Automated Controls for the Hard X-Ray Split & Delay System at LCLS

SLAC NATIONAL ACCELERATOR LABORATORY

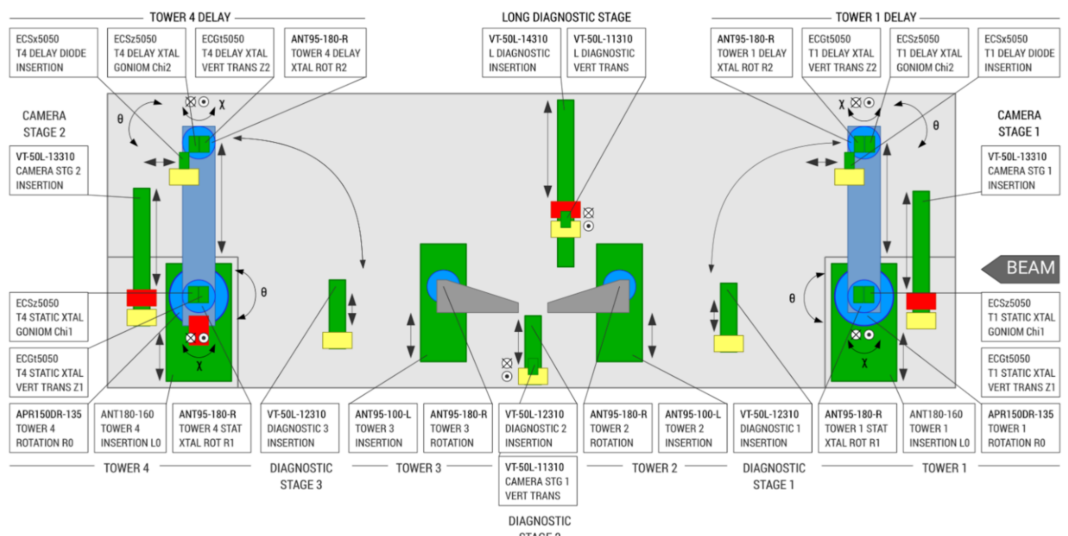
A.P. Rashed Ahmed, M.C. Browne, D.L. Flath, K.L. Gumerlock, T.K. Johnson, L. Lee, Z. Lentz, T.H. Rendahl, H. Shi, H.H. Slepicka, Y. Sun, T.A. Wallace, D. Zhu

Abstract

The hard x-ray split and delay (HXRSnD) system at the Linear Coherent Light Source (LCLS) was designed to allow for experiments requiring two-pulse based x-ray photon correlation spectroscopy. The system consists of eight silicon crystals split between two optical branches, with over 30 degrees of freedom. To maintain system stability and safety while easing system operation, we expand the LCLS Skywalker software suite to provide a python-based automation scheme that handles alignment, operations and engineer notification. Core safety systems such as collision avoidance are handled at the controller and Experimental Physics and Industrial Control System (EPICS) layer. Higher level functionality is implemented using a stack of open-source python packages (ophyd, bluesky, transitions) which provide a comprehensive and robust operational environment consisting of virtual motors, plans and finite state machines (FSM).

