MotionCode™
Step-by-step solutions for motion applications

Galil MotionCode Solution:
Computer-Controlled Loaf Slicer (Cut-to-Length)

Contents

Figure 1.

1. Machine Description

Food processing equipment is becoming more automated as companies strive for quality, consistency, throughput, and waste-savings. Automation also eliminates human intervention and contamination. Many companies have developed machines that automate the entire process of mixing, forming, slicing, preparing, and packaging food items destined for consumer distribution. These machines often employ the use of centralized motion control to coordinate all of these processes. The centralized control is responsible for many tasks including coordinating feed conveyors, accurate portion control (size and weight), dispensing fluids, regulating cut lengths, stacking, sorting, packaging, and temperature control. With centralized control, the machine can store product-specific information so that it can quickly handle a new product.

In this application, we focus on one specific food-processing machine: the Loaf Slicer *(figure 1)*. The task of the Slicer is to control the feed of the product (a loaf of deli meat) into a constantly rotating blade so that a desired thickness is consistently obtained at high speed. The machine controls the thickness by synchronizing the feed transport with the blade rotation. Hence, by varying the gear ratio between the two axes, various thicknesses can be set.
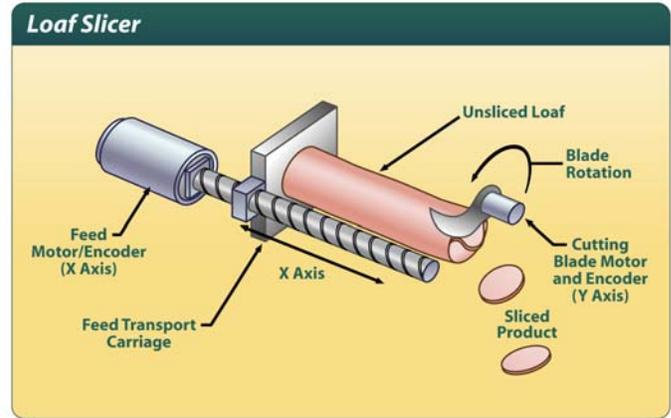
2. Requirements

This section summarizes the requirements for the machine described above:

- *X-axis linear transport with position feedback and forward and reverse limit switches*
- *Y-axis rotary blade control with position feedback*
- *Consistent thickness (regardless of feed rate)*
- *Standalone operation with HMI interface*
- *Graphical User Interface allows the user to do the following:*
  (1) Select product from a menu to set the machine feed rate, cut thickness, and loaf length
  (2) start the operation
  (3) monitor the total slice count and feed position

3. Components Selected

This section describes the Galil hardware and software products chosen to implement the machine. Below is a complete bill of materials followed by a description of major components.

Table 1. Bill of Materials

Part Number	Description	Unit Price (U.S.) Qty 1 / qty 100
DMC-2123-DC48-DIN	2 axis stand-alone motion control with DIN mount, Ethernet communications, and DC power input	$1095/ $785
AMP-20520	2-axis 500 W brushless amplifier	$595/ $395
BLM-N23-50-1000 or equivalent	Brushless servo motor with 1000 ppr encoder X2	Consult mfg.
CPS-6-48 or equivalent	6 A, 48 VDC power supply	Consult mfg.
WSDK tuning software	Windows Servo Design Kit for controller setup and tuning	$195 (one time)
QSI G124	HMI panel with Ethernet port (www.qsicorp.com)	Consult mfg.
Qlarity Foundry	HMI development software	free

Controller: DMC-2123-DC48-DIN

This application requires stand-alone operation; hence, an Ethernet controller with DIN mount is appropriate. In addition, since the application requires two axes of servo motor control, the DMC-2123 with attached AMP-20520 is most appropriate. The Ethernet port allows for easy connectivity and communications from the HMI panel to the controller.

Figure 2. DMC-2123 Motion Controller

Motor: BLM-N23-50-1000

For maintenance-free operation, we choose brushless motors. Galil's NEMA 23 #BLM-N23-50-1000 brushless motors, or equivalent, are appropriate because all axes require less than 0.3 Nm of continuous torque. Incremental encoders with 1000 cycles per revolution are installed on the motors resulting in 4000 quadrature counts per revolution. Hall sensors are not required on the motors as the incremental encoders provide commutation tracks for input to the amplifiers.

Amplifier: AMP-20520

To drive the two motors, we choose the very compact AMP-20520, which is a two-axis brushless amplifier (500 W per axis) that directly mounts to the top of the controller.

Figure 3. DMC-2123-DC48-DIN with attached AMP-20520

QSI G124 HMI panel

QSI offers a very affordable HMI panel *(figure 4)* that can easily be integrated into a Galil control system. The Qlarity Foundry software development environment

(figure 5) includes Galil communications drivers, which allow the user to quickly create an application that communicates with the Galil controller via Ethernet or serial.
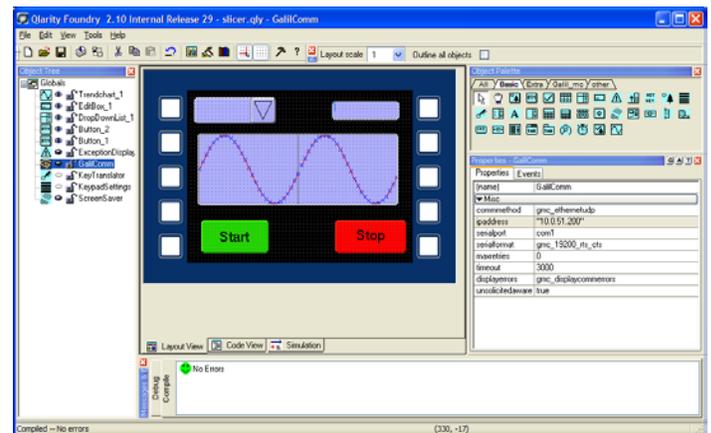

Figure 4. QSI G124 HMI panel

Figure 5. Qlarity Foundry HMI Development Software

4. Implementation

This section details how the components selected above were used to implement the control system.

