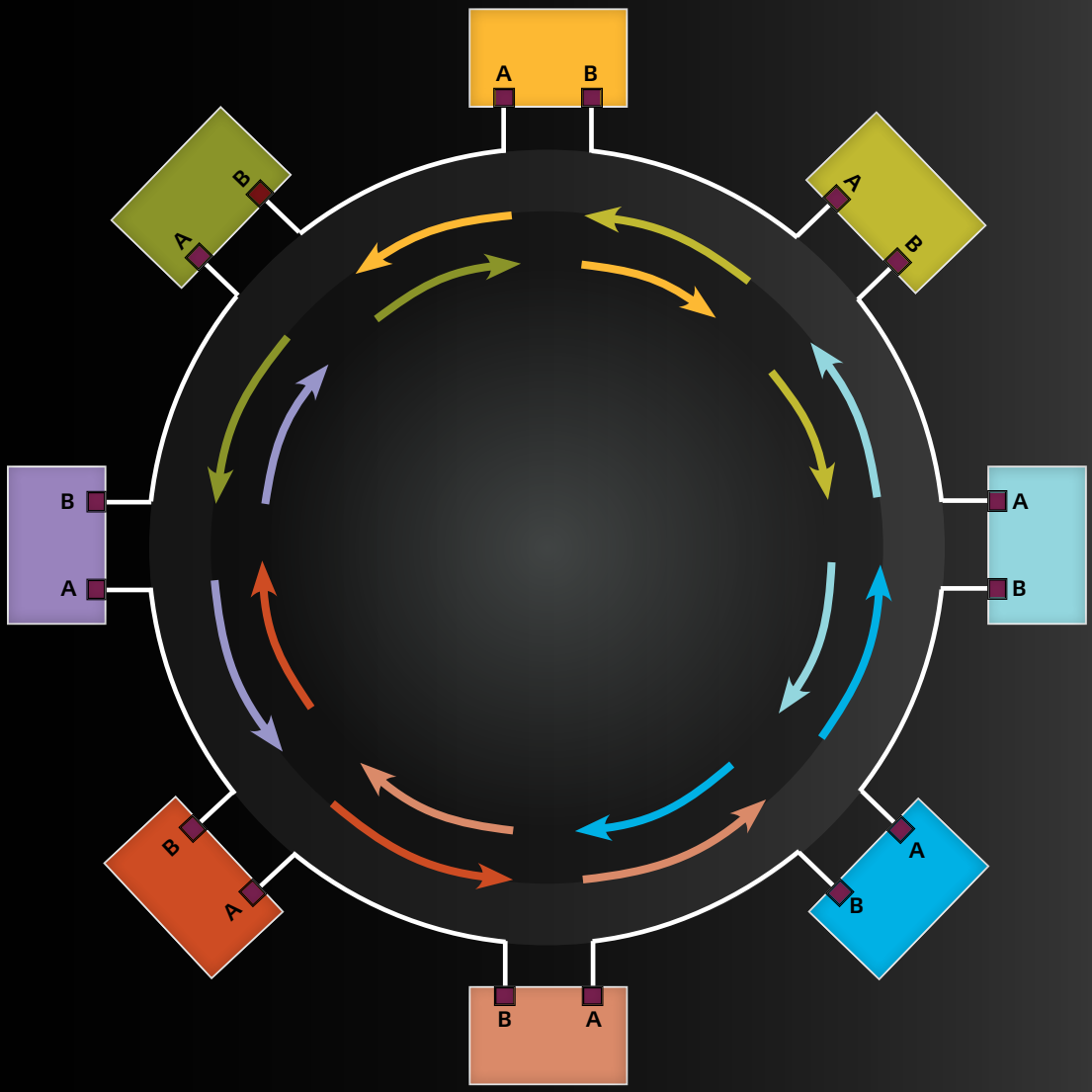
Determinism



High-priority traffic at the start in the HSR ring

by Hubert Kirrmann, Solutil, Switzerland, and Ioannis Sotiropoulos, ABB Switzerland Ltd, Switzerland

in Substation Automation with IEC 61850 (HSR / PTP)

Measurement, protection and control in the electrical grid rely upon transmission of time-critical sensor measurement and actor commands within a guaranteed time. To this effect, the signal chain from sensors to actors must provide deterministic data delivery from end to end. This article explains the principles of deterministic operation and the methods to obtain it. It shows how the Process Bus (IEC 61850-9-2) implemented as an HSR ring can be scheduled using the precision time protocol (IEC/IEEE 61850-9-3) to provide deterministic transmission of Sampled Values and maximize the available bandwidth for commands and other messages that have relaxed timing requirements.

