

Pulse Picker MCode Pseudocode & Reference

- Global Variable Declarations
 - Legacy MCode Variables
 - Pulse Picker Variables
- Program 1
 - Initialization (LB SU)
 - Mode Select Loop (LB Za)
 - Mode 1: One-Shot (LB Z1)
 - One-Shot ISR 1 (P1 to P2 move) (LB J1)
 - One-Shot ISR 2 (P2 to P1 move) (LB J2)
 - Mode 2: Flip-Flop (LB Z2)
 - Flip-Flop ISR 1 (P1 to P2 move) (LB J5)
 - Flip-Flop ISR 2 (P2 to P1 move) (LB J6)
 - Mode 3: Burst (LB Z3)
 - Burst Open ISR (LB J3)
 - Burst Close ISR (LB J4)
 - Mode 4: Fast Open (LB Z4)
 - Mode 5: Fast Close (LB Z5)
 - Mode 6: Follower (LB Z6)
 - Follower Open ISR (LB J7)
 - Follower Close ISR (LB J8)
 - Mode 7: Home to Index Mark (LB Z7)
 - Subroutines
 - Blocking move to nearest closed position (LB X5)
 - Realign by computing new open-loop position (LB X7)
- Mode/State Tables
 - Se - Mode Select
 - Sd - Status Indicator
 - Df - Slit Open/Closed Status
- Velocity Tables
 - Maximum Velocity/Acceleration (V0)
 - One-Shot Velocity Defaults (V1)
 - Flip-Flop Velocity Defaults (V2)
 - Burst Velocity Defaults (V3)
 - Fast Open Velocity Defaults (V4)
 - Fast Close Velocity Defaults (V5)
 - Follower Velocity Defaults (V6)
 - Homing/Alignment Velocity Defaults (V9)
- Magic Offset (Mo) Table
- Changelog
 - Version Ns=96 (pcds_motion R2.5.0)
 - Version Ns=94 (pcds_motion R2.3.14)
 - Version Ns=93 (pcds_motion R2.3.13)
 - Version Ns=92 (pcds_motion R2.3.12)
 - Version Ns=91 (Non-Release)



Revision Info

This page currently corresponds to the **Ns=96** revision of the MCode. This revision resides in release **R2.5.0** of `pcds_motion`.



Motors with PCDS MCode Loaded

Any motors with standard PCDS MCode loaded should be purged and reset with the **FD** command before being loaded with pulse picker MCode. This avoids any interference between variables and code labels.

Global Variable Declarations

Legacy MCode Variables

Kept for compatibility. These are:

- **Ve** - MCode version. if **Ve != DVER**, MCode gets uploaded by IOC.
- **Pu** - Power-up status flag

Pulse Picker Variables

- **Se** - Pulse Picker mode [Se]lect. Refer to Mode List for a complete list of Se modes.
- **Sd** - Status indicator register. Used for debugging. Refer to Status List for a complete list of statuses.

- **Sw** - [Sw]EEP distance. Number of motor steps to equal 11.25 degrees of motion.
- **Sr** - Encoder sweep distance. Number of encoder steps to equal 11.25 degrees of motion.
- **Dr** - Encoder [Dr]ift, or the difference between the encoder count the predicted encoder position.
- **DI** - Encoder [D]rift [L]imit, Ud or Ld flags will be raised if encoder drift [Dr] exceeds this limit. This can be changed by the user, but is reset to 46 encoder ticks (4 motor steps, roughly 1 degree) if the motor is power-cycled.
- **P0** - Center (open) position, in motor steps.
- **P1** - Positive closed position, in motor steps, nominally P0+Sw. (Repurposed from standard MCode program.)
- **P2** - Negative closed position, in motor steps, nominally P0-Sw.
- **N0** - Center (open) position, in encoder counts.
- **N1** - Positive closed position, in encoder counts, nominally N0+Sr.
- **N2** - Negative closed position, in encoder counts, nominally N0-Sr.
- **Ec** - Error checking flag. (0=OFF, 1=ON)
- **Ns** - Pulse Picker MCode version. (Repurposed from standard MCode program.)
- **Hb** - Heartbeat counter, incremented approx. every 100ms.
- **Cs** - Encoder counter shadow register (=C2)
- **Ud** - Drift violation counter, incremented at each + drift violation. Cleared upon homing the motor.
- **Ld** - Drift violation counter, incremented at each - drift violation. Cleared upon homing the motor.
- **Df** - Motor position flag (see interpretation below)
- **Mo** - Experimentally-determined magic offset, in motor steps, that is the distance between the negative side of the homing index mark and P1. This number will be *negative* so that it feeds directly into an MR command. *This has no initializer and depends on being correctly set by the engineer!*

