

MRO Delay Line

Requirement specifications for the MROI “production” delay line software

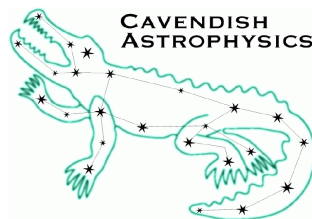
INT-406-CON-0101

John Young
jsy1001@cam.ac.uk

Living Document

rev 1.3

13th January 2010



Cavendish Laboratory
JJ Thomson Avenue
Cambridge CB3 0HE
UK

Change Record

Revision	Date	Authors	Changes
0.4	2009-05-11	JSY	Incorporated comments from DFB.
0.5	2009-05-14	JSY	Incorporated comments from MF.
0.6	2009-05-19	JSY	Minor changes.
1.0	2009-06-03	JSY	Changes agreed at requirements review.
1.1	2009-12-03	JSY	Changes to ISS interface requirements, change to max. frequency of offset demand signal. This revision not released.
1.2	2010-01-07	JSY	Further changes to ISS interface requirements. Supervised and/or standalone mode now specified for each use case. Added "baseline solution observation" use case (variation on "astronomical observation"). Added "FT spectral calibration" use case.
1.3	2010-01-13	JSY	Added requirement to transmit DL positions to FT over low-latency link. Communication latency now specified.

Objective

To state the technical requirements for the "production" delay line software that will be delivered by Cavendish Laboratory.

Reference Documents

RD1 Delay Line Production Software Development Statement of Work

RD2 [System Overview](#) INT-406-VEN-0100

RD3 [Requirements Specification for the MROI Delay Line System](#) INT-406-TSP-0002

RD4 [List of Tests](#) INT-406-VEN-0108

RD5 [Trolley Electronics Design Description](#) INT-406-VEN-0112

RD6 [Shear Camera Software Functional Description](#) INT-406-VEN-0105

RD7 [Metrology System & VME Hardware Design Description](#) INT-406-VEN-0113

RD8 ICD: Delay line to ICS INT-406-ENG-0002

RD9 Design of an MROI System INT-409-ENG-0020

Applicable Documents

AD1 [Commissioning Plan and Performance Verification Milestones for the MROI](#) INT-414-TSP-0004

AD2 Prototype [Control Software Architecture](#) INT-406-VEN-0101

AD3 [Coding and Documentation Guidelines for Prototype Delay Line Software](#)

AD4 [ICS/Delay Line Information Flow](#)

AD5 [Software Requirements Specification for Automated Alignment System](#) INT-410-TSP-0100

AD6 Fringe Tracker Control Software: Requirements and Conceptual Architecture INT-407-TSP-0020

Acronyms and Abbreviations

AAS	Automated Alignment System
API	Application Programming Interface
DL	Delay lines
dlmsg	Cavendish delay line messaging protocol
FT	Fringe Tracker
GUI	Graphical User Interface
ISS	Interferometer Supervisory System
MCBD	ISS Monitor and Configuration DataBase
MOB	Magic Optical Box
MROI	Magdalena Ridge Observatory Interferometer
OPD	Optical Path Difference
PVM	Performance Verification Milestone
RTC	Jet Propulsion Laboratory's Real Time Control framework
SI	Science Instrument
SOW	Statement of Work
TBC	To Be Confirmed
TBD	To Be Determined
UT	Unit Telescope

Table of Contents

1	Introduction and Context.....	6
1.1	Document Conventions.....	6
2	System Overview.....	7
2.1	Software Components.....	10
3	Use Cases.....	12
3.1	Reset metrology use case.....	12
3.2	Align delay line pipe use case.....	12
3.3	Align metrology laser use case.....	12
3.4	Align primary fiducial to delay line use case.....	12
3.5	Adjust cat's-eye focus use case.....	13
3.6	Baseline solution observation use case.....	13
3.7	Astronomical observation use case.....	13
3.8	Baseline solution use case.....	14
3.9	Internal fringes use case.....	14
3.10	FT Spectral calibration use case.....	15
3.10.1	Alternative A: Spectral calibration by single-pass internal fringes	15
3.10.2	Alternative B: Spectral calibration by double-pass internal fringes	15
3.11	Performance Verification Milestone use cases.....	15
3.12	Trolley acceptance test use cases.....	16
4	Functional Requirements.....	17
4.1	Delay Line System Objects.....	17
4.2	Data transmission to ISS.....	17
4.3	Fault/Alert transmission to ISS.....	17
4.4	Axial Control.....	18
4.4.1	FOLLOW system mode.....	18
4.4.2	STOP/IDLE system mode	18
4.4.3	DATUM system mode	18
4.4.4	DIRECT SLEW system mode.....	19
4.4.5	Fringe tracking.....	19
4.4.6	Fringe searching.....	19
4.5	Shear control.....	19
4.6	Roll control.....	19
4.7	PMAC configuration.....	20
4.8	Sub-system emulation.....	20
4.9	Workstation data recording.....	20
5	Performance Requirements.....	20
6	Computations and Algorithms.....	20
6.1	Trajectory Computation.....	20
6.2	Slew Algorithm.....	21
6.3	Fringe tracking step pre-filter.....	21
6.4	Common Offset Computation.....	21
6.5	Baseline Solution Computation.....	21

6.6 Shear Sensor Algorithm.....	21
7 Hardware Interfaces.....	21
8 User Interfaces.....	22
8.1 Trolley software.....	22
8.1.1 Engineering user interface.....	22
8.2 Shear sensor software.....	22
8.2.1 Live image display.....	22
8.3 Workstation software.....	22
8.3.1 Engineering control GUI.....	22
8.3.2 Standalone analysis GUI.....	23
8.3.3 Baseline solution UI.....	23
9 Communications Interfaces.....	23
9.1 Configuration Interface.....	23
9.1.1 Configuration for standalone use.....	23
9.2 Command/Data Interface.....	24
9.3 Low-latency FT Interface.....	24
10 Non-functional requirements.....	24
10.1 Operating Environments.....	24
10.2 Coding Standards.....	24

1 Introduction and Context

Following on from the Final Design Review for the MROI delay lines, the need for “production” software to both commission and operate the delay-lines for their astronomical role has been identified. The team at the Cavendish Laboratory who have designed the delay-line system are now contracted to deliver this production software, the basis for the contract being the agreed Statement Of Work [RD1].

