NGAO

Real Time Controller

Design Document

Revision History			
Version	Author	Date	Details
1.1	Marc Reinig and Donald Gavel	May 8, 2009	First draft version. Derived from MR's draft version 4.17
Ver. 1.1.347		Oct 2, 2009	Updated Draft

Table of Contents

Table of Figures

Table of Tables

1. Introduction and Organization

This document covers the design of the Keck Next Generation Adaptive Optics Real Time Controller (NGAO RTC).

This section is an overall introduction and summary of the RTC architecture and implementation and the organization of the document.

Section 2 describes the overall system characteristics.

Section 2.4.3 covers the latency and timing requirements that drive the architecture and implementation decisions.

Sections 4 through 11 cover the individual sub-systems in the RTC.

Subsequent sections cover RTC issues of a general nature or related topics.

For details on the algorithm used, see [ref NGAO Algorithm Document].

1.1 *Background and Context*

The design of the Real-Time Processor for the Keck Next Generation Adaptive Optics system represents a radically new computational framework that blazes new territory for astronomical AO systems. The need for a new approach is due to the considerable step up in the amounts of data to be processed and controls to be driven in a multiple guide star tomography system. Furthermore, KNGAO has set ambitious new goals for wavefront correction performance that puts further pressure on processing rates and latency times that traditional compute paradigms have trouble scaling up to. Current processing requirements are in the Terra Operations per second region. Instead, the KNGAO RTC is structured around a massively parallel processing (MPP) concept, where the highly parallelizable aspects of the real-time tomography algorithm are directly mapped to a large number of hardware compute elements. These compute elements operate separately and simultaneously. With this approach, increasing problem size is handled roughly by increasing the number of processor elements, rather than processor speed.

In short, in this design, the structure of the algorithm drives the structure of the hardware. Therefore, it is necessary to understand the algorithm itself to understand the hardware architecture, so please review the companion volume [ref Algorithms Design Document], which presents the technical details of the massively parallel algorithm for AO tomography.

For the purpose of introducing the hardware design, the algorithm and hardware have three main top-level component steps: wavefront sensing, tomography, and DM control. These physically map to component blocks of hardware, see **Figure 2**. The actual physical layout of boards and connecting cables mimic this geometry. Inside the blocks, further parallelization is achieved by use of Fourier domain processing, where each spatial frequency component is processed independently. Fourier-domain algorithms have been developed in recent years for wavefront phase reconstruction [ref Poyneer] and for tomography [ref Tokovinin, Gavel].

As stated, the real-time processing takes place on small compute elements, either Field-Programmable Gate Array (FPGA) logic chips, or Graphical Processing Units (GPUs).

FPGA devices have been used in the digital electronics industry since the 1980's when they were first used as alternatives to custom fabricated semiconductors. The arrays of logic were "field-programmed" to produce any digital logic function. They grew in a role as support chips on board computer motherboards and plug-in cards. Recent versions incorporate millions of transistors and some also incorporate multiple on-chip conventional arithmetic processors. Their use is now common in many products from high performance scientific add in boards to consumer portable electronics including cell phones. Today these chips represent a fast growing, multi-billion dollar industry.

Graphical Processing Units were first introduced to offload the video processing for computer displays from the computer's CPU. They have since developed as a key processor for digital displays, which have demanding applications in computer-aided design and video gaming. GPUs specialize in geometric rotations, translations, and interpolation using massive vector-matrix manipulations. Beyond graphics, GPUs are also used in analyzing huge geophysical signal data sets in oil and mineral exploration and other scientific calculations requiring extremely high performance. GPUs can be adapted to the wavefront reconstruction problem through these means and, at this point, they remain a possible low-cost option in AO for wavefront sensor front-end video processing, but they are still under evaluation for this purpose.

The RTC uses a mix of GPUs and FPGAs to best match their individual strengths to the RTC's needs.

1.2 *Design Philosophy*

To insure the maximum reliability, serviceability, and maintainability, we will maximize the use of commodity parts and industry standard interfaces.

We have attempted to match the structure of the problems to the form of the solutions to the implementation. This maintains the visibility of the various aspects of the original problem as they are transformed through the algorithms and implementations.

It is not often you can examine several alternate implementations in full detail. Each of several possible algorithms has several possible implementations. We have tried within the budgeted time and resources to investigate these as far as possible.

1.3 *Functions of the RTC System*

Measure the atmosphere, producing wave fronts based on these measurements

Compute the tomographic reconstruction of the atmosphere

Drive DMs

Compensation for Vibrations (Reference the section)

Provide diagnostic and telemetry data.

The RTC System is a real-time digital control system that operates on a fixed clock cycle, referred to as the Frame Clock. Each Frame, the RTC takes data from various cameras, processes it, and updates the various mirrors and DMs for which it is responsible.

It runs the same program on each frame, with different data, but the same parameters, repeatedly, until the AO Control causes it to change.

The RTC interacts with the telescope system and the AO Control through two Ethernet connections: the RTC Control Processor (CP) and the RTC Disk Sub-system.

The CP is based on a conventional Linux box. It acts as the synchronizing interface between asynchronous requests coming from the AO Control and the synchronized parameter loading to the interconnected RTC components.

The "state" of the RTC is affected by the AO Control via a parameter loading process. This loading occurs in the background to the real-time cycles. Parameter updates to the PE's [define] program occur synchronized to the frame clock. The companion document [ref Processing Engine to AO Control Interface Design Document] describes the available control parameters and the update process in detail.

In order to accommodate differing natural guide star's (NGS) brightness, the rate of the camera clocks can be set differently on each wavefront sensor with respect to the tomography engine. This is useful to allow overall system optimization by trading individual wavefront or tip/tilt/focus sensor signal-to-noise ratio with sensor bandwidth. All sensor sample periods however must be less than the tomography sample period in order for the system to remain synchronized at its maximum rate. The master frame clock is derived from one of the HOWFS cameras.

Figure 1 shows a top-level view of the RTC Processor System in its computational and data flow environment.

1.4 *Functional Architecture*

Figure 1 and **Figure 2** depicts the overall design, emphasizing real-time signal flow. **Figure 21** highlights the flow of telemetry data and **Figure 10** highlights the flow of parameter loads and status feedback to and from the control processor.

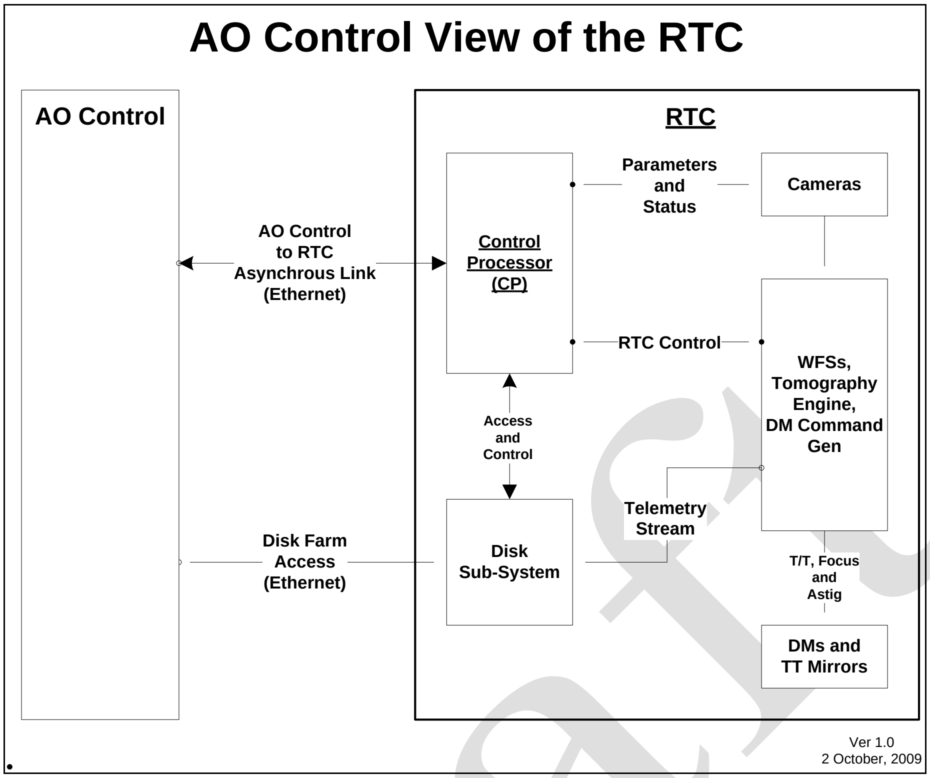


Figure 1 Top-level view of the AO Control's View of the RTC

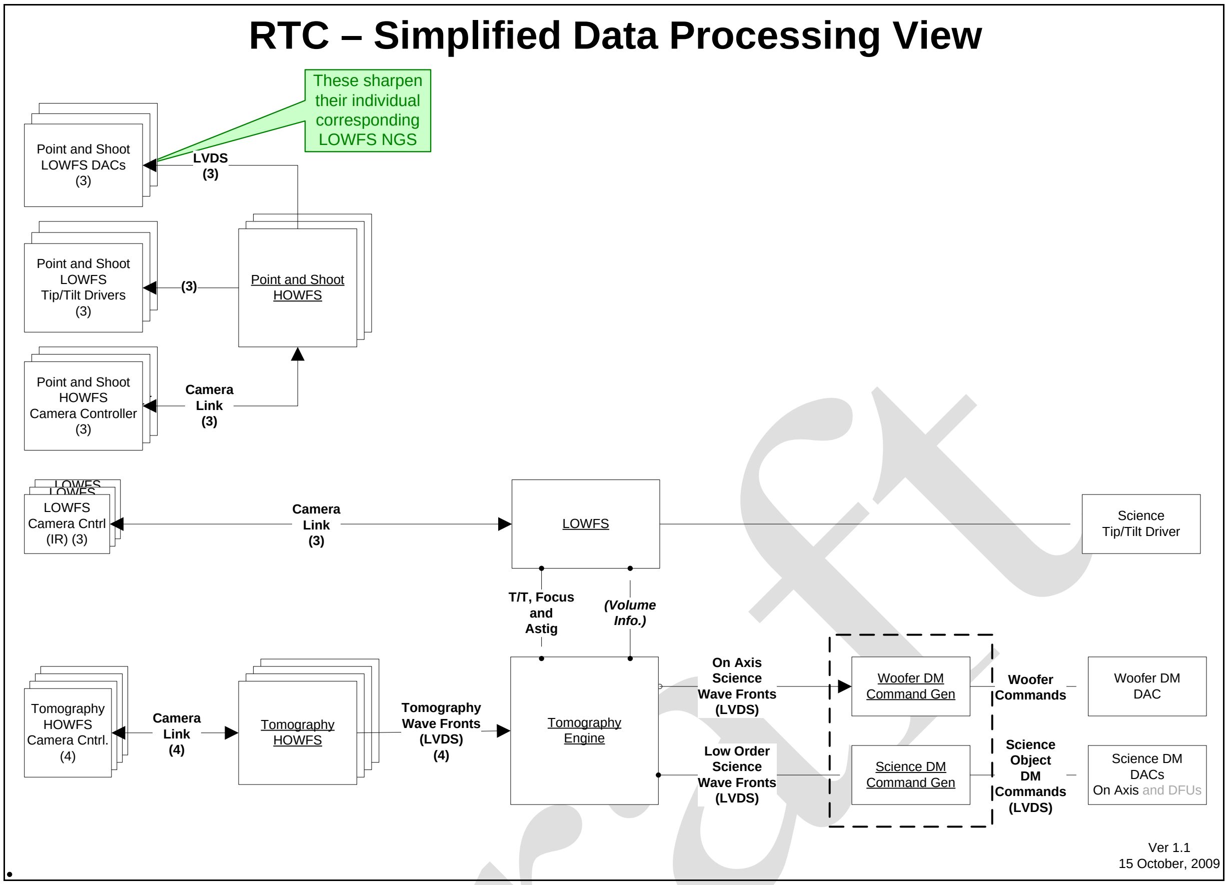


Figure 2 Simplified View of the RTC

1.4.1 Wave Front Sensors (WFS)

A short overview: There are two types of WFS in the NGAO RTC: low order (LOWFS) and high order (HOWFS).

1.4.1.1 LOWFS

LOWFS use an IR natural guide star (NGS) to estimate tip/tilt, focus, and astigmatism for the science field.

A low-order wavefront sensor (LOWFS) processor is responsible for converting raw camera pixel data into a tip/tilt signal, plus focus and astigmatism numbers for the one TTFA sensor. Algorithm details are given in the [Algorithm Design document]. Each LOWFS camera operates in parallel and feeds its data to the LOWFS. The LOWFS processes the data and sends commands to the science tip/tilt actuator on the Woofer and to the tomography engine.

1.4.1.2 HOWFS

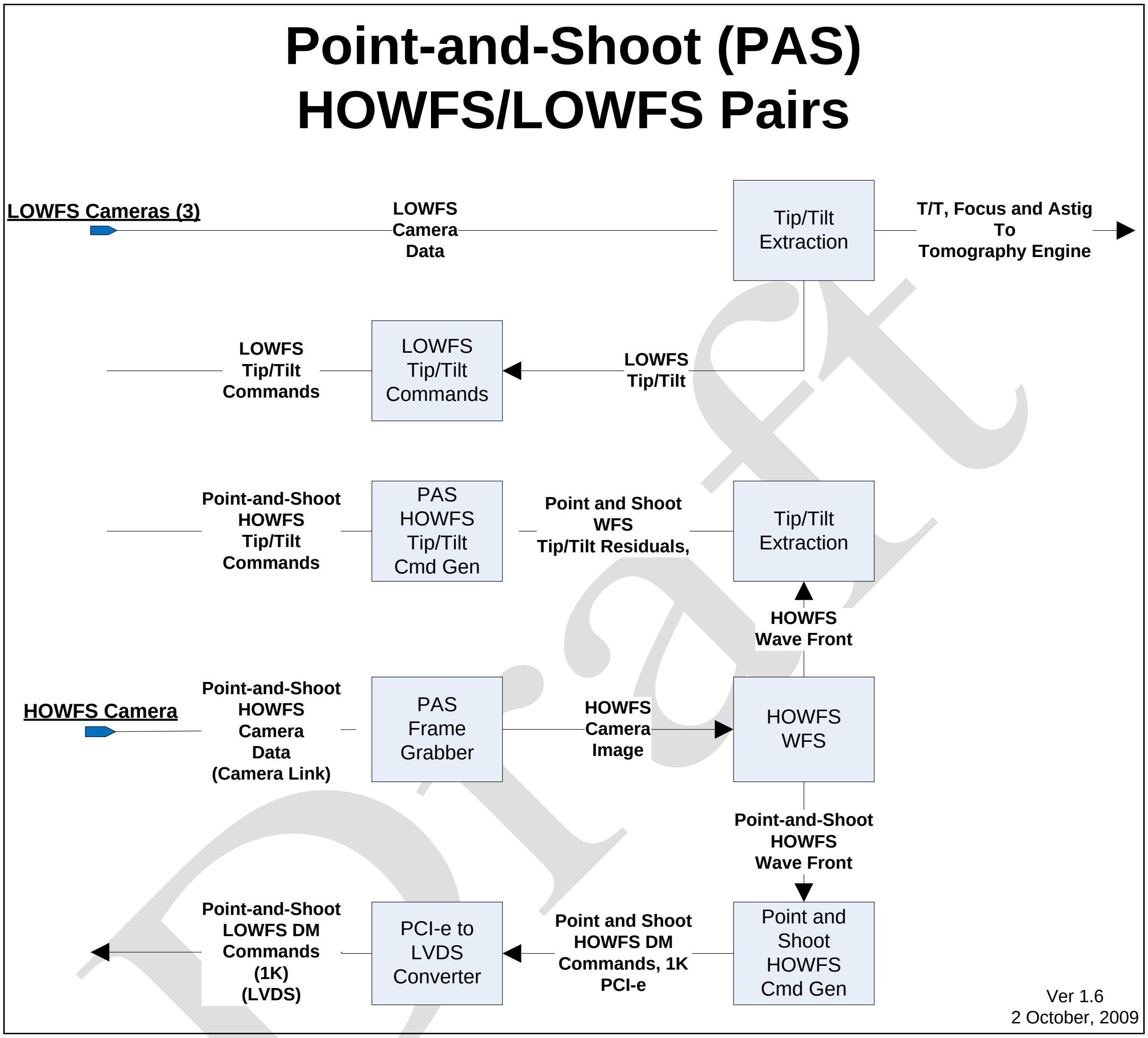


Figure 3 Point-and-Shoot HOWFS/LOWFS Pair

A high-order wavefront sensor processor is responsible for converting raw Hartmann WFS camera pixel data into an array of wavefront phase measurements spaced on a regular grid in a coordinate system that defined as common throughout the AO system. HOWFS use a laser guide star (LGS) or an NGS to probe the atmosphere and determine the high order atmospheric distortion in that direction. There are two types of HOWFS:

The Point-and-shoot HOWFS first generates a wave front to correct light in the direction of the LGS. This wave front is then converted to DM commands that are used to determine the shape of a DM. This DM is used to sharpen their associated LOWFS NGS.

Tomography HOWFS provide probing of the atmospheric volume containing the science object and generate a wave front in the direction of the LGS. These wave fronts are processed by the tomography engine and used to estimate the volume in the direction of the science object. The estimate of the volume is in turn used to provide a wave front to correct light in that direction. This wave front is passed to a DM command generator to place the desired shape on the science DM.

