3 Low-Level Details

3.1 Timing System

Timing systems are critical to the ability of Project-X to coordinate beam acceleration and transfer between the various accelerators that will make up the complex. They are also essential to the ability of the control system to provide correlated data acquisition. In order to provide these timing capabilities, there will be two types of clock systems in Project-X.

The first will be a basic accelerator clock that provides high-level timing for the entire complex and is common to all machines. In the legacy systems this function is accomplished via the TCLK system. It is an 8-bit, 10 MHz, modified Manchester encoded serial transmission of clock events that provide basic accelerator timing information. As more timing functionality is necessary for the new accelerators of Project-X, a new clock system (herein referred to as XCLK) is required.

The second type of clock systems will be machine-specific RF timing systems (Beam Sync Clocks.) These systems will allow for the transmission of the individual machine's RF and beam synchronization markers to facilitate high precision (RF bucket level) timing for such things as instrumentation and kicker triggering. In the legacy systems this function is handled via the individual accelerator's Beam Sync Clocks (MIBS, RRBS, etc.). As with TCLK, the legacy beam sync clocks are 8-bit, modified Manchester encoded serial transmissions. However, their base frequencies are subharmonics of the machine's RF frequencies rather than the 10 MHz of TCLK. The new Linac of Project-X will require its own beam sync clock.

3.1.1 Basic Accelerator Clock (XCLK)

This section only covers requirements for XCLK.

No.	Requirement	Source	Priority
CXR-LL1-100	Basic accelerator clock timing will be sourced via a single Timeline Generator (TLG) and transmitted on optical fiber.	G.Vogel 12-2007	Expected
CXR-LL1-110	TCLK events shall play synchronously on both TCLK and XCLK.	G.Vogel 12-2007	Critical
CXR-LL1-120	XCLK will run on a 1 GHz, or higher, carrier phase-locked to the TCLK's 10 MHz carrier.	G.Vogel 12-2007	Expected
CXR-LL1-130	8 bits of the XCLK frame represent the event (which is compatible with the current TCLK event.) The XCLK frame will have an additional *n* bits for payload.	G.Vogel 12-2007	Expected
CXR-LL1-140	The XCLK frame size will not exceed 1.2uS.	G.Vogel 12-2007	Critical

No.	Requirement	Source	Priority
CXR-LL1-150	Events occurring on XCLK cannot affect events in the legacy system.	G.Vogel 12-2007	Critical
CXR-LL1-160	The data in the event payload will be self-describing.	C.Briegel 12-2007	Desired
CXR-LL1-170	8 bits of the XCLK frame define a "subevent" type field.	C.Briegel 12-2007	Desired
CXR-LL1-180	32 bits of the XCLK frame will be reserved for a per-event counter	R. Rechenmacher 12-2207	Desired
CXR-LL1-190	2 fibers will be run from repeater to repeater.	G.Vogel 12-2007	Expected
CXR-LL1-200	The repeaters will constantly monitor and compare the two transmissions and have auto-switchover if one carrier fails.	G.Vogel 12-2007	Expected
CXR-LL1-210	The repeaters will inhibit beam if total clock is lost.	G.Vogel 12-2007	Expected
CXR-LL1-220	The hardware group will provide hardware XCLK simulators for front-end developers.	G.Vogel 12-2007	Desired
CXR-LL1-230	Front-end software will be able to simulate the clock system to allow development on machines that don't have access to XCLK signals.	R.Rechenmacher 12-2007	Desired

3.1.2 Beam Sync Clock

This section only covers requirements for the Beam Sync Clock.

No.	Requirement	Source	Priority
CXR-LL1-240	Main Injector and Recycler will continue to use the existing beam sync clock.	G.Vogel 12-2007	Expected
CXR-LL1-250	Linac RF reference with a bucket marker will be made available on fiber in all linac locations.	G.Vogel 12-2007	Expected

3.2 Equipment Interface/Instrumentation

The Controls System needs to interface to a variety of equipment both purchased and designed in-house. These include instrumentation, vacuum, power supplies, water systems, etc. In order to ensure that the Controls System is reliable and easy to diagnose, a limited number of standard interfaces should be implemented and duplication should be avoided. The controls group will provide a standard equipment interface for other groups to incorporate into their equipment designs.