The delay line software provides real-time control, monitoring and data acquisition for the MROI delay line system [RD2,RD3]. The software also provides a data analysis capability for use in commissioning the delay line system.

The use cases for the production software can be summarized as:

- operation of the delay lines for the planned system commissioning tasks, i.e. the Performance Verification Milestones (PVMs) 1 to 13 [AD1];
- operation of the delay-lines for regular astronomical observations;
- acceptance tests of the second and subsequent production delay line trolleys.

These use cases are elaborated in Sec. 3 of this document. The software will be used with any number of installed delay lines, up to a maximum of 10.

The code to be delivered will be based on the prototype control and analysis software [AD2] that has been used to demonstrate the performance of the prototype delay-line trolley, but is to be re-architected and written so as to integrate with the MROI supervisory control system (ISS) and to provide the functionalities needed for PVM testing and science operations.

The requirements that were discussed and agreed upon at the requirements review in May 2009 were listed in revision 1.0 of this document. This version incorporates changes to reflect the subsequent agreement with NMT that a custom framework will be used to interface the delay line software with the ISS, rather than RTC.

We expect that the requirements document will be treated as a “living document” to which further details and clarifications are added during the early phases of the project. As part of this effort we will write a verification plan defining the factory and site acceptance tests that will be performed. Each distinct requirement mentioned in this document will be assigned a unique identifier at this point. The draft verification plan will be presented at the Preliminary Design Review.

This document supersedes INT-406-TSP-0003 revision 3.

1.1 Document Conventions

Where we have identified a choice of possible requirements this is highlighted using a coloured box, for example:

CHOICE: This could be specified as A or B.

Where requirements should be refined once the project is underway this is indicated using a grey box:

Open Issue: We need to decide which component handles C.

In this document we use the word **system** to refer to a *single* delay line as an entity, notwithstanding the fact that some hardware and software components are associated with multiple delay lines (see Fig. 3).

We use the words **user** and **operator** interchangeably to refer to a human user.

2 System Overview

The delay lines are required to introduce vacuum optical paths between zero and 380m using a single stroke, with an absolute precision of 10 μ m and jitter below 15nm over 10ms intervals.

The design architecture adopted for the MROI delay lines incorporates a single-stroke movable carriage carrying retro-reflecting optics and running in an evacuated aluminium pipe. A 3-d cutaway diagram showing the basic elements of the design is presented in Fig. 1. Further details of the additional hardware components for the full delay line system are shown schematically in Fig. 2.

The optical assembly utilises a standard configuration of a parabolic primary mirror with a flat secondary located at the M1 focus, and is itself supported on flexures (in an inverted pendulum geometry) mounted on a motorised carriage. The location of the cat's eye is monitored by a commercial Agilent laser metrology system where the only modification has been to expand the laser beam to approximately 20mm diameter to accommodate the long propagation path. However, the design also incorporates a number of unusual features which are significant departures from the norm. The most important of these are:

1. The motorised carriage runs directly on the interior surface of the aluminium pipe that provides the vacuum vessel for the system. To accommodate the imperfections that characterise the interior finish of the extruded pipe, the carriage uses compliant polyurethane tyres rather than hard metal wheels.
2. Because the aluminium pipes used to define the trajectory of the carriage are themselves not guaranteed to be straight, the secondary mirror is adjustable in tip and tilt so as to correct for the shear introduced by any lack of pipe straightness. The shear is detected by picking off a fraction of the return metrology beam and sensing its location on a CCD camera. Even at slewing speeds, only a low control bandwidth is necessary because the length scales on which the vacuum pipe is "bent" are relatively large (i.e. many tens of cm).
3. The whole of the optical path in the MROI delay lines is kept under vacuum. By operating at a pressure of \sim 1 millibar, problems associated with internal seeing and longitudinal dispersion are minimized while still allowing sufficient heat transfer via conduction to take place between the delay line electronics chassis and the vacuum pipe walls.
4. Because rails are not used to define its trajectory, the delay line carriage is unconstrained in rotation about the vacuum pipe axis. This "roll" degree of freedom is controlled both by design — the center of gravity of the delay line carriage is below the vacuum pipe centerline — and also through an on-board tilt sensor which measures the roll angle in real time so that the undriven rear wheel of the carriage can be steered to correct any roll errors. As is the case for the shear variations, only low-bandwidth control (a few Hz) is necessary.
5. The distribution of power and control signals to the carriage is performed in a "contactless" manner. Power is transmitted inductively, using a wire lying in the bottom of the vacuum pipe which slides through a long transformer core attached to the bottom of the carriage. On-board power storage to manage periods of high load is also made available through the use of a bank of super-capacitors. Control signals to the carriage use two separate radio links. The first is a commercial 2.4 GHz wireless Ethernet link used for high data-rate

communication between the on-board micro and external control computers, while the second is a dedicated low-latency 900 MHz link used to close the servo-loop controlling the cat's-eye position, e.g. when following astronomical targets.

The novel features of the hardware design imply that the control system must differ from those of other optical delay lines in several respects:

- There is a need to manage additional servo loops, for control of beam shear and trolley roll.
- The metrology (OPD) loop must be closed wirelessly with less than 40 μ s latency.
- To mitigate the risks associated with commissioning a new design, it is prudent to provide the capability to capture large volumes of engineering telemetry data.