Program 1

Initialization (LB SU)

Code:

- If Se<>0, we are likely experiencing a hard reset. In that case:
 - Signal state change to 80 (Hard Reset Initialization)
 - Set Ve (MCode version) to 3 and Pu (Power-Up flag) to 0.
 - Set Se (Mode select) to 99. This means little now, but it gets saved to NVRAM, so that in event of a power cycle, this hard reset code section runs.
 - Set EL (Encoder lines) to 4096, giving $4096 \text{ lines} * 4 \text{ steps/line} = 16384 \text{ steps/rev}$
 - Set MS (Microsteps) to 8, giving $8 \text{ uSteps/step} * 200 \text{ step/rev} = 1600 \text{ uSteps/rev}$
 - Set EE (Encoder enabled) to 0, forcing open-loop step mode. Closed-loop mode is too slow because of a 1kHz loop rate.
 - Set S3 (I/O 3 settings) to 16,0,1. This is used for encoder power - sourcing output, active low.
 - Set S4 (I/O 4 settings) to 16,0,0. This is used for fan power - sinking power, active low.
 - Set S13 (I/O 13 settings) to 60,0,0. This is used for input signaling, and makes it available to the input capture trip (interrupt)
 - Set FC (Filter Capture) to 9. This sets a debounce filter on input 13 with a 12.9uSec delay time.
 - Set HT (Hold Time) to 5, setting the hold current delay time to 5ms.
 - Set RC (Run Current) to 100, making 100% of current available for movement.
 - Set HC (Holding Current) to 5.
 - Set NE (Numeric Enable) to 0, to prevent errant communications as being interpreted as movement commands.
 - Set Sw (Sweep Distance) to 50 steps.
 - Set Sr (Encoder Sweep Distance) to 512 counts.
 - Reset Dr (Drift) to 0.
 - Reset DI (Drift limit) to 23.
 - Reset P0 (Open position) to 0.
 - Reset P1 (Positive closed position) to P0+Sw = 512.
 - Reset P2 (Negative closed position) to P0-Sw = -512.
 - Reset N0 (Open encoder position) to 0.
 - Reset N1 (Positive closed encoder position) to N0+Sr.
 - Reset N2 (Negative closed encoder position) to N0-Sr.
 - Pre-load shadow register Cs=C2 (*this doesn't actually do anything and may be eliminated in a future revision*)
 - Set Ec=1 to turn error checking on.
 - Reset drift violation counters Ud and Ld to 0.
 - Reset Df to 99 (current slit position is unknown)
 - Save all settings to NVRAM (S command)
- For both hard and soft resets, we do the following
 - Signal state change to 81 (Soft Reset Initialization)
 - Disable any trips (TE=0)
 - Hold for 100ms to let things stabilize.
 - Reset Hb (heartbeat) to 0.
 - Set Se (Mode Select) to 0
 - Set O3 (Output 3) to 1, turning the encoder on. *This is necessary because O3 is not saved in NVRAM.*
 - Set O4 (Output 4) to 1, turning the fan on. *This is necessary because O4 is not saved in NVRAM.*
 - Goto Mode Select Loop

Mode Select Loop (LB Za)

- Mode Select Loop START (forever):
 - Signal state change to 0 (Mode Select)
 - If Se=1, branch to Mode 1: One-Shot

- If Se=2, branch to Mode 2: Flip-Flop
 - If Se=3, branch to Mode 3: Burst
 - If Se=4, branch to Mode 4: Fast Open
 - If Se=5, branch to Mode 5: Fast Close
 - If Se=7, branch to Mode 7: Home
 - Increment heartbeat
 - Hold for 100ms
- Mode Select Loop END

Mode 1: One-Shot (LB Z1)