Each HOWFS wavefront sensor has an associated wavefront sensor processor. These operate in parallel then feed their aggregate results to the tomography engine.

The architecture for the hardware for all WFSs follows the massively parallel approach. First, each wavefront sensor has an associated wave front sensor processor card, operating in parallel with but otherwise independent of (other than the frame synchronization clock) every other processor card.

At the present time, it is not established whether the WFSs will be implemented using FPGAs or with GPUs.

1.4.2 Tomography Engine

The tomography engine's processors are mapped "horizontally" over the aperture. A given processing element (PE) on this map is assigned a piece of the aperture and alternates processing a portion of either the spatial or Fourier domain of it.

All computational elements run the identical program, albeit with different parameters and data. Each processor in the tomography engine is connected to its four neighboring elements (representing the next spatial frequency over in both directions) because is it necessary to shift data to neighbors in order to implement the Fourier transform and interpolation steps in the tomography algorithm.

1.4.3 DM Command Generators

The Point-and-Shoot HOWFS's and the tomography engine's wavefronts are sent to the deformable mirror command generators for their DMs. These are dedicated units, one per DM. They take desired wavefronts in and produce DM command voltages that result in the correct wavefront, taking into account actuator influence functions and nonlinearities of each DM.

1.4.4 Control Processor (CP)

1.4.4.1 Synchronization of the RTC with the AO Control

1.4.5 Disk Sub-System

The RTC Disk Sub-System is an extremely high performance, large storage system that is designed to capture system Telemetry data that can be generated by the RTC. This data can be generated at over 36 GBytes/sec. This system is designed to be able to capture this data for an extended period, but it is not intended to be an archival system for long-term storage or access.

The system has a capacity to store up to 60 terra bytes of data, which could be filled in a matter of days of heavy use.

Likewise, no data base facilities are provided beyond normal directory services.

1.4.6 Global RTC System Timing

Sub msec accurate time stamping of data will be provided for the telemetry stream. This will provide

1.4.7 Diagnostics and Telemetry Streams

All major data signals can be sent to the Telemetry/Diagnostic data path at the request of the AO Control. Any data in this path can be sent to either-or-both the AO Control, for diagnostics, or the RTC disk sub-system, for storage. See: Section 8.

Data that can be saved through Telemetry or viewed through Diagnostics include (See Section 9):

1. Raw Camera Data
2. Centroids
3. Wave Fronts
4. Tomographic Layers
5. Science on-axis High Order Wave Front
6. Science on-axis Woofer Wave Front
7. Science on-axis High Order Commands
8. Science on-axis Woofer Commands
9. RTC Current Parameter Settings

All Diagnostic information is time stamped accurate to one Frame Time.

1.5 *Physical Architecture*

The RTC is physically divided between the Naismith and the computer room below the telescope. The cameras and their controllers and the DMs and their are located on the Naismith. The rest of the RTC, the WFS, tomography engine, disk sub-system, DM command generators, control processor, and clock generator are located in the computer room. The two sections are connected by high speed fiber links.

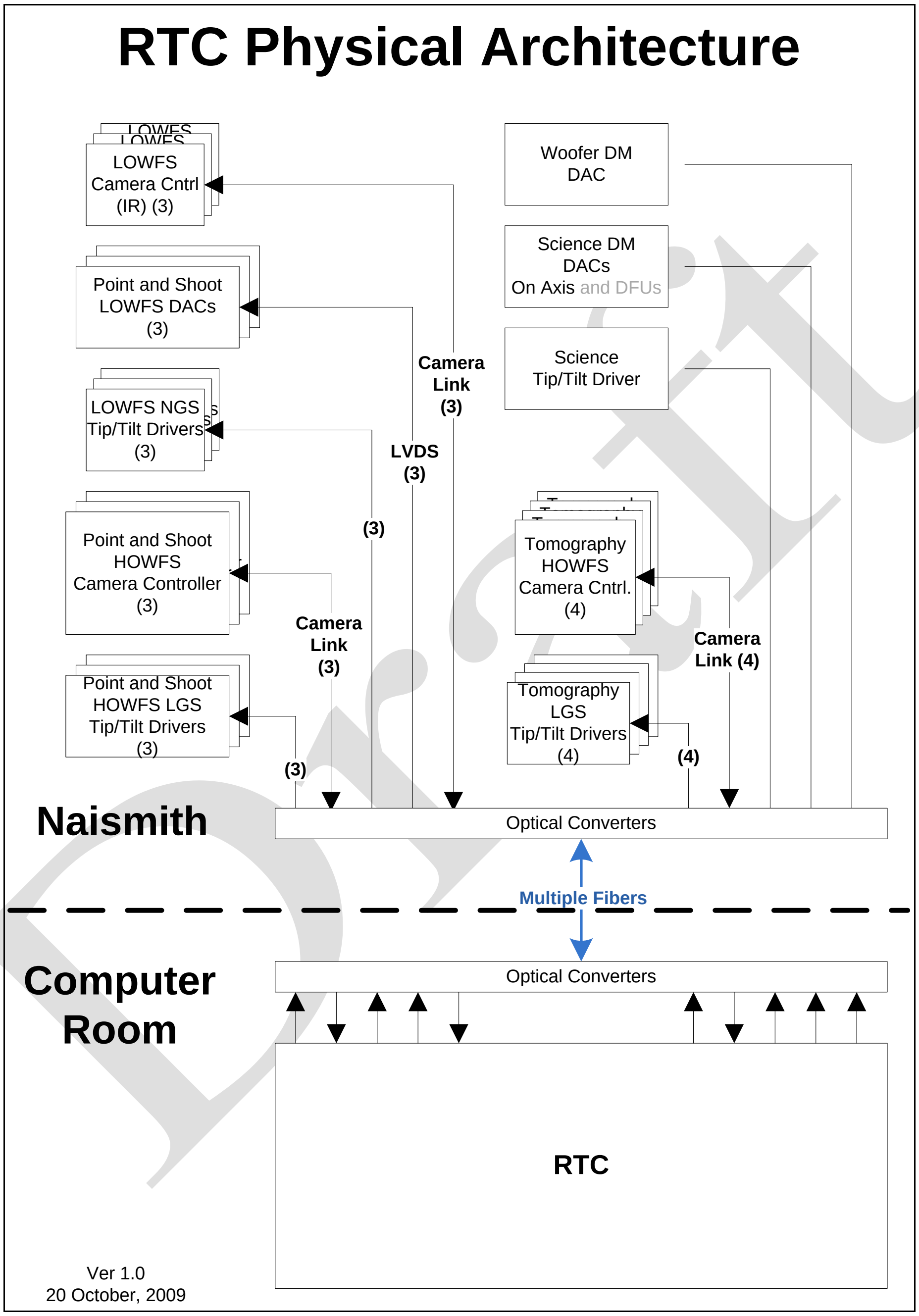


Figure 4 RTC Physical Architecture

1.6 *Implementation Alternatives*

1.6.1 Multi-core CPUs

1.6.2 FPGAs

XXXX

FPGAs have a significant advantage over traditional CPUs in that you can change both their hardware and software.

1.6.3 GPUs

XXX

We are examining different candidate technology implementations for the WFS and the DM Command Generators.

The WFS can be implemented using conventional CPUs, FPGAs, or GPUs. We are currently determining the best fit between the requirements and implementation technology.

The Tomography Engine is currently being developed using Field Programmable Gate Arrays (FPGAs). No alternate implementation is being pursued.

The DM Command Generation can be implemented using custom logic or GPUs. We are currently determining the best fit between the requirements and implementation technology.

2. System Characteristics

2.1 *Assumptions and Performance Requirements*

Item No.	Item	Value	Units	Ref
1	Wind		m/sec	
2	r0		cm	
3	Max Zenith Angle	54	degrees	
4	Frame Rate	2	Hz	
5	Stare Time HOWFS	500	µsec	
6	Stare Time LOWFS	4,000	µsec	
7	Sub Apertures	64		
8	Number of Tomography Layers	5		
9	Number of Tomography WFSs	4		
10	Number of WFSs for Tip/Tilt, focus, and astigmatism	3		
11	Number of Science Objects	1		
12	MTBF		per KHr	
13	MTTR		per KHr	

Table 1 NGAO RTC System Assumptions

Item No.	Item	Value	Units	Ref
1	Wind		m/sec	
2	r0		cm	
3	Max Zenith Angle	54	degrees	
4	Frame Rate	2	Hz	
5	Stare Time HOWFS	500	µsec	
6	Stare Time LOWFS	4,000	µsec	

7	Sub Apertures	64		
8	Number of Tomography Layers	5		
9	Number of Tomography WFSs	4		
10	Number of WFSs for Tip/Tilt, focus, and astigmatism	3		
11	Number of Science Objects	1		
12	MTBF		per KHr	
13	MTTR		per KHr	

Table 2 NGAO RTC System Performance Requirements

2.2 *RTC States Visible to the AO Control*

There are multiple levels of states in the AO system. This section describes only those states that the RTC keeps track of. These states are limited and insure only that the RTC cannot damage itself, will not give invalid data, and that it can be controlled to move between states become functional or be shut down.

Figure 5 shows the states visible to the AO control.

Finer grained information may be available for diagnostics, but are not presented here.

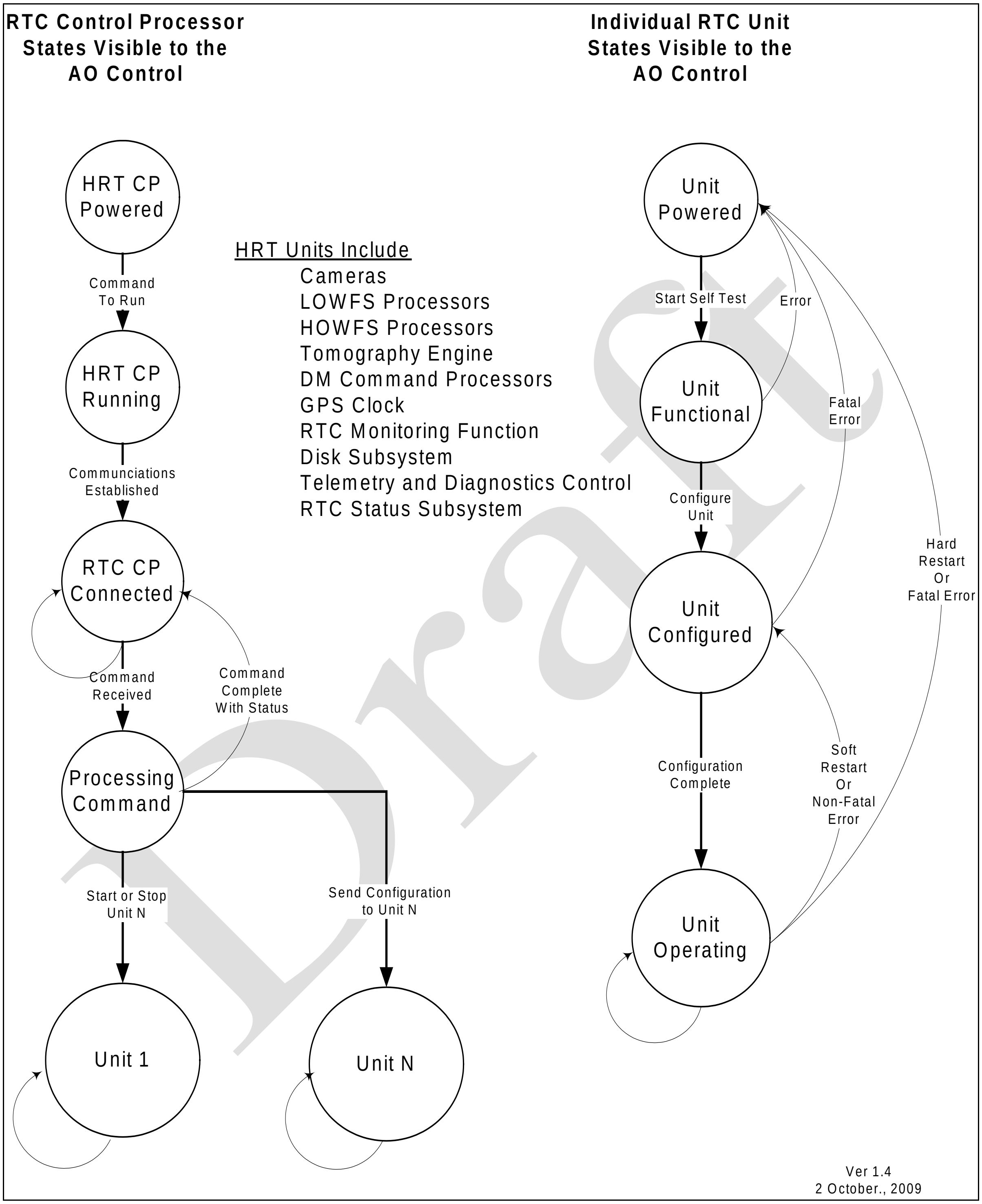


Figure 5 RTC and Sub-System States Viewable by the AO Control

The RTC CP will not honor a request to do something that is not valid, given its current state, and will return an appropriate error to the AO Control.

Examples:

1. If a sub-system has not been configured and the AO Control attempts to start the sub-system, the RTC will return an error indicating “Not Configured”.
2. If the AO Control attempts to set a parameter with a value that is out of the allowable bounds for that parameter, the RTC will return and error indicating “Out of Bounds” with an additional message that indicates the allowable bounds. This is not really a “state” issue, but is mentioned here for clarity.
3. However, if the AO Control attempts to set a parameter with a valid value but one that does not make sense in the presence of other parameters that have been set, the RTC will simply set that parameter. The states represented by various combinations of settable parameters have meaning to a higher level of control than the RTC and should be controlled by that level.

2.3 *Timing and Control of Internal States of the RTC*

See Section 2.2 for a discussion of the States of the RTC visible to the AO Control.

Several sub-systems of the RTC run at different rates or start at different times during operation. The LOWFSs run at approximately 250 Hz while the HOWFSs run at approximately 2 KHz. In addition, commands to change parameters come from the AO Control asynchronously. The RTC synchronizes these changes in parameters or data to occur at the start of the basic frame, ~2 KHz. This ensures that all processing that occurs during a frame is done with the same parameters and input data.

Anything arriving too late to be applied at the start of a frame will be applied at the start of the next frame.

2.4 *Sizing and Scaling*

2.4.1 Sizing the Problem

2.4.2 Sizing the Solution

2.4.3 Scaling to Different Sized Problems

2.4.4 Scaling to Different Speed Requirements

2.5 *Reconfigurability*

2.6 *Reliability and Recoverability*

2.6.1 SEUs

2.6.2 BER

2.6.3 Diagnostic Tools

3. Error Budgets, Latency, and Data Rates

3.1 *Error Budgets*

Ref [______________________]

3.1.1 Latency

3.1.2 Accuracy

The accuracy of a system is determined by several factors. In our system, they are the accuracy of the:

- Camera data
- Parameters used in calculations
- Algorithms used
- Arithmetic operations
- Output hardware

3.1.2.1 Camera Data

The camera data supplied is 16 bits. We have assumed that the camera data is accurate to 14 bits.

3.1.2.2 Parameters

All fixed parameters and arrays used in calculations are represented as 18-bit numbers for real values with an added 18-bit imaginary part if they are complex. Examples of the parameters are C_n^2, Fourier coefficients, Kolmogorov filter coefficients, asterism parameters, DM modeling tables, etc.

3.1.2.3 Algorithms

Don, do you want to say something here?

3.1.2.4 Arithmetic Operations

Individual arithmetic operations take 18-bit values and accumulate a 45-bit result.

3.1.2.5 DACs

The DM's are supplied with 14-bit values.

3.2 *Latency*

The most significant element to overcome in an AO system is the latency between sampling the atmospheric turbulence and applying the compensation for it. See Appendix C for calculations on the impact of latency components on the wavefront error.

It and the spatial sampling determine the rate at which calculations and data transfers must be made to achieve a given level of compensation.

These in turn determine the size of the computation engine that is needed to handle the problem. For the current NGAO specification, this is a terra operation per second problem, which directly affects and limits the choices of algorithms, architectures, and implementations.

The total RTC latency is made up of several components, illustrated below (not to scale).