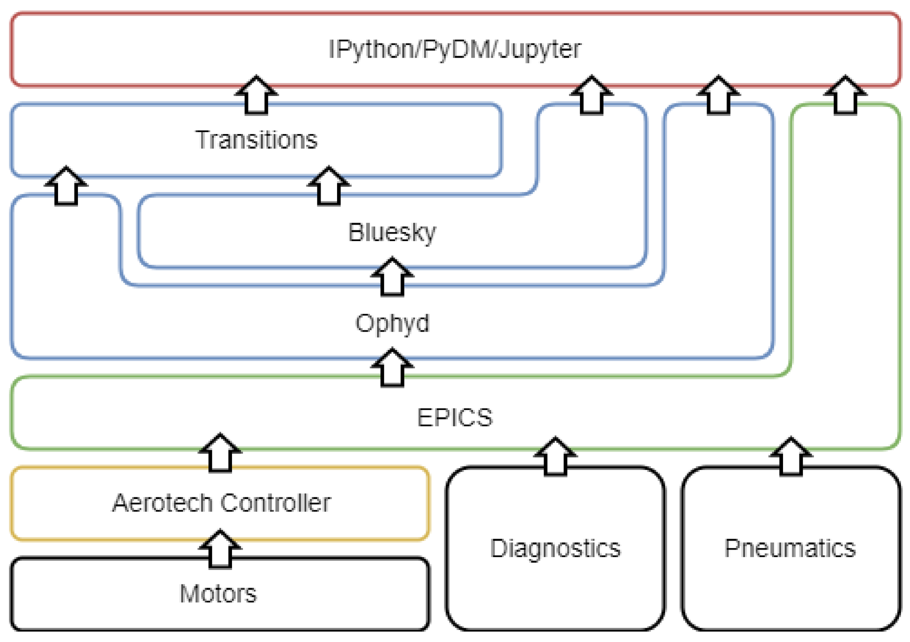
System Design and Hardware

- Consists of four towers, an assortment of diagnostics and pneumatics.
- Towers 1 and 4 and diagnostic dd comprise the delay branch.
- Towers 2 and 3 and diagnostics dci, dcc and dco comprise the channel cut branch.

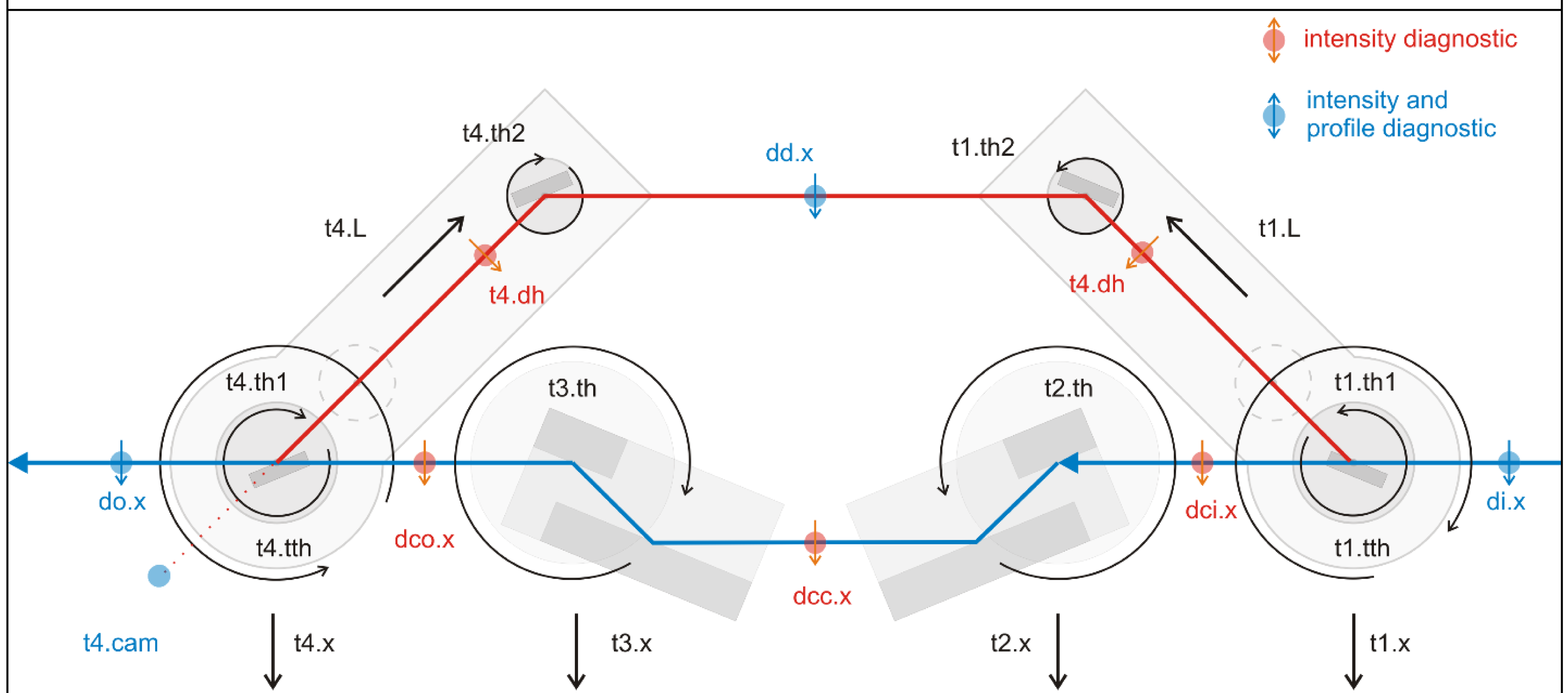


Software Stack

- ***Aerotech Controller*** - Motor controller that handles low-level safety to prevent collisions.
- ***EPICS*** - Hardware communications layer that implements some higher level safety interlocks.
- ***Ophyd*** - Python interface for hardware. Implements low and high-level devices.
- ***Bluesky*** - Python package that implements system state transitions and procedures.
- ***Transitions*** - Python package for creating finite state machines (FSMs).



