROUTINE Press4Unload
Begin
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot move to press 4 clear',2
  MOVE TO PRESS4_CLR1
  --! LABELEVENT 'Wait for Press 4 unload request',0
  WAIT FOR DIN[1]=ON
  $SPEED=1000
  --! LABELEVENT 'Robot Move part pierce grasp',2
  MOVE TO OP4_PIERCE_CLR
  $SPEED=250
  MOVE TO OP4_PIERCE_GSP
  --! LABELEVENT 'Robot part pierce grasp',5
  -- ! MakeVisible,1,'OP4_PIERCE_R6'
  DELAY 1500
  -- ! MakeInvisible,1,'OP4_PIERCE'
  --! LABELEVENT 'Robot Move press 4 clear',2
  $SPEED=500
```

```
  MOVE TO OP4_PIERCE_CLR
  MOVE TO PRESS4_CLR1
  DOUT[1]=ON
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R6_HOME
  DOUT[1]=OFF
END Press4Unload

ROUTINE Press5Load
BEGIN
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
   --! LABELEVENT 'Robot Move press 5 clear',2
  MOVE TO PRESS5_CLR
  --! LABELEVENT 'Wait for Press 5 load request',0
  WAIT FOR DIN[2]=ON
  $MOTYPE=Linear
  $SPEED=1000
  --! LABELEVENT 'Robot Move part pierce release',2
  MOVE TO OP4_PIERCE_CLR1
  $SPEED=250
  MOVE TO OP4_PIERCE_RLS
  --! LABELEVENT 'Robot part pierce release',5
  -- ! MakeVisible,1,'OP4_PIERCE1'
  DELAY 1500
  -- ! MakeInvisible,1,'OP4_PIERCE_R6'
  --! LABELEVENT 'Robot Move press 3 clear',2
  $SPEED=500
  MOVE TO OP4_PIERCE_CLR1
  $SPEED=1000
  MOVE TO PRESS5_CLR
  DOUT[2]=ON
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R6_HOME
  DOUT[2]=OFF
END Press5Load

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R6#
  $UTOOL=POS(790.4,0,243.8,0,0,0,'')
```

```
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  -- ! MakeInvisible,1,'OP4_PIERCE_R6'
  MOVE TO R6_HOME
Repeat
  Press4Unload
  Press5Load
UNTIL FALSE
END Robot_R6
```

PROGRAM Robot_R7

```
--  Workspace LT KAREL 2 Program for ROBOT_R7 Robot

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- R7 unload part pierce, R7 clear press 5
  --! SIGNALDEF DOUT[2],TRACK     -- R7 load part pierce, R7 clear press 6
  --! SIGNALDEF DIN[1],Press_5.KL,2  -- Press 5requests R7 unload part pierce
  --! SIGNALDEF DIN[2],Press_6.KL,1  -- Press 6 requests R7 unload part pierce

ROUTINE Press5Unload
Begin
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  --! LABELEVENT 'Robot move to press 5 clear',2
  MOVE TO PRESS5_CLR1
  --! LABELEVENT 'Wait for Press 5 unload request',0
  WAIT FOR DIN[1]=ON
  $SPEED=1000
  --! LABELEVENT 'Robot Move part pierce grasp',2
  MOVE TO OP5_PIERCE_CLR
  $SPEED=250
  MOVE TO OP5_PIERCE_GRSP
  --! LABELEVENT 'Robot part pierce grasp',5
  -- ! MakeVisible,1,'OP5_PIERCE_R7'
  DELAY 1500
  -- ! MakeInvisible,1,'OP5_PIERCE'
  --! LABELEVENT 'Robot Move press 5 clear',2
  $SPEED=500
  MOVE TO OP5_PIERCE_CLR
  MOVE TO PRESS5_CLR1
  DOUT[1]=ON
  $MOTYPE=Joint
  $SPEED=2000
```

```
  --! LABELEVENT 'Robot Move home',2
  $MOTYPE=Joint
  MOVE TO R7_HOME
  DOUT[1]=OFF
END Press5Unload