A real-time system responds to a change of its input by a corresponding change of its output within a given time. All automation systems are "real-time"- a term introduced in the punched card age to distinguish from "simulated time" and "batch" computing. Real-time systems are divided into hard-real-time systems that always react before a given deadline (after which the produced output is worthless or even dangerous) and in soft-real-time systems that have a low probability to exceed this deadline (after which their output loses value).

Protection functions in the electrical grid belong to the hard-real-time systems. A change in their input (measured value) is reflected as a change in the output (action on circuit breaker or power device) within a given deadline under fault-free conditions. If the output is not produced at the deadline, this indicates a failure and corrective actions are taken. Such a behavior is called deterministic with respect to the deadline. Determinism increases reliability of protection and control since it allows an early detection of malfunctions and better control of the execution.

Many other functions in a substation such as logging, monitoring, file transfer and maintenance belong to the class of soft-real-time systems. These functions are executed on a best-effort basis with a low probability of exceeding the deadline, but nothing special happens if they do. Voice and video surveillance are special since their quality degrades when the delay experiences too much jitter. Figure1 shows the probability that a task completes at time t. The upper, deterministic task always finishes before the deadline, while the lower, not deterministic task has a low probability to exceed the deadline. Variations according to the load have no influence on the concept.

Hard- and soft-real-time functions share the automation system, in particular the processors and the network. The challenge is to provide determinism for the hard-real-time functions in spite of soft-real-time functions that take a significant share of the same resources. To this effect, the hard-real-time functions require resource reservation, which makes deterministic operation costly, and only affordable for the critical data that can endanger the grid if not received in time. This especially applies to substation networks, which are shared by hard-real-time and soft-real-time traffic. On the Process Bus, the bulk of the traffic is deterministic. On the Station Bus, most of the traffic is non-deterministic, except when the Station Bus is used for busbar protection.

The network protocols specified by IEC 61850, allow seamless operation when using the PRP and HSR protocols. They can also provide deterministic delivery if the traffic is properly scheduled thanks to the Power Utility Profile of the Precision Time Protocol IEEE 1588. The devices that provide determinism are backward compatible with the present PRP and HSR devices. The implementation in FPGA is simple.

Hubert Kirrmann

Ioannis Sotiropoulos

by Hubert Kirrmann, Solutil, Switzerland, and Ioannis Sotiropoulos, ABB Switzerland Ltd, Switzerland

Determinism

Train vs. Car: To understand determinism, consider the travel time by train or by car.

■ ***Train:*** A train with a regular schedule is an example of a deterministic system based on a cyclic operation: its timetable. For example, a traveler arrives at the station at a random time to catch a train that departs every 20 mins and which takes exactly 14 mins to reach the destination. Figure 2 expresses the probability to reach the destination before time t (pink).

This probability is 0% below 14 mins and 100% after 34 mins, since in the worst case, a train is leaving when the traveler arrives at the station, but the next train will come within 20 mins, so the bound is 34 mins. This travel time is deterministic with respect to a deadline of 40 mins - under fault-free conditions. But if the deadline is 32 mins, the probability to arrive is only 90% and the travel time is not deterministic. Figure 2 shows also the derivative of the probability (in blue), called the probability density function (pdf), that expresses the probability to arrive between time t and t + δt. In the interval from 14 to 34 mins, the pdf is uniform over these 20 mins and equal to 0.05/min. For instance, the probability to arrive between 21 and 24 mins is 0.05 × 3 = 15%. The pdf is an easier representation to work with. *Note* that the surface under a pdf (its integral) is always equal to 100% - one always arrives after infinite time.

■ ***Car*** is an example of a non-deterministic system. From the parking lot to the destination, driving would take 14 mins (same as the train) with no traffic lights and no other traffic.