3.2.1 Calculating Latency

Reference __________

3.2.1.1 ½ Frame for stare time

3.2.1.2 ½ Frame for DM hold time

Description of each element

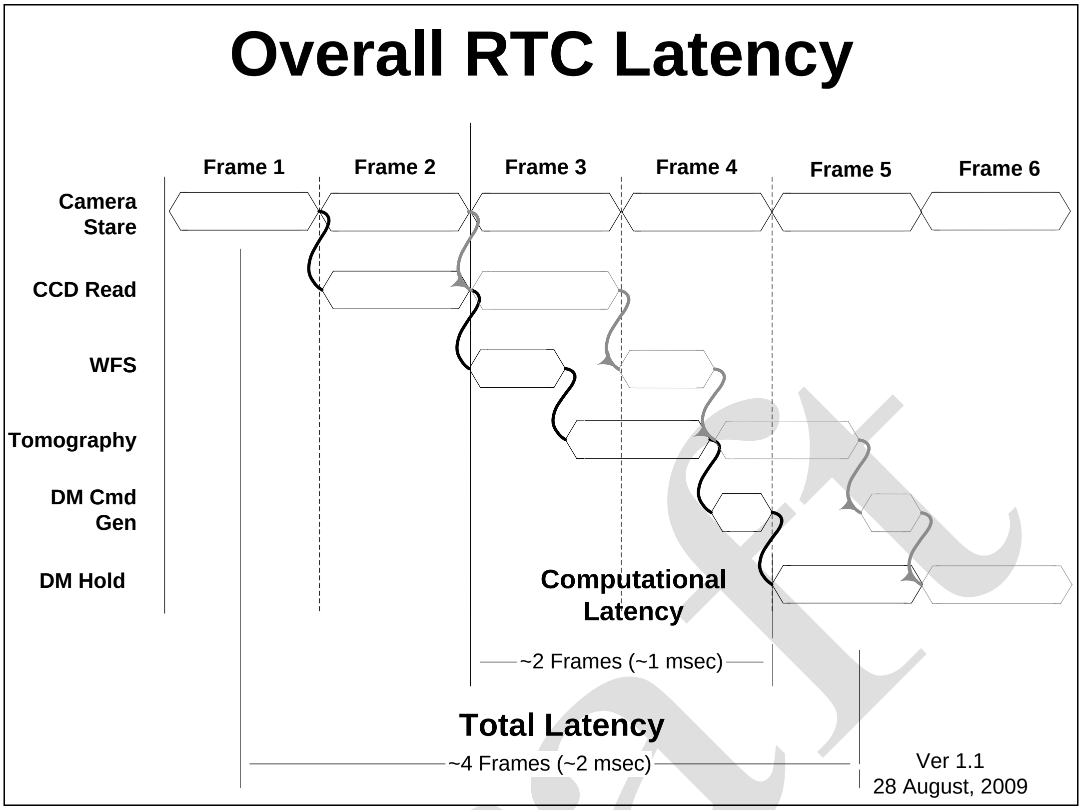


Figure 6 Overall RTC Latency Components (not to scale)

Example Timing

The total latency is calculated from the midpoint of the camera stare time to the midpoint of the subsequent DM hold-time. Computational Latency includes only the portion of time due to our actual processing of the data.

3.3 *Non-Pipelined vs. Pipelined Architectures and Latency*

Processing system architectures can be divided into two major categories: non-pipelined and pipelined.

3.3.1 Non-Pipelined Architectures

In a non-pipelined system, data is brought in, processed by a single processing unit, and the results are used, with latency, L (refer to non-pipelined diagram), and an effective cycle time of L. New data is brought in at time T_0 and corrections are made at time $T_0 + L$, whereupon new data is brought in again. The update rate is 1/L.

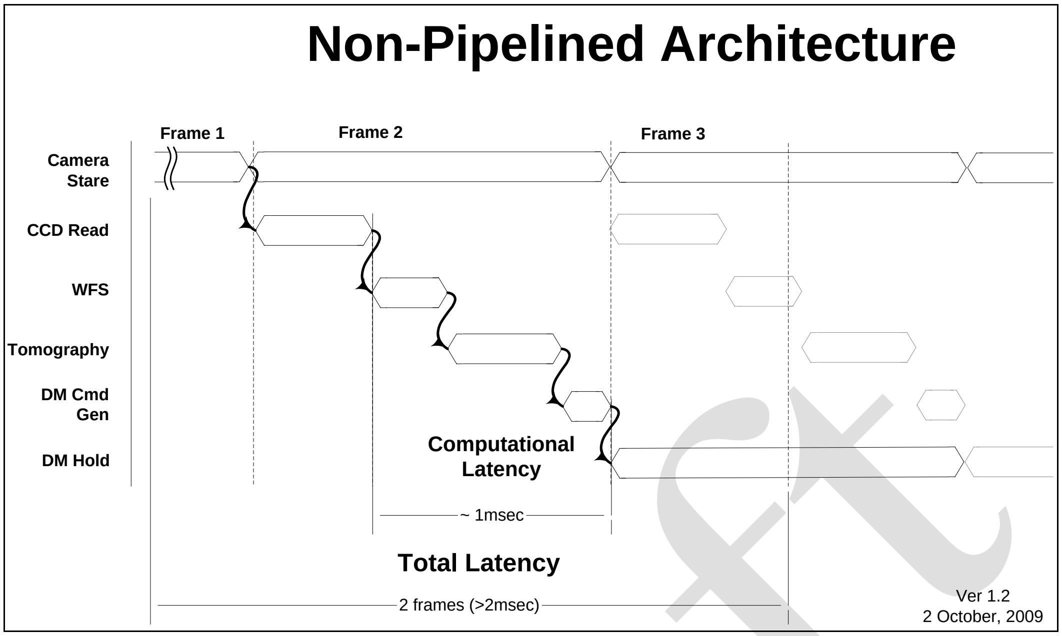


Figure 7 Non-Pipelined Latency Issues (not to scale)

Example Timing

In a non-pipelined system, the frame rate is determined by the total latency of handling the data. In this case, it is the time to read the CCD plus the computational latency.

All calculations must be finished before new calculations can start. The total latency is two frames (1 frame to handle the data, plus ½ a frame on the front end to account for the integration time of the camera and ½ a frame at the end to account for the integrated effect of the DM hold time).

3.3.2 Pipelined Architectures

In a pipelined system, the processing is divided amongst several units. Each unit processes the data and passes it on to the next unit with the total time still being L, as above (assuming transfer times between units are negligible). This means the computational latency is the same, L. However, it can be seen that new data can be brought in as soon as the first unit is through processing its last data, and the output can be updated as soon as the last unit has processed its data. The update rate here is 1/M where M is the largest time spent by any individual unit. This rate is considerably faster than that in the non-pipelined case.

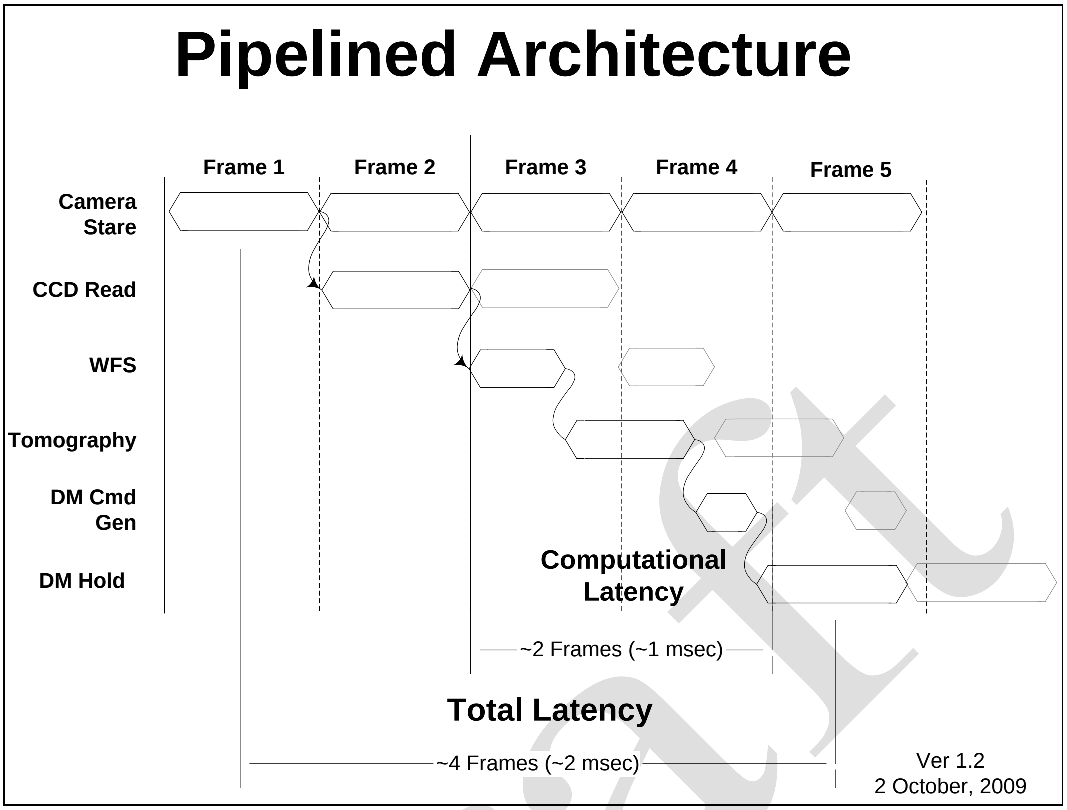


Figure 8 Pipelined Architecture Latency Issues (not to scale)

Example Timing

In this system, each element of the data processing is handled by separate hardware. This allows us to have several frames of camera data being processed somewhere in the pipe at the same time, just not in the same piece of hardware. The maximum frame rate is determined by the longest time it takes for any single element to process its data.

In our situation, the frame rate is determined by the CCD read time, which happens to be 500 µsec. All other individual computational elements are less than this. The time to handle the data is the time to read the CCD plus the Computational Latency (~3 Frames). The total latency is 4-2KH frames (3 frames to handle the data, plus ½ a frame on the front end to account for the integration time of the camera and ½ a frame at the end to account for the integrated effect of the DM hold time).

While in Figure 8, the computational Latency ends on a frame boundary, it is actually asynchronous to the frame clock. This means the DM operates on a clock that is the same frequency as the frame clock, but shifted by a not necessarily integral amount. See Figure ________.

Assuming the time to handle the data is the same for both the pipelined and the non-pipelined case, the pipelined architecture will provide less total latency. It can require more hardware, however.

For the NGAO RTC, we use a pipelined architecture.

3.4 *Latency Calculations*

Latency can be divided into three categories: transfer, compute, and fixed latencies.

Transfer latency is the result of moving data from one place to another, usually between physically distinct hardware.

Compute latency is due to the time it takes to apply an algorithm on the data, and includes any minor transfer latency that may occur during that process.

Fixed latency is due to system architecture issues that are separate from the above. In the case of the RTC, it is the result of the value of the frame clock and the time read the data out of the CCD subsequent from the actual time to transfer it to the WFS frame grabber.

Detailed latency analysis can be found in Appendix C.

Table 3 shows the system parameters used in the following latency analysis.

Extended Sub Apps are the total subapertures we use in computations to account for the FOV and zenith angle.

Frame Rate (Hz)	Sub Apps	Ext Sub Apps
2,000	64	88

Table 3 NGAO RTC System Parameters.

The frame rate of 2 KHz is driven by the maximum tomography error allowed (see Appendix C and [Requirements Doc]).

There are 64 subapertures across the primary aperture.

In order to avoid artifacts due to wrapping while processing Fourier data, the number of apertures used internally is 88.

3.4.1 Transfer Latency

Table 4 shows the transfer latencies that affect the total latency of the RTC. Also shown are the data rates needed to support the required Telemetry of system data during operation.

Ver 1.1	1 Sept, 2009	Tomographic Data Rates And Latency Calculations							
Transfer Latencies	Number of Items	Size per Frame (B)	Rate for Diag/Telem (MB/Sec)	Total Diag/Telem Rate (MB/Sec)	Operational Xfer Latency Budget (µSec)	Operation Rate (MB /Sec)	Transfer Rate (GB/Sec)	Total Op. Rate (MB/Sec)	Notes
WFS CCD Read Out	1	131,072	N/A	N/A	500	N/A	N/A	N/A	
HOWFS Camera to Frame Grabber	4	131,072	262.14	1,049	50	2,621	N/A	10,486	1
HOWFS Frame Grabber to Centroider	4	131,072	262.14	N/A	50	2,621	N/A	10,486	2
HOWFS Centroider to Telemetry	4	30,976	61.95	248	N/A	N/A	N/A	N/A	3
HOWFS Wave Front	4	15,488	30.98	124	8	2,000	2.00	8,000	4
Tomography Layer	5	15,488	30.98	155	8	2,000	2.00	10,000	5
DM Cmd	4	8,192	16.38	66	50	N/A	N/A	N/A	6
Total Tomographic Transfer Time				1,641	665			38,972	

Table 4 Tomography Transfer Latencies and Rates

Ver 1.1	1 Sept, 2009	NGS Data Rates And Latency Calculations							
Transfer Latencies	Number of Items	Size per Frame (B)	Rate for Diag/Telem (MB/Sec)	Total Diag/Telem Rate (MB/Sec)	Operational Xfer Latency Budget (µSec)	Operation Rate (MB /Sec)	Transfer Rate (GB/Sec)	Total Op. Rate (MB/Sec)	Notes
WFS CCD Read Out	1	131,072	N/A	N/A	500	N/A	N/A	N/A	
HOWFS Camera to Frame Grabber	3	131,072	262.14	786	50	2,621	N/A	7,864	1
HOWFS Frame Grabber to Centroider	3	131,072	262.14	N/A	50	2,621	N/A	7,864	2
HOWFS Centroider to Telemetry	3	30,976	61.95	186	N/A	N/A	N/A	N/A	3
HOWFS Wave Front	3	15,488	30.98	93	8	2,000	2.00	6,000	4
DM Cmd	4	8,192	16.38	66	50	N/A	N/A	N/A	6
Total NGS Transfer Time				1,131	658			21,729	

Table 5 NGS Transfer Latencies and Rates

Notes:

Operational transfer times must take place in a small part of a frame (~50 µsec ~10% of a frame) whereas Diagnostic/Telemetry transfers may take an entire frame (~500 µsec). This leads to a much higher rate for Operational transfers.

The two key parameters calculated are the "Total Diagnostic/Telemetry Rate" and the "Operational Transfer Time". The former determines the characteristics of the RTC Disk Sub-System and the later is part of the Total RTC latency calculation.

1. This is the time for a single camera to transfer the camera data to the Frame Grabber. A time of 50 µsec has been allocated which is consistent with a transfer using Camera Link full configuration. Also shown is the total data rate needed to save the camera data for all 7 cameras through the Diagnostic/Telemetry port.
2. If the frame grabber is a separate piece of hardware from the rest of the WFS, this is the time to transfer the camera data from it to the next stage of the WFS. Since camera data has already been saved if desired, there is no load indicated for the Diagnostic/Telemetry port.

3. If it is desired to save the centroids, this number is the amount of time to transfer the data over the Diagnostic/Telemetry port. No additional operational transfer load is required.
4. After the WFS has calculated the wave front, these numbers are the amount of time needed to transfer it to the tomography engine and the Diagnostic/Telemetry Port.
5. After the Tomography Engine has estimated the atmospheric volume, the data needs to be transferred to the DM Command Generator. If it is also desired to save the tomographic layer information, the amount of time to transfer the data over the Diagnostic/Telemetry port is also given.
6. The DM information from the Tomography Engine is in a spatial measure and the DM Command Generator generates the correct DM actuator voltages to best match the desired DM shape. These numbers are the amount of time it takes to transfer the correct shape to the DM. Additionally, the data rate for the Diagnostic/Telemetry Port is shown if the data is to be saved.

3.4.2 Non-Transfer Latencies

Non-Transfer Latencies	(µSec)	Type
Camera Stare	500	Fixed
WFS	300	Compute
Tomography	450	Compute
DM Cmd Gen	200	Compute
DM Hold	500	Fixed
Total Fixed Time	**500**	
Total Compute Time	**950**	

Table 6 Non-Transfer Latencies

3.4.3 Total Latency for Tomography Operations

See Figure 8 for a representative timing diagram.

Total Tomographic Latency Calculations	
Total Computation Time	950
Total Fixed Time	500
Total Op. Xfer Time	665
Total Tomographic Latency	**2,115**

Table 7 Total Latency Calculations

3.4.4 Total Latency for NGS Operations

Total NGS Latency Calculations	
Total Computation Time	500
Total Fixed Time	500
Total Op. Xfer Time	658
Total NGS Latency	**1,658**

Table 8 Total NGS Latency Calculations

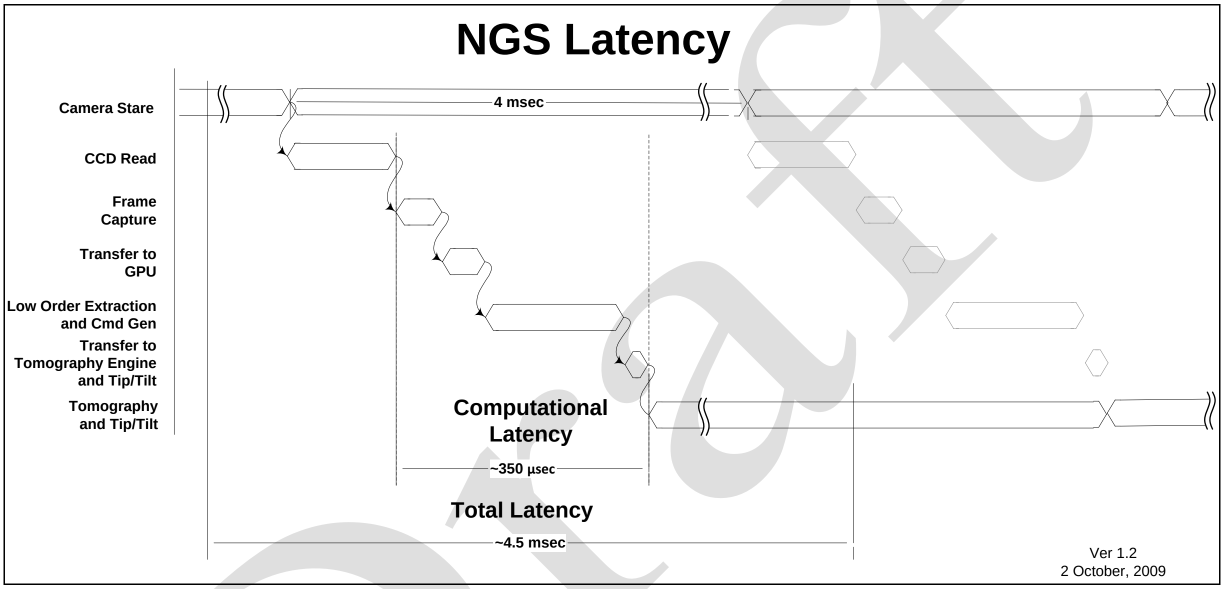


Figure 9 Natural Guide Star Latency

Preliminary Timing

3.5 *Telemetry and Diagnostics Data Rates*

In addition to the latency associated with processing the AO data and controlling the actuators, the NGAO system needs to be able to save key system telemetry data. This feature allows subsequent processing of the science data, atmospheric and system analysis, and the display real time diagnostics information.