ROUTINE Press6Load
BEGIN
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
   --! LABELEVENT 'Robot Move press 6 clear',2
  MOVE TO PRESS6_CLR
  --! LABELEVENT 'Wait for Press 6 load request',0
  WAIT FOR DIN[2]=ON
  $MOTYPE=Linear
  $SPEED=1000
  --! LABELEVENT 'Robot Move part pierce release',2
  MOVE TO OP5_PIERCE_CLR1
  MOVE TO OP5_PIERCE_RLS
  -- ! MakeVisible,1,'OP6_RESTRIKE'
  DELAY 1500
  -- ! MakeInvisible,1,'OP5_PIERCE_R7'
  --! LABELEVENT 'Robot Move press 6 clear',2
  $SPEED=500
  MOVE TO OP5_PIERCE_CLR1
  MOVE TO PRESS6_CLR
  DOUT[2]=ON
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R7_HOME
  DOUT[2]=OFF
END Press6Load

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R7#
  $UTOOL=POS(790.4,0,230.1665,0,0,180,'')
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=1000
  -- ! MakeInvisible,1,'OP5_PIERCE_R7'
  MOVE TO R7_HOME
Repeat
  Press5Unload
```

```
   Press6Load
UNTIL FALSE
END Robot_R7
```

PROGRAM Robot_R8

```
PROGRAM Robot_R8
--  Workspace LT KAREL 2 Program for ROBOT_R8 Robot

CONST
   PartsTotal=120
   PartsPerBin=12

VAR
   PartsCount : INTEGER
   BinPartCount : INTEGER
   LoadBinNo:INTEGER

   -- Signal Declarations
   --! SIGNALDEF DOUT[1],TRACK     -- R8 unload part Re-strike, R8 clear press 6
   --! SIGNALDEF DOUT[2],TRACK     -- C1 Bin Full
   --! SIGNALDEF DOUT[3],TRACK     -- C3 Bin Full

   --! SIGNALDEF DIN[1],Press_6.KL,2  -- Press 6 requests R8 unload part Re-strike
   --! SIGNALDEF DIN[2],Conveyor_1.KL,2  -- R8 Load bin on C1, ready
   --! SIGNALDEF DIN[3],Conveyor_3.KL,2  -- R8 Load bin on C3, Ready


ROUTINE Press6Unload
Begin
   $MOTYPE=Joint
   $TERMTYPE=FINE
   $SPEED=2000
   --! LABELEVENT 'Robot move to press 6 clear',2
   MOVE TO PRESS6_CLR1
   --! LABELEVENT 'Wait for Press 6 unload request',0
   WAIT FOR DIN[1]=ON
   $SPEED=1000
   --! LABELEVENT 'Robot Move part Door grasp',2
   $MOTYPE=LINEAR
   MOVE TO OP5_PIERCE_CLR1
   $SPEED=250
   MOVE TO OP5_PIERCE_RLS
   --! LABELEVENT 'Robot part Door grasp',5
   --! Makevisible,1,'OP6_RESTRIKE'
   --! COPYOBJECTALT 'OP6_RESTRIKE','DOOR'
```

```
  --! ATTACHOBJALT 'R8_TUBE','NOPARENT.DOOR'
  DELAY 1500
  -- ! MakeInvisible,1,'OP6_RESTRIKE'
  --! LABELEVENT 'Robot Move press 6 clear',2
  $SPEED=500
  MOVE TO OP5_PIERCE_CLR1
  MOVE TO PRESS6_CLR1
  DOUT[1]=ON
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'Robot Move home',2
  MOVE TO R8_HOME
  DOUT[1]=OFF
END Press6Unload

ROUTINE Conveyor1Load
BEGIN
  Press6Unload
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'R8 move to C1 clear',2
  MOVE TO CONV1BINLOAD
  $MOTYPE=LINEAR
  $SPEED=1500
  --! LABELEVENT 'Wait for C1 Bin load ready',2
  WAIT FOR DIN[2]=ON
  --! LABELEVENT 'R8 Load Bin',2
  MOVE RELATIVE VEC(((-1640)+BinPartCount*140),0,0)
  $SPEED=250
  MOVE RELATIVE VEC(0,0,-700)
  --! LABELEVENT 'R8 part Door Release',5
  --! DETACHOBJALT 'R8_TUBE.DOOR'
  IF LoadBinNo=1 THEN
    --! ATTACHOBJALT 'BIN_1','NOPARENT.DOOR'
  ENDIF
  IF LoadBinNo=3 THEN
    --! ATTACHOBJALT 'BIN_3','NOPARENT.DOOR'
  ENDIF
  DELAY 1500
  --! LABELEVENT 'R8 move C1 clear',2
  MOVE RELATIVE VEC(100,0,0)
  $SPEED=1000
  MOVE RELATIVE VEC(0,0,700)
  MOVE TO CONV1BINLOAD
  --! LABELEVENT 'Robot Move press 6 clear',2
  $MOTYPE=Joint
```