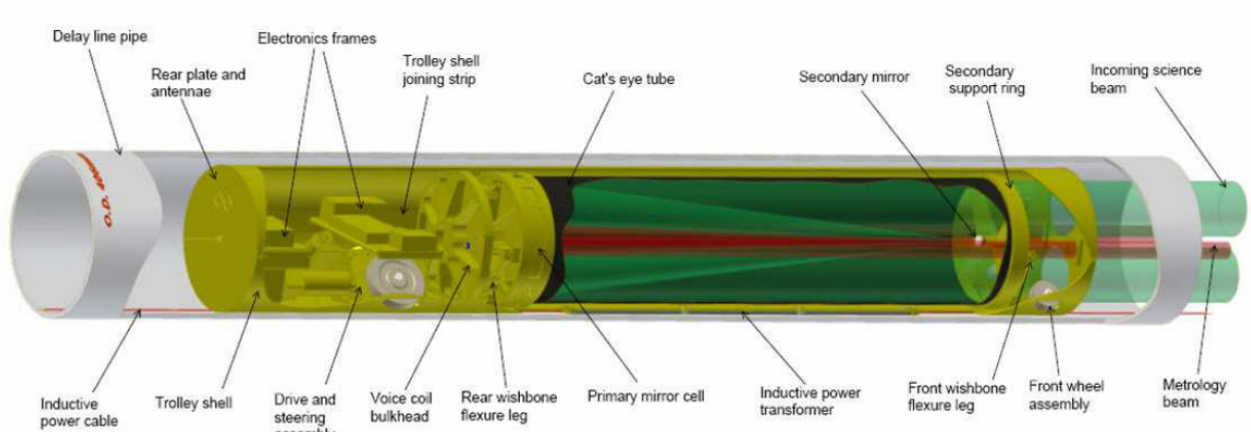


Figure 1: A 3-d cutaway view showing the principal design features of the MROI delay line system with the optical assembly mounted within its concentric mechanical carriage and the whole trolley running within an evacuated pipe. In this schematic the incoming science beam enters at right, is retro-reflected by the cat's eye, and returns to the right underneath the incoming beam. The input and output metrology beams enter and exit at right too, but remain in the same horizontal plane. The carriage motor, control electronics and communications and power systems are all located to the left of the delay line trolley, behind the primary mirror and voice-coil bulkhead. The supports carrying the vacuum pipe are not shown here.

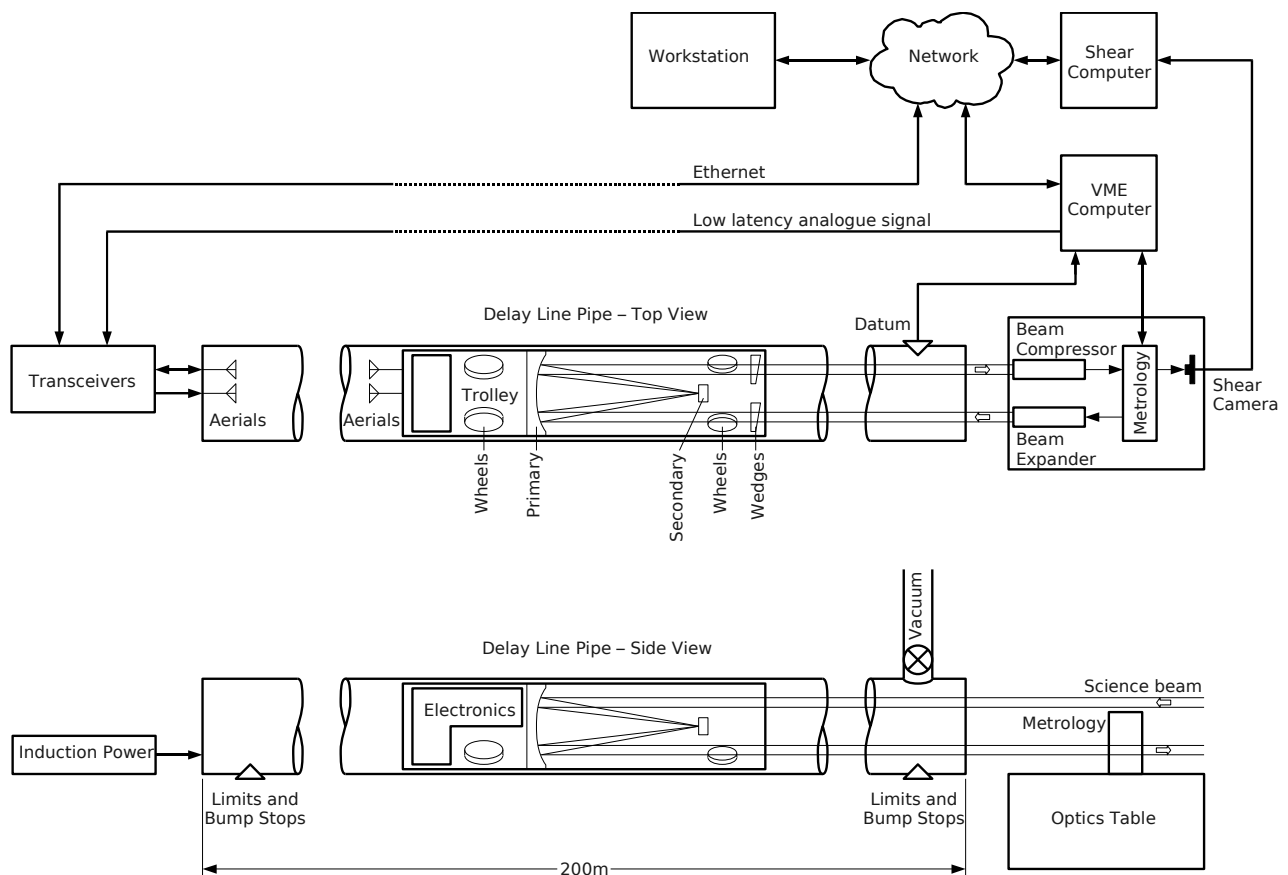


Figure 2: Schematic top and side views of a complete delay line system, showing the physical locations of the major components.

2.1 Software Components

The components of the production software are specified in the SOW [RD1]. For convenience these are enumerated below. This list should be read in conjunction with Fig. 3 which illustrates how these components relate to the overall control architecture for the delay-lines. The terms highlighted in bold will be used to refer to the software components in later sections of this document.

The list below now assumes that the interface between the delay line system and the MROI supervisory control system (ISS) uses a C code-generation framework that implements the custom TCP/IP protocols described in RD9.

In addition to the supervised mode used for regular operations, the delay line system will also be able to operate in a standalone mode, independent of the ISS. In this mode the user will be able to command the delay lines using the supplied engineering control GUI.

1. **Trolley software**: resides on-board each delay-line trolley and interfaces to on-board hardware. Responsible for closing the roll loop, and applying correction signals from the shear sensor in order to close the shear loop. The trolley software also implements commanded trolley actions such as changes in drive velocity and secondary focus.
2. **Shear sensor software**: performs capture and centroiding of CCD images to provide closed-loop control of the delay line output beam shear.
3. VME **metrology software**: closes the fast position loops of the delay lines by sampling the Agilent metrology hardware and transmitting correction signals over the low-latency wireless links to the trolleys' cat's-eye drive electronics. The metrology system will incorporate a low-latency wired interface to the Fringe Tracker system for fringe tracking and fringe searching.
4. Workstation software: used to co-ordinate the overall delay-line system and manage communications with the MRO ISS. This will include:
 - a **system controller** for each delay line, responsible for generating trajectory demands for the metrology software and managing the state of the delay line sub-systems. This would provide interfaces to the ISS, implemented using the C code-generation framework, for receiving delay line system commands and transmitting system and sub-system status and telemetry (including shear camera video) as monitor data;
 - an **engineering control GUI** with telemetry recording capability and live displays of system and sub-system status;
 - a standalone **analysis GUI** (written in Matlab) that reads recorded telemetry
5. **Baseline solution software**: used to refine an initial baseline model, using fringe acquisition data for a set of stars widely distributed across the sky. We anticipate that this will be supplied as a software library that can be integrated into the ISS by NMT, together with a simple user interface that can be used prior to this integration taking place.

