UPDATED for TEST 2

|  |  |  |  |
| --- | --- | --- | --- |
| PARAMETER | Purpose? | Used |  |
| #<P> | argspec for M11/M12 | M10: reads for #<steps> calc  M11/M12: reads #<p> |  |
| #5170 | current pocket | M13: writes to "1" after homing |  |
| #<steps> | calculate pockets to move | M10: writes after (#5170 - #<p>) | Sets argspec "P" for M11/M12 |
| #<steps\_to\_move> | calculate pockets to move | M11/M12: writes #<steps\_to\_move> = #p> |  |
| #5171 | "0" ATC not homed  "1" ATC is homed | M10: reads for homed, M13 if not  M13: writes to "1" after homing |  |
|  |  |  |  |
| #5189 | unknown | M6: reads in tool-in-pocket calc  M13: reads in formula for widget |  |
| #5231 | current spindle tool | M13: reads for M61 (LCNC set tool) |  |
| #5190-#5207 | represent pocket #'s | M6: writes to #1-#18 |  |
| #100 | #<selected\_tool> | M6: writes selected tool to #100 |  |
| #110 | #<tool\_in\_spindle> | M6: writes tool in spindle to #110 |  |
| #120 | #<\_selected\_pocket> | M6: writes selected pocket to #120 |  |
| #<selected\_tool> | stdglue | M6: reads for #100 |  |
| #<tool\_in\_spindle> | stdglue | M6: reads for #110 |  |
| #<\_selected\_pocket> | stdglue | M6: reads for #120  M10: reads for carousel destination |  |
| #<next\_pocket> |  | M6: writes #<next\_pocket> = 0 for calc  M6: writes #<next\_pocket> = number of pockets (#15)  M6: reads #<next\_pocket> for tool in ATC check  M6: reads #<next\_pocket> to set M10 P argspec  M6: reads #<next\_pocket> to move widget  M6: reads #<next\_pocket> |  |
| #<open\_pocket> |  | M6: writes #<open\_pocket> = 0 for calc  M6: writes #<open\_pocket> = number of pockets (#15)  M6: reads #<open\_pocket> for open pocket check  M6: reads #<open\_pocket> to set M10 P argspec  M6: reads #<open\_pocket> to move widget  M6: reads #<open\_pocket> to write to #30 |  |
| #<\_my\_current\_pocket> |  | M11/M12: writes#<\_my\_current\_pocket> = #5170 | Not used anywhere |
| #30 |  | M6: writes #30 = #<open\_pocket>  M6: writes ##30 = #110 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NAME | FUNCTION | MCODES INSIDE | NEW FUNCT | NEW MCODES |
| toolchange.ngc | main program | M10 M21 M21 | main program |  |
| M10 | Check and home carousel then  move carousel best direction to pocket | M11 M12 M13 | check and home carousel |  |
| M11 | Move carousel CCW parameter P number of steps  Update current pocket | M13 | JOG FWD | M13 |
| M12 | Move carousel CW parameter P number of steps  Update current pocket | M13 | JOG REV | M13 |
| M13 | Home carousel | none | home carousel | none |
| M21 | Lower head to ZLow  orient spindle  move carousel to tool change  M24 unclamp tool (activate PDB)  Raise head to ZHigh  Update no tool in spindle | M19 M24 | Unload ATC to spindle | M24 |
| M22 | orient spindle  M24 unclamp tool (activate PDB)  Lower head to ZLow  clamp tool (deactivate PDB)  retract carousel | M19 M24 | Load tool from ATC to spindle | M24 |
| M24 | activate PDB | none | same |  |
| M25 | extend carousel | none |  |  |
| load\_spindle\_safety.ngc | check for tool number already in carousel  update tool in spindle number |  |  |  |
| store\_tool\_in\_carousel.ngc | cancel tool length comp G49  set tool in spindle to 0  call T0 M6 | M6 |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

2022-07-20 notes

* M10
  + Change pocket number line to M68 E0 Q#<P> (set desired pocket to argspec "P" from toolchange.ngc
  + Re-incorporate "steps" calc for ATC widget
  + Figure out WTF #5189 is for