```
  BinPartCount=BinPartCount+1
  IF BinPartCount=PartsPerBin THEN
    LoadBinNo=LoadBinNo+1
    BinPartCount=0
    DOUT[2]=ON    --Bin Full
    MOVE TO PRESS6_CLR1
    DOUT[2]=OFF
  ENDIF
END Conveyor1Load

ROUTINE Conveyor3Load
BEGIN
  Press6Unload
  $MOTYPE=Joint
  $SPEED=2000
  --! LABELEVENT 'R8 move to C3 clear',2
  MOVE TO CONV3BINLOAD
  $MOTYPE=LINEAR
  $SPEED=1500
  --! LABELEVENT 'Wait for C3 Bin load ready',2
  WAIT FOR DIN[3]=ON
  --! LABELEVENT 'R8 Load Bin',2
  MOVE RELATIVE VEC(((-1640)+BinPartCount*140),0,0)
  $SPEED=250
  MOVE RELATIVE VEC(0,0,-700)
  --! LABELEVENT 'R8 part Door Release',5
  --! DETACHOBJALT 'R8_TUBE.DOOR'
  IF LoadBinNo=2 THEN
  -- ! ATTACHOBJALT 'BIN_2','NOPARENT.DOOR'
  ENDIF
  IF LoadBinNo=4 THEN
  -- ! ATTACHOBJALT 'BIN_4','NOPARENT.DOOR'
  ENDIF
  DELAY 1500
  --! LABELEVENT 'R8 move C3 clear',2
  MOVE RELATIVE VEC(100,0,0)
  $SPEED=1500
  MOVE RELATIVE VEC(0,0,700)
  $SPEED=2000
  MOVE TO CONV3BINLOAD
  $MOTYPE=Joint
  BinPartCount=BinPartCount+1
  IF BinPartCount=PartsPerBin THEN
    LoadBinNo=LoadBinNo+1
    BinPartCount=0
    DOUT[3]=ON    --Bin Full
```

```
    --! LABELEVENT 'Robot Move press 6 clear',2
    MOVE TO PRESS6_CLR1
    DOUT[3]=OFF
  ENDIF
END Conveyor3Load


BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Robot_R8#
  $UTOOL=POS(690.4,0,230.7725,0,0,0,'')
  $MOTYPE=Joint
  $TERMTYPE=FINE
  $SPEED=2000
  PartsCount=0
  BinPartCount=0
  LoadBinNo=1
  MOVE TO PRESS6_CLR1
Repeat
  IF LoadBinNo=4 THEN
    Conveyor3Load
  ENDIF
  IF LoadBinNo=3 THEN
    Conveyor1Load
  ENDIF
  IF LoadBinNo=2 THEN
    Conveyor3Load
  ENDIF
  IF LoadBinNo=1 THEN
    Conveyor1Load
  ENDIF
  IF LoadBinNo=5 THEN
    LoadBinNo=1
  ENDIF
  PartsCount=PartsCount+1
UNTIL FALSE
END Robot_R8
```

Appendix C - Conveyor Tracks

PROGRAM Conveyor_1

```
--  Workspace LT KAREL 2 Program for CONVEYOR1 conveyor

CONST
  PartsTotal=120
```

```
VAR
  PartsCount : INTEGER

  C1_HOME : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- C1 ready for transfer of bin to C2
  --! SIGNALDEF DOUT[2],TRACK     -- C1 Ready for Bin load by R8

  --! SIGNALDEF DIN[1],Conveyor_2.KL,1  -- Wait for C2, ready
  --! SIGNALDEF DIN[2],Robot_R8.KL,2  -- Bin full

ROUTINE Bin1ToCon2
Begin
  --! LABELEVENT 'Rollers raise bin',4
  MOVE AXIS 1 BY 40.00
  -- ! Attachobject,1,'BIN_1','LIFTROLLERS1'
  MOVE AXIS 1 BY 50.00
  --! LABELEVENT 'Wait for C2 Ready',0
  WAIT FOR DIN[1]=ON      -- Wait for C2 - ready
  --! LABELEVENT 'Index bin to C2',4
  MOVE AXIS 2 BY 1930.4
  -- ! Detachobject,1,'BIN_1'
  DOUT[1]=ON
  --! LABELEVENT 'Rollers lower',4
  MOVE AXIS 1 BY -90.00
  DOUT[1]=OFF
END Bin1ToCon2