- Signal state change to 10 (One-Shot Init)
- Load One-Shot speeds (CL V1)
- Move the shutter to the nearest closed position (CL X5)
- If the shutter is in the negative closed position:
 - Specify One-Shot ISR 2 (J2) to run on Input Trip
- Else
 - Specify One-Shot ISR 1 (J1) to run on Input Trip
- EndIf
- Arm Input Trip (TE=4)
- Signal state change to 11 (One-Shot Loop)
- One-Shot Loop START (forever):
 - Increment heartbeat
 - Hold for 100ms
 - If user sets the soft reset (Se=0)
 - Branch to Mode Select (Mode 0)
 - EndIf
 - If user requests fast close (Se=5)
 - Branch to Fast Close (Mode 5)
 - EndIf
- One-Shot Loop END

NOTE: Once in the one-shot loop, you must set Se=0 to get back to Mode Select.

One-Shot ISR 1 (P1 to P2 move) (LB J1)

- Move to negative closed position P2
- If error checking is enabled (Ec=1)
 - Calculate drift at previous position P1: $Dr=N1-Cs$
 - If drift is at or beyond limit ($Dr \geq Dl$)
 - Increment the lower drift violation counter Ld
 - EndIf
- EndIf
- Hold until movement is complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=-1 to signal at negative closed position
- Return

One-Shot ISR 2 (P2 to P1 move) (LB J2)

- Move to positive closed position P1
- If error checking is enabled (Ec=1)
 - Calculate drift at previous position P2: $Dr=Cs-N2$
 - If drift is at or beyond limit ($Dr \geq Dl$)
 - Increment the upper drift violation counter Ud
 - EndIf
- EndIf
- Hold until movement is complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=1 to signal at positive closed position
- Return

Mode 2: Flip-Flop (LB Z2)

- Signal state change to 20 (Flip-Flop Init)
- Load Flip-Flop speeds (CL V2)
- Move the shutter to the nearest closed position (CL X5)
- If the shutter is in the negative closed position:
 - Specify Flip-Flop ISR 2 (J6) to run on Input Trip
- Else
 - Specify Flip-Flop ISR 1 (J5) to run on Input Trip
- EndIf
- Arm Input Trip (TE=4)
- Signal state change to 21 (Flip-Flop Loop)
- Flip-Flop Loop START (forever):

- Increment heartbeat
 - Hold for 100ms
 - If user sets the soft reset (Se=0)
 - Branch to Mode Select (Mode 0)
 - EndIf
 - If user requests fast close (Se=5)
 - Branch to Fast Close (Mode 5)
 - EndIf
- Flip-Flop Loop END

NOTE: Once in the flip-flop loop, you must set Se=0 to get back to Mode Select.

Flip-Flop ISR 1 (P1 to P2 move) (LB J5)

- Move to negative closed position P2
- If error checking is enabled (Ec=1)
 - Calculate drift at previous position P1: $Dr=N1-Cs$
 - If drift is at or beyond limit ($Dr \geq DI$)
 - Increment the lower drift violation counter Ld
 - EndIf
- EndIf
- Set Flip-Flop ISR 2 (J6) to run on next Input Trip
- Hold until movement is complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=-1 to signal at negative closed position
- Re-arm Input Trip (TE=4)
- Return

Flip-Flop ISR 2 (P2 to P1 move) (LB J6)

- Move to positive closed position P1
- If error checking is enabled (Ec=1)
 - Calculate drift at previous position P2: $Dr=Cs-N2$
 - If drift is at or beyond limit ($Dr \geq DI$)
 - Increment the upper drift violation counter Ud
 - EndIf
- EndIf
- Set Flip-Flop ISR 1 (J5) to run on next Input Trip
- Hold until movement is complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=1 to signal at positive closed position
- Re-arm Input Trip (TE=4)
- Return

Mode 3: Burst (LB Z3)

- Signal state change to 30 (Burst Mode Init)
- Load Burst speeds (V3)
- Move the shutter to the nearest closed position (CL X5)
- Specify Burst Open ISR (J3) to run on Input Trip
- Arm Input Trip (TE=4)
- Signal state change to 31 (Burst Loop)
- Burst Loop START (forever):
 - Increment heartbeat
 - Hold for 100ms
 - If user sets the soft reset (Se=0)
 - Branch to Mode Select (Mode 0)
 - EndIf
 - If user requests fast close (Se=5)
 - Branch to Fast Close (Mode 5)
 - EndIf
- Burst Loop END