2022-07-16 notes

* #5170 is current pocket
* M13 new calc
  + move carousel to home
  + rotate widget to pocket 1
* M10 new calc
  + determine how many pockets to move
  + then output to carousel an exact pocket number
  + then update current pocket #5170
* toolchange.ngc
  + O100 do
  + O110 if [#[5189 + #15] EQ #100] ; checks all pockets to see if it contains tool number requested as the new tool
  + #<next\_pocket> = #15 ; if tool is found in pocket, assigns the next pocket
  + O110 endif
  + O115 if [#[5189 + #15] EQ 0] ; checks if the pocket is empty, last pocket checked will be the lowest empty pocket number, for putting tool in spindle away.
  + #<open\_pocket> = #15
  + O115 endif
  + #15= [#15 - 1]
  + O100 while [#15 GT 0]
  + O100 do
  + O110 if ATC pockets = desired tool
  + next pocket = number of total ATC pockets
  + O115 if ATC pockets = 0
  + open pocket = number of total ATC pockets
    - #15= [#15 - 1] (decrement atc pockets to keep checking through the pockets)
  + O100 while [#15 GT 0] (keep doing this until you run out of pockets)

toolchange.ngc

* <parameter> Assign tool numbers to pocket numbers based on parameters
* <parameter> Assign M6 parameters to local parameters
* Check if tool in spindle = selected tool
  + M2 if true
* <parameter> Assigns 0 to the next pocket for a later check if the tool is found in the carousel
* <parameter> Assign test parameter the number of pockets in the carousel
* Checks all pockets to see if it contains tool number requested as the new tool
  + If tool is found in pocket, assigns the next pocket
* Checks if the pocket is empty, last pocket checked will be the lowest empty pocket number, for putting tool in spindle away.
* If tool is not found, aborts and sends a message
* If tool in spindle is not 0, put it in ATC
  + Check if there is an open pocket
  + Abort if no open pocket
  + **M10 -** Rotate to selected pocket
  + **M21**
    - Lower head to ZLow
    - orient
    - turn on extend solenoid
    - check ATC extended
    - **M24 -** activate PDB (unclamp tool)
      * check PDB active
    - Raise head to ZHigh
    - <parameter> empty tool # in the spindle
  + <parameter> save tool number in pocket
  + <parameter> empty tool # in the spindle
* Raise head to ZHigh
* If selected tool **is not** tool 0, grab it from ATC
  + **M10 -** Rotate to selected pocket
  + **M65 P2** - turn off retract solenoid
  + **M64 P1** - turn on extend solenoid
  + **M66 P1** - Check extend sensor
  + **M65 P1** - Turn off extend solenoid
  + **M22**
    - orient
    - **M24** activate PDB (unclamp tool)
      * check PDB active
    - Lower head to ZLow
    - Deactivate PDB (clamp tool)
    - retract ATC
  + <parameter> pocket now empty
  + <parameter> Set persistent variable to remember tool in spindle after power cycle
* If selected tool **is** tool 0
  + deactive drawbar
  + turn off extend solenoid
  + turn on retract solenoid
  + check ATC retracted
  + turn off retract solenoid
* <parameter> assign the tool numbers to the pockets

Parameters

* #15 (number of pockets in carousel)
* #5170 (current pocket #)
* #5171 (ATC homed)
* #5189
* #5231 (tool in spindle)
* #5399 (timer from M66 wait on input)
* #<Steps\_to\_move>
* #<\_my\_current\_pocket>
* #<steps> (= #5170 - #<p>)
* #<open\_pocket>

UPDATED for TEST

|  |  |  |
| --- | --- | --- |
| M6 | main program |  |
| M10 | Home carousel, check if homed | DONE |
| M11 | Jog carousel FWD | DONE |
| M12 | Jog carousel REV | DONE |
| M13 | Store tool in carousel |  |
| M14 | Load spindle safety check |  |
| M21 | Extend ATC, check for in position | DONE |
| M22 | Retract ATC, check for in position | DONE |
| M23 | Raise head |  |
| M24 | Lower head |  |
| M25 | Unclamp PDB, check for unclamped |  |
| M26 | Clamp PDB, check for clamped |  |

EXISTING for PROBE BASIC

Parameters

* #15 (number of pockets in carousel)
* #5170 (current pocket #)
* #5171 (ATC homed)
* #5189
* #5231 (tool in spindle)
* #5399 (timer from M66 wait on input)
* #<Steps\_to\_move>
* #<\_my\_current\_pocket>
* #<steps> (= #5170 - #<p>)
* #<open\_pocket>