With 10 traffic stops with a cycle of 2 mins each, the travel time raises in the worst case to 34 mins. The travel time is deterministic with respect to a deadline of 40 mins. If the traffic light cycles are independent, the resulting pdf is the convolution of 10 pulses of 2 mins each, resulting in the dotted violet line in Figure 3. A shorter travel time is achieved by synchronizing all traffic lights so that the car can ride through (ignoring traffic jams), which requires an overall scheduling.

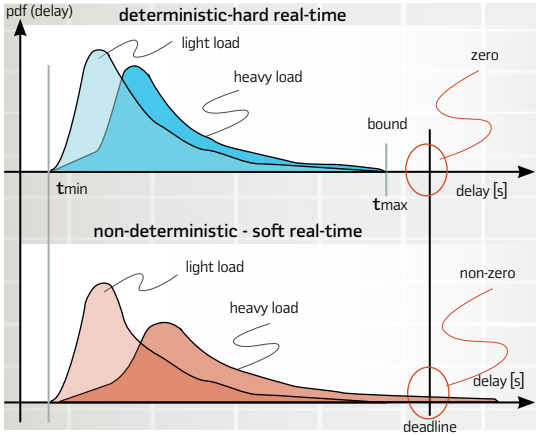
While traffic lights are predictable, traffic jams are not and so neither is the travel time. Figure 3 shows the traffic-dependent pdf as the solid red curve. The probability to reach the destination before 40 mins is only 90%, no bound can be predicted.

This is because the road is a shared resource that is not reserved, anyone uses it on a first-come, first serve basis. But, even though overall car traffic is stochastic, determinism could still be achieved for a high-priority traffic by limiting the other traffic, for instance by closing the highway entrances to second-class vehicles when the traffic nears congestion, i.e. reserving the road time for high-priority traffic. If all traffic is high-priority, there is no remedy except limiting all traffic or building more roads.

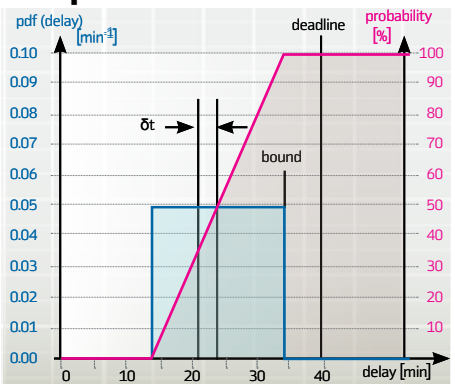
The difference in behavior between deterministic and not deterministic is fundamental and most felt at the destination: a person waiting for the train knows that the traveler will arrive with a specific train or with the next, and if he is not on the expected trains, then something went wrong.

By contrast, a person expecting a car has no special deadline to wait for, at some point, she will lose patience, but it is unclear when, since traffic jams could explain the delay.

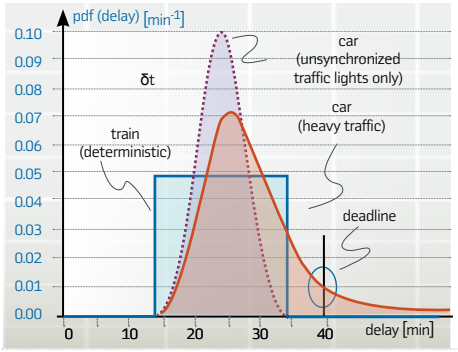
1 Delay probability



2 Delay probability and pdf of train travel time



3 Delay pdf of train, car with and with no traffic



While a train is considered a deterministic transport, this applies only to those who could board the train. When considering the chain of the commuters from the office to the station, the traffic jam may well be in the elevators to the station when everyone wants to squeeze into the train at rush hour.

Determinism with multiple components: A signal processing chain stretches from the sensors to the actors over several devices. An example is a chain composed of two IEDs connected by a Process Bus according to IEC 61850-9-2 (Figure5). IED1 operates with period T1; in each cycle, it reads the value of the instrument transformer, processes it and deposits the result into the communication buffer of the process bus. The process bus operates with a period T2 and deposits the value into the buffer of the IEDs.

The protection IED3's task has period T3; it reads the value from the network buffer, calculates the protection algorithm and acts on the circuit breaker. For instance, all compo-

nents have the same basic execution period, T1=T2=T3=250 µs (4000 Hz). The processing and propagation delays are ignored, since they introduce a fixed delay.