ROUTINE Bin3ToCon2
Begin
  --! LABELEVENT 'Rollers raise bin',4
  MOVE AXIS 1 BY 40.00
  -- ! Attachobject,1,'BIN_3','LIFTROLLERS1'
  MOVE AXIS 1 BY 50.00
  --! LABELEVENT 'Wait for C2 Ready',0
  WAIT FOR DIN[1]=ON      -- Wait for C2 - ready
  --! LABELEVENT 'Index bin to C2',4
  MOVE AXIS 2 BY 1930.4
  -- ! Detachobject,1,'BIN_3'
  DOUT[1]=ON
  --! LABELEVENT 'Rollers lower',4
  MOVE AXIS 1 BY -90.00
  DOUT[1]=OFF
END Bin3ToCon2
```

```
ROUTINE Bin1Conv1Index
Begin
  -- ! Attachobject,1,'BIN_1','ROLLERS1'
  --! LABELEVENT 'Index bin to load ready position',4
  MOVE AXIS 3 BY 2140.8
  -- ! Detachobject,1,'BIN_1'
  MOVE AXIS 3 BY -2140.8
END Bin1Conv1Index

ROUTINE Bin3Conv1Index
Begin
  -- ! Attachobject,1,'BIN_3','ROLLERS1'
  --! LABELEVENT 'Index bin to load ready position',4
  MOVE AXIS 3 BY 2140.8
  -- ! Detachobject,1,'BIN_3'
  MOVE AXIS 3 BY -2140.8
END Bin3Conv1Index

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Conveyor_1#
  PartsCount=0
  MOVE AUX TO C1_HOME
Repeat
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R8, Bin Full',0
  WAIT FOR DIN[2]  -- Bin Full
  DOUT[2]=OFF
  Bin1ToCon2
  Bin3Conv1Index
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R8, Bin Full',0
  WAIT FOR DIN[2]  -- Bin Full
  DOUT[2]=ON
  Bin3ToCon2
  Bin1Conv1Index
  PartsCount=PartsCount+1
UNTIL FALSE
END Conveyor_1
```

PROGRAM Conveyor_2

```
--  Workspace LT KAREL 2 Program for CONVEYOR2 conveyor

CONST
  PartsTotal=120
```

```
  PartsPerBin=12

VAR
  PartsCount : INTEGER
  DeleteCount : INTEGER
  BinCount : INTEGER

  C2_HOME : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- C2 ready to receive from C1, bin
  --! SIGNALDEF DOUT[2],TRACK     -- C2 ready to receive from C3, bin

  --! SIGNALDEF DIN[1],Conveyor_1.KL,1  -- C1 TRANSFER TO C2, complete
  --! SIGNALDEF DIN[2],Conveyor_3.KL,1  -- C3 TRANSFER TO C2, complete

ROUTINE Bin1Convey
Begin
  --! LABELEVENT 'Rollers raise',4
  MOVE AXIS 1 BY 90
  --! LABELEVENT 'Wait for C1 Ready',0
  DOUT[1]=ON  -- C2 ready to receive from C1, bin1
  WAIT FOR DIN[1]  -- C1 TRANSFER TO C2, complete
  DOUT[1]=OFF
  --! LABELEVENT 'Rollers lower bin',4
  --! Attachobject,1,'BIN_1','LIFTROLLERS2'
  MOVE AXIS 1 BY -50.00
  --! Detachobject,1,'BIN_1'
  MOVE AXIS 1 BY -40.00
  --! Attachobject,1,'BIN_1','ROLLERS2'
  --! LABELEVENT 'Index bin to off- load position',4
  MOVE AXIS 2 BY -2140.8
  --! Detachobject,1,'BIN_1'
  DeleteCount=PartsPerBin
  WHILE DeleteCount>0 DO
    --! DeleteObjectalt 'BIN_1.DOOR'
    DeleteCount=DeleteCount-1
  ENDWHILE
  --! Placeobject,1,'BIN_1',-32962.8,-1930.4,385.6,90,0,0
  MOVE AXIS 2 BY 2140.8
  BinCount=BinCount +1
END Bin1Convey

