

HYRobotics EPIK, UNIK Line by line Programming Samples

Axis X : Traverse

Axis Y : Kick

Axis Z : Vertical

Command

1. WaitIMM : Wait molding machine signal
 - a. WaitIMM(E_IS_BACK) : Wait Ejector is back
 - b. WaitIMM(M_IS_Open) : Wait Mold is open
 - c. WaitIMM(M_IS_Closed) (Need to be in program)
2. SETIMM : Set Molding Machine interlock
 - a. SetIMM(M_Close, False) : Set Molding machine not to close mold
 - b. SetIMM(M_Area_Free,True) : Robot Mold area free
 - c. SetIMM(E_FOWRAD, TRUE) : Set Machine ejector allow to forward
3. SetDO : Set Digital Output
 - a. SetDO (douVacuum, on, false, , ,false) : Set Digital Output : Vacuum on)
4. WaitDIN(DigVacuumConfirm, On) : Wait Digital Input
5. MoveHY : HY made movement program
6. MoveAxis (y,200.0, v40) : Move Axis
7. WaitTimeSec(1.0) : Wait time

*** If condition statement

8. IF (Rjccount := 3, then
Rejccount := Rejccount+1
Move(RejPosition, v20)
WaitTimeSec (1.0)
Else
Palletize (Pallet1, v 20)
End_if
9. If (digUserInput1.val=TRUE) THEN
HYMOVE OR move HY

ELSEIF (digUserInput2.Val=TRUE) THEN
HYMOVE OR move HY
END_IF
10. If (digUserInput1.val=TRUE) AND (digUserInput2.Val=TRUE)
THEN
HYMOVE OR move HY
ELSEIF (digUserInput3.Val=TRUE) THEN
HYMOVE OR move HY
END_IF
11. SoftTorqueoff(y) : Kick arm servo motor : low torque. Ejector can push robot arm
12. **HYHoming () : HYRobotics made Homing Subroutine**
13. **HYStartup () : HYRobotics programmed start up subroutine**
14. Rejccount : = 0 (Set counter to be 0)

**Below is 4 basic step sample
(without adding any special steps, only using mode select for each step)**

```
// KAIROVersion 2.01
IF bDebugMode = FALSE THEN
  HYHoming()
  HYStartUp()
ELSE
  CALL homing()
  CALL StartUp()
END_IF

IF bSimpleFlowChart = TRUE THEN
  Dynamic(10, 10, 10)
ELSE
  Dynamic(100, 100, 100)
END_IF

WHILE bAutoRun AND NOT ((bContinueToProduce = FALSE) AND (dinReachedQuantityToGoal.val = TRUE)) AND (bDebugMode = FALSE) DO
  // START_EDIT
  // 2.00a.8
  HYTakeOutPosition(TakeOutPos, vpTakeOutPos, rKickReturnDelay)
  HYUpPosition(UpPos, vpUpPos, rUpDelay)
  HYRunnerRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease)
  HYMainRelease(MainArmReleasePos, vpMainArmReleasePos, rMainArmRelease)
  HYRunnerReturnRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease)
  HYToWaitingPos(startPos, vpWaitingPos, rDownDelay)
  // END_EDIT
  ##HYTakeOutPosition(TakeOutPos, vpTakeOutPos, rKickReturnDelay) : Extra copy of programming line
  ##HYUpPosition(UpPos, vpUpPos, rUpDelay) : Extra copy of programming line
  ##HYRunnerRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease) : Extra copy of programming line
  ##HYMainRelease(MainArmReleasePos, vpMainArmReleasePos, rMainArmRelease) : Extra copy of programming line
  ##HYRunnerReturnRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease) : Extra copy of programming line
  ##HYToWaitingPos(startPos, vpWaitingPos, rDownDelay) : Extra copy of programming line
  IF bOnlyOneAutoRun THEN
    // bSimpleFlowChart := FALSE //170526 delete WJM
    BREAK
  END_IF
END_WHILE

WHILE bDebugMode = TRUE DO
  Debug()
  IF bOnlyOneAutoRun THEN
    //bSimpleFlowChart := FALSE //170526 delete WJM
    BREAK
  END_IF
END_WHILE
```