The amount of data it is possible to save is huge (many Giga Bytes per second) and a very large, fast disk sub-system is needed to capture this data prior to analysis.

4. RTC Control Processor (CP)

The RTC Control Processor (CP) provides an asynchronous interface between the RTC system the AO Control.

It is built on a normal Linux box.

We recognize that the RTC system is a key component to the overall AO system and the rest of the telescope environment and the CP interface will adhere to the component architecture defined in [_____________].

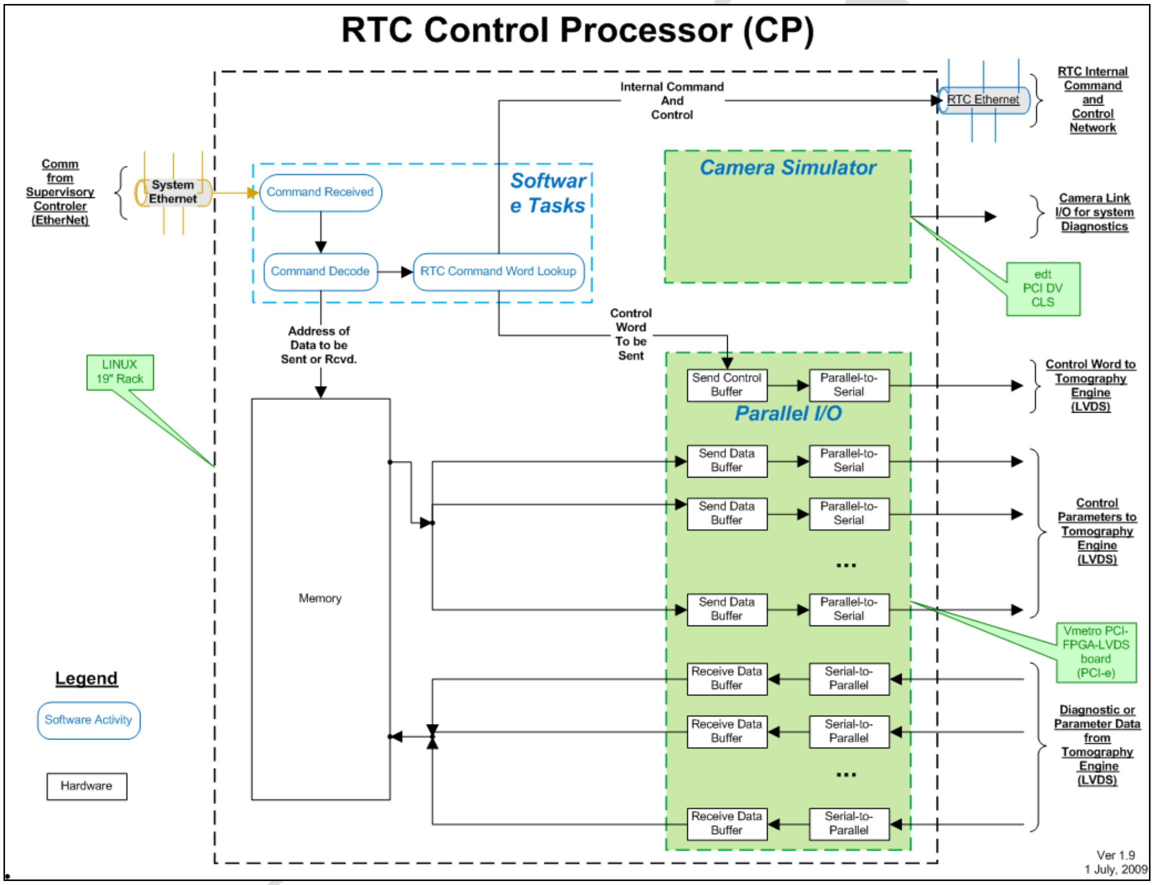


Figure 10 RTC Control Processor (CP)

4.1 *AO Control System Feature Support*

See the AO Control Interface Document (ref_______) for details on these features.

4.1.1 Commands

The CP accepts commands from the system AO Control (or alternatively from a telnet session) over sockets. Data is passed as part of the command or as a system address from which the CP will fetch the data.

See [ref CP interface spec]

4.1.2 Monitoring

4.1.3 Events

Events within the RTC are generated by the individual sub-systems and monitored by the CP. The CP in turn generates events for the AO Control as required by the AO Control.

4.1.4 Alarms

4.1.5 Log Messages

4.1.6 Configuration

Low-level configuration is handled by the CP under control of the AO Control

4.1.7 Archiving

AO data is saved short term in the RTC disk sub-system. Long term archiving is provided by the telescope archiving systems.

4.1.8 Computational Load

Boxcar averaging of camera frame or other data will be provided prior to transfer to the AO Control.

This may require some added hardware, on the order of an additional processor. This is not significant and an analysis is ongoing to determine the exact implementation and hardware/software impact.

4.2 *Interfaces and Protocols*

4.2.1 Interface to the AO Control

The interface is over Ethernet using sockets and the details of this interface are covered in a separate document [ref CP interface spec].

The compute load of the CP is expected to be relatively low. Therefore, it is possible that the AO Control functions and the CP functions could be integrated into a single hardware platform if desired.

4.2.2 Interface to the rest of the RTC

The CP's interface to the rest of the RTC is through Gbit Ethernet, LVDS, or Camera Link busses as appropriate.

4.3 *Data Flow and Rates*

4.3.1 To AO Control

4.3.2 To the Disk Sub-system

4.3.3 To the rest of the RTC

5. Cameras

5.1 *HOWFS (Tomography and Point-and-Shoot)*

5.1.1 Interfaces and Protocols

The interface between the HOWFS camera sub-systems and the RTC will be through Camera Link™ using the Camera Link™ Full configuration.

5.1.2 Data Flow, Rates and Latency

The following are the specifications the camera systems must meet in order for the RTC to meet the required error budget.

5.1.2.1 Camera Frame Rate

The HOWFS camera must be able to sustain a 2 KHz frame rate.

5.1.2.2 CCD Read Time

After each frame is exposed, the CCD transfers the frame data to holding registers and starts exposing the next frame. The CCD must be able to transfer this data to the camera controller at a 2 KHz rate without affecting the simultaneous science object acquisition.

5.1.2.3 Camera Transfer Latency

The data latency between the last byte transferred from the CCD and the last byte sent to the frame grabber must be less than 50 µsec. See: **Table 4**.

5.2 *LOWFS (IR)*

5.2.1 Interfaces and Protocols

The interface between the camera sub-systems and the RTC will be through Camera Link™ using the Camera Link™ Full configuration.

5.2.2 Data Flow, Rates and Latency

The following are the specifications the camera systems must meet in order for the RTC to meet the required error budget.

5.2.2.1 Camera Frame Rate

The LOWFS IR cameras must be able to sustain a 250 Hz frame rate [REF].

5.2.2.2 CCD Read Time

After each frame is exposed, the CCD transfers the frame data to holding registers and starts exposing the next frame. The CCD must be able to transfer this data to the camera controller at a ___ rate without affecting the simultaneous science object acquisition.

5.2.2.3 Camera Transfer Latency

The data latency between the last byte transferred from the CCD and the last byte sent to the frame grabber must be less than 50 µsec. See: **Table 5**.

5.3 *Master Clock Generation for Camera Synchronization*

Cameras will be able to synchronized to an external Frame Clock or be able to generate a Frame Clock to which other cameras can be synchronized. Ideally, one camera will be designated as the master RTC camera. All other cameras will be synchronized to this camera.

5.3.1 Inter Camera Capture Jitter

The time between the synchronization signal and the start of the camera exposure will be a Maximum of _____ µsec for cameras of the same frame rate.

5.4 *Control*

The AO Control will control all camera parameters through facilities provided by the CP. The following controls will be supported be each camera:

5.4.1 Gain

5.4.2 Frame rate

The frame rate is the frequency at which new frames are captured.

Whatever rates are required by the system, i.e. 2 KHz, 1 KHz, 500 Hz, 100 Hz.

5.4.3 Stare Time

Whatever times are required by the system. Ref (________)

5.4.4 Frame Transfer rate

The frame transfer rate is the speed of the link between the camera and the frame capture card.

6. Wave Front Sensors (WFSs)

We intend for all WFSs to have a common architecture and implementation. This brings the advantage of lower spares cost, easier trouble shooting and development, and faster debug time during operation.

6.1 *WFS Interfaces*

The interfaces to the WFSs are shown in the following figures: LOWFS in **Figure 11**, Point-and-Shoot HOWFS in **Figure 15**, and Tomography HOWFS in **Figure 13**. Input will be via the machine-vision industry's standard Camera Link cabling and communications protocol using the full configuration, unless otherwise noted. Output will connect to the tomography engine also via Camera Link communication. The 24 bits data width in this standard can represent two camera pixels each with a dynamic range of up to 2^12 = 4096 counts per pixel, which is above the maximum counts anticipated in NGAO wavefront sensing. The computed result from the WFSP does not increase the measurement limited signal-to-noise ratio so this 24 bit word width is also sufficient for transfer to the tomography engine, even though the tomography engine will ultimately be using a 36 bit internal word width to maintain computational accuracy.

The Camera Link interface cable is serialized low-voltage differential signaling (LVDS) pairs on an engineered standard cable (e.g. 3M's Mini D Ribbon). The base configuration allows a throughput of up to 2.04 Gbit/s. For the 256x256 WFS chip running at 2 kHz frame rate, and 12 bits per pixel, the pixel data rate is 1.573 Gbit/s, which is under the base configuration limit and so does not demand an enhanced bandwidth configuration. Since the bit rate of Camera Link is generally faster than the processor clock (in the case of FPGAs), standard serializer/deserializer (SERDES) transceivers are needed to terminate the connections on each board.

Wavefront sensor processor cards are assigned one per wavefront sensor, while the tomography engine processors are mapped over the aperture, it is necessary for the four tomography WFSPs (each containing a full aperture's worth of data) to distribute their results over the tomography engine processors. The distribution is accomplished using a specialized interface located along the left side of the tomography engine array, as shown in **Figure 17**. This interface makes sure the correct data are shifted in along each row of tomography processors. The cable connection from the WFSPs to the tomography engine uses the same LVDS / Camera Link 2 Gbit/s standard as used for the input to the WFSPs, only this time there is far less data to transfer. The full-frame transfer of 64x64x2x12bit numbers (an extra factor two for complex Fourier coefficients) is accomplished in 50 microseconds, about 10% of a 2 KHz frame cycle. A doublewide Camera Link interface can be used if this is deemed to take too much of the latency budget.

6.2 *Data Flow and Rates*

6.3 *LOWFSs (Low Order Wave Front Sensors) (for T/T, Astigmatism and Focus)*

The LOWFSs take inputs from the 3 IR Tip/Tilt/focus and astigmatism cameras. The cameras feeding the LOWFSs are focused on natural guide stars (NGS). They generate low order control for the on-axis correction of the light for both science and AO paths.

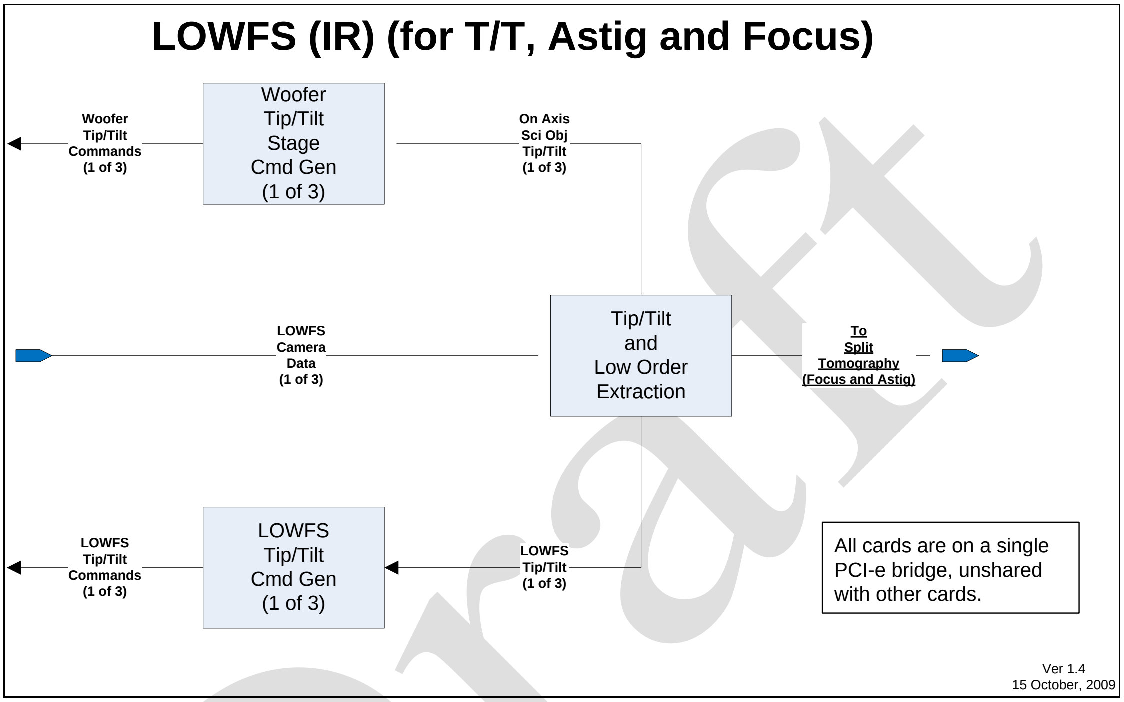


Figure 11 LOWFS

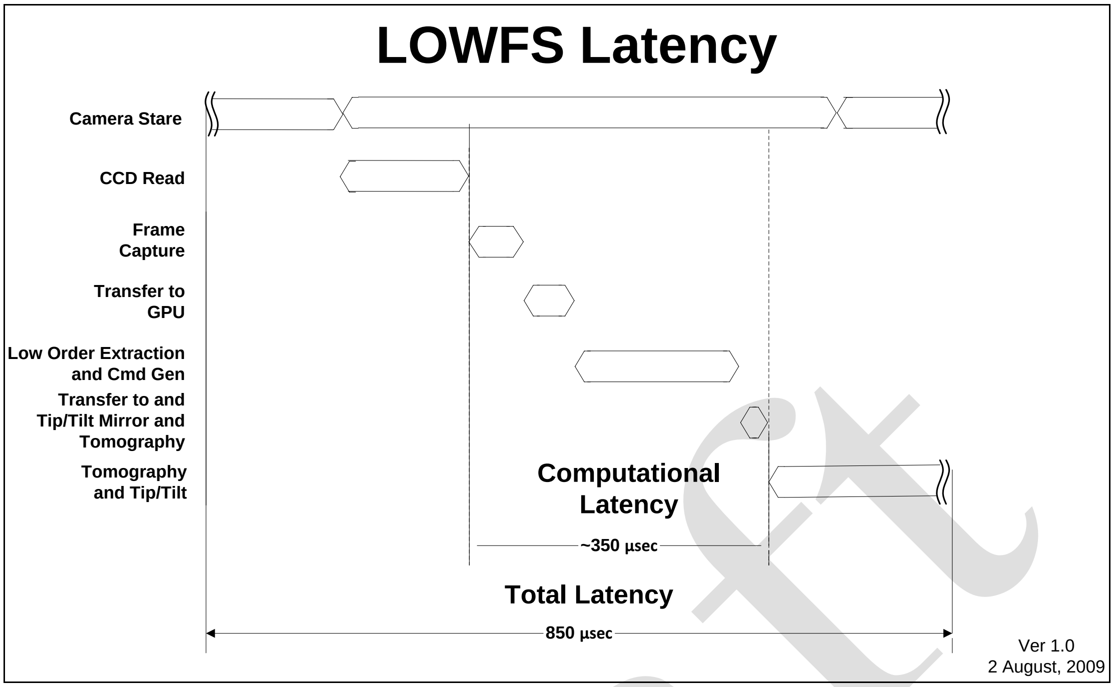


Figure 12 LOWFS Latency (not to scale)

Preliminary Timing

6.3.1 Interfaces and Protocols

6.3.1.1 Inputs

The Inputs are from 3 IR cameras over Camera Link base configuration at 250 Hz frame rate:

- Two tip/tilt cameras (Size?)
- One focus/astigmatism camera (Size?)