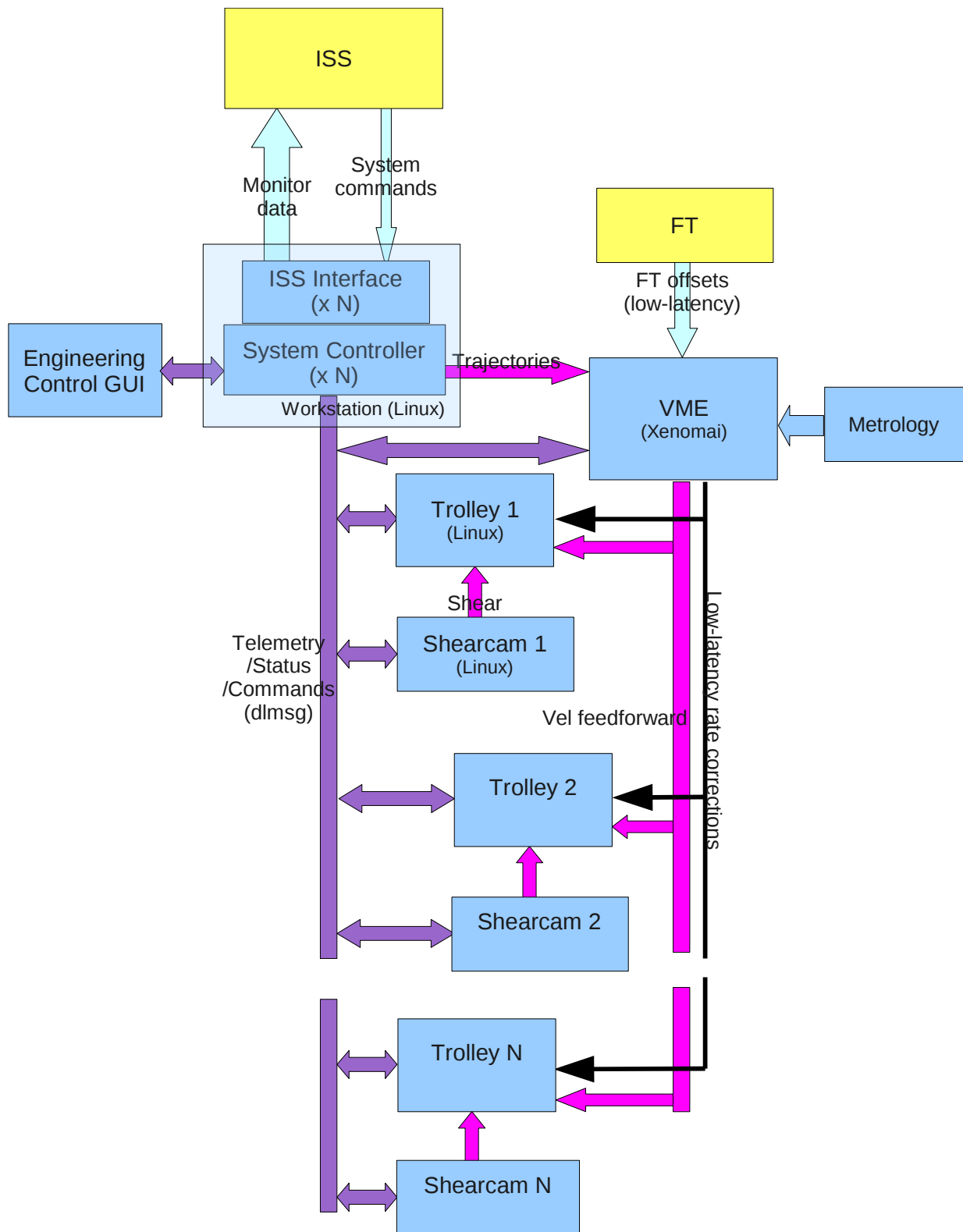


Figure 3: Architecture of delay line control system as integrated into MROI. Different coloured arrows refer to different types of communication. The components shown in yellow correspond to NMT-supplied deliverables (which interface to the Cambridge software).

3 Use Cases

The unique use cases that have been identified for the delay line systems are described in this section. Several of the use cases mentioned below involve other interferometer systems besides the delay lines and ISS, for example the Automated Alignment System (AAS) and Fringe Tracker. For such use cases, the focus of this document is on the role played by the delay lines. We expect that the interactions with other systems will be fleshed out as the control software for those systems becomes better defined.

3.1 Reset metrology use case

Context of Use: Immediately after startup of delay line system, and whenever metrology signal is lost. The delay lines operate in supervised or standalone mode.

Description: ISS or operator commands system to seek datum switch and reset metrology counts at the datum position.

3.2 Align delay line pipe use case

Context of Use: When delay line is installed or lengthened. Subsequently at ~yearly intervals to accommodate subsidence. May also be used to check pipe alignment, either if persistent errors occur (for instance loss of metrology signal at a particular DL position) or to measure the rate of change of pipe alignment. The delay lines operate in standalone mode.

Description: Operator deactivates shear loop. Operator commands delay line to acquire datum in order to reset coarse position sensor. Operator drives trolley at constant velocity (using DIRECT SLEW mode) over full length of pipe, while recording telemetry. Operator reads telemetry file into analysis GUI and plots shear error against trolley position. Operator uses plot to decide whether and how to adjust delay line pipe supports. Iterate until shear error is within specifications.

3.3 Align metrology laser use case

Context of Use: Nightly to weekly. The delay lines operate in standalone mode.