Below is an extra step sample on existing 4 basic step

```
// KAIROVersion 2.01
IF bDebugMode = FALSE THEN
  HYHoming()
  HYStartUp()
ELSE
  CALL homing()
  CALL StartUp()
END_IF

IF bSimpleFlowChart = TRUE THEN
  Dynamic(10, 10, 10)
ELSE
  Dynamic(100, 100, 100)
END_IF

WHILE bAutoRun AND NOT ((bContinueToProduce = FALSE) AND (dinReachedQuantityToGoal.val = TRUE)) AND (bDebugMode = FALSE) DO
  // START_EDIT
  // 2.00a.8
  SetDO(douSpareOutput2, ON, FALSE, , , FALSE)
  SetDO(douSpareOutput4, ON, FALSE, , , FALSE)
  HYWaitForIMM()
  MoveHY(Down, 70, , 0.0, RotRETURN, SwiRETURN, FALSE)
  SetDO(douSpareOutput2, OFF, FALSE, , , FALSE)
  SetDO(douSpareOutput4, OFF, FALSE, , , FALSE)
  MoveHY(Kick, 30, , 0.0, RotRETURN, SwiRETURN, FALSE)
  SetDO(douSpareOutput1, ON, FALSE, , , FALSE)
  SetDO(douSpareOutput3, ON, FALSE, , , FALSE)
  WaitTimeSec(0.5)
  SetDO(douChuck, ON, FALSE, , , FALSE)
  SetDO(douSpareOutput1, OFF, FALSE, , , FALSE)
  SetDO(douSpareOutput3, OFF, FALSE, , , FALSE)
  SetIMM(E_FORWARD, TRUE)
  WaitTimeSec(0.5)
  SetDO(douMainArmGrip, ON, FALSE, , , FALSE)
  HYTakeOutPosition(TakeOutPos, vpTakeOutPos, rKickReturnDelay)
  SetIMM(E_BACK, FALSE)
  SetDO(douSpareOutput2, OFF, FALSE, , , FALSE)
  WaitIMM(E_IS_FORWARD)
  SetDO(douSpareOutput2, ON, FALSE, , , FALSE)
  SetDO(douSpareOutput4, ON, FALSE, , , FALSE)
  WaitTimeSec(1.0)
  MoveHY(KickReturn, 50, , 0.0, RotRETURN, SwiTURN, FALSE)
  MoveHY(Down2, 70, , 0.0, RotRETURN, SwiRETURN, FALSE)
  WaitTimeSec(0.5)
  IF Count < 20 THEN
    // Insert Count
    Count := Count + 1
  MoveHY(Kick2, 50, , 0.0, RotRETURN, SwiRETURN, FALSE)
  SetDO(douSpareOutput2, OFF, FALSE, , , FALSE)
  SetDO(douSpareOutput4, OFF, FALSE, , , FALSE)
  SetDO(douSpareOutput3, ON, FALSE, , , FALSE)
  WaitTimeSec(0.5)
  SetIMM(E_BACK, TRUE)
  WaitTimeSec(1.0)
```

```

SetDO(douSpareOutput3, OFF, FALSE, , , FALSE)
WaitIMM(E_IS_BACK)
WaitTimeSec(0.0)
ELSE
  //Reset
  Count := 0
END_IF

IF bSimpleFlowChart = FALSE THEN
  HYSetDI(digChuckConfirm, 0.0, FALSE, 5, iWhichToolError)
END_IF

SetDO(douChuck, OFF, FALSE, , , FALSE)
WaitTimeSec(0.5)
SetIMM(E_BACK, TRUE)
WaitIMM(E_IS_BACK)
SetDO(douSpareOutput3, OFF, FALSE, , , FALSE)
SetDO(douSpareOutput2, ON, FALSE, , , FALSE)
SetDO(douSpareOutput4, ON, FALSE, , , FALSE)
HYUpPosition(UpPos, vpUpPos, rUpDelay)
HYRunnerRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease)
HYMainRelease(MainArmReleasePos, vpMainArmReleasePos, rMainArmRelease)
HYRunnerReturnRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease)
HYToWaitingPos(startPos, vpWaitingPos, rDownDelay)
// END_EDIT
##HYTakeOutPosition(TakeOutPos, vpTakeOutPos, rKickReturnDelay)
##HYUpPosition(UpPos, vpUpPos, rUpDelay)
##HYRunnerRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease)
##HYMainRelease(MainArmReleasePos, vpMainArmReleasePos, rMainArmRelease)
##HYRunnerReturnRelease(SubArmReleasePos, vpSubArmReleasePos, rSubArmRelease)
##HYToWaitingPos(startPos, vpWaitingPos, rDownDelay)
IF bOnlyOneAutoRun THEN
  // bSimpleFlowChart := FALSE      //170526 delete WJM
  BREAK
END_IF
END_WHILE
WHILE bDebugMode = TRUE DO
  Debug()
  IF bOnlyOneAutoRun THEN
  //bSimpleFlowChart := FALSE      //170526 delete WJM
  BREAK
END_IF
END_WHILE

```

////////////////////////////////////

1st couple shot reject sample

HYHoming ()

HYStartup ()

Rejcount := 0

While B AutoRun and Not ((b ContinueToProduce=False) And (dinReachedQuantityToGoal.val=Tues)) Do

HYWaitforIMM()

MoveAxis (x,20,v60)

MoveAxis (y,200.0, v40)

SetIMM(M_Close, False)

WaitIMM(M_IS_Open)

SetDO (douVacuum, on, false, , false)

MoveAxis(z, 600.0, v60)

Move Axis (y, 300.0, v20)

Softtorque(Y, 0)

SetIMM(E_FOWRAD, TRUE)

WaitTimeSec(1.0)

WaitDIN(DigVacuumConfirm, On)

SoftTorqueoff(y)

MoveAxis(y,200,0,v20)

SetIMM(E_Back, True)

WaitTimeSec(1.0)

SetIMM(E_Back, False)

MoveAxis(z,0.0,V60)

SetDO(douVacuum, On, False, digVacuumConfirm, 2.0 False)

SetIMM(M_Close,True)

SetIMM(M_Area_Free,True)

MoveAxis(x,1400.0, v60)

SetDO (douRotation, On, False, digRotationComplete, 3.0, True)

SetDO (douRotation, Off, False,,, False)

MoveAxis (y, 300.0, v60)

WaitIMM(M_IS_Closed)

IF (Rejcount := 3, then

Rejcount := Rejcount+1

Move(RejPosition, v20)

WaitTimeSec (1.0)

Else

Palletize (Pallet1, v 20)

End_If

SetDO(douvacuum, Off, False,,, False)

MoveAxis(Z, 200.0,V20)

SetDO (douRotationReturn, On, False, digRotationReturnComplete, 3.0, True)

Move Axis (z,0.0, V40)

Move Axis (x,0.0,V60)

End_While

- Sam Lee – 09-27-19
www.hyrobotics.com