6.3.1.2 Outputs

The outputs are:

- An analog X/Y controls to the on-axis Tip/Tilt stage of the Woofer at a 250 Hz frame rate.
- 3 analog X/Y controls to the tip/tilt mirrors associated with the 3 LOWFS cameras at a 250 Hz frame rate.
- Digital Tip/tilt, focus, and astigmatism information sent to the tomography engine at a 2 KHz frame rate to synchronize with the Tomography's frame rate.

6.3.2 Data Flow, Rates and Latencies

6.3.2.1 Camera Data

Each LOWFS camera has an image size of ______ and a data width of ______. This yields a data rate of ______ B/frame and a total rate of ______ B/sec. This data is sent to a frame grabber in the LOWFS and alternatively to the Telemetry/Diagnostic signal path.

The LOWFS IR camera has a stare time of 4 msec [REF] for a frame rate of 250 Hz.

6.3.2.2 Tip/Tilt Signals

The tip/tilt signals are analog voltages < ±10 V, with a rise/fall time of < 10 µsec, and a frame rate of 2 KHz.

6.3.2.3 Astigmatism and Focus

The astigmatism and Focus data are sent to the tomography engine as digital data over 1 LVDS port.

6.3.3 Low Order Reconstruction Engine

See the ______ document for the details on the algorithms for extracting tip/Tilt, focus, and astigmatism from the camera data.

6.4 *HOWFSs (High Order Wave Front Sensors)*

Seven HOWFSs generate seven wave fronts used by the AO RTC system.

- Four tomography HOWFSs generate wavefronts used by the tomography engine that in turn generates on axis wavefronts used by the woofer and high order DM(s) to correct for atmospheric aberration affecting the science object(s). Currently there is only one science object, but we have the possibility of supporting more if needed.
- Each of three point-and-shoot HOWFSs compensate for atmospheric aberrations affecting a corresponding LOWFS NGS used for T, astigmatism, and focus.

To generate these wave fronts, the HOWFSs take inputs from the tomography or point-and-shoot cameras. These cameras are focused on laser guide stars (LGS).

The images are first corrected for dark current, thresholded, and then processed to produce centroids corrected for reference centroids and offsets.

Tip/Tilt is then extracted from the centroids and subtracted from them.

The Tip/Tilt removed centroids are then processed to produce a corresponding tip/tilt removed wavefront.

6.4.1 Centroider and Wave Front Reconstruction Engine

6.4.1.1 CPU Implementation

This would be a conventional implementation using vector-matrix multiplication for the reconstruction. It is the current implementation plan for the HOWFS reconstructors and uses standard off the shelf components.

6.4.1.2 GPU implementation

A 64 subaperture across (88 across for zero padding) reconstructor has been implanted on a single Nvidia Graphics Processor card. This presents a simple solution for the HOWFSs that could be built with standard off the shelf components. It would potentially provide lower latency than the CPU implementation and fewer racks of equipment since multiple HOWFSs could be supported by a single computer.

The GPU would reside on a PCI-E backplane slot on a CPU motherboard. The CPU can be used to configure and run the GPU. Additional PCI-E slots would house Camera Link interface cards, which in turn connect to the WFS camera for input and to the tomography engine for output.

We are currently analyzing whether we can utilize the GPU in the system at a 2 KHz frame rate under a real-time OS.

6.4.1.3 FPGA implementation

Using FPGAs, it is possible to implement a centroider and reconstructor in a small number of chips that implement a variety of algorithms. These would be custom boards but have the potential for the lowest latency implementation.

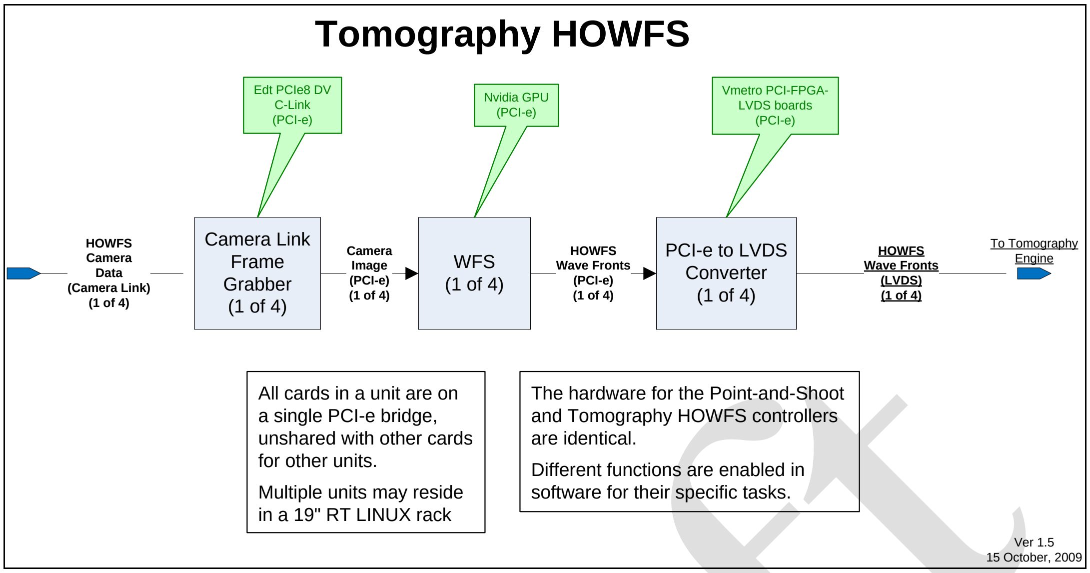
The Tip/Tilt extracted by each HOWFS is used to control the tip/tilt mirror associated with that particular HOWFS camera.

There is a small difference between a Tomography HOWFS and a Point-and-Shoot HOWFS.

6.4.2 Tomography HOWFS

The Tomography HOWFS sends the reconstructed wave front to the Tomography Engine. See **Figure 13**.

The latency contributed to the total RTC latency in **Table 6** is the computational Latency shown in **Figure 14**.



All cards in a unit are on a single PCI-e bridge, unshared with other cards for other units.

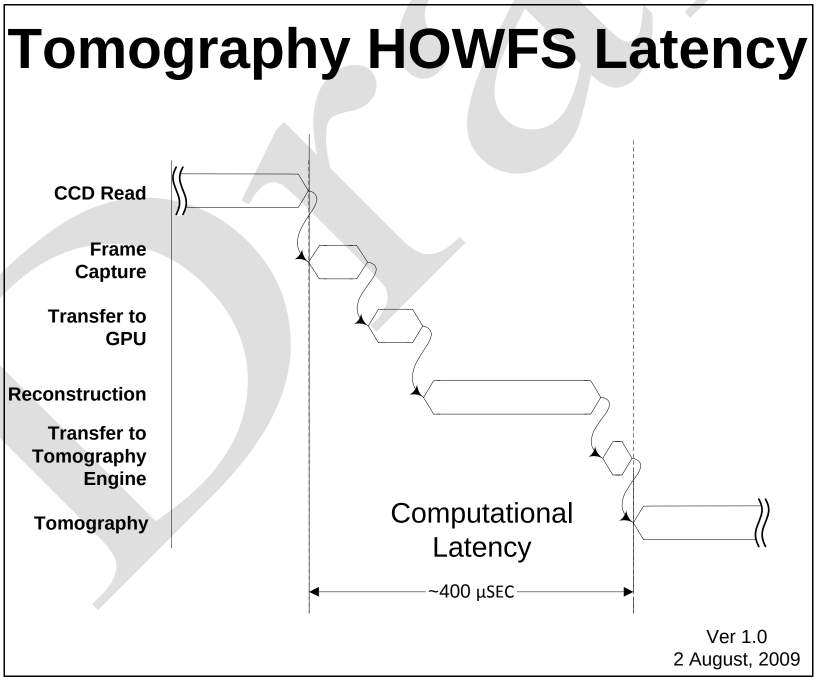
Multiple units may reside in a 19" RT LINUX rack

The hardware for the Point-and-Shoot and Tomography HOWFS controllers are identical.

Different functions are enabled in software for their specific tasks.

Ver 1.5
15 October, 2009

Figure 13 Tomography HOWFS



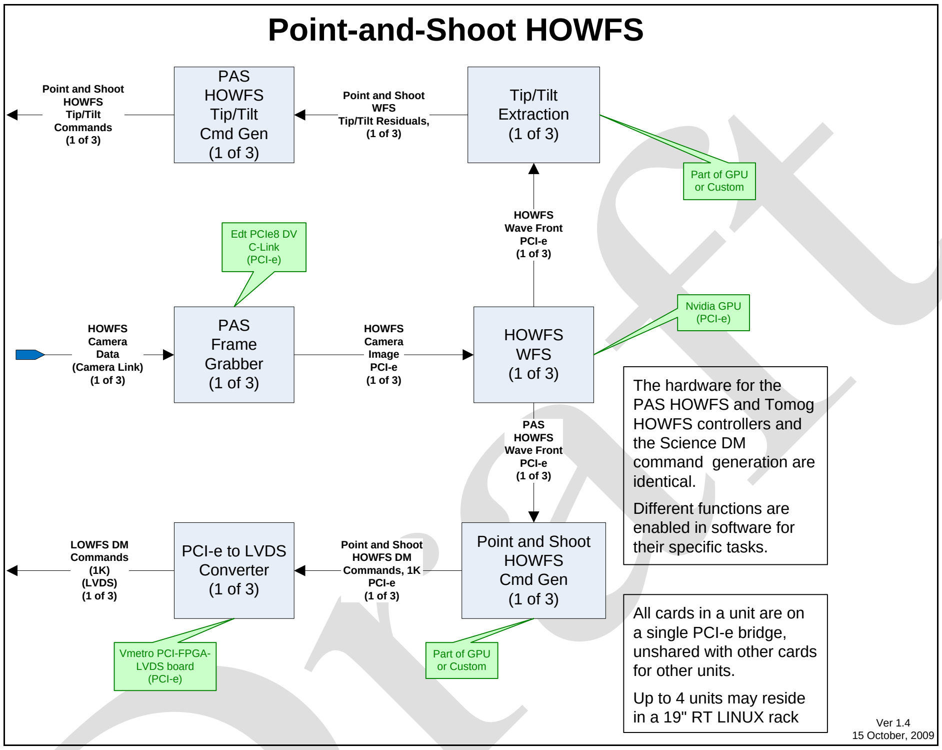
Ver 1.0
2 August, 2009

Figure 14 Tomography WFS Latency (not to scale)

Preliminary Timing

6.4.3 Point-and-Shoot HOWFS

The Point-and-Shoot HOWFS, however, further processes the reconstructed wave front to generate DM commands which are sent to a DM that corrects the laser spot associated with its corresponding camera, see **Figure 15** and Section 8. **Figure 15** illustrates the Point-and-Shoot HOWFS latency.



The hardware for the PAS HOWFS and Tomog HOWFS controllers and the Science DM command generation are identical.

Different functions are enabled in software for their specific tasks.

All cards in a unit are on a single PCI-e bridge, unshared with other cards for other units.

Up to 4 units may reside in a 19" RT LINUX rack

Figure 15 Point-and-Shoot HOWFS

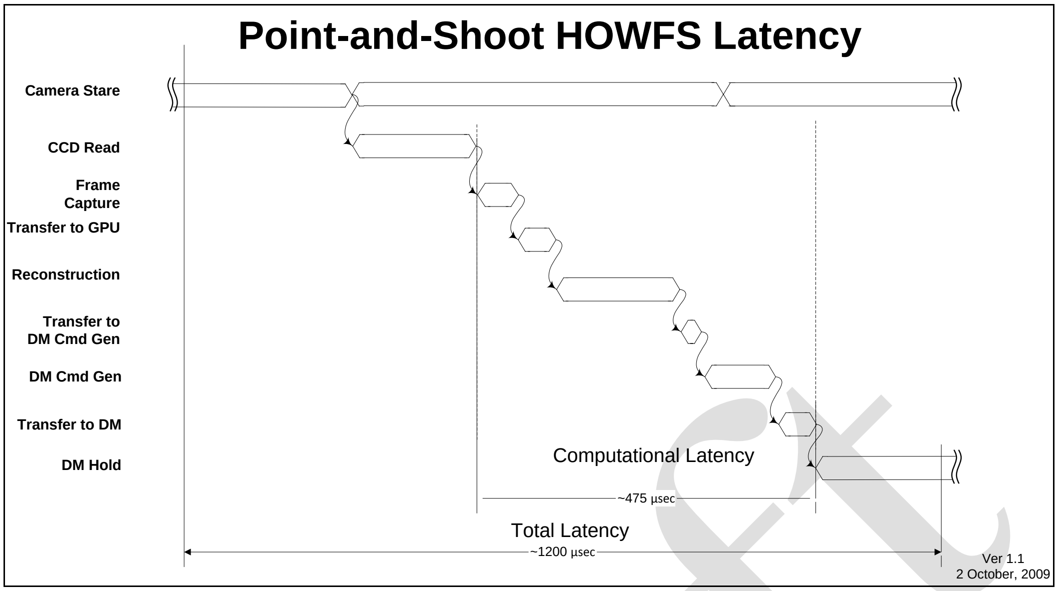


Figure 16 Point-and-Shoot HOWFS Latency (not to scale)

Preliminary Timing

7. Tomography Engine

The tomography engine is implemented as a systolic array mapped to the physical volume of the atmosphere above the telescope. This is not your typical tower or rack-mounted PC. It is specifically designed to implement very high-speed vector/matrix operations with very small latency requirements. The total throughput rate is in the terra operation per second range with latencies of less than a microsecond.

7.1 *Tomography Error Requirements*

Don, do we need to specify the Tomography error requirements separately?

7.2 *Algorithm*

The tomography algorithm is an iterative solver, using Arithmetic Reconstruction Tomography (ART) [1]. For details of the algorithm see ___________________.

7.3 *Design*

The design ________

7.4 *Architecture*

We have modeled the algorithm implementation after the tomography problem itself. We divide the atmosphere into regions called voxels. We use an iterative algorithm to calculate the influence of each voxel. These calculations are highly parallel, very simple, have minimum memory requirements, and minimum communication requirements with other voxels. Our implementation follows this model. We assign a very simple processor called a processing element (PE) to a small number of voxels. We place a large number of these processors on a field programmable gate array (FPGA). Each FPGA contains all of the voxels for one column of the atmosphere above the telescope. Multiple FPGAs, then, are used to hold all of the voxels for the space above the telescope.

This implementation maintains the visibility of all aspects of the original problem through the various phases of the solution.

The Tomography Engine PE is a wide word SIMD architecture.

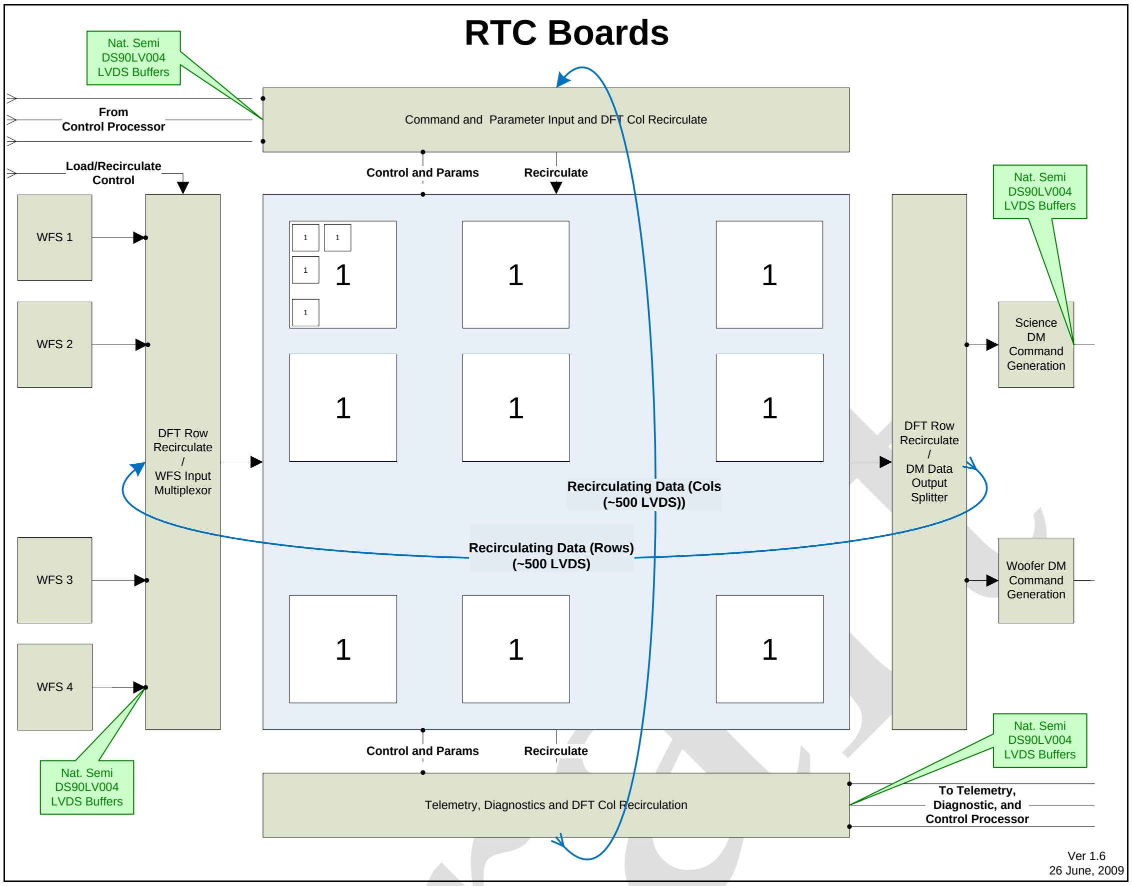


Figure 17 RTC Boards

7.4.1 Systolic Arrays

7.4.2 Voxels

7.4.3 Processing Elements (PEs) and their Functions

One PE can process data for multiple sub apertures. This allows us to trade off processing speed for system size.