M-codes

* M10 - rotate carousel, formula to select M11 or M12
  + M13, or
  + M11/M12
* M11 - Rotate carousel CCW Parameter P number of steps
  + M13, or
  + M64 P4 (ATC motor rev)
  + M66 P4 (wait for index input)
  + M65 P4 (stop ATC motor)
* M12 - Rotate carousel CW Parameter P number of steps
  + M13, or
  + M64 P3 (ATC motor fwd)
  + M66 P4 (wait for index input)
  + M65 P3 (stop ATC motor)
* M13 - Rotate tool carousel CW until ATC zero (homing)
  + M64 P3 (ATC motor fwd)
  + M66 P3 (wait on input)
  + M65 P3 (stop ATC motor)
  + #5170 = 1
  + #5171 = 1
  + M61 Q5231 (set current tool in parameters)
* M21 - Move carousel OUT, unclamp tool from spindle
  + M65 P1 (carousel in solenoid OFF)
  + M66 P1 (wait on input: carousel in sensor)
  + M5, M9
  + G0 G53 Z-3.1900 (move Z to LOW position)
  + M19 (orient spindle)
  + M64 P0 (carousel out solenoid ON)
  + M66 P1 (wait on input: carousel out sensor)
  + M24 (unclamp tool)
  + M5 (turn off spindle lock)
* M22 - Move carousel IN, clamp tool in spindle
  + M19 (orient spindle)
  + M24 (unclamp tool)
  + M19 (orient spindle)
  + G0 G53 Z-3.9 (move to above the toolchange position)
  + M65 P2 (clamp tool)
  + M5
  + M66 P5 (wait on input: tool clamped sensor)
  + M65 P0 (move carousel home)
  + M66 P0 (wait on input: carousel home sensor)
* M24 - Unclamp tool
  + M64 P2 (unclamp tool)
  + M66 P2 (wait on input: tool unclamped sensor)
    - M65 P2 (turn off drawbar if timed out)
* M25 - Move carousel out
  + M64 P0 (move carousel out)
  + M66 P1 (wait on input: carousel out sensor)

toolchange macros

|  |  |  |
| --- | --- | --- |
| toolchange.ngc | main program |  |
| M10 | rotate carousel |  |
| M11 | rotate CCW | Change to position mode |
| M12 | rotate CW | Change to position mode |
| M13 | home carousel | Change to homing |
| M21 | move to spindle, unclamp | need |
| M22 | release tool, move to home | need |
| M24 | unclamp tool then turn off drawbar |  |
| M25 | move to spindle (manual control) |  |
|  |  |  |
|  |  |  |

toolchange\_test01

(M73) restore modal settings after toolchange

CHECK: Z-homed

Home-Z if not

CHECK: B-homed

Home-B if not

PARAMETERS:

newtool = selected\_tool

oldtool = tool\_in\_spindle

oldpocket = #5190

M66 E0 (gets prep-pocket)

newpocket = #5399 (result of M66)

#5190 = #5399 (change 5190 to newpocket

O100 => #<\_task> return loop

G90 G40 G49 (safety codes)

CHECK: if oldtool = newtool

CHECK: if oldtool > 0

CHECK: if oldpocket > 0

CHECK: if oldpocket is valid

SPINDLE FULL:

MOVE: G53 Z-low

ORIENT spindle

CHECK: spindle oriented

IF oriented, proceed

IF not, ABORT

TRIGGER: carousel extend ON (M64 Px)

CHECK: carousel extended (M66 Px)

IF extended, proceed

IF not, ABORT

TRIGGER: unclamp (M64 Px)

CHECK: unclamped (M66 Px)

IF unclamped, proceed

IF not, ABORT

MOVE: G53 Z-high

MOVE: rotate to newtool

MOVE: G53 Z-low

TRIGGER: clamp (M65 Px)

CHECK: clamped (M66 Px)

IF clamped, proceed

IF not, ABORT

TRIGGER: carousel extend OFF (M65 Px)

TRIGGER: carousel retract ON (M64 Px)

CHECK: carousel retracted (M66 Px)

IF retracted, proceed

IF no, ABORT

TRIGGER: carouseld retract OFF (M65 Px)

MOVE: G53 Z-high

UPDATE tool number & offset (G61 Q#<newtool>, G43 H#<newtool>

SPINDLE EMPTY:

MOVE: G53 Z-high

ORIENT spindle

CHECK: spindle oriented

IF oriented, proceed

IF not, ABORT

TRIGGER: carousel extend ON (M64 Px)

CHECK: carousel extended (M66 Px)

IF extended, proceed

IF not, ABORT

MOVE: rotate to newtool

TRIGGER: unclamp (M64 Px)

CHECK: unclamped (M66 Px)

IF unclamped, proceed

IF not, ABORT

MOVE: G53 Z-low

TRIGGER: clamp (M65 Px)

CHECK: clamped (M66 Px)

IF clamped, proceed

IF not, ABORT

TRIGGER: carousel extend OFF (M65 Px)

TRIGGER: carousel retract ON (M64 Px)

CHECK: carousel retracted (M66 Px)

IF retracted, proceed

IF no, ABORT

TRIGGER: carouseld retract OFF (M65 Px)

MOVE: G53 Z-high

UPDATE tool number & offset (G61 Q#<newtool>, G43 H#<newtool>

HAL/INI changes

* (HAL)connect iocontrol.tool-prep-pocket through s32float to motion.analog-in-00
* (INI) number of pockets
* (HAL) connect halui.joint.N.home to motion.digital-out-XX
* (HAL) connect halui.joint.N.is-homed to motion.digital-in-XX (net a-is-homed)