No.	***Requirement***	***Source***	***Priority***
CXR-LL2-100	General purpose digitizing hardware must allow all of its channels to be plotted simultaneously.	R.Neswold 12-2007	Expected
CXR-LL2-110	Software and hardware will be designed with high-availability in mind.	C.Briegel 12-2007	Desired
CXR-LL2-120	The hardware group will support and provide a preferred digitizer.	G.Vogel 12-2007	Expected
CXR-LL2-130	Front-end hardware will support, as a minimum, full-duplex, gigabit copper communications.	T.Zingelman 12-2007	Critical
CXR-LL2-140	Front-end platforms will support IPv6 communications.	T.Zingelman 12-2007	Expected
CXR-LL2-150	Front-end platforms must support port scans gracefully.	T.Zingelman 1-2008	Expected
CXR-LL2-160	Supported hardware and software will be enumerated.	C.Briegel 12-2007	Expected
CXR-LL2-170	All new support (requested or required) will be sent to a committee for review to aid in maintaining an appropriate set of solutions.	C.Briegel 12-2007	Expected

3.3 Development Environment

Due to technical and historical reasons, the front-end development environment is separate from ones used by other groups. We hope a consolidation of toolsets is possible, so that working on different layers of the control system doesn't require a differing set of skills.

In addition to the requirements in this section, we would like to see the front-end development also follow the requirements expressed in the Software Development section.

No.	**Requirement**	**Source**	**Priority**
CXR-LL3-100	The development and build environments for front-ends are the same used by central and console developers.	R.Neswold 12-2007	Expected
CXR-LL3-110	Front-ends will run on platforms that support memory protection.	R.Neswold 12-2007	Expected
CXR-LL3-120	Front-end software will use memory protection features in its design to improve reliability.	R.Neswold 12-2007	Expected
CXR-LL3-130	Front-end systems will run on platforms that support portable APIs (e.g. pthreads, unix system calls)	R.Neswold 12-2007	Desired

No.	Requirement	Source	Priority
CXR-LL3-140	The framework used to create front-end software will be usable by central service programmers.	R.Neswold 12-2007	Desired
CXR-LL3-150	The front end provides a user application framework allowing user code in the front-end.	C.Briegel 12-2007	Expected
CXR-LL3-160	All front-ends will have debugging facilities available to developers ("post-mortem" facilities, access to internal state, etc.)	R.Rechenmacher 12-2007	Expected
CXR-LL3-170	All front-ends will have a common set of devices to monitor the front-end (CPU Temperature, system load, memory usage, version devices, etc.)	R.Neswold 12-2007	Expected

3.4 Data Acquisition/Setting

Fermilab has a distributed control system, meaning the data used by an application is generally acquired from a remote machine. A standardized network protocol is used to express how to request data and how the data is returned to the requestor.

No.	Requirement	Source	Priority
CXR-LL4-100	One network protocol will be used to collect control system data.	R.Rechenmacher 1-2008	Expected
CXR-LL4-110	"Policies" active in the control system need to be honored across all acquisition methods.	R.Rechenmacher 1-2008	Expected
CXR-LL4-120	A device can be viewed as an object with many attributes and data types.	C.Briegel 12-2007	Expected
CXR-LL4-130	Device data may be acquired at any rate a user specifies. If the rate exceeds the capabilities of the device, data is returned at the device's maximum rate.	R.Neswold 12-2007	Expected
CXR-LL4-140	Acquisition protocols must provide a way to correlate the data.	R.Rechenmacher 1-2008	Expected
CXR-LL4-150	Acquisition protocol will support multicast requests to acquire data across multiple front-ends.	M.Sliczniak 1-2008	Expected
CXR-LL4-160	Replies to a multicast request must arrive before a deadline prior to the next cycle and the reply data must come from the current cycle.	M.Sliczniak 1-2008	Expected
CXR-LL4-170	Data acquisition protocols will use a self-describing data format.	R.Rechenmacher 12-2007	Expected

No.	Requirement	Source	Priority
CXR-LL4-180	All device data can be described by a data definition derived from the front-end and consistent with the application environment.	C.Briegel 12-2007	Expected
CXR-LL4-190	A time stamp of when the data was captured will accompany all device data. The acquisition protocol will also have a global beam cycle count and a global error/status field.	R.Neswold 12-2007	Expected
CXR-LL4-200	The latest device data can be retrieved (known as a "one-shot" request.)	C.Briegel 12-2007	Expected
CXR-LL4-210	Data can be retrieved at a periodic rate.	C.Briegel 12-2007	Expected
CXR-LL4-220	"One-shot" and repetitive data can be retrieved based on an XCLK event occurrence followed by an optional delay.	C.Briegel 12-2007	Expected
CXR-LL4-230	"One-shot" and repetitive data can be retrieved based on a state change.	C.Briegel 12-2007	Expected
CXR-LL4-240	Data can be retrieved based on an event with a specified event counter.	C.Briegel 12-2007	Expected
CXR-LL4-250	Repetitive data can be retrieved whenever the data changes by a delta (or range, tolerance, etc.)	C.Briegel 12-2007	Desired
CXR-LL4-260	Front-ends will be able to acquire data from each other, directly from the remote machine -- not through a consolidator.	R.Neswold 12-2007	Expected
CXR-LL4-270	While device data can be returned either in raw or scaled values, all front-ends must be capable of scaling the data for internal use. This includes data acquired from other nodes.	C.Briegel 12-2007	Expected
CXR-LL4-280	Devices will be able to handle rapid (~15Hz) settings.	C.Briegel 12-2007	Expected
CXR-LL4-290	Setting protocol will allow optional reading of a parameter.	C.Briegel 12-2007	Desired
CXR-LL4-300	All data on the network will be in network byte order.	R.Neswold 12-2007	Desired
CXR-LL4-310	There will be an acquisition protocol that returns data suitable for a real-time plot.	C.Briegel 1-2008	Expected
CXR-LL4-320	There will be an acquisition protocol that returns an array of data on an event (i.e. a snapshot of a waveform.)	C.Briegel 1-2008	Expected
CXR-LL4-330	The waveform snapshot will also support collection on alarm activation.	C.Briegel 1-2008	Desired