When several unsynchronized transfer functions are connected in series, the overall pdf is the convolution of the individual pdf, assuming that the devices run independently. The pdf resulting of the three cyclic operations is shown in Figure 4. The overall behavior is deterministic with respect to a deadline larger than 750 µs (3 x 250 µs).

Note that this is the same calculation that was used for the traffic lights in series of Figure 3. When the number of independent cycles in series increases, the shape resembles a Gaussian bell - but it is not. When the number of elements in series increases, the peak shifts right and the deadline may be exceeded, so a synchronization of the cycles is required. Since the resulting pdf is the convolution of the pdf of all components, any component that does not behave deterministically ruins determinism. In the case of a protection system, it makes no sense to carefully provide a deterministic network transmission, but to ignore the behavior of the attached IEDs: determinism is a system concept.

Jitter

The system of Figure 4 is deterministic, but not jitter-free. A jitter-free system would exhibit a pdf that is a small, tall pulse. Jitter-free systems require all the components in the chain to be jitter-free and strictly synchronized, which is beyond the requirements of most applications. Jitter-free operation does not request a jitter-free transmission, except for the distribution of time.

A global time such as GPS or IEC/IEEE 61850-9-3 allows to precisely time-stamp the measurements at the source and to resample them with other streams at the destination. For instance, the transmission of Synchrophasor values according to IEC TR 61850-90-5 does not require a jitter-free network transmission. Voice can also be resampled. In the example of Figure 5, the jitter in the sensor to actor chain can be kept small without resampling by synchronizing the execution of the tasks in the IEDs with the bus.

All IEDs must sample the input at the same time and be finished before the bus transmission of their values starts. The receiving IEDs must immediately process the received data and act on the process within time T3. This requires an overall scheduling, not just a network scheduling. The jitter requirements of IEC 61850-5 applied mainly to differential protection that relied on a precise propagation time along a telephone line. With a global synchronization and determinism, jitter-free transmission is not an objective in substation automation networks.

Providing determinism

Non-determinism: Non-determinism is introduced by unpredictable delays, such as: time sharing the processor in event-driven operating systems, enabling interrupts, queued inter-task communication, programming in languages that allow garbage collection or random path selection, using virtual memory and algorithms with no predictable execution time, for instance recursive algorithms. It is then mathematically impossible to prove that the system will always perform, simulations are needed to estimate the probability of success. This is acceptable for many tasks.

Providing deterministic processing: Determinism requires that the delay that every component in the chain introduces is bounded. This requires reserving all resources needed for the operation exclusively for the operation that needs determinism. There resources are e.g. time, processing power, memory or access to peripherals.

The simplest way to achieve deterministic processing time is to pre-allocate a time slot to each time-critical task with a sufficient period.

4 Convolution of 3 components operating with a cycle of 250 µs

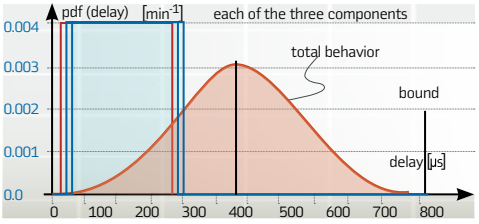


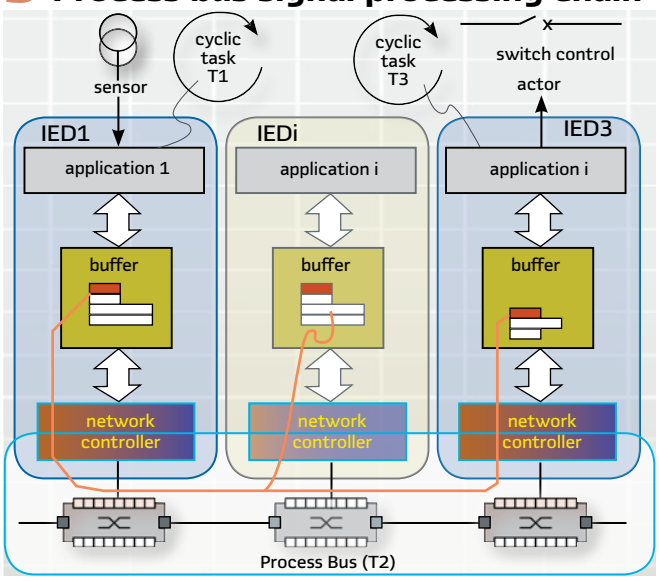
Figure 4: The system on the left is deterministic, but not jitter-free.