ROUTINE Bin2Convey
Begin
  --! LABELEVENT 'Rollers raise',4
```

```
  MOVE AXIS 1 BY 90
  --! LABELEVENT 'Wait for C1 Ready',0
  DOUT[2]=ON  -- C2 ready to receive from C3, bin2
  WAIT FOR DIN[2]  -- C3 TRANSFER TO C2, complete
  DOUT[2]=OFF
  --! LABELEVENT 'Rollers lower bin',4
  --! Attachobject,1,'BIN_2','LIFTROLLERS2'
  MOVE AXIS 1 BY -50.00
  --! Detachobject,1,'BIN_2'
  MOVE AXIS 1 BY -40.00
  --! LABELEVENT 'Index bin to off- load position',4
  --! Attachobject,1,'BIN_2','ROLLERS2'
  MOVE AXIS 2 BY -2140.8
  --! Detachobject,1,'BIN_2'
  DeleteCount=PartsPerBin
  WHILE DeleteCount>0 DO
    --! DeleteObjectalt 'BIN_2.DOOR'
    DeleteCount=DeleteCount-1
  ENDWHILE
  --! Placeobject,1,'BIN_2',-32962.8,1930.4,385.6,90,0,0
  MOVE AXIS 2 BY 2140.8
  BinCount=BinCount +1
END Bin2Convey

ROUTINE Bin3Convey
Begin
  --! LABELEVENT 'Rollers raise',4
  MOVE AXIS 1 BY 90
  --! LABELEVENT 'Wait for C1 Ready',0
  DOUT[1]=ON  -- C2 ready to receive from C1, bin1
  WAIT FOR DIN[1]  -- C1 TRANSFER TO C2, complete
  DOUT[1]=OFF
  --! LABELEVENT 'Rollers lower bin',4
  --! Attachobject,1,'BIN_3','LIFTROLLERS2'
  MOVE AXIS 1 BY -50.00
  --! Detachobject,1,'BIN_3'
  MOVE AXIS 1 BY -40.00
  --! LABELEVENT 'Index bin to off- load position',4
  --! Attachobject,1,'BIN_3','ROLLERS2'
  MOVE AXIS 2 BY -2140.8
  --! Detachobject,1,'BIN_3'
  DELAY 5000  -- Delay required for C1 and C2 ready
  DeleteCount=PartsPerBin
  WHILE DeleteCount>0 DO
    --! DeleteObjectalt 'BIN_3.DOOR'
    DeleteCount=DeleteCount-1
```

```
  ENDWHILE
  --! Placeobject,1,'BIN_3',-32962.8,-1930.4,385.6,90,0,0
  MOVE AXIS 2 BY 2140.8
  BinCount=BinCount +1
END Bin3Convey

ROUTINE Bin4Convey
Begin
  --! LABELEVENT 'Rollers raise',4
  MOVE AXIS 1 BY 90
  --! LABELEVENT 'Wait for C1 Ready',0
  DOUT[2]=ON  -- C2 ready to receive from C3, bin 4
  WAIT FOR DIN[2]  -- C3 TRANSFER TO C2, complete
  DOUT[2]=OFF
  --! LABELEVENT 'Rollers lower bin',4
  --! Attachobject,1,'BIN_4','LIFTROLLERS2'
  MOVE AXIS 1 BY -50.00
  --! Detachobject,1,'BIN_4'
  MOVE AXIS 1 BY -40.00
  --! LABELEVENT 'Index bin to off- load position',4
  --! Attachobject,1,'BIN_4','ROLLERS2'
  MOVE AXIS 2 BY -2140.8
  --! Detachobject,1,'BIN_4'
  DeleteCount=PartsPerBin
  WHILE DeleteCount>0 DO
    --! DeleteObjectalt 'BIN_4.DOOR'
    DeleteCount=DeleteCount-1
  ENDWHILE
  --! Placeobject,1,'BIN_4',-32962.8,1930.4,385.6,90,0,0
  MOVE AXIS 2 BY 2140.8
  BinCount=BinCount +1
END Bin4Convey


BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Conveyor_2#
  PartsCount=0
  MOVE AUX TO C2_HOME
Repeat
  Bin1Convey
  Bin2Convey
  Bin3Convey
  Bin4Convey
  PartsCount=PartsCount+1
UNTIL FALSE
```

```
END Conveyor_2
```

PROGRAM Conveyor_3

```
--  Workspace LT KAREL 2 Program for CONVEYOR1 conveyor

CONST
  PartsTotal=120

VAR
  PartsCount : INTEGER

  C3_HOME : AUXPOS

  -- Signal Declarations
  --! SIGNALDEF DOUT[1],TRACK     -- C3 ready for transfer of bin to C2
  --! SIGNALDEF DOUT[2],TRACK     -- C3 Ready for Bin load by R8