Automatic Alignment



Rocking-Curve Maximization

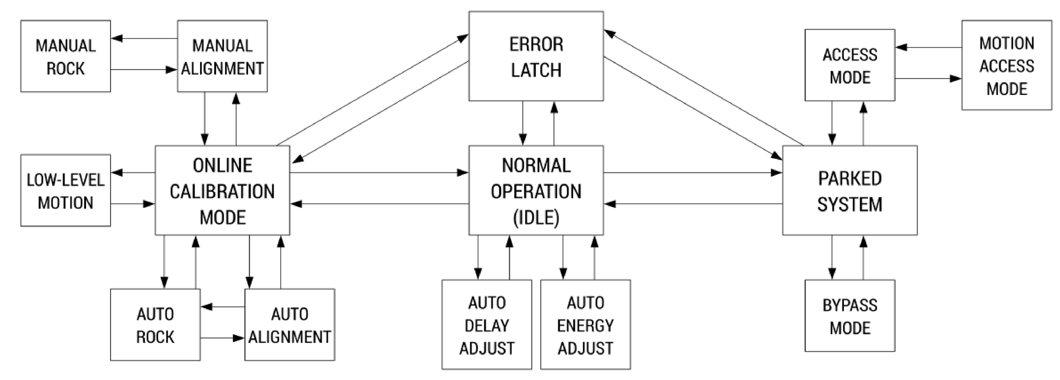
- Begin a linear scan with desired crystal.
- At each step record upstream and downstream beam intensity.
- Fit Lorentzian curve to the ratio of downstream to upstream intensity.
- Move to maximum position, and repeat.

Alignment Procedure

- Insert the delay and channel cut branches.
- Adjust t1.th1 to cut the beam in half.
- Move the delay branch to the desired energy and delay using the virtual motors.
- Perform a rocking curve maximization through the delay line.
- Move the channel cut branch to the desired energy using the virtual motor.
- Perform a rocking curve maximization through the channel cut line.

Finite State Machines

- ***Normal Operation*** – Delay towers "flying", primarily using virtual motors.
- ***Parked System*** – Delay towers in "Parked", all motors disabled.
- ***Online Calibration*** – Delay towers "flying", alignment and homing procedures available.



Virtual Motors

Delay - Moves delay motors t1.L and t4.L to L, where:

$$L = \frac{1}{(1 - cos(2\theta_1))}\left(\frac{ct}{2} + \frac{d_G(1 - cos(2\theta_2))}{sin(\theta_2)}\right)$$

E1 - Moves motors t1.th1, t1.th2, t4.th1, t4.th2 to θ_1, t1.tth and t4.tth to $2\theta_1$, and dd.x to dd_L where:

$$dd_L = Lsin(2\theta_1)$$

E2 – Moves motors t2.th and t3.th to θ_2, and dcc to dcc_L where:

$$dcc_L = 2d_G cos(\theta_2)$$

Acknowledgments

Use of the Linac Coherent Light Source (LCLS), SLAC National Accelerator Laboratory, is supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences under Contract No. DE-AC02-76SF00515.

References

- Zhu, D. et al., "Development of a hard x-ray split-delay system at the Linac Coherent Light Source", Proc. SPIE 10237, Advances in X-ray Free-Electron Lasers Instrumentation IV, 102370R (June 2017)
- M.L. Gibbs et al., "PyDM: A Python-Based Framework for Control System Graphical User Interfaces", ICALEPCS 2017, Barcelona (October 2017)
- T.H. Rendahl et al., "Skywalker: Python Suite for Automated Photon Alignment at LCLS", ICALEPCS 2017, Barcelona (October 2017)
- N. W. Brown, et al., "Developing EASE, A remote EPICS monitoring tool", SLAC National Laboratories (2016).
- National Synchrotron Light Source II, Ophyd v0.7.0 [Python Package]. Retrieved from http://nsls-ii.github.io/ophyd/ (December 2014)
- National Synchrotron Light Source II , Bluesky Data Collection Framework v0.10.0 [Python Package]. Retrieved from http://nsls-ii.github.io/bluesky (June 2015)
- T. Yarkoni, A. Neumann, et al., Transitions v0.6.1 [Python Package]. Retrieved from https://github.com/pytransitions/transitions (October 2014)