Release Notes for Oak/Allin1DC/Mpu11 CNC12 v4.22 12-6-22

Changes, Fixes and Additions from previous version (v4.20)

- Fixed an issue with systest.cnc where it does not recognize if one of the axes (5-8) is a rotary and therefore allowed to have no travel limit.

- Fixed execution times for PLC programs displayed in live PLC diagnostics to be compatible with faster hardware. The displayed times were based upon a 400 MHz Blackfin CPU speed (older systems) and now work with the faster 528 MHz hardware.

- Change: any report_*.zip files that are placed into the extra-report-files folder will not be included in the report zip file. The extra-report-files folder is the place to put files that a user wants included in the report zip file that are not already included in a report, but it is inviting some users to store report zip files there by default, with the unintended consequence of making the report zip files grow increasingly larger. In addition, when choosing a location to save the report, choosing the extra-report-file folder or one of it's subdirectories is not allowed and will result in a message prompt to choose another location.

- MPUCOMP/MPU: support for 128 timers in the PLC program, PLC Detective, and in G-code user variables #90001-#90128.

- Lathe: RTG will show a triangle tool tip for nose vectors 5 and 7 when using a turning type tool.

- Changes to optimize RTG/backplotting time= Faster Backplots!

- Lathe: fixed tool offset measuring bugs.

- Implemented the Proportional Coefficient Scale Algorithm which uses machine parameters 620-635. Parameters 620-627 specify a proportional constant (the P in PID control) for axes 1-8 and machine parameters 628-635 specify a max correction velocity in encoder counts per interrupt. If the proportional constant is non-zero, then the scale velocity is computed using the proportional constant multiplied by the scale error and is bounded to the max correction velocity. If the proportional constant is zero, then the scale velocity specified in the Scale Settings menu in CNC software is used. This feature may be helpful for systems that have large backlash amounts for which a set velocity reacts too slowly and raising the scale velocity to compensate results in instability. Note that a non-zero value for the fourth or fifth axis will override the dual velocity scale algorithm for these axes (see machine parameters 600-605).

- Added the ability to perform on-the-fly switching between fast and slow jog. This was to support a retrofit that had a jogging joystick. The joystick could be moved in four directions and would move slow at first but then it could be pushed further to go into a fast mode.

- Changed the wording for machine parameter 180 in the parameters menu and in the manual from "Clear Limit Switch Distance" to "Clear Home Switch Distance".

- Change: user variables #25024-#25026 (added in Rev2) have been moved to #25029-#25031 to resolve a future merge conflict.

- Added user variables #27801 - #27808 for position mode error correction values, the same that is in the PID Encoder Menu. There is also access in the PLC program via system variables SV_POSITION_MODE_ERROR_AXIS_1-8.

- Added user variable #25026 to access lifetime spindle on time in floating point hours.

- Added user variable #25025 to access lifetime job in progress time in floating point hours.

- Added user variable #25024 to access lifetime power on time in floating point hours.

- Skinning API: Added function Tool.GetToolLibrary. See the Skinning API documentation for more information.

- SKinning API: Added function Job.ContinueExecution. See the Skinning API documentation for more information.

- Added machine parameter 440 (Stop for Jogging Continue Bit). A value of 0 is the stadard behavior wherein we wait for CYCLE START or TOOL CHECK when at an M0/M200/M201 prompt (and M0 jogging is allowed) before continuing execution. A value of 1-255 will also continue execution when the system varibale SV_SKIN_EVENT_1 - 255 is set. A value of 256 will also continue execution after the Skinning API Job.ContinueExecution function call.

- Fixed Lathe spindlebenchtest.cnc trying to write to a text file with the wrong (cncm) path.

- Fixed intermittent issue where the PLC non-volatile variables (SV_NV_Wx and SV_NV_FWx) were not getting saved.

- Change: to enable M0 jogging, parameter 10 (Macro M function handling) bit 5 (+32) must be set. Note that for servo controls, a trial code or Pro/Ultimate license is also required.

- A warning message is displayed and the job is stopped if trying to execute a G53 command in incremental (G91) mode.

- Lathe: moved the F9 Exit/CNC12 function key to F7 to prevent unintended exit from the CNC software when trying to use the F9 Tool Wear from the main menu (which can register as an F10 Shut/Down followed by an F9 Exit/CNC12 if one is not careful).

- Lathe Intercon: a tool with spindle direction of NSP (No SPindle) will be post with an M5, even if the Setup option "Stop spindle during tool change" is set to "No". More specifically, this fixed an issue using a bar puller tool where the spindle would keep turning according to the previous tool's spindle direction.

- Added the PLC system variable bits SV_HOME_SET_AXIS_1 - SV_HOME_SET_AXIS_8. These are for the PLC to access the individual home set state. There are some short-comings of this implementation right now. For example, if CNC software un-homes an axis, that does not get reflected, but we have G-code user variables to access that information if needed. The way they behave now is that they get set when the MPU receives a command to set home for an axis and they stay set until MPU reboot.

- Change: if the USB MPG feature is not unlocked, then do not force parameter 218 (USB MPG Options) to zero. This change will allow the USB MPG to start working after a trial code or license re-enables it without requiring the user to set parameter 218 again.

- Added the PLC system variable bits SV_ENCODER_INDEX_PULSE_1 - SV_ENCODER_INDEX_PULSE_21 that return the status of the encoder index pulse signals.

- Added the PLC system variable bits SV_LATCHED_ENCODER_INDEX_PULSE_1 - SV_LATCHED_ENCODER_INDEX_PULSE_21. These variables get set when an index pulse occurs and remain set until they are read by the PLC program.

- Added the PLC system variable bits SV_SCALE_INITIALIZED_1 - SV_SCALE_INITIALIZED_8 that are set once the axis scale is initialized, which occurs after the axis is homed. It is also normal for a scale to become uninitialized when starting a homing command. There is also user variable acces via #27601 - #27608.

- Added the PLC system variable bits SV_SCALE_ENABLED_1 - SV_SCALE_ENABLED_8 that reflect when the scale is enabled, the same that is viewed in the Scale menu of CNC software. There is also user variable acces via #27701 - #27708.

- Added two 32-bit integer word PLC system variables, SV_ENCODER_DIFF_BITS and SV_ENCODER_QUAD_BITS, that each have a bit for each encoder (up to 21) to indicate whether the encoder has a differential or quadrature error.

- Change: when using an absolute motor encoder with an incremental scale encoder and home is set, the scale is set to the equivalent motor position. While this will prevent a slow drift away after home is set, which is what happens if you set the scale to zero, such a machine configuration should be reconsidered due to repeatability issues.

- Changed the behavior of the PLC system variables SV_DAC_OUTPUT_ENABLE_x and SV_DAC_OUTPUT_VALUE_x so that if the DAC output is enabled and the value corresponds to a zero output, the axis is disabled.

- Fixed issue where laser tables/lash was not working correctly if the axis was configured for an absolute encoder.

- Lathe change: when the spindle encoder counts (P34) or spindle axis (P35) is changed, the updated information is sent to the MPU. This change was to support an application where the updated information was needed without requiring a restart.

- Added the M291 command to reset the MPG offset. This works for the first MPG and the Z axis, though we expect to add the capability to use it for another axis and a second or third MPG.

- Changed the bound on the maximum MPG encoder input from 15 to 21. This should allow MPG encoder input to be connected to ENCEXP/ENCEXP2 hardware, which have encoder inputs that are mapped to values 16 – 21.