All time-critical tasks are executed periodically, whether their execution is needed or not. For instance, although the input variable did not change, the task is nevertheless executed. The execution time of each critical task is bounded. In addition, the critical task uses a dedicated memory partition and never waits on another task, so inter-task communication relies on buffered shared memory rather than message queues.

This is how programmable logic controllers work according to IEC 61131. Programs consist of a fixed quantity of function blocks whose execution time is previously known. The frequency and execution time of a task is fixed. Whoever programs a block in some procedural language is responsible to meet the bound, this requires special testing and validation tools. Non-critical tasks such as download, debugging or display are executed in the remaining time.

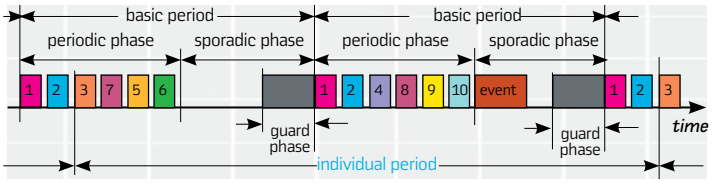
Network partition and priorities: In local area networks, bridges control the traffic with two instruments as specified in IEEE 802.1Q: filtering

5 Process bus signal processing chain



by Hubert Kirrmann, Solutil, Switzerland, and Ioannis Sotiropoulos, ABB Switzerland Ltd, Switzerland

6 Bus operation in a field bus



by VLAN or Multicast domains and priority queues.

Multicast and VLAN filtering are useful to reduce traffic on some network segments and reduce congestion, but they do not solve the fundamental problem: they do not increase bandwidth, they just reduce traffic.

Although priorities are common in data networks, they only increase the probability of high-priority traffic to reach the destination earlier at the expense of lower-priority traffic. However, a low-priority frame could stall the high-priority traffic since once its transmission started, it cannot be interrupted. Even a modest amount of low-priority, long frames can disrupt the transmission of small, high-priority frames. This is a well-known issue in operating systems called priority inversion, responsible for numerous incidents.

Priorities can only give determinism to one, highest priority class when the production rate of this highest priority class is limited and in absence of lower priority traffic. But when priority traffic increases, sending at random high priority frames could well let the delay exceed the deadline. Priorities are useful, but do not replace resource reservation.

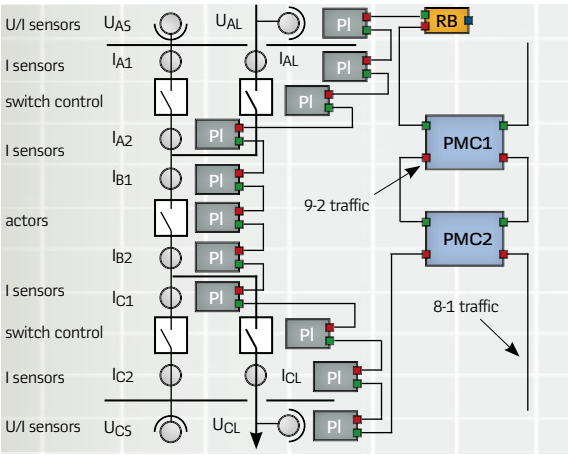
Cyclic bus operation: The most used scheme to reserve network bandwidth is TDM (time division multiplexing). For instance, SDH/SONET applies TDM to all traffic. This made sense for telephony traffic, but this wastes bandwidth when event-driven traffic such as IP traffic is to be carried.

Field busses used in industrial networks support both critical and non-critical traffic. They achieve this by reserving a portion of the bus time for the data that need periodic transmission, called the periodic phase and another portion for event data, called the sporadic phase (Figure 6). Periodic data are always transmitted, whether they changed or not, which at the same time indicates liveliness of the source.

Event data are transmitted in the sporadic phase, when required. The sporadic phase however may not overlap the next periodic phase. So, a guard phase with no new transmission is introduced to ensure that the medium is free when the next periodic phase starts.