Graphical User Interface

The QSI G124 panel is programmed with a touchscreen application *(Figure 6)*, which features a Start button, a product menu, a counter display, and a progress bar (actually feed position).

Figure 6. HMI Display at Runtime

Upon power-up, the feed axis is homed to a limit switch and the QSI panel automatically connects via Ethernet to the DMC-2123 stand-alone two-axis controller. It then prompts the operator to enter the product code from the pull-down menu. The QSI panel automatically updates the speed, thickness, and length into the Galil controller when the menu item is selected.

The machine starts when the operator pushes the "Start" button. The operation ends when the product feed transport reaches the end of travel, where it automatically returns to the beginning.

Values

Table 2. below details the values used in the application code that characterize the loaf slicer:

Table 2. Slicer Values

Requirement	Value	Units
Range of cut thickness	0.25 to 25	mm
Feed transport resolution	800 (5 mm lead pitch)	counts/mm
Maximum feed speed	300	mm/s
Maximum blade speed	480	RPM
Feed Acceleration	5000	mm/s^2

Gearing (GA, GR)

By utilizing the gearing mode (GA) of the Galil controller, the thickness is made consistent by gearing the feed transport travel (slave) to the blade rotation (master). By selecting a proper gear ratio (which is derived from the feed resolution and the blade modulo), the feed transport can advance forward a specific distance for every cycle of the blade regardless of feed rate. If the desired thickness is 0.25" (or 5080 cnts of the transport travel) and the blade modulo cycle is 4000 counts, the gear ratio would be 5080/4000 (or 1.27).

Galil Code

The DMC program below is downloaded to the DMC-2123 via the Ethernet port and is saved to non-volatile memory using the BP command. It consists of three routines: #AUTO initializes variables and runs when the controller boots, #Go runs when the user hits the Start button and ends when the loaf is fully sliced, and #Monitor runs continuously to update the Cut Count and Progress displays on the touchscreen. The comments below explain the code in detail.

Figure 7. Galil Application Code for Loaf Slicer

```
#AUTO                           ;'Label for auto-start (runs on power up)
  'Initialize variables passed to touchscreen
  CutCount = 0                  ;'Initialize cut counter variable
  Progress = 0                  ;'Initialize progress status variable

  'These variables are set by the touchscreen. Default is Bologna.
  GrRatio = 1.27                ;'Gear ratio variable (defaults to 1.27)
  BladeSpd = 16000              ;'Blade speed set point (defaults to 16000)
  EndPoint = 243840             ;'End of transport travel (default 12")

  AC 4000000,4000000            ;'Acceleration in counts/sec2
  DC 4000000,4000000            ;'Deceleration in counts/sec2

  'Home feed to reverse limit then index
  JGX = -10000                  ;'Jog feed towards reverse limit
  BGX                           ;'Begin motion
  AMX                           ;'Wait until motion is complete

  JGX  = 500                    ;'Set slow feed speed to find index
  FIX                           ;'Find feed index
  BGX                           ;'Begin feed motion
  AMX                           ;'Wait until feed motion is complete

  XQ #Monitor, 1                ;'Start monitor program in thread 1
EN                              ;'End of initialization
                                 routine
```

(Continued next page)

```
#Go                                  ;'Label for main motion program (called by touchscreen)

   'Cut the product
   GAY                               ;'Set gearing axis master (Y is master of X)
   GRX = GrRatio                     ;'Set gear ratio
   JGY = BladeSpd                    ;'Set speed of blade
   BGY                               ;'Begin motion on feed transport
   MFX = EndPoint                    ;'Wait for end of travel
   STY                               ;'Stop at end
   AM                                ;'Wait for motion complete
   GRX = 0                           ;'Disengage gearing

   'Send feed back
   SPX = 100000                      ;'Bring the feed back quickly
   PAX = 0                           ;'Send feed transport back to 0
   BGX                               ;'Begin motion on feed axis
   AMX                               ;'Wait for motion complete
EN                                   ;'End program

#Monitor                             ;'Monitors cut count and progress
   CutCount = @INT[_TPY / 4000]      ;'Calculate # of cuts
   Progress = 100 * (_TPX / EndPoint) ;'Calculate Progress (0 to 100 percent)
JP #Monitor                          ;'Loop program
```

QSI Code

This section describes the QSI code that allows the user to select between Bologna and Ham. This is the only code written for the QSI application. The rest was implemented by dragging objects onto the screen and setting their properties.

Within the QSI panel, a DropDownList object is programmed such that a specific gear ratio, feed rate, and length are automatically sent to the controller when a selection is made. The Qlarity code below shows the structure for selecting and sending the required parameters to the controller.

Figure 8. QSI DropDownList code. 12" loaf of bologna cut at 1/4" at 4 cuts/sec. 9" loaf of ham cut at 1/8" at 6 cuts/sec

```
func valuechanged()
   if DropDownList_1.selected == 0 then 'user picked bologna
     GMCCommandExec_1.command = "GrRatio=1.27;BladeSpd=16000;EndPoint=243840"
   GMCCommandExec_1.executecommand=true
   endif

   if DropDownList_1.selected == 1 then 'user picked ham
     GMCCommandExec_1.command = "GrRatio=0.635;BladeSpd=24000;EndPoint=182880"
     GMCCommandExec_1.executecommand=true
   endif
endfunc
```