NOTE: Once in the burst loop, you must set Se=0 to get back to Mode Select. *Burst mode is continuous, that is, each subsequent trigger pulse will open and close the shutter.*

Burst Open ISR (LB J3)

- Move (MA) to open position (P0)
- If error checking is enabled (Ec=1) **AND** last position was P1 (Df==1)
 - Calculate drift at previous position P1: $Dr=N1-Cs$
 - If calculated drift is out of spec ($Dr \geq DI$)
 - Increment lower drift violation counter Ld
 - EndIf
- EndIf
- Specify Burst Close ISR (J4) to run on Input Trip

- Hold until movement complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=0 to indicate slits are at open position
- Re-arm input trip (TE=4)
- Return

Burst Close ISR (LB J4)

- Move (MA) to positive closed position (P1)
- If error checking is enabled (Ec=1)
 - Calculate drift at previous position P0: $Dr = Cs - N0$
 - If calculated drift is out of spec ($Dr \geq DI$)
 - Increment upper drift violation counter Ud
 - EndIf
- EndIf
- Specify Burst Open ISR (J3) to run on Input Trip
- Hold until movement complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=1 to indicate slits are at positive closed position
- Re-arm input trip (TE=4)
- Return

Mode 4: Fast Open (LB Z4)

- Signal state change to 40 (Fast Open Init)
- Load fast open speeds (CL V4)
- Move (MA) to the open position (P0)
- Hold until movement complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=0 to indicate that slits are in open position
- Signal state change to 41 (Fast open complete)
- Goto mode select

Mode 5: Fast Close (LB Z5)

- Signal state change to 50 (Fast close init)
- Load fast close speeds (CL V5)
- If neither at Df=1 (positive closed) nor at Df=-1 (negative closed), then
 - Move (MA) to the positive closed position (P1)
 - Hold until movement complete
 - Copy current encoder count to Cs for future drift checks
 - Set Df=1 to indicate that slits are in positive closed position
- EndIf
- Signal state change to 51 (Fast close complete)
- Goto mode select

Mode 6: Follower (LB Z6)

- Signal state change to 60 (Follower Mode Init)
- Load Follower speeds (V6)
- Move the shutter to the nearest closed position (CL X5)
- Set hi-speed trip input to trigger on HIGH-going edge
- Specify Follower Open ISR (J7) to run on Input Trip
- Arm Input Trip (TE=4)
- Signal state change to 61 (Follower Loop)
- Burst Loop START (forever):
 - Increment heartbeat
 - Hold for 100ms
 - If user sets the soft reset (Se=0)
 - Branch to Mode Select (Mode 0)
 - EndIf
 - If user requests fast close (Se=5)
 - Branch to Fast Close (Mode 5)
 - EndIf
- Burst Loop END

NOTE: Once you enter follower mode, you should cycle the EVR line (high-low-high or low-high-low) at least once to synchronize the shutter and the input. This limitation will be fixed in a later picker hardware revision.

NOTE: Once in the follower loop, you must set Se=0 to get back to Mode Select. *Follower mode is continuous, that is, each subsequent trigger pulse will open and close the shutter.*

Follower Open ISR (LB J7)

- Move (MA) to open position (P0)
- If error checking is enabled (Ec=1) **AND** last position was P1 (Df==1)
 - Calculate drift at previous position P1: Dr=N1-Cs
 - If calculated drift is out of spec (Dr>=DI)
 - Increment lower drift violation counter Ld
 - EndIf
- EndIf
- Specify Follower Close ISR (J8) to run on Input Trip
- Hold until movement complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=0 to indicate slits are at open position
- Set high-speed trip input to trigger on LOW-going edge
- Re-arm input trip (TE=4)
- Return

Follower Close ISR (LB J8)