7.4.4 Interfaces and Protocols

7.4.5 Data Flow and Rates

7.4.6 Timing and Events

7.4.7 Control and Orchestration of Processor States

7.4.7.1 On-Line Parameter Loading

7.4.7.2 Off-Line Parameter Loading

7.5 *Design*

7.5.1 Physical Architecture of the Tomography Engine

7.5.2 FPGA Design

Xilinx Virtex 5 or 6

Higher logic density may lead to timing and pin-out issues

Higher density logic may lead to more susceptibility to SEU’s

7.5.2.1 Basic Cell Layout

7.5.2.2 FPGA Configuration

7.5.2.3 System Monitor

The System Monitor monitors the core temperatures and voltages of each FPGA. They are linked serially, so that the health of each FPGA in the tomography processor can be monitored independent of the operation state of the system.

7.5.2.4 Clocks

7.5.2.5 Data

7.5.2.6 System Signaling

Diagnostics and Telemetry

7.5.2.7 Position Detection

7.5.2.8 Power

7.5.2.9 Power Supply Bypassing

7.5.2.10 Heat Removal

7.5.3 Board Design

7.5.3.1 Stack-up

7.5.3.2 FPGA Configuration

7.5.3.3 Power Regulators

Each board will have its own power regulator for the critical core voltages: _______

These voltages must be < ±___ for all FPGA’s on the board.

7.5.3.4 Power Supply Bypassing

7.5.3.5 Voltage Monitors

7.5.3.6 Temperature Monitor

7.5.3.7 Signal Routing

7.5.3.8 Connectors

Power

Clock

Data

7.5.3.9 System Signaling

Diagnostics and Telemetry

7.5.3.10 Position Detection

7.5.3.11 Clock Issues

7.5.3.12 Heat Removal

7.5.4 System (Multi-Board)

7.5.4.1 Board to Board

7.5.4.2 Data

7.5.4.3 Clock

7.5.4.4 System Signaling

Diagnostics and Telemetry

7.5.5 System Facilities

7.5.5.1 Power Supplies

All power supplies are to be digitally programmable and monitored.

Number and size

Do they need to be redundant?

7.5.5.2 Clocks

7.5.5.3 Communication to and from the CP

7.5.5.4 Communications to the Telemetry Stream

7.5.5.5 Communications to the Diagnostic Stream

7.5.6 Mechanical Design

7.5.6.1 Ease of use

7.5.6.2 Cooling

7.5.6.3 Cabling

7.5.6.4 Power

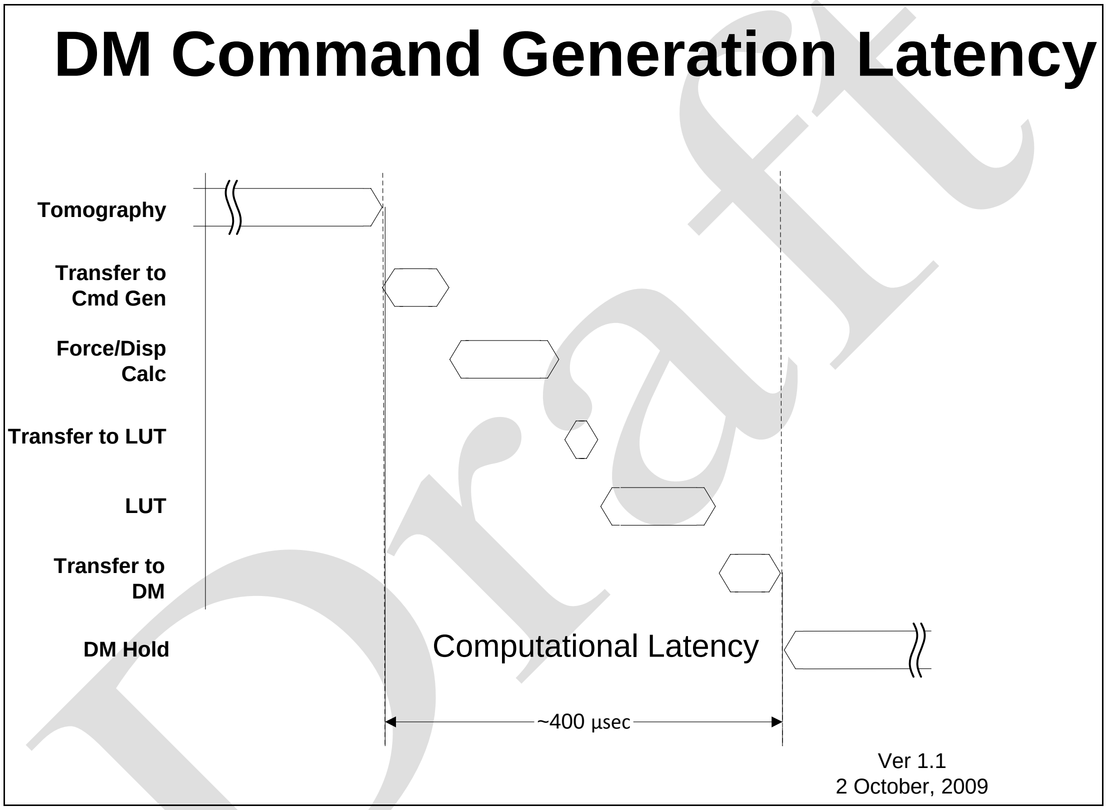
7.5.6.5 Data In and Out

7.5.6.6 Power Control (Pulizzi)

8. DM and T/T Command Generation

The DM and T/T command generator sub-systems convert wavefront information into mirror voltage commands to result in the desired wavefront.

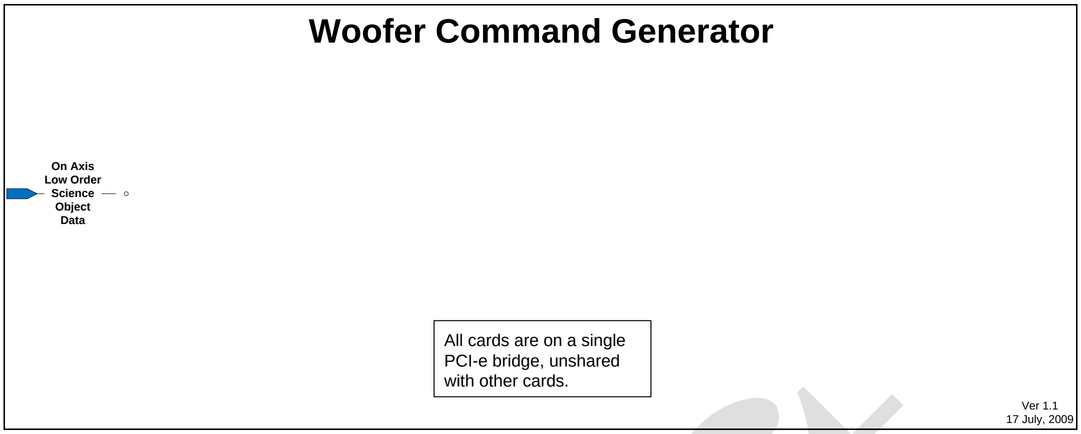
8.1 *DM Control Engine*



Ver 1.1
2 October, 2009

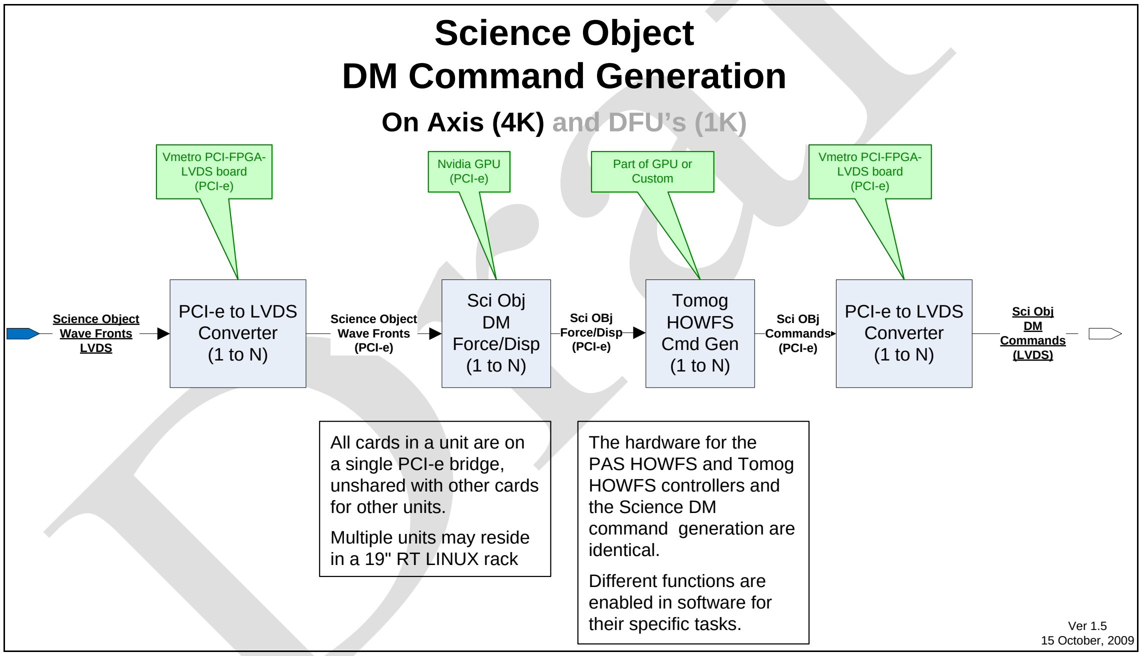
Figure 18 DM Command Generation Latency (not to scale)

Preliminary Timing



All cards are on a single PCI-e bridge, unshared with other cards.

Figure 19 Woofer Command Generation



All cards in a unit are on a single PCI-e bridge, unshared with other cards for other units.

Multiple units may reside in a 19" RT LINUX rack

The hardware for the PAS HOWFS and Tomog HOWFS controllers and the Science DM command generation are identical.

Different functions are enabled in software for their specific tasks.

Figure 20 Science Object DM Command Generation

Add HOWFS Command Gen Diagrams

8.2 *Low Order (Woofer) (Closed Loop Operation)*

The low order DM (woofer) operates closed loop.

8.2.1 Interfaces and Protocols

8.2.2 Data Flow and Rates

8.3 *Open Loop Operations*

The high order science DM and the HOWFS DMs all operate open loop.

8.3.1 High Order (Science and HOWFS) Open Loop Operations

8.3.1.1 Algorithm Description (Brief)

Ref(_____)

8.3.1.2 Non Linear Look-Up Tables

8.3.1.3 Interfaces and Protocols

8.3.1.4 Data Flow and Rates

8.4 *Tip/Tilt Mirrors*

The RTC controls 10 tip/tilt mirrors (one for each WFS) and 1 on-axis tip/tilt actuator on the Woofer:

- 3 T/T mirrors associated with the 3 LOWFS cameras
- 7 T/T mirrors associated with the 7 HOWFS cameras

- 1 T/T actuator on the Woofer

8.4.1 Tip/Tilt Control Engine

8.4.2 Interfaces and Protocols

8.4.3 Data Flow and Rates

9. Telemetry and Diagnostics Data Streams

All major data signals can be sent to the Telemetry/Diagnostic data path at the request of the AO Control. Any data in this path can be sent to either-or-both the AO Control, for diagnostics, or the RTC disk sub-system, for storage. (See: Section 8)

These signals include:

1. Raw Camera Data for 4 tomography WFSs, 3 point-and-shoot WFSs, 2 T/T, and 1 TT/focus/astigmatism sensor cameras
2. Centroids all HOWFS
3. Reconstructed Wave Fronts supplied as input to the Tomography engine from the Tomography HOWFS.
4. Tomography Layers
5. Science on-axis High Order Wave Front produced by the Tomography engine after forward propagating from the science object through the tomographic estimate of the atmosphere. This is the open loop control wave front for the High Order DM.
6. Science on-axis Woofer Wave Front produced by the Tomography engine after forward propagating from the science object through the tomographic estimate of the atmosphere. This is the closed loop control wave front for the Low Order DM.
7. Science on-axis High Order Commands
8. Science on-axis Woofer Commands
9. All RTC Current Parameter Settings including the settings of the following: shutters, laser fiber In/Out actuators, laser On/Off state and intensity, filter wheel position, interlocks.

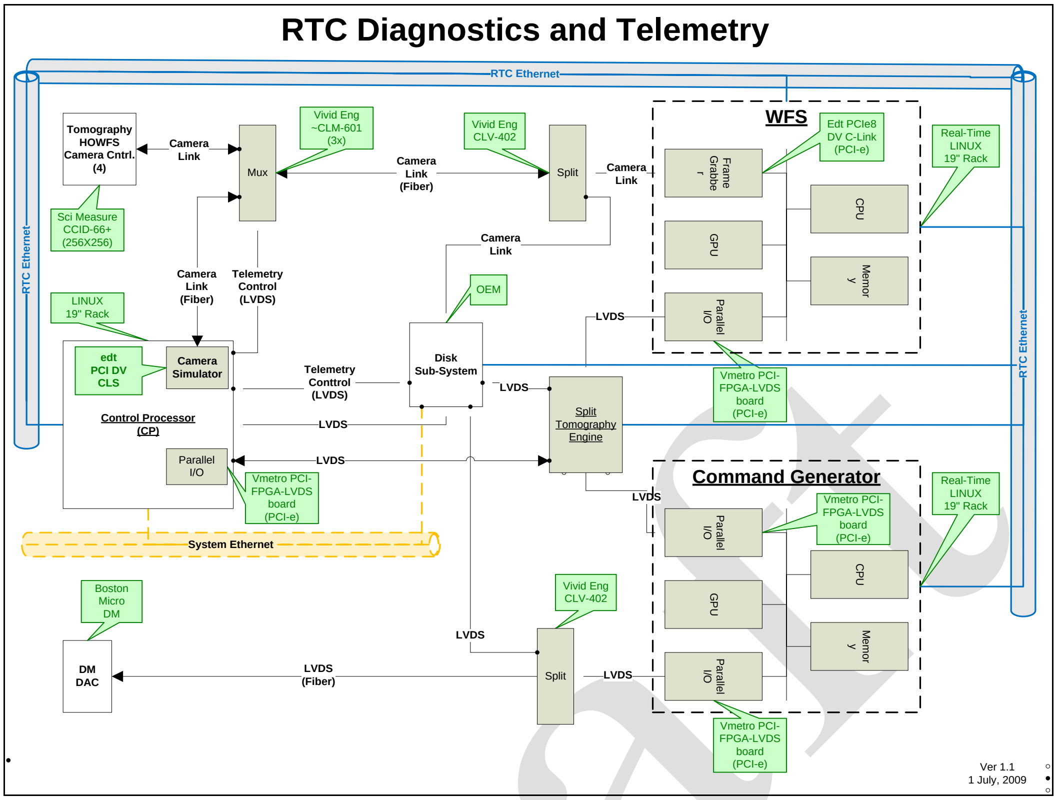


Figure 21 RTC Diagnostics and Telemetry

Diagnostic data is captured at rates of approximately one sample per second, as determined by the AO Control.

A separate frame grabber associated with the disk sub-system captures camera data for diagnostics and telemetry. This data is a parallel stream supplied by a Camera Link splitter, shown in **Figure 21 RTC Diagnostics and Telemetry**.

9.1 *Interfaces and Protocols*

9.2 *Data Flow and Rates*

9.3 *Timing and Events*

9.4 *Synchronization of AO Data Time Stamps with the Science Image Time Stamps*

The RTC system will have a sub millisecond accurate clock, GPS based to time stamp the AO data for telemetry and diagnostics and provide to the SRT.

- How will we use atmospheric data collected during the capture of science images for post processing (e.g. PSF)?
- To what accuracy must the time stamps on the science image and the AO data match?
- How to determine what time slice of the atmospheric data corresponds to a science image?
- Currently, science image shutter open and close times are time stamped to a precision nominally of a small fraction of a second but may be as much as ±30 seconds in error (see Appendix A, Time-related keywords in Keck Observatory FITS files).
- To what do we reference the RTC data and how?
- How to relate the AO data time stamps to the times in the science data at which the shutter was actually opened and closed?

9.5 *An Important Note on the Telemetry Rates*

Since the total storage rate can exceed 36GB/sec, if all possible data sets are captured, care should be taken to insure there is enough space remaining in the disk sub-system to save the data. The disk sub-system can support multiple TB of data, but after a number of nights of heavy use, a great deal of the storage may be used.

The disk sub-system is not meant to archive data. Its purpose is merely to store it temporarily until the desired data can be archived to another location and the unneeded data can be deleted.

9.6 *Handling System Diagnostic Functions*

The RTC does not need to know that a wave front was generated by a star or a laser diode (with or without simulated turbulence). Its job is the same; calculate the wave front and tomography. No special features are in place for system diagnostics beyond the normal diagnostics described here.

10. RTC Disk Sub-System

All major data signals can be sent to the Telemetry/Diagnostic data path at the request of the AO Control. Any data in this path can be sent to either-or-both the AO Control, for diagnostics, or the RTC disk sub-system, for storage.

