PRODUCT BRIEF

QNX Loosely Coupled Lock Step

Autonomous driving systems use the most powerful CPUs and the most complex software ever seen in the automotive industry. On the hardware front, semiconductor manufacturers are pushing technology to the point where, for the first time in recent history, hardware is becoming less reliable. The miniaturization used in modern processors and memory makes them more susceptible to errors caused by electromagnetic interference and cross talk between neighboring cells and particle bombardment such as alpha rays. The complex role that software is being asked to take with regards to simultaneous image processing, deep learning, precise localization and intelligent control algorithms can lead to errors caused by subtle software interaction problems. These errors, known as Heisenbugs are non- reproducible, elusive bugs that can go undetected even with the most rigorous testing.

The occurrence of such hardware and software errors in an autonomous driving system can impact the system's safety. To achieve ISO 26262 safety certification, these errors must be detected and handled. To help address these errors and functional safety system challenges, BlackBerry QNX has developed QNX Loosely Coupled Lock Step (LCLS).

The need for software compensation

To detect and recover from the errors described above, system designers must implement compensation mechanisms. In previous generation systems, hardware lock step has been used to detect faulty CPU operation. This fault detection was done, by having duplicate CPUs execute the same code. If one of the CPUs misbehaved one could detect that something had gone wrong. However, since both CPUs will "correctly" execute the same code, hardware lock step does not compensate for random bit flips in memory or Heisenbugs. One could also use a hardware analyzer to check the internal states and determine if something has gone wrong. This technique is not practical for today's high-performance hardware, where there are far too many internal states for a hardware checker to analyze in real time.

Clearly, hardware diagnostics on its own is not enough to detect all these errors. When paired with realtime software checking an efficient and complete means of verifying the system operation can be achieved. Such a system uses redundant copies of the software each of which perform safety-critical calculations, and the output of these copies is compared to perform the verification.

This, in essence, is the concept of QNX Loosely Coupled Lock Step.

Managing redundancy – the need for flexible configurations

QNX LCLS provides a means to test whether the hardware or software has been pushed beyond their safe operation capability. Inherently, autonomous driving is about building a functionally safe system, BlackBerry QNX LCLS helps detect potential errors that may interfere with that safe operation.

Depending on the safety design, redundant copies of safety-critical computations may need to be deployed on the same core (using temporal separation), or on different cores of a multi-core processor, or on different processors on the same board or on different ECUs over a network. In each of these different deployment configurations, synchronization and output comparison of the software copies becomes a challenge. With QNX LCLS, multiple software copies can be deployed transparently, dramatically reducing the development effort required to implement a redundancy scheme.

QNX LCLS middleware has been designed to offer many flexible configurations. Servers can be deployed on different cores in a multicore processor, on different processors within an electronic control unit (ECU) or across cores over an Ethernet network. In addition, the number of servers in a group can be flexibly and dynamically arranged in a redundant pair, in a two out of three-majority logic voting scheme or in other arrangements.

QNX Loosely Coupled Lock Step - Operating principles

The Loosely Coupled Lock Step design operates as follows:

- The QNX LCLS middleware is implemented as two C libraries that are used by safety critical applications. This duplication creates a redundant implementation to avoid a single point of failure.
- A LCLS client makes a request for a safety critical calculation to a group of distributed servers.
- Each server (a POSIX process) in a server group (as determined by the safety design) conducts a safety critical calculation.
- The LCLS middleware guarantees, that all messages from clients are delivered in the same order to all servers in a group.
- Furthermore, the LCLS middleware synchronizes the server states as new servers join or leave group.
- Since all events observed by a server are ordered, all servers will independently arrive at the same state (assuming error free operation) albeit at differing times, thereby providing protection against Heisenbugs. That is, if servers don't arrive at the same state something is wrong.
- The results from the servers can be compared by the LCLS middleware, by the client or a developer-defined comparison function.

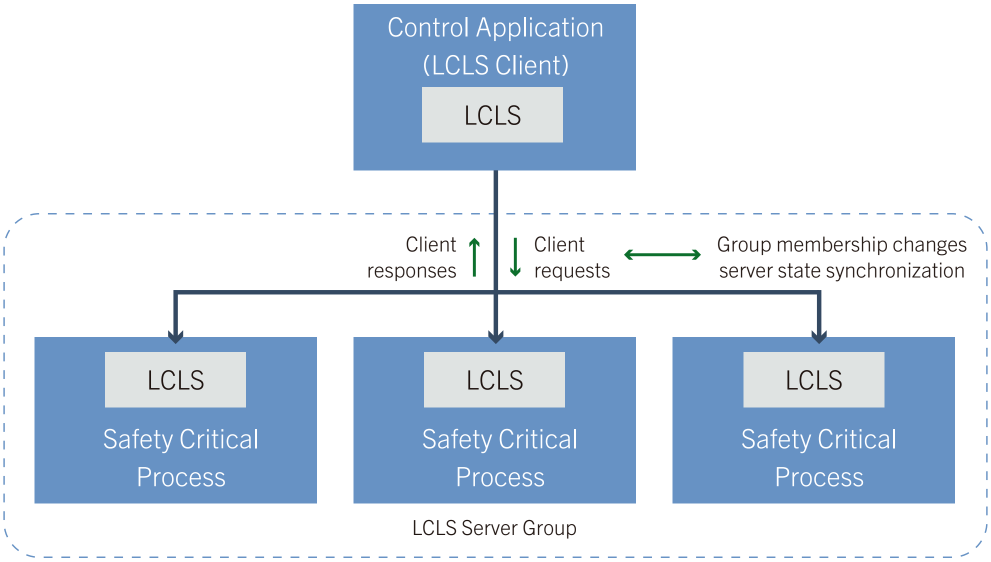

Figure 1. The LCLS middleware replicates the client messages that trigger a safety critical computation to each server in a server group. The LCLS middleware guarantees that all messages are delivered to all servers in the same order. The LCLS middleware also offers services to synchronize server state when new servers join a group.

Benefits

Since QNX LCLS uses software to handle redundant implementations, several benefits are realized in comparison with the traditional hardware lock step approach. These benefits include:

- Diverse implementations: Servers do not need to be identical replications; they can be implemented using different programming languages or by different developers from a common specification. With LCLS, servers can be implemented as monitors, providing simple sanity checks on more complex calculations.
- Separation of responsibilities: The computation logic is separated from the redundancy scheme. Software algorithms can be implemented, by domain experts, without concern for the final redundancy configurations. Safety experts can flexibly and dynamically control the redundancy approach to achieve the desired safety goal.

QNX Loosely Coupled Lock Step – Package details

QNX LCLS middleware is provided as client and server libraries supporting 32 and 64-bit ARM and x86 processors.

About BlackBerry QNX

BlackBerry QNX, is a leading supplier of safe, secure, and trusted operating systems, development tools, and professional services for connected embedded systems. Global leaders such as Ford, Audi, Cisco, General Electric, Lockheed Martin, and Siemens depend on BlackBerry QNX technologies for their next generation of secure vehicle software platforms, network routers, medical devices, industrial automation systems, security and defense systems, and other mission and/or life-critical applications. This includes full software lifecycle management via secure over the air software updates. Founded in 1980, BlackBerry QNX is headquartered in Ottawa, Canada, with its products distributed in over 100 countries worldwide.