- Move (MA) to positive closed position (P1)
- If error checking is enabled (Ec=1)
 - Calculate drift at previous position P0: Dr=Cs-N0
 - If calculated drift is out of spec (Dr>=DI)
 - Increment upper drift violation counter Ud
 - EndIf
- EndIf
- Specify Follower Open ISR (J7) to run on Input Trip
- Hold until movement complete
- Copy current encoder count (C2) to shadow register Cs
- Set Df=1 to indicate slits are at positive closed position
- Set high-speed trip input to trigger on HIGH-going edge
- Re-arm input trip (TE=4)
- Return

Mode 7: Home to Index Mark (LB Z7)

- Signal state change to 70 (Homing init)
- Load homing/alignment speeds (CL V9)
- Execute home-to-index (HI) with option 3 (slew at VM in + direction, creep in - direction)
- Hold until movement complete
- Relative move (MR) by Mo to positive closed position
- Hold until movement complete
- Set C1 to P1 - tell the motor where it is
- Set C2 to N1 - tell the encoder where it is
- Set Cs to C2 - copy the encoder count to shadow register
- Set Df to 1 (positive closed position, known)
- Reset limit violation counters Ld and Ud to 0
- Signal state change to 79 (Homing complete)
- Goto mode select

Subroutines

Blocking move to nearest closed position (LB X5)

This routine closes the pulse picker to the positive closed position P1 if it is not already closed.

- If not already in the positive or negative closed position (Df<>1 AND Df<>-1)
 - Move to positive closed position (P1)
 - Hold until movement is complete
- - Copy current encoder count to Cs for future drift checks
 - Signal at positive closed position (Df=1)
- EndIf

Realign by computing new open-loop position (LB X7)

This routine updates the open-loop position based on the encoder reading. The target position of any subsequent move will be better aligned.

- Declare local variables L1, L2
- Compute the new open-loop position: $C2 * \text{<encoder counts per rev>} / \text{<microsteps per rev>} = (C2 * EL * 8) / (MS * 200)$
 - $L1 = C2 * EL * 8$
 - $L2 = MS * 200$
- Write the new open-loop position to C1. ($C1 = L1 / L2$)
- Set Df=98 (Realigned)
- Return

Mode/State Tables

Se - Mode Select

Se	Mode Description
0	Soft Reset - interpreted at start-up to bypass hard reset initialization routine
1	One-Shot mode: one pulse on the input line results in a single sweep of the pulse picker. Does not automatically return to mode select.
2	Flip-Flop mode: will sweep the pulse picker each time a pulse is received on the input line. Does not automatically return to mode select.
3	Burst mode: will open the pulse picker on the first input pulse received, and close on the second. Does not automatically return to mode select.
4	Fast Open: moves the pulse picker to the full open position (P0). Automatically returns to mode select.
5	Fast Close: moves the pulse picker to the positive closed position (P1). Automatically returns to mode select.
7	Homing: moves the pulse picker to the home index mark, and then by magic offset to the positive closed position P1. Automatically returns to mode select.

Sd - Status Indicator

Sd	Status Description
0	Waiting for Se mode selection
10	One-Shot Initialization
11	One-Shot Loop Waiting
42	One-Shot Complete
20	Flip-Flop Initialization
21	Flip-Flop Loop Waiting
22	Flip-Flop Loop Active
30	Burst Initialization
31	Burst Armed and Waiting
32	Burst Open Pulse Received
33	Burst Close Pulse Received
39	Burst Complete
40	Fast Open Initialization
41	Fast Open Complete
50	Fast Close Initialization
51	Fast Close Complete
60	Follower Mode Initialization
61	Follower Mode Loop
70	Homing Initialization
79	Homing Complete
80	Hard Reset/Power-Up Initialization
81	Soft Reset/Power-Up Initialization
90	Toggle Move Complete
91	Move to Positive Closed Position Complete
92	Move to Negative Closed Position Complete

94	Move to Open Position Complete
95	Close move did not execute because pulse picker is already closed

Df - Slit Open/Closed Status

Df	Status Description
-1	Closed in the negative position
0	Open
1	Closed in the positive position
98	Realigned ; slits are in known position but not necessarily open or closed
99	Slit status is unknown

Velocity Tables

Maximum Velocity/Acceleration (V0)

Var	Value [step/s(^2)]	[deg/s(^2)]	[rev/s(^2)]
VI	5300	1192.5	3.3125
VM	10000	2250	6.25
A	1875000	421880	1171.9
D	1875000	421880	1171.9

The VI is slightly lower for the V0 set of speeds for more reliable 2-shot burst operation. These speeds are used wherever there is motion from closed to open position, or vice versa.