Since the total storage rate can exceed 36GB/sec if all possible data sets are captured, care should be taken to insure there is enough space remaining in the disk sub-system to save the data. The disk sub-system can support multiple TB of data, but after a number of nights of heavy use, a great deal of the storage may be used.

The disk sub-system is not meant to archive data, merely to store it temporarily until the desired data can be archived to another location and the undesired data can be deleted.

10.1 *Interfaces and Protocols*

10.1.1 Interface to the RTC

10.1.2 Interface to the System AO Control

10.2 *Data Flow and Rates*

10.2.1 RTC Telemetry Port

10.2.2 System Network

10.3 *Storage Capacity*

11. Timing Generation and Control

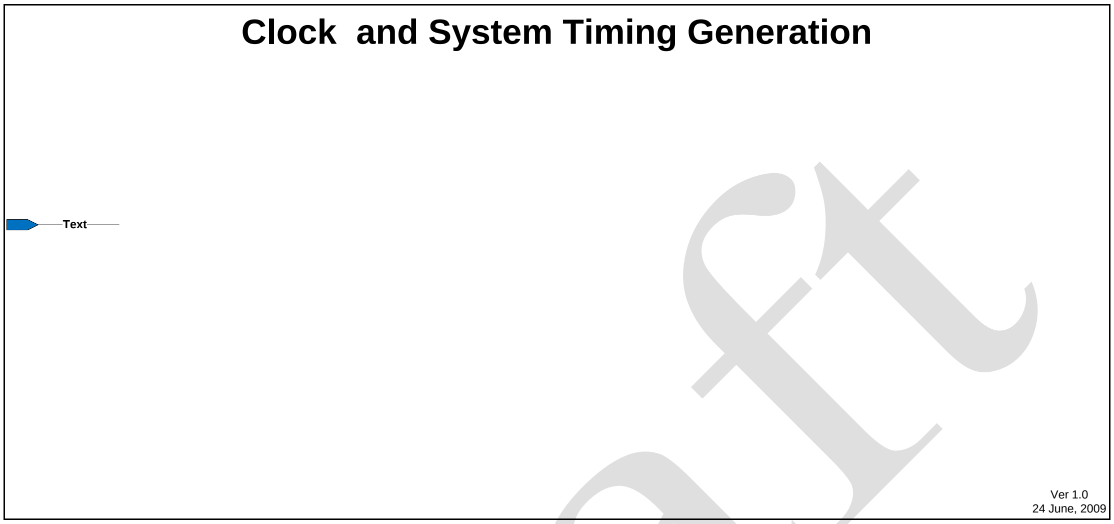


Figure 22 Camera Synch and System Timing Generation

11.1 *Camera Synchronization and Timing*

11.2 *GPS Sub millisecond Time Stamp Source*

11.3 *Pipeline coordination*

11.4 *Global Synch*

11.4.1 Global Reset

11.5 ***Tomography System Clock***

11.5.1 Tomography Synch

11.5.2 Tomography Clock (100 MHz)

11.5.3 Tomography LVDS Clock (600 MHz)

12. RTC Test Bench

RTC Test Bench System

Test Bench System

The Test Chassis is designed to perform bench tests on RTC major or sub units.

Two permanent benches will be maintained: one at Keck and one at Santa Cruz.

Their purpose is to provide rapid and accurate acceptance, functional, development, diagnostic and debug test of all major RTC components

It can plug into the LVDS or Camera Link output from a Camera, a WFS, a tomography engine card, or a DM Command Generator and capture and analyze their outputs.

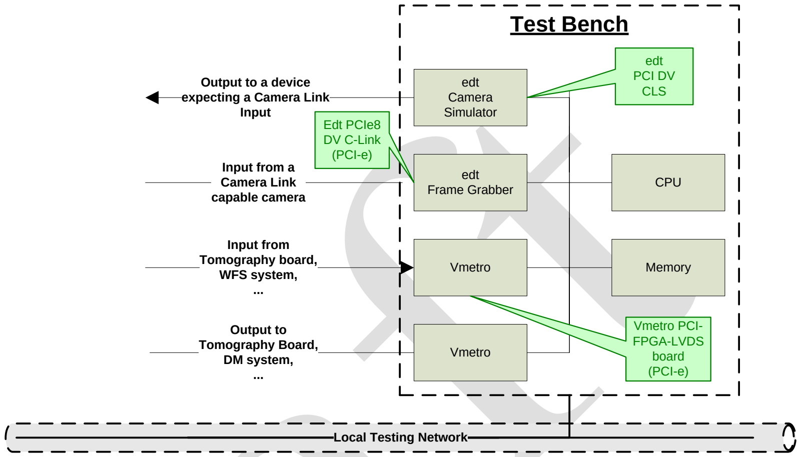
It can supply simulated input information to WFS, tomography engine cards or DM Command Generators and simultaneously capture their outputs to validate unit functionality.

It is meant as a bench tool in the lab rather than as an operational piece of the RTC.

It has all facilities necessary to provide the correct control to the devices through either LVDS, camera link or Ethernet.

Two types of tests will be provided: 1. push button (Go/No Go) confidence tests to validate the basic functionality of the unit; 2. manually run tests to de detailed analysis of the functionality or performance of the unit.

To test the Tomography Engine boards, a special jig will be supplied that includes power supplies and cables, clock generators and speed selection, monitoring and ***cooling***.



Ver 1.3
18 July, 2009

Figure 23 RTC Test Bench

13. General Requirements for System Components (HW and SW)

13.1 *All Components*

13.1.1 Acceptance Testing

All components must have an established acceptance test requirement as part of their individual design package.

13.1.2 Long Term Maintenance Plan

The cost of ownership of a complex system such as the NGAO RTC is significant and may exceed the acquisition cost over time. A clear plan must be in place for any custom hardware or software that will describe how the product will be maintained over time. This must include:

1. Types of personnel required for various levels of maintenance including upgrades, feature additions, or bug fixes; i.e., what tasks would it be expected for astronomers, telescope technicians, software or hardware staff
2. A maintenance agreement must be included by the vendor that covers the delivered product for bug fixes for a minimum of 1 year after acceptance.

13.1.3 Software, Material and Documentation Control

All elements of the system are revision controlled with CVS or a functional equivalent.

13.1.4 Spares

13.1.5 Panel Retention

All panels shall be held in place with self-retaining screws.

13.2 *Custom Components*

Custom components include systems that are built from an aggregate of standard components, but combined into a custom configuration.

13.2.1 Documentation and Training

These products will need a variety of levels of documentation and training targeted at different audiences:

13.2.1.1 Operator Manuals

13.2.1.2 Diagnosis of Problems

13.2.1.2.1 Development of New Features

13.2.2 Diagnostics Capability

Each board must have the ability to have a minimum set of diagnostics performed on it while in the system and operating normally. The following are required for all boards. Requirements for specific board types are specified in their section.

13.2.2.1 Software Module Identification

All software modules must contain a revision number and build date that can be queried during diagnostics to track currently installed module revisions. This applies to each module in any compiled unit and any .DLL or .so, which is a custom piece of code.

Further, any executable must have a command that can be given during diagnostics that will list the information recursively for ***all*** such modules with which it is built or to which it is linked.

13.2.2.2 Board ID, S/N, location reporting

13.2.2.2.1 Board ID

13.2.2.2.2 S/N

13.2.2.2.3 Location

If there are multiple identical board of identical types installed (i.e., Tomography boards or WFS boards) there must be a means of identifying in which position the board sits in the arrangement of identical boards. To insure that it this cannot be incorrectly set, this should be set automatically by its position in the installation.

13.2.2.3 Power supply monitoring

Sampling rate, sample depth, trigger capabilities, threshold capabilities.

13.2.2.4 Clock monitoring

Clock monitors

13.2.3 Trouble shooting capability

Basic trouble shooting procedures will need to be performed occasionally: voltage or waveform viewing with a meter or scope,

If an extender is to be used to perform these operations, the operation of the board in a system during normal operations must be demonstrated using such an extender.

13.2.4 Cable Attachment Verification

Cables and connectors should be designed in such a manner that a device at one end can determine whether the cable is plugged in to the receptacle at the other end.

Circuitry will be included in each board for each cable to perform this analysis and report its result during diagnostics.

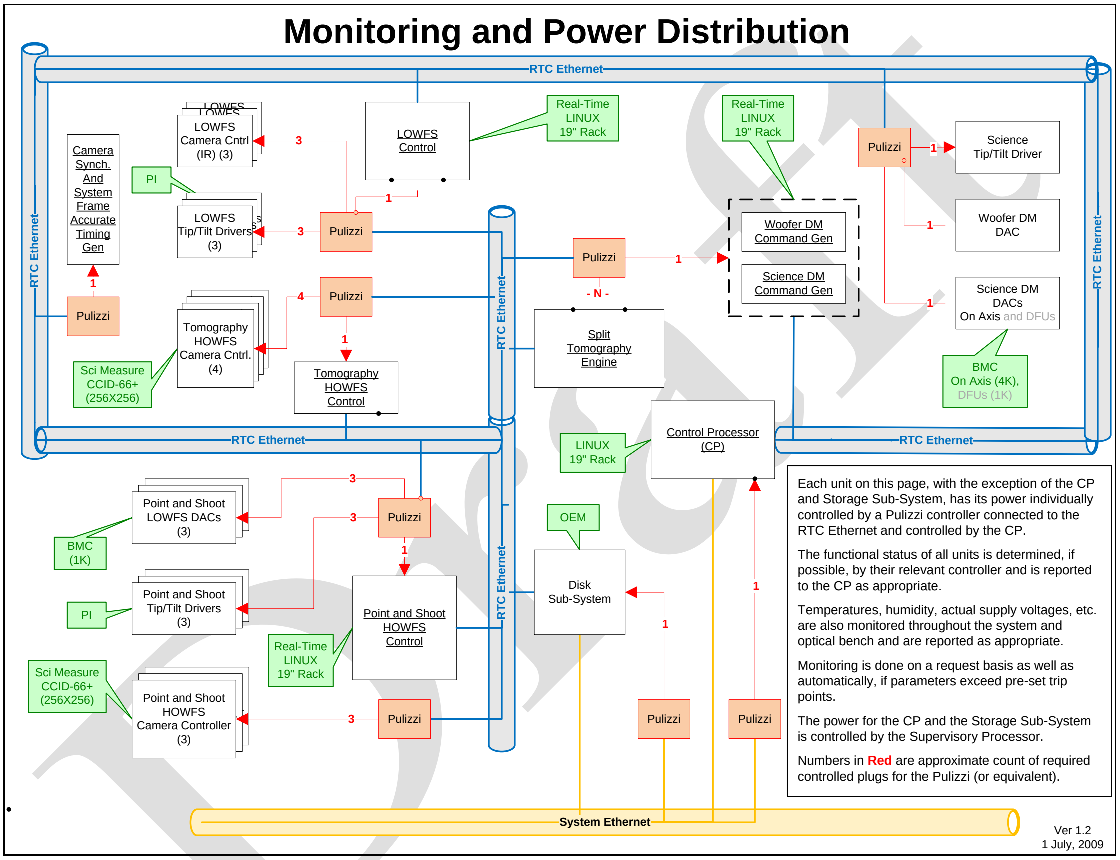
Note: If this feature cannot be implemented, then this requirement must be specifically waived for the specific board type and connector in question.

13.2.5 Temperature

14. Power Distribution, Environment and Cooling

Power to the various sub-systems will be controlled by a networked power control system such as a Pulizzi. These are controlled by commands through the CP by the AO Control. The power to the CP and the Disk Sub-System will be controlled directly by the AO Control.

Figure 24 shows the power control for the RTC.



Each unit on this page, with the exception of the CP and Storage Sub-System, has its power individually controlled by a Pulizzi controller connected to the RTC Ethernet and controlled by the CP.

The functional status of all units is determined, if possible, by their relevant controller and is reported to the CP as appropriate.

Temperatures, humidity, actual supply voltages, etc. are also monitored throughout the system and optical bench and are reported as appropriate.

Monitoring is done on a request basis as well as automatically, if parameters exceed pre-set trip points.

The power for the CP and the Storage Sub-System is controlled by the Supervisory Processor.

Numbers in Red are approximate count of required controlled plugs for the Pulizzi (or equivalent).

Figure 24 Power Distribution

14.1 *Power Sequencing*

14.2 *WFS*

14.3 *Tomography Engine*

14.3.1 FPGA Power Dissipation

The majority of the power dissipated by the tomography engine is in the FPGA's. There will be approximately 150 FPGAs, each dissipating a maximum of approximately 15 Watts for a total of 2.25 KW.

14.3.1.1 Sample Chip and Configuration

The initial evaluation of the FPGA design was based on the following estimated parameters in **Table 9**:

MAKE THIS A LINK

Item	**Value**	**Comment**
Chip:	VSX95T	
System Clock:	100 MHz	
LVDS Clock:	600 MHz	
Sub apertures per chip:	100 (4 sub apertures per PE)	
Process Parameters	Nominal	
Ambient Air Temp	30° C	
Air Flow	250 LFPM (moderate air flow)	
Heat Sink	High Profile (I like cool junctions)	
Junction Temp (θJ)	52° C (Nice)	
I/O utilization:	100%	
DSP-48 Mult/Acc utilization	40%	
Logic utilization	56%	

Table 9 Preliminary FPGA Power Dissipation Analysis

14.3.1.2 Initial Power Estimates per Chip

It results of the preliminary power estimate per chip was a worst case of < 15 Watts.

14.3.2 Tomography Engine Cooling

The initial results of the pre-design power estimator tool look good. We are not pushing any limits yet. No exotic cooling requirements seem indicated. A more detailed examination will be done later with a sample design actually synthesized and routed into a chip.

14.4 *DM command Generation*

14.5 *Disk Sub-System*

14.6 *Control Processor (CP)*

14.7 *Environment*

14.8 *Power and Cooling Requirements*

15. Documentation

16. Other Hardware and Software not Otherwise Covered

16.1 *Internal Networking*

16.2 *Shutters*

16.3 *Laser fiber In/Out actuators*

16.4 *Laser On/Off and intensity*

16.5 *Filter wheel operation*

16.6 *Interlocks*

16.7 *Inter-Unit Cables*

Cables between units must provide the ability to verify the physical connection has been made between both units from a remote location. An example would be a loopback pair in the cable with a permanent jumper between the pins on the connector. Other methods can be used, but the requirement is a direct method to insure connectivity from a remote location.

17. Appendices

Appendix A Glossary

Item	**Description**
Junction Temperature	
Container/component	
CP	Control Processor
Pipelined	
MTBR	Mean Time Before Repair
MTBF	Mean Time Between Failures
Camera Link™	
Centroid	
DAC	Digital to Analog Converter
DM	Deformable Mirror
FPGA	Field Programmable Gate Array
GPU	
HOWFS	High Order Wave Front Sensor
Latency	
LGS	Laser Guide Star
LOWFS	Low Order Wave Front Sensor
LVDS	Low Voltage Differential Signaling
NGS	Natural Guide Star
RTC	Real Time Computer
Tip/Tilt Mirror	
Tomography	
WFS	Wave Front Sensor

Appendix B Time-related keywords in Keck Observatory FITS files [2]

As an overview of the time-related keywords found in FITS files from the optical instruments at Keck Observatory please, see this computer-generated documentary list.

The two meanings of "keyword" at Keck

In the subsequent discussion, it is important to distinguish between KTL keywords and FITS keywords.

KTL keywords are obtained by interprocess (and often inter-machine) communication when a client process indicates interest. The values of the KTL keywords communicate information about the state of the various subsystems of the Keck telescopes and instruments. Depending on the nature of the quantity described by the keyword it may be continuously varying during an observation or it may never change. Each KTL keyword is associated with a "service" that corresponds to some subsystem of the operations at Keck. Some KTL services are the responsibility of the Keck staff and describe observatory-wide systems (e.g., DCS describes the telescope pointing and dome, ACS describes the primary mirror). Other KTL services are associated with the particular instrument (e.g., CCD subsystems, motor control subsystems, etc.).

FITS keywords are recorded in the headers of the files, which are produced by the instrument systems when they obtain an image. In many cases, there is a one-to-one correspondence between KTL keywords and FITS keywords. There are, however, a number of FITS keywords inserted into every image that do not correspond to any KTL keyword. There can also be FITS keywords, which correspond to KTL keywords, but where the FITS keyword name does not match the KTL keyword.

Finally, it is relevant to note that the FITS `DATE` keyword indicates a time stamp related to the construction of the FITS file itself. There should be no expectation that the value of the FITS `DATE` keyword is related to the time at which the data in the FITS file were acquired.

Historical review of Keck timing keyword systems

During the development of the keyword handling systems at Keck Observatory, there was no requirement for a method of obtaining a precise time stamp. None of the specifications for the initial optical instruments at Keck (HIRES, LRIS, ESI, DEIMOS) included any requirements for precise timing. As a result, the ability of these instruments at Keck to indicate the time for any exposure-related event is haphazard.

The Keck telescope pointing control system (DCS) uses commercial GPS receivers as the time base. The time provided by these receivers has experienced "interesting" events during the GPS W1K rollover in 1999 and during theater-level GPS jamming experiments conducted by DoD. It is not clear what the DCS system reports during a leap second. Other than in these exceptional conditions, the time base used by DCS is relatively reliable.