As cyclic transmission costs bandwidth, the time-critical variables are transmitted with a period that corresponds to their criticality, usually a power or two multiple of the base period. In Figure 6, variables 1 and 2 have the highest priority, while variables 3 and 4 have second priority and are transmitted every second period. Changing the cycle structure at run-time is a tricky issue, scheduling does not only concern the network.

8 Process bus in 1½ CB topology



Deterministic Process Bus

The process bus is the most demanding network in a substation automation system. It transports the Sampled Values (SV) coming from the instruments. The most extended case is a 1 ½ circuit breaker topology (Figure 8), in which up to 21 devices are connected to the protection and control IEDs (PMC1 and PMC2).

IEC 61869-9 (digital instrument transformers) foresees transmission of voltage and current samples at 4 kHz (for a 50 Hz grid). At 100 Mbit/s, the longest Ethernet frame takes 123 µs to transmit and the sporadic phase must accommodate at least one. When sampling at 4 kHz, this leaves 127 µs for the periodic phase. Since transmission of an SV frame takes some 12 µs, at most 7 SV frames can be sent (see IEC 61850-90-4). The deadline is the beginning of the next cycle, after that, samples start to overtake each other.

If all nodes would transmit at random, no deterministic operation would be possible. To give more room for event messages, several samples can be packed into one SV frame, as IEC 61869-9 foresees, but this does not change the basic limitation. At 100 Mbit/s, the configuration of Figure 8

7 Precision frame sending in HSR (one direction)

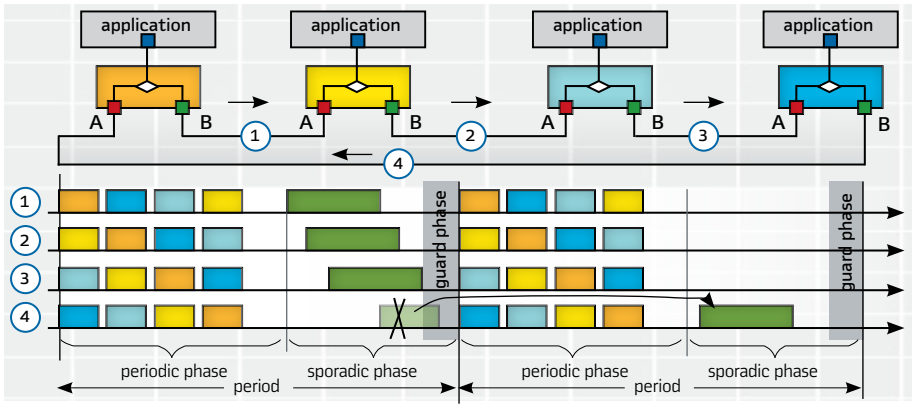


Figure 7: Sending takes place in both directions, but for simplicity, the Figure below shows only one direction.

is only possible with proper scheduling and with a rate of 2400 Hz.

Precision Frame Sending (PFS) in HSR

Periodic phase: HSR (IEC 62439-3) foresees a method for transmitting deterministically sampled values with a cyclic time-division multiplexing. All nodes transmit periodically at the same instant one predefined SV frame prepared in a buffer. Sending takes place in both directions, but for simplicity, Figure 7 shows only one direction.

This method became feasible after the IEC/IEEE 61850-9-3 PTP profile allowed an accuracy better than 1 µs. This is sufficient since the jitter between the nodes must be less than the minimum SV frame duration, which is about 12 µs @ 100 Mbit/s, or 1.2 µs @ 1 Gbit/s.

While a node is transmitting, it receives and buffers the frame from its neighbor, which started at the same time. As soon as it finishes sending its own frame, a node forwards the buffered frame from the neighbor, while it receives the frame of the next neighbor, until all critical frames have been sent.

All SV frames are appended contiguously with minimum inter-frame gap. Their sequence in the cycle varies depending on the link, but a node always sees the same sequence under error-free conditions.

SV frames do not need to have the same length, but there shall be no gaps in the periodic phase. This means that all nodes participate in the round, even nodes that do not have SV to send, such as PMC1 in Figure 8.

These nodes send a dummy SV frame or a GOOSE frame with the same priority as the SV frames. This also allows sending the circuit breaker commands with the shortest possible time delay.

