.PROGRAM pick\_place()

PARAMETER HAND.TIME = 0.5

SPEED 50 ALWAYS

MOVE safe

OPENI

BREAK

APPRO pick, 70

BREAK

SPEED 10

MOVES pick

BREAK

CLOSEI

SPEED 10

DEPARTS 70

BREAK

APPRO place, 70

BREAK

SPEED 10

MOVES place

BREAK

OPENI

SPEED 10

DEPARTS 70

BREAK

MOVE safe

BREAK

.END

.PROGRAM program()

PARAMETER HAND.TIME = 0.5

SPEED 100 ALWAYS

MOVE safe

OPENI

BREAK

SIGNAL -3, -4, -5, -6

WAIT SIG(1001)

IF SIG(1002) THEN

SET baza = nl\_storage

ELSE

SET baza = l\_storage

END

n = 0

IF SIG(1003) THEN

n = n+2

END

IF SIG(1004) THEN

n = n+1

END

TIMER 1 = 0

FOR i = 0 TO n-1

SET pick = SHIFT(baza BY 0,0,(3-i)\*4.3)

SET place = SHIFT(baza2 BY 0,0,i\*4.3)

CALL pick\_place()

END

d = INT(TIMER(1))

TYPE "Timpul necesar : ", d, " secunde."

IF d > 7 THEN

d = 7

END

IF d BAND 4 THEN

SIGNAL 4

END

IF d BAND 2 THEN

SIGNAL 5

END

IF d BAND 1 THEN

SIGNAL 6

END

SIGNAL 3

.END

=======================================

.PROGRAM prob(base\_pallet, loc\_rel\_TE, loc\_rel\_LIG, loc\_rel\_piesa3, grip.trans.te, grip.trans.lig, safe)

DECOMPOSE griplig[] = grip.trans.lig

DECOMPOSE gripte[] = grip.trans.te

MOVE safe

OPENI

VPICTURE(cam, -1) 2

VWAIT

VFINDER(cam, 1, -1, -1)

VWAIT

FOR i=1 TO 100

SET offset\_paleta = SHIFT base\_pallet BY (((i-1) MOD 10)\*200, INT((i-1) / 10)\*200, 0)

VLOCATE(cam, 2, 0) "LIG", L.pos

IF VFEATURE(1) == FALSE THEN

GOTO 10

END

SET grip = TRANS(0, 0, griplig[2], 0, 180, 0)

SET dest = offset\_paleta:loc\_rel\_LIG

CALL p\_p(L.pos, dest, grip)

VLOCATE(cam, 2, 0) "TE", TE.pos

IF VFEATURE(1) == FALSE THEN

GOTO 10

END

SET grip = TRANS(0, 0, gripte[2], 0, 180, 0)

SET dest = SHIFT offset\_paleta BY loc\_rel\_TE

CALL p\_p(TE.pos, dest, grip)

gasit\_cerc = FALSE

WHILE gasit\_cerc == FALSE DO

VLOCATE(cam, 2, 0) "?", NEC.pos ;iau pe rand toate obiectele necunoscute

IF VFEATURE(1) == FALSE THEN

GOTO 10

END

SET arie = VFEATURE(10) ;aria obiectului

SET gauri = VFEATURE(17) ;numarul de gauri al obiectului

SET perim = VFEATURE(41) ;perimetrul obiectului

if( (Amin <= arie AND arie <=Amax) AND (Pmin <= perim AND perim <=Pmax) AND gauri == 3) ;daca respecta toate regulile atunci este cerc

SET grip = TRANS(0, 0, 0, 0, 180, 0)

SET dest = SHIFT offset\_paleta BY loc\_rel\_piesa3

CALL p\_p(NEC.pos, dest, grip)

gasit\_cerc = TRUE

END

END

END

MOVE safe

OPENI

RETURN

10: TYPE "Nu avem suficiente piese"

.END

.PROGRAM p\_p(source, dest, grip)

PARAMETER HAND.TIME=0.7

SPPED 85 ALWAYS

SET piesa.loc = to.cam:source:grip

APPRO piesa.loc, 100

SPPED 55

BREAK

MOVES piesa.loc

CLOSEI

DEPARTS 100

BREAK

APPRO DEST,100

SPPED 55

BREAK

MOVES dest

OPENI

DEPARTS 80

BREAK

RETURN

.END

=======================================

.PROGRAM paletizare(ordine)

$ip = "172.16.200.20"

PARAMETER VTIMEOUT =5

PARAMETER HAND.TIME = 0.5

RIGHTY

MOVE safe, 2

BREAK

VPARAMETER(1, 2, 530, 1, 1) $ip = ordine

loop = TRUE

WHILE(loop) DO

MOVE safe

BREAK

loop = false

VRUN $ip, 1

nrobj = VRESULT($ip, 1, 2, 1, 1310, 1, 1)

type nrobj

IF(nrobj > 0) THEN

FOR i = 0 TO nrobj

amp1 = VRESULT($ip, 1, 3, 1, 1506, 1, i)

amp2 = VRESULT($ip, 1, 3, 1, 1506, 1, i)

tip = VRESULT ($ip, 1, 2, i, 1312, 1, 1)

IF(am1 > 200) AND (amp2 > 200) AND tip == 0 THEN

loop = TRUE

SET sursa = VLOCATION ($ip, 1, 2, 1, 1311, 1, 1)

APPRO sursa, 50

BREAK

SPEED 20

MOVES sursa

CLOSEI

DEPARTS 50

BREAK

APPRO place, 50

BREAK

SPEED 20

MOVES place

OPENI

DEPARTS 50