One-Shot Velocity Defaults (V1)

Var	Value [step/s(^2)]	[deg/s(^2)]	[rev/s(^2)]
VI	7500	1687.5	4.6875
VM	10000	2250	6.25
A	1875000	421880	1171.9
D	1875000	421880	1171.9

These speeds correspond to the fastest experimentally-determined speeds for continuous 60Hz operation. These speeds are used wherever there is motion from one closed position to the other closed position.

Flip-Flop Velocity Defaults (V2)

Var	Value [step/s(^2)]	[deg/s(^2)]	[rev/s(^2)]
VI	7500	1687.5	4.6875
VM	10000	2250	6.25
A	1875000	421880	1171.9
D	1875000	421880	1171.9

These speeds correspond to the fastest experimentally-determined speeds for continuous 60Hz operation. These speeds are used wherever there is motion from one closed position to the other closed position.

Burst Velocity Defaults (V3)

V3 speeds are identical to V0 speeds.

Fast Open Velocity Defaults (V4)

V4 speeds are identical to V0 speeds.

Fast Close Velocity Defaults (V5)

V5 speeds are identical to V4 speeds.

Follower Velocity Defaults (V6)

V6 speeds are identical to V0 speeds.

Homing/Alignment Velocity Defaults (V9)

Var	Value [step/s(^2)]	[deg/s(^2)]	[rev/s(^2)]
VI	640	144	0.4
VM	2560	576	1.6
A	640	144	0.4
D	640	144	0.4

These speeds are arbitrarily selected to be slow.

Magic Offset (Mo) Table

Installation	Date	Mo	Spindle PV	Verified By
XPP	2013-10-14	-245	XPP:SB2:MMS:29	klg
MEC	2013-12-20	-240	MEC:HXM:MMS:18	alexnd
XCS	2014-01-13	-246	XCS:SB2:MMS:09	klg & alexnd
CXI	2013-12-20	-286	CXI:ATC:MMS:29	alexnd

The magic offsets are normally stored in IMS NVRAM, but in case they are lost, refer to the table above.

Changelog

Version Ns=96 (pcds_motion R2.5.0)

- Added Follower mode (mode 6), in which the state of the picker shutter follows the state of the EVR input line, with 0V == closed and 5V == open.
- Added the ability to trigger Fast Close directly from One-Shot, Flip-Flop, Burst, and Follower modes

Version Ns=94 (pcds_motion R2.3.14)

- Changed the hard reset routine so that it saves to NVRAM at the end. This handles the edge case of a virgin motor being loaded with MCode for the first time, then being power-cycled without ever having its settings saved to NVRAM again.
- Added Se=99 to the hard reset routine, to be saved to NVRAM so that subsequent power-cycles **should** invoke the hard reset routine.
- Wishlist for next revision:
 - Test realignment subroutine (X7) and add it to every mode change (especially fast open/close modes)
 - Further velocity tuning? We could possibly go faster.

Version Ns=93 (pcds_motion R2.3.13)

- Moved O3 and O4 initializers to run on soft resets as well as on hard resets. This avoids a power-up edge case where, if the motor is powered up with Se=0, the encoder would not power up, making the spindle run haywire upon homing.

Version Ns=92 (pcds_motion R2.3.12)

- Release version of Ns=91
- Adjusted default drift limit D1 to 46 encoder counts (~1 degree)

Version Ns=91 (Non-Release)

- Reduced call depth by moving code from subroutines into their respective ISRs/functions. This was done for speed, as we gain a significant fraction of a millisecond for each CL/RT pair eliminated.
- Eliminated a few Sd states that are no longer being used (for speed purposes).
- Burst mode now drift checks the open position, to allow it to be used ad infinitum without risk of it drifting in the positive direction.
- Drift limit violation counters are now reset upon homing as well as upon power cycle.
- Eliminated no-longer-used subroutines.
- Removed all unneeded spaces from conditional branch/call statements.