Any of the Keck instruments is able to ask DCS for its current values of its timing keywords. Most notable among these KTL keywords are `DATE-OBS`, `UTC`, and `MJD`. Reading the KTL `DATE-OBS` keyword from DCS provides a FITS Y2K-agreement-compliant value of the calendar date according to Coordinated Universal Time. Reading the KTL `UTC` keyword from DCS provides a sexagesimal character string representation of the value of Coordinated Universal Time. It is important to note that these are two separate keywords. It is not possible to request both keywords simultaneously, nor to receive values, which are guaranteed to refer to the same instant. As a result, it is possible that around 0 hours UTC the values of the KTL `DATE-OBS` and `UTC` keywords from DCS can refer to different calendar days -- thus resulting in a one-day discrepancy in the available notion of the time. (Given that 0 hours UTC occurs around mid-day in Hawaii this will probably never be an issue for observational data.)

The manner by which DCS can be queried for date and time is a KTL keyword request. In the current scheme, the KTL keyword read generates a EPICS request that is sent over the network to the DCS systems. After that request is received and processed in DCS the resulting value is sent back over the network to the KTL requestor. The design of the system provides no guarantees about how long this round trip takes. Under normal circumstances the response to the KTL read is received within a small fraction of a second, but under adverse circumstances the DCS system has been observed to take as long as 30 seconds. In these adverse circumstances, it is not clear how to ascertain what point in the transaction is represented by the time value.

Gathering the time-related keywords at Keck

The initial optical instruments at Keck (HIRES, LRIS, ESI, DEIMOS) monitor the sequence of events of an exposure using a process named `watch_ccd`. The `watch_ccd` process is responsible for gathering most of the KTL keywords, which will be written, into the FITS file along with the image data of an exposure. The `watch_ccd` process has a list of KTL keywords, which indicates when each keyword should be gathered during the exposure sequence. The three points during the exposure are erase, shutter open, and shutter close.

In these instruments, the computer, which has command of the CCD, is called the CCD crate. The CCD crate issues signals to the CCD controller to initiate all operations of the CCD electronics and the shutter. When the CCD crate issues exposure-related signals to the CCD controller, it also transmits a MUSIC broadcast message to the `traffic` process. The `traffic` process then re-transmits that MUSIC broadcast message to every other process, one of which is `watch_ccd`. Each of these hops between processes on the network takes time.

When `watch_ccd` receives an exposure-related event it proceeds to issue KTL read requests for each of the keywords whose value is desired at that event. These KTL read requests are issued sequentially. Each request must complete before the next KTL read is done. A successful KTL read involves a round trip between `watch_ccd` and the system(s), which know(s) the value(s) of each keyword. Usually each KTL read succeeds in a small fraction of a second, but under adverse circumstances, the servers for the keywords may respond slowly, or not respond at all.

Each KTL keyword read has a timeout. If the server does not respond within that timeout a KTL read will fail, and `watch_ccd` will proceed to read the next keyword in its list. There are often dozens, even hundreds of keywords in the lists that `watch_ccd` has to acquire. The total time to gather the keywords, even under ideal circumstances can be several seconds, and under adverse circumstances, it has been monitored to be as many as 30 seconds or more. If the delay is long then the values of the KTL `DATE-OBS` and (more importantly) `UTC` keywords from DCS may not bear much relation to the instant at which the shutter events happened.

Strategies employed to get the best possible time stamps

These issues regarding the validity of the time-related keywords came to the attention of the UCO/Lick Scientific Programming Group during the retrofit, which added the exposure meter to HIRES. Starting with that deployment the CCD readout software has been tailored to provide the best possible indication of event times given the constraints of systems, which were not specified with that as a design requirement. These principles are now in use with HIRES and DEIMOS. They will be applied to LRIS during the Red Mosaic upgrade.

The first remedy was to carefully arrange the order of the keywords in the lists provided to `watch_ccd`. In the more recent deployments, the list has the KTL `UTC` keyword as the first in the list. There can be no quicker way of getting that keyword from DCS.

Another remedy was to enhance `watch_ccd` to distribute the keyword collection over three events rather than two. Initially the keywords were only gathered in response to shutter open and shutter close. Those lists were rather large, and not optimally arranged. In the more recent versions of `watch_ccd,` it is possible to gather keywords in response to the initiation of the erase of the CCD prior to opening the shutter. The objective is to gather keywords whose value tends not to change during erase, and thus reduce the length of the lists of keywords desired at shutter open and shutter close.

Another remedy was to enhance `watch_ccd` to be able to rename a KTL keyword when writing it to the FITS file. This renaming is often done when it is desired to sample the value of a KTL keyword more than once during an observation. In the case of HIRES the KTL `UTC` and `DATE-OBS` keywords are read in response to both shutter open and shutter close. For shutter open, they are written to the FITS file with the same name, and for shutter close, they are written to the FITS file with the names `UTC-END` and `DATE-END`. This has been in use for all instruments deployed or upgraded since the HIRES exposure meter. (Note that in the unusual case of an exposure very near to 0 h UTC the combination of all four of these keywords can probably be used to ascertain whether the pair of keywords for one of the shutter events may have straddled the day boundary.)

The most definitive remedy was to create an independent method for `watch_ccd` to give some indication of the times of the shutter events. In recent versions when `watch_ccd` receives the shutter events, it immediately queries the system to get the UNIX system time. The values of these queries are inserted into the KTL keywords `DATE_BEG` (for shutter open) and `DATE_END` (for shutter close). These keywords are only as precise as one second. Furthermore, these keywords rely on the UNIX system clock being set correctly. In normal operation, the Keck systems should be using NTP to keep their clocks on time, but this is not guaranteed.

Interpretation of time keywords in Keck FITS files

In any instrument, which supplies the FITS `DATE_BEG` and `DATE_END` keywords along with the `UTC` and `UTC-END` keywords, it is prudent to inspect their values. If the various subsystems were operating nominally then the keywords for shutter open will be in reasonable agreement, as will the keywords for shutter close. Furthermore, the difference between the times of shutter close and shutter open should be in reasonable agreement with the various keywords that indicate the duration of the exposure. If all of these agree, then their values are probably reliable. If they do not agree, then their discrepancies give some indication of how unreliable the time stamps may be.

Steve Allen <sla@ucolick.org>
$Date: 2008/12/08 $

Appendix C Wavefront Error Calculations Due to Latency

Appendix D Split Tomography Issues

Appendix E System Synchronization

Appendix F RTC Sensors and Actuators

Ver 1.1	1 Sept, 2009		NGAO Current Target Design			
Element	WFS Cameras	DM Size (K)	LOWFS Cameras	Tip/Tilt Mirrors	Tip/Tilt Actuators	Comments
LOWFS 1 *	N/A	1024	1	1	N/A	LOWFS DMs get their wavefront correction from the POS WFS
LOWFS 2 *	N/A	1024	1	1	N/A	LOWFS stars are at 2am to preserve sky coverage.
LOWFS 3 *	N/A	1024	1	1	N/A	LOWFS get their T/T from the LOWFS, not a LGS
PAS 1 **	128x128	N/A	N/A	1	N/A	Point and Shoot LGSs are used only for correcting the
PAS 2 **	128x128	N/A	N/A	1	N/A	wavefront of their LOWFS star, not for tomography
PAS 3 **	128x128	N/A	N/A	1	N/A	
Tomog 1	256x256	N/A	N/A	1	N/A	
Tomog 2	256x256	N/A	N/A	1	N/A	
Tomog 3	256x256	N/A	N/A	1	N/A	
Tomog 4	256x256	N/A	N/A	1	N/A	
Woofer	N/A	400	N/A	0	1	Woofer T/T value is calculated from the on-axis
						tomography
MOAO 1	N/A	4096	N/A	0	N/A	Woofer will serve as T/T for the central science object
MOAO 2	N/A	(1K)	N/A	(1)	N/A	MOAO T/T values are calculated from off-axis tomography
MOAO 3	N/A	(1K)	N/A	(1)	N/A	
MOAO 4	N/A	(1K)	N/A	(1)	N/A	
MOAO 5	N/A	(1K)	N/A	(1)	N/A	
MOAO 6	N/A	(1K)	N/A	(1)	N/A	
MOAO 7	N/A	(1K)	N/A	(1)	N/A	
Totals	**7**	**5**	**3**	**10**	**1**	

() Indicates possible future expansion

MOAO 2 - 7 may not be installed in initial configuration, but the RTC must be able to handle them, if and when they are installed, while still meeting the PRD requirements: 2KHz, etc.

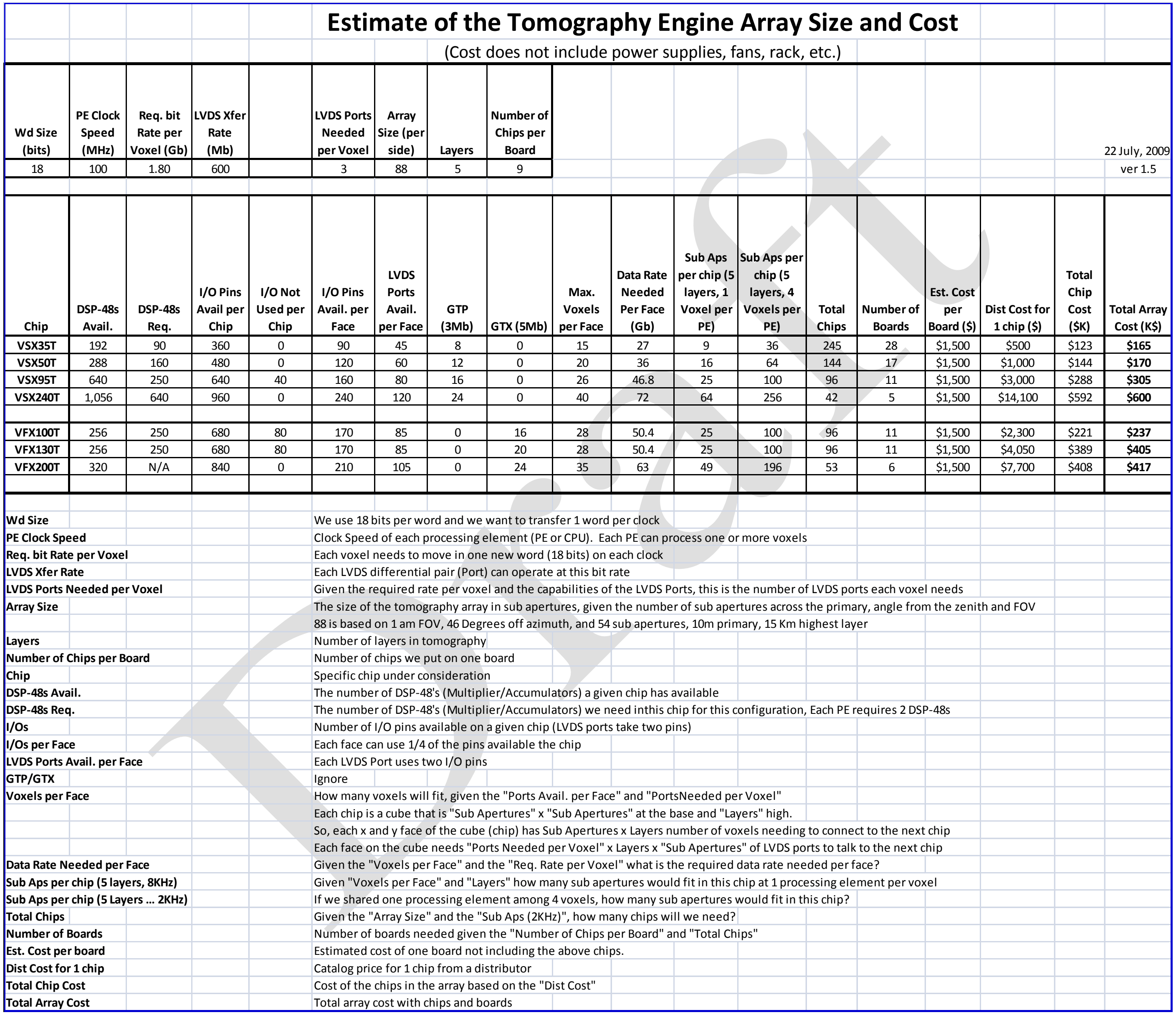
* What size are the T/T and Truth cameras?

** T/T mirrors for the LOWFS LGS get their value from their LOWFS LGS, not their LOWFS star

*** T/T mirrors for the HOWFS get their values from their HOWFS, not from tomography

Table 10 List of sensors and actuators connected to the RTC

Appendix G Tomography Array Size and Cost Estimate



Estimate of the Tomography Engine Array Size and Cost

(Cost does not include power supplies, fans, rack, etc.)

22 July, 2009
ver 1.5

Wd Size (bits)	PE Clock Speed (MHz)	Req. bit Rate per Voxel (Gb)	LVDS Xfer Rate (Mb)		LVDS Ports Needed per Voxel	Array Size (per side)	Layers	Number of Chips per Board
18	100	1.80	600		3	88	5	9

Chip	DSP-48s Avail.	DSP-48s Req.	I/O Pins Avail per Chip	I/O Not Used per Chip	I/O Pins Avail. per Face	LVDS Ports Avail. per Face	GTP (3Mb)	GTX (5Mb)	Max. Voxels per Face	Data Rate Needed Per Face (Gb)	Sub Aps per chip (5 layers, 1 Voxel per PE)	Sub Aps per chip (5 layers, 4 Voxels per PE)	Total Chips	Number of Boards	Est. Cost per Board ($)	Dist Cost for 1 chip ($)	Total Chip Cost ($K)	Total Array Cost (K$)
VSX35T	192	90	360	0	90	45	8	0	15	27	9	36	245	28	$1,500	$500	$123	**$165**
VSX50T	288	160	480	0	120	60	12	0	20	36	16	64	144	17	$1,500	$1,000	$144	**$170**
VSX95T	640	250	640	40	160	80	16	0	26	46.8	25	100	96	11	$1,500	$3,000	$288	**$305**
VSX240T	1,056	640	960	0	240	120	24	0	40	72	64	256	42	5	$1,500	$14,100	$592	**$600**
VFX100T	256	250	680	80	170	85	0	16	28	50.4	25	100	96	11	$1,500	$2,300	$221	**$237**
VFX130T	256	250	680	80	170	85	0	20	28	50.4	25	100	96	11	$1,500	$4,050	$389	**$405**
VFX200T	320	N/A	840	0	210	105	0	24	35	63	49	196	53	6	$1,500	$7,700	$408	**$417**

Wd Size	We use 18 bits per word and we want to transfer 1 word per clock
PE Clock Speed	Clock Speed of each processing element (PE or CPU). Each PE can process one or more voxels
Req. bit Rate per Voxel	Each voxel needs to move in one new word (18 bits) on each clock
LVDS Xfer Rate	Each LVDS differential pair (Port) can operate at this bit rate
LVDS Ports Needed per Voxel	Given the required rate per voxel and the capabilities of the LVDS Ports, this is the number of LVDS ports each voxel needs
Array Size	The size of the tomography array in sub apertures, given the number of sub apertures across the primary, angle from the zenith and FOV
	88 is based on 1 am FOV, 46 Degrees off azimuth, and 54 sub apertures, 10m primary, 15 Km highest layer
Layers	Number of layers in tomography
Number of Chips per Board	Number of chips we put on one board
Chip	Specific chip under consideration
DSP-48s Avail.	The number of DSP-48's (Multiplier/Accumulators) a given chip has available
DSP-48s Req.	The number of DSP-48's (Multiplier/Accumulators) we need inthis chip for this configuration, Each PE requires 2 DSP-48s
I/Os	Number of I/O pins available on a given chip (LVDS ports take two pins)
I/Os per Face	Each face can use 1/4 of the pins available the chip
LVDS Ports Avail. per Face	Each LVDS Port uses two I/O pins
GTP/GTX	Ignore
Voxels per Face	How many voxels will fit, given the "Ports Avail. per Face" and "PortsNeeded per Voxel"
	Each chip is a cube that is "Sub Apertures" x "Sub Apertures" at the base and "Layers" high.
	So, each x and y face of the cube (chip) has Sub Apertures x Layers number of voxels needing to connect to the next chip
	Each face on the cube needs "Ports Needed per Voxel" x Layers x "Sub Apertures" of LVDS ports to talk to the next chip
Data Rate Needed per Face	Given the "Voxels per Face" and the "Req. Rate per Voxel" what is the required data rate needed per face?
Sub Aps per chip (5 layers, 8KHz)	Given "Voxels per Face" and "Layers" how many sub apertures would fit in this chip at 1 processing element per voxel
Sub Aps per chip (5 Layers … 2KHz)	If we shared one processing element among 4 voxels, how many sub apertures would fit in this chip?
Total Chips	Given the "Array Size" and the "Sub Aps (2KHz)", how many chips will we need?
Number of Boards	Number of boards needed given the "Number of Chips per Board" and "Total Chips"
Est. Cost per board	Estimated cost of one board not including the above chips.
Dist Cost for 1 chip	Catalog price for 1 chip from a distributor
Total Chip Cost	Cost of the chips in the array based on the "Dist Cost"
Total Array Cost	Total array cost with chips and boards

Table 11 Estimate of the Tomography Engine Array Size and Cost

18. References

---

1 Kak, TOMOGRAPHY

2 Steve Allen, http://www.ucolick.org/~sla/fits/kecktime.html, Jan 28, 2009

PAGEREF