Description: Operator deactivates shear loop. Operator commands delay line to acquire datum in order to reset coarse position sensor. Operator drives trolley at constant velocity (using DIRECT SLEW mode) over full length of pipe, while recording telemetry. Operator reads telemetry file into analysis GUI and plots shear error against trolley position. Operator fits straight line using analysis GUI, and uses parameters of best-fit line to adjust tip/tilt of metrology laser beam (if necessary) and shear sensor fiducial point. Iterate as necessary. Adjustment of the metrology laser tip/tilt could be done using the Newport Agilis hand controller or MRO-provided software.

Open Issue: Do we need to perform this use case in supervised mode?

3.4 Align primary fiducial to delay line use case

Context of Use: Nightly to weekly. The delay lines operate in supervised mode.

Prerequisites: Metrology laser must be aligned as per previous use case.

Description: This use case is UC-010 in AD5. Activate AAS MOB and Nasmyth table

shear camera. With delay line shear correction activated, use AAS Nasmyth table shear camera to measure change in shear with delay line position (probably by moving between two pre-set positions). Adjust tilt of MOB beamsplitter to cancel change in shear. Iterate as necessary.

Open Issue: The description of this use case in AD5 needs to be re-written.

Open Issue – all alignment use cases: Will these eventually be done using ISS data displays, under the control of the ISS?

3.5 Adjust cat's-eye focus use case

Context of Use: Will be done prior to observing each night and several times during the night as the temperature of the cat's-eye changes. The delay lines operate in supervised mode.

Description: The ISS periodically monitors the cat's-eye temperature as measured by the onboard sensors. If the temperature has changed sufficiently since the previous focus adjustment, the ISS waits until next time the trolley stops tracking (for example, when slewing between targets) then commands the delay line to adjust the focus accordingly.

Open Issue: Adjustment may only be needed much less frequently. If so, monitoring the temperature and making focus adjustments could be manual tasks.

3.6 Baseline solution observation use case

Context of Use: After reconfiguring the UT array and subsequently at ~weekly intervals. The delay lines operate in supervised mode.

Description: The ISS slews the UTs to a specified astronomical target and commands all or a subset of the delay lines to follow appropriate sidereal trajectories. The workstation software calculates the common offset [AD4].

The ISS commands the FT software to activate modulation and the FT fringe camera readout. When the UTs and delay lines have reached their tracking state, the ISS commands the FT to perform an automated (XXX option for manual search?) fringe search on a specified set of baselines. The FT commands the delay lines to offset from the sidereal trajectory in an appropriate manner as a function of time and informs the ISS the time at which fringes are seen on each baseline and the offset of the white-light fringe positions from the sidereal trajectory (together with appropriate data-quality measures) at that moment. The ISS stores this information in the monitor and configuration database (MCDB) as well as the contemporaneous delay line baseline model.

The ISS repeats this procedure for ~10 targets evenly distributed over the sky. The ISS then invokes the baseline solution software on the stored data.

3.7 Astronomical observation use case

Context of Use: Multiple times during night. The delay lines operate in supervised mode.

Description: The ISS slews the UTs to a specified astronomical target and commands all or a subset of the delay lines to follow appropriate sidereal trajectories. The workstation software calculates the common offset [AD4].

The ISS commands the FT software to activate modulation and the FT fringe camera

readout. When the UTs and delay lines have reached their tracking state, the ISS commands the FT to perform an automated (XXX option for manual search?) fringe search on a specified set of baselines. The FT commands the delay lines to offset from the sidereal trajectory in an appropriate manner as a function of time and informs the ISS the time at which fringes are seen on each baseline and the offset of the white-light fringe positions from the sidereal trajectory (together with appropriate data-quality measures) at that moment. The ISS stores this information in the MCDB as well as the contemporaneous delay line baseline model. These data may be used offline to determine a baseline solution.

The FT switches to fringe tracking mode (this may be automatic or require another ISS command, depending on the definition of the command used to initiate searching). Science data may now be acquired using the FT or SI. When data acquisition is complete the use case is repeated for a new target.

3.8 Baseline solution use case

Context of Use: After reconfiguring the UT array. Subsequently at ~weekly intervals. Typically the observational data required are obtained from a dedicated baseline solution observation (see Sec. 3.6), but can also come from regular observing (Sec. 3.7).

Eventually the baseline solution software will be invoked automatically by the ISS, but prior to full integration with the ISS, the baseline solution software will be run manually using the supplied user interface, as follows:

Description: Operator retrieves delay line positions, times and baseline models recorded on fringe acquisition from the MCDB and uses these as input to the standalone baseline solution software. The baseline solution software determines a new set of baseline models. The operator may examine the fit residuals and hence flag suspicious data prior to re-fitting. The operator stores the new baseline models in the MCDB. The new baseline models are made available to the DL system controller software, either as DL system command parameters the next time a sidereal trajectory is commanded, or by restarting the DL system causing it to re-read its configuration data (TBD).

3.9 Internal fringes use case

Context of Use: Daytime, after any adjustment/maintenance that may have significantly altered the optical path lengths within the interferometer. The delay lines operate in supervised mode.

The tasks performed by the delay lines in this use case are identical to those needed for the "OPD alignment" use case defined in AD6, hence this document excludes that use case.

Description: The ISS commands the AAS and UTs to set up all or a subset of the beam trains for internal fringes using the MOB broad-band light source retroreflected at the UT tertiary mirrors (this includes: set up MOB light source, set unit telescopes to retro-reflection mode, XXX). The ISS commands all or a subset of the delay lines to follow fixed position trajectories. The ISS commands the FT to activate modulation and readout of the FT fringe camera, then commands the FT to perform an automated (XXX option for manual search?) fringe search on a specified set of baselines. The FT commands the delay lines to offset from the fixed position trajectory in an appropriate manner as a function of time and informs the ISS the time at which fringes are seen on each baseline and the offset of the white-light fringe positions from the fixed position trajectory (together with appropriate data-

quality measures) at that moment. The ISS stores this information in the MCDB.

3.10 FT Spectral calibration use case

Context of Use: After any realignment of the FT dewar optics. May also need to be performed at the start of each observing night, and/or after switching between H and K_s bands in the fringe tracker. The delay lines operate in supervised mode.

There are two alternative ways of generating internal fringes in order to calibrate the FT spectral response. In both use cases the FT system is used as a Fourier Transform Spectrograph (FTS), measuring the fringe envelope shape and hence the spectral response of each spectral channel. The amplitude of the path modulation used to generate the fringes must be larger than the coherence length of a spectral channel, hence the FT modulators cannot be used.