  --! SIGNALDEF DIN[1],Conveyor_2.KL,2  -- Wait for C2, ready
  --! SIGNALDEF DIN[2],Robot_R8.KL,3  -- Bin full

ROUTINE Bin2ToCon2
Begin
  --! LABELEVENT 'Rollers raise bin',4
  MOVE AXIS 1 BY 40.00
  -- ! Attachobject,1,'BIN_2','LIFTROLLERS3'
  MOVE AXIS 1 BY 50.00
  --! LABELEVENT 'Wait for C2 Ready',0
  WAIT FOR DIN[1]=ON      -- Wait for C2 - ready
  --! LABELEVENT 'Index bin to C2',4
  MOVE AXIS 2 BY 1930.4
  -- ! Detachobject,1,'BIN_2'
  DOUT[1]=ON
  --! LABELEVENT 'Rollers lower',4
  MOVE AXIS 1 BY -90.00
  DOUT[1]=OFF
END Bin2ToCon2

ROUTINE Bin4ToCon2
Begin
  --! LABELEVENT 'Rollers raise bin',4
  MOVE AXIS 1 BY 40.00
  -- ! Attachobject,1,'BIN_4','LIFTROLLERS3'
  MOVE AXIS 1 BY 50.00
  --! LABELEVENT 'Wait for C2 Ready',0
  WAIT FOR DIN[1]=ON      -- Wait for C2 - ready
```

```
  --! LABELEVENT 'Index bin to C2',4
  MOVE AXIS 2 BY 1930.4
  -- ! Detachobject,1,'BIN_4'
  DOUT[1]=ON
  --! LABELEVENT 'Rollers lower',4
  MOVE AXIS 1 BY -90.00
  DOUT[1]=OFF
END Bin4ToCon2

ROUTINE Bin2Conv3Index
Begin

  -- ! Attachobject,1,'BIN_2','ROLLERS3'
  --! LABELEVENT 'Index bin to load ready position',4
  MOVE AXIS 3 BY 2140.8
  -- ! Detachobject,1,'BIN_2'
  MOVE AXIS 3 BY -2140.8
END Bin2Conv3Index

ROUTINE Bin4Conv3Index
Begin
  -- ! Attachobject,1,'BIN_4','ROLLERS3'
  --! LABELEVENT 'Index bin to load ready position',4
  MOVE AXIS 3 BY 2140.8
  -- ! Detachobject,1,'BIN_4'
  MOVE AXIS 3 BY -2140.8
END Bin4Conv3Index

BEGIN
  $USEMAXACCEL=TRUE
  %INCLUDE Conveyor_3#
  PartsCount=0
  MOVE AUX TO C3_HOME
Repeat
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R8, Bin Full',0
  WAIT FOR DIN[2]  -- Bin Full
  DOUT[2]=OFF
  Bin2ToCon2
  Bin4Conv3Index
  DOUT[2]=ON
  --! LABELEVENT 'Wait for R8, Bin Full',0
  WAIT FOR DIN[2]  -- Bin Full
  DOUT[2]=OFF
  Bin4ToCon2
  Bin2Conv3Index
```

```
   PartsCount=PartsCount+1
UNTIL FALSE
END Conveyor_3
```

Conveyor_3#.KL, Teach points

```
C3_HOME[1] = 0
C3_HOME[2] = 0
C3_HOME[3] = 0
-- END C3_HOME
```

Appendix D - Macros

PROGRAM DunnageSetup

```
--  Workspace LT KAREL 2 Program for DunnageSetup Macro

Begin
   -- ! Placeobject,1,'BIN_1',-30822,-1930.4,385.6,90,0,0
   -- ! Detachobject,1,'BIN_1'
   -- ! Placeobject,1,'BIN_3',-32962.8,-1930.4,385.6,90,0,0
   -- ! Detachobject,1,'BIN_3'

   -- ! Placeobject,1,'BIN_2',-30822,1930.4,385.6,90,0,0
   -- ! Detachobject,1,'BIN_2'
   -- ! Placeobject,1,'BIN_4',-32962.8,1930.4,385.6,90,0,0
   -- ! Detachobject,1,'BIN_4'
End DunnageSetup
```

Appendix E - Mechanical Press Specifications

Press Specifications	
Capacity (Tons)	500
Dia. Of Crankshaft at Main and Eccentric	10.5, 13.75
Stroke (in)	18
Shut Height SDAU Over Bolster (in)	30
Bolster Thickness	7.5
Slide Adjustment	10
Speed (Strokes per min)	30
Bed, Bolster and Slide (in)	96 x 48