3.4.1 Network Protocol Policies

As an evolution of our current control system, we envision the improved protocols allow "policies" to be enabled. For instance, we might decide that settings require authentication, or that devices can be temporarily owned by users.

No.	Requirement	Source	Priority
CXR-LL4-340	"Policies" active in the control system need to be honored across all acquisition methods.	R.Rechenmacher 1-2008	Expected
CXR-LL4-350	Device access will have "ownership". Should be able to query who is current "owner" of the device.	C.Briegel 12-2007	Expected
CXR-LL4-360	There will be a policy of access rights for users to restrict who is allowed to make changes, mainly for preventing interference in conflicting uses of the machine.	R.Rechenmacher 12-2007	Expected
CXR-LL4-370	Setting protocol will support optional transaction semantics (i.e. settings can be queued for later commit or rollback.)	R.Neswold 12-2007	Desired

3.4.2 Alarm support

Alarm reporting is a very important apect of our current control system. A huge number of devices are scanned frequently to ensure they are operating within their constraints. This section of requirements covers alarm support, along with some improvements.

No.	Requirement	Source	Priority
CXR-LL4-380	Front-ends will scan devices periodically and compare the reading with alarm constraints. Readings that violate their constraint are said to be in alarm. Alarm reports are forwarded to a central alarm service, which reports the alarms to operators.	C.Briegel 12-2007	Expected
CXR-LL4-390	There is a default alarm scan frequency, which can be overridden for each device.	C.Briegel 12-2007	Expected
CXR-LL4-400	There is a default alarm scan routine, which can be overridden by a user-written version.	C.Briegel 12-2007	Expected
CXR-LL4-410	The central alarm service will be designed with high-availability in mind.	R.Neswold 1-2008	Expected
CXR-LL4-420	A device can be constrained by maximum and minimum values.	C.Briegel 12-2007	Expected
CXR-LL4-430	A device can be constrained by a bit-mask and pattern-match.	C.Briegel 12-2007	Expected

No.	Requirement	Source	Priority
CXR-LL4-440	A device can be constrained by a user-defined constraint.	C.Briegel 12-2007	Desired
CXR-LL4-450	All alarms have a consecutive alarm threshold.	C.Briegel 12-2007	Expected
CXR-LL4-460	All alarms have the ability to pull a software abort.	C.Briegel 12-2007	Expected
CXR-LL4-470	All alarms have the ability to set a device.	C.Briegel 12-2007	Expected
CXR-LL4-480	All alarms have the ability to provide an unsolicited alarm notification to a specified set of servers.	C.Briegel 12-2007	Expected
CXR-LL4-490	There are event alarms, which are notifications without a clear of the event.	C.Briegel 12-2007	Expected
CXR-LL4-500	There are exception alarms, such as analog reading, digital status, and alarm setting alarms.	C.Briegel 12-2007	Expected
CXR-LL4-510	Alarms can be individually bypassed.	C.Briegel 12-2007	Expected
CXR-LL4-520	Alarms can be grouped together and be bypassed as a group.	C.Briegel 12-2007	Expected

3.4.3 Reliability Requirements

The following requirements improve the reliability of the network protocol.

No.	Requirement	Source	Priority
CXR-LL4-530	Network packets need to be acknowledged (or the reply itself be sent) within 100ms of reception.	R.Neswold 12-2007	Expected
CXR-LL4-540	Long-term, slow frequency connections will provide a “keep-alive” status to be able to detect broken or closed connections.	R.Neswold 12-2007	Expected
CXR-LL4-550	The front-end infrastructure will be able to report, within a few seconds, the flows/connections to front-ends that do not involve "standard" front-end interface ports.	R.Rechenmacher 12-2007	Desired