3.10.1 Alternative A: Spectral calibration by single-pass internal fringes

In this alternative, the MOB slides will be used to scan through the fringe envelope. The delay lines are not involved in this scenario.

3.10.2 Alternative B: Spectral calibration by double-pass internal fringes

In this alternative, the delay lines will be used to scan through the fringe envelope. Unit Telescopes must be installed and functional in order to retroreflect the light source back along the relevant beam trains.

Description: The ISS commands the AAS and UTs to set up all or a subset of the beam trains for internal fringes using the MOB broad-band light source retroreflected at the UT tertiary mirrors (this includes: set up MOB light source, set unit telescopes to retro-reflection mode, XXX). The ISS commands the FT to activate modulation and readout of the FT fringe camera. The ISS commands all or a subset of the delay lines to follow fixed position trajectories. Internal fringes are found on the relevant baselines as in the "internal fringes" use case, with the FT commanding the delay lines to offset from their fixed position trajectories.

The ISS commands the FT to deactivate modulation. The ISS commands the delay lines to follow constant velocity trajectories that perform a single scan through the entire fringe envelope. The ISS determines the parameters of these trajectories using the white-light fringe positions that have just been found and the nominal width of the fringe envelope.

The ISS invokes the FT offline data processing software on the monitor data containing the FT fringes, and stores the results in the MCBD.

3.11 Performance Verification Milestone use cases

The technical commissioning plan for MROI involves a series of Performance Verification Milestones, defined in AD1. Each PVM specifies a task-oriented subset of the final interferometer functionality. The following PVMs involve increments to the delay line functionality required:

- (PVM1, 2): No delay line functionality required.
- PVM3: Capability to align one delay line.
- PVM4: Sidereal trajectories and manual fringe searching for two delay lines.

Automated fringe searching desirable but not obligatory.

- PVM5: Sidereal trajectories, manual and automated fringe searching for two delay lines. Closed-loop group-delay tracking using offsets from Fringe Tracker.
- PVM6: As PVM5 but with three delay lines.
- PVM11: Operation with four delay lines.
- PVM12: Operation with six delay lines.

The production software will handle up to 10 delay lines. The remaining delay line functionality required for the PVMs is included in the use cases already defined in previous sections, specifically:

- Alignment of a delay line means both the “align delay line pipe” use case (Sec. 3.2) and the “align metrology laser” use case (Sec. 3.3).
- Sidereal trajectories, fringe searching and fringe tracking are part of the “baseline solution observation” (Sec. 3.6) and “astronomical observation” (Sec. 3.7) use cases.

3.12 Trolley acceptance test use cases

The production software will be used for factory, campus, site, and on-sky acceptance tests of 2nd and subsequent production trolleys. Acceptance tests for the first production trolley will be performed using the prototype software.

Acceptance tests will consist of appropriate subsets (for example there will be no metrology system at the campus test facility) of the tests defined in RD4. The trolley will initially be tested in air, and a subset of the tests will then be repeated in vacuum. The tests are categorized as follows:

- Slew tests: these test the repositioning time of the trolley, the maximum velocity and acceleration, the power consumption and trolley temperatures, and show that the metrology system maintains lock when the shear system and steering loops are closed.
- Tracking tests: these comprise constant velocity tests over a range of tracking velocities (both positive and negative) with tracking speeds from 0.1mm/s to 15mm/s, sets of 10 minute long tracking tests to represent normal observation times and tracking over joins to assess the effect of the join.
- Trajectory tests: these test the re-positioning of a trolley from tracking at one position to tracking at another position, tracking at constant accelerations and tracking reversal and tracking offsets of 0.5 μ m to 10 μ m.
- Roll and shear tests: these check the operation of the shear loop, the secondary tip/tilt servo and the effect of the steering in open and closed loop. They also provide information on the deviations of the delay line pipe.
- Datum tests: these test the repeatability of the datum switch and the overall stability of the datum over a period of time.
- Focus tests: these test the resolution and stability of the cat’s eye focussing system.
- Limit tests: these test the functionality of the limits and the characteristics of the trolley and cat’s eye servos when a limit is encountered.
- Vacuum vessel integrity tests: (no software functionality needed).

The acceptance tests will be performed in standalone mode using the engineering control GUI.

4 Functional Requirements

This section presents the functional requirements for the software implied by the use cases described above.

4.1 Delay Line System Objects

The workstation system controller software shall expose a number of delay line system objects, each of which represents one of the installed delay lines as a single entity. Each system object shall conform to the standard characteristics of an MROI system as described in RD9. The systems shall be “monitoring systems” as defined in RD9.

The systems shall be implemented using the C system code-generation framework. Generic system functionality that is provided as standard by the frameworks (such as the state model) is assumed to be implemented and will not be mentioned further in this document.

The system object shall implement synchronous and asynchronous commands that provide the required delay line system functionality.

Other required functional capabilities that are either optional features or extensions of the system framework are presented in the following sections.

4.2 Data transmission to ISS

The workstation system controller software shall transmit monitor data (this encompasses several of the classes of data defined for the prototype DL software, namely telemetry and status for both the delay line system and its sub-systems) for the delay lines to ISS Data Collector(s). The data available for transmission shall include video from the DL shear cameras.

The data streams available for transmission and their datatypes and units will be listed in the system's Monitor Worksheet (see RD9). These properties are fixed.

Command(s) shall be provided to deactivate individual monitor points. To provide a further capability to reduce the network bandwidth requirements, the workstation shall provide a facility to decimate monitor data by only transmitting every Nth sample to the ISS. It is a goal to provide a command to change the decimation factor for each monitor point individually.

4.3 Fault/Alert transmission to ISS

The workstation system controller software shall transmit fault notifications to the ISS. The information transmitted shall include:

- The name of the fault
- The time at which the fault was detected
- A human-readable message describing the fault condition
- The monitored property, if any, that produced the fault
- The state of the system at the time the fault was produced (the current state according to the standard MROI System state model is provided by the

framework code)

- Optionally, an array of data items that might be relevant to the fault
- Optionally, an array of other faults associated with the fault

If a serious condition occurs that requires the immediate action of the telescope operator, the software shall transmit an alert notification to the ISS. The information transmitted shall include:

- The name of the alert
- The time at which the alert condition was detected
- A human-readable message describing the alert condition
- The level (seriousness) of the alert
- The monitored property that produced the alert (XXX is this optional?)
- The fault, if any, that is associated with the alert

4.4 Axial Control

There shall be four “system modes” that will routinely be commanded for a delay line. These are: FOLLOW, STOP/IDLE, DATUM and DIRECT SLEW. Changes in system mode are initiated by the ISS or from the engineering control GUI running on the workstation.

The reader should note that these system modes are distinct from the lower-level Track and Slew Modes. Track and Slew Modes are labels for different behaviours of the delay line axial control loops, which are used by certain sub-systems in their implementation of the system modes. For example, in FOLLOW mode the metrology sub-system selects either Track or Slew Mode depending upon how far the cat's-eye is from the desired position, whereas the STOP/IDLE and DIRECT SLEW system modes only use Slew Mode.

4.4.1 FOLLOW system mode

In this system mode the workstation commands the metrology sub-system to cause the trolley to follow the trajectory defined by the position and velocity data the workstation is currently sending. The metrology software decides whether or not the trolley must be in Slew Mode in order to reach the condition when Track Mode can be entered so that the trajectory can then be followed precisely.

4.4.2 STOP/IDLE system mode

In this system mode the workstation directly commands the trolley to enter Slew Mode with a velocity of zero. In this situation the trolley is stationary but the metrology loop is not closed.

4.4.3 DATUM system mode

In this system mode the metrology sub-system is commanded to initiate a sequence that will drive the trolley in a safe way so as to eventually reach the datum position and then acquire datum in a consistent fashion.

The trolley onboard coarse position sensor shall be reset to zero on acquiring the datum, and when the trolley sub-system is initialized. Readings from the coarse position sensor will primarily be used in DIRECT SLEW mode.

4.4.4 DIRECT SLEW system mode

In this mode that carriage is driven in Slew mode at a specified (non-zero) velocity, by means of a command sent from the workstation direct to the trolley micro. DIRECT SLEW mode will be used for testing without the metrology software (e.g. in the campus test facility). During normal operations, this mode will be employed for use cases which require the shear loop to be deactivated (when FOLLOW mode might fail due to loss of metrology signal).

4.4.5 Fringe tracking

The DL shall support the closed-loop tracking of atmospheric OPD fluctuations when used in combination with the MROI Fringe Tracker system, by responding to path-correction signals from the FT in real time when in FOLLOW system mode.

To support group delay and fringe phase tracking the DL shall accept an external offset demand signal with sampling rate in the range 1Hz to 1kHz. The upper limit accommodates transmission of an updated demand every quarter-wave modulator step when operating with one FT fringe camera in very poor temporal seeing.

The DL shall transmit timestamped delay line positions to the Fringe Tracker with a maximum latency of 500 μ s (including communication latency), for use in the FT real-time data processing.

4.4.6 Fringe searching

The DL shall support fringe searching in combination with the MROI Fringe Tracker system, by executing commands from the FT to offset from the defined trajectory when in FOLLOW system mode.

Open Issue: How will the offset as a function of time for each DL be parameterized?

4.5 Shear control

Closed-loop shear correction shall be implemented, with the shear sensor software capturing images from the shear camera, centroiding the images (Sec. 6.6), and transmitting correction signals to the on-board trolley software.

It shall be possible to deactivate the shear loop so that the shear sensor measures the non-straightness of the delay line pipe (see Sec. 3.2) or misalignment of the metrology beam (Sec. 3.3).

The target position on the shear camera CCD shall be adjustable, so that small adjustments to the shear servo alignment can be made without physically moving the camera (see use case in Sec. 3.3). The shear servo shall transmit neutral data if the metrology beam is accidentally blocked to protect the trolley electronics and tip-tilt stage.

4.6 Roll control

Closed-loop correction of trolley roll shall be implemented, using the trolley on-board tilt sensor to sense the roll angle.

The target roll angle shall be adjustable to compensate for trolley machining tolerances.

4.7 PMAC configuration

The software shall provide an interface to the trolley's embedded PMAC controller. This interface shall provide the ability to read and write PMAC variables for engineering purposes and a method for downloading control programs to PMAC for software management purposes.

4.8 Sub-system emulation

It shall be possible to perform appropriate tests of the sub-system software when delay line hardware is not available. The software shall respond to commands and generate random-value or fixed-value monitor data. In particular:

- it shall be possible to run the trolley software on a single board computer identical to the on-board micro but without a PMAC controller or input/output boards;
- it shall be possible to run the metrology software on a VME CPU without metrology or input/output boards.

4.9 Workstation data recording

The software shall provide a recording capability for monitor data, for use during standalone testing when the ISS is not available. Recording shall be activated using the engineering control GUI. Incoming data shall be buffered so that recording of recent data can be commanded by the operator if they notice unexpected system behaviour.

Non-image data shall be recorded to FITS binary tables. Shear camera image data shall be recorded to a lossy compressed video format such as MPEG2. The maximum duration of a recording shall be at least 600 sec.

Open Issue: Does the set of data items recorded by the workstation need to be reconfigurable?

5 Performance Requirements

The software shall be capable of controlling the delay line such that the top-level performance requirements listed in Sec. 5.2 of RD3 are satisfied.

Commanded changes to trajectory parameters shall be applied with a delay of no more than 3 seconds. This delay contributes to the 30 sec allowed for a 15m slew [RD3].

6 Computations and Algorithms

6.1 Trajectory Computation

The software shall implement the following classes of trajectory for FOLLOW mode:

- Sidereal trajectory (astronomical observation use case).
- Zero velocity i.e. fixed position (internal fringes use case).
- Constant velocity (trolley acceptance test use case).
- Constant acceleration (trolley acceptance test use case).

The system shall accept a minimal set of parameters defining the trajectory (e.g. for sidereal trajectory these would be the baseline parameters and target coordinates) and use these to calculate demand positions and velocities for the delay line.

In addition, a constant velocity in DIRECT SLEW mode will be used for the align delay line pipe use case.

6.2 Slew Algorithm

The metrology software shall implement a two-stage servo for repositioning of a delay line prior to entering Track Mode:

- For small position errors (less than ~ 10 mm), a damped first-order servo.
- For larger position errors, a constant deceleration servo.

6.3 Fringe tracking step pre-filter

The filter is TBD.

6.4 Common Offset Computation

The common offset is defined in AD4. There shall be a facility to command a change in the assumptions used in the calculation. XXX possibilities for the assumptions.

Open Issue: Various types of assumptions for the common offset calculation are possible, for example: keep a specified delay line in a fixed position; avoid certain position ranges for certain delay lines; or avoid certain ranges of tracking speed

6.5 Baseline Solution Computation

The baseline solution software shall determine a best-fit set of baseline parameters and their uncertainties, given a set of observed fringe positions (ideally for stars evenly distributed over the accessible sky) and the baseline parameters used for the measurements. The software shall output the fit residuals for individual observations.

It shall be possible for the user to fix the values of a subset of the baseline parameters. It shall be possible to solve for a fixed time offset between the correct and actual delay line trajectories.

6.6 Shear Sensor Algorithm

Shear camera images shall be centroided as follows: Ignore the lowest quartile of all pixel brightnesses. Calculate the barycentre of the remaining pixels.

The shear servo shall incorporate a damping term.

7 Hardware Interfaces

The software shall interface to the delay line system hardware components enumerated in the following documents:

- Trolley hardware: RD5 [Trolley Electronics Design Description](#)
- Shear camera hardware: RD6 [Shear Camera Software Functional Description](#)
- Metrology sub-system hardware: RD7 [Metrology System & VME Hardware Design](#)

8 User Interfaces

8.1 Trolley software

8.1.1 Engineering user interface

A CURSES-based user interface to interrogate and set digipots shall be provided. We expect that adjustment of digipots will only be performed when a trolley is first commissioned and when certain components need to be replaced as part of a repair.

8.2 Shear sensor software

8.2.1 Live image display

The shear sensor software shall provide live display of shear camera images on the shear computer console, or on a remote display using the X11 protocol. Note that shear camera images will also be transmitted to the ISS (Sec. 4.2), and could hence be displayed by other (non-deliverable) software.

8.3 Workstation software

8.3.1 Engineering control GUI

An engineering control GUI with data recording capability (Sec. 4.9) shall be provided. This GUI will be used for initial commissioning of the delay line, infrequent alignment tasks, and for trolley acceptance tests (for the 2nd production trolley onwards). The GUI shall allow multiple delay lines to be operated simultaneously. It shall be possible to display one instance of the GUI on any X-windows-capable display.

It shall be possible to run the delay line system controllers without displaying the engineering control GUI. It shall be possible to start and stop the engineering control GUI (typically on a remote display) while the system controllers are running.

8.3.1.1 Controls

GUI elements (buttons, entry fields etc.) shall be provided for:

- Control of system mode and demand trajectory.
- Activation and aborting of standalone data recording.

The GUI shall provide convenient controls for defining demand trajectories suitable for all of the tests listed in RD4. In particular it shall allow the user to initiate a transition between two constant velocity trajectories with offset positions.

8.3.1.2 Displays

The following displays shall be incorporated into the GUI:

- Display of system mode and sub-system states.
- Live text display of monitor data.

Note that shear camera images may be displayed by running the shear sensor software over an X11 connection, hence there is no need to integrate image display

into the engineering control GUI.

Open Issue: Do we need live data plots? This capability was to have been provided by the RTC Engineering GUI.

8.3.2 Standalone analysis GUI

A standalone analysis GUI shall be provided that reads data recorded in FITS format by the engineering control GUI. The analysis GUI shall incorporate controls for:

- Plotting all available data for a specified delay line sub-system, against time.
- Plotting two specified data streams (perhaps from different sub-systems), either with a common time axis or against one another.
- Plotting the power spectrum of a specified data stream.
- Plotting the integrated power spectrum of a specified data stream.
- Saving specified data to Matlab workspace for further processing/analysis or export

8.3.3 Baseline solution UI

A simple standalone user interface to the baseline solution software shall be provided. This shall read input data from ASCII files.

Open Issue: What kind of standalone user interface is needed: terminal, command-line, or minimal GUI?

9 Communications Interfaces

This section presents the specifications for the communication interfaces between the delay line system and the ISS and Fringe Tracker.

A preliminary data dictionary (written for the prototype software) is available in RD2.

9.1 Configuration Interface

In normal operations, the delay line systems shall retrieve all configuration data on startup from the ISS monitor and configuration database. Any data needed to get the system into a state in which it can access the database is excluded from this requirement.

The system may use the standard protocols provided by the system code-generation framework to communicate with the ISS Database Manager in order to retrieve data from the MCDB. Alternatively, a facility provided by the ISS may be used to write configuration file(s) containing ASCII “[key]=[value]” pairs using information retrieved from the MCDB. These configuration file(s) would be read by the system on initialization.

There is no requirement for reconfiguration while the system is running.

9.1.1 Configuration for standalone use

The ISS facility to write ASCII configuration file(s) will be used to configure a single delay line for standalone use. It shall be possible to use these configuration file(s) to configure a delay line for standalone tests when the ISS database is not available.

Open Issue: do we need to configure *multiple* delay lines for simultaneous use without the ISS?

9.2 Command/Data Interface

The system shall use the standard protocols provided by the adopted system code-generation framework to receive commands from the ISS and to return data on completion of a command.

9.3 Low-latency FT Interface

There shall be a dedicated interface with latency $< 200 \mu\text{s}$ for transmission of tracking offset demands for each delay line from the MROI fringe tracking system to the delay line metrology sub-system. The interface will also be used to transmit measured delay line positions back to the fringe tracker.

In order to allow development of the production delay line metrology software to proceed, MRO and Cavendish Laboratory have agreed to use the RTnet real-time ethernet stack for low-latency communications between the FT Fringe Engine and the delay line metrology software.

10 Non-functional requirements

10.1 Operating Environments

The software shall be able to operate the delay lines in the following environments, which have the unique features mentioned:

- Cambridge test rig (short length, with or without metrology hardware).
- Campus test rig (short length, no metrology hardware).
- Magdalena Ridge (1 to 10 delay lines).

It shall be possible to run the software and capture monitor data with the trolley mounted on the handling gurney or resting on a laboratory bench. Note that in this case special connections to the trolley for power and communications may be required (TBD).

It shall be possible to operate two delay lines simultaneously using the production metrology software in conjunction with the *prototype* trolley, workstation and shear sensor software, during the period prior to delivery of the complete production software package (please refer to the schedule in RD1).

10.2 Coding Standards

The components of the delay line software that will be written from scratch (e.g. metrology software) or extensively re-worked (e.g. workstation software) shall be coded in C++ where appropriate. Linux kernel modules must be written in C, so device drivers and hard-real time code will be written in C.

We expect to carry forward the majority of the coding guidelines described in AD3, with adjustments reflecting the adoption of C++. A revised document will be issued in due course.