Nodes keep on sending while their input buffer contains high-priority frames. After that, they can transmit their next priority data, e.g. GOOSE, voice or MMS, or forward such on the ring.

table 1 FPGA usage

Function	Logical Units (LUT)	Memory (BRAM)
PTP	44%	2%
HSR	36%	82%
PES	20%	16%

Sporadic phase and guard phase: The HSR network must be free of traffic when the periodic phase starts. This is achieved by the guard phase. The HSR node withholds a frame that would overlap the start of the next periodic phase until the next cycle. At 100 Mbit/s, the longest frame takes 123 µs and with a 4 kHz transmission, about half of the bandwidth would need to be reserved for the guard phase.

To improve on this, HSR takes advantage that the length of the frame is announced in the HSR tag. Therefore, the logic decides if there is sufficient time for transmission of an own or foreign frame before the start of the next period and the guard phase is reduced to the size of the frame to be sent, provided it does not overlap with the start of the periodic phase. To recover from an abnormal situation, a receiving node is resilient to an overlap.

The network engineer ensures that there is room to send the longest Ethernet frame in the sporadic phase, by limiting the rate of the sampled values or the number of devices in the ring.

Considering both directions: The prior considerations showed one direction in the ring, but HSR sends the frames in both directions. The Figure on page 38 shows the whole traffic, assuming all nodes send SV frames. The order of reception is different in both directions, but under error-free conditions, a node always receives the SV frames in the same order, so the jitter is small. Mode X of HSR increases bandwidth and refrains sending a frame that duplicates a frame already received from the opposite direction. This bandwidth optimization is not applicable in the periodic phase since it would introduce gaps.

A deterministic system can be mathematically verified, in contrast to a non-deterministic system, for which only a probability of success can be given.

Other Topologies: The principle is not limited to ring topologies. HSR can also be used in a meshed topology. The scheme only requires that the whole network be synchronized and that all nodes observe the same guard phase.

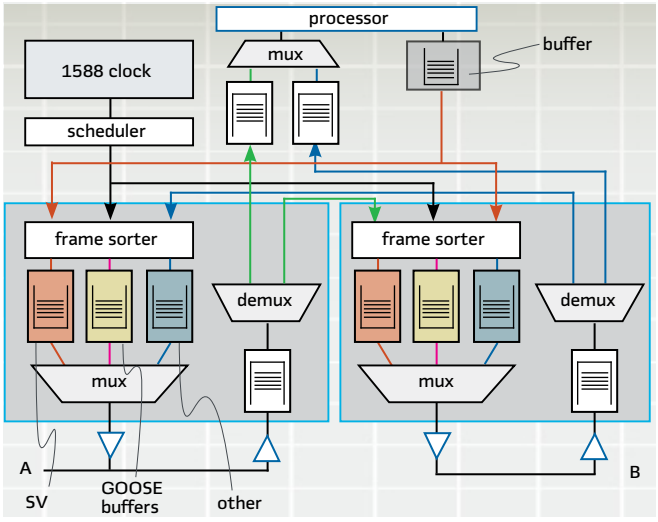
Application Synchronization: The HSR clock sends a signal to synchronize the sampling of the sensors ahead of the bus cycle start, and starts the protection task to act timely on the circuit breaker.

Implementation

The logic for deterministic operation is straightforward once the PTP clock is implemented. The implementation done by ABB and the Swiss Federal Institute of Technology (EPFL) considers two high-priority traffics: Sampled Values at a variable period and GOOSE, which is transmitted in the sporadic phase with a high priority. To this effect, the SV, GOOSE and soft-real-time traffics are separated into three buffers, depending on the 802.1Q tag as Figure 9 shows. The time-critical SV frames wait in the buffer until the clock tick comes, and then are sent in both directions. GOOSE and soft-real-time messages are withheld to observe the guard phase. The usage of the FPGA is indicated in Table 1.

Determinism requires limiting the production rate to the throughput that the weakest link in the chain can process.

9 Principle circuit of an HSR node



Conclusions

1) Determinism is defined with respect to a given deadline and under error-free conditions
2) Determinism is a Boolean property. Making a system faster may be necessary to achieve determinism but it does not make it "more deterministic"
3) A deterministic system can be mathematically verified, in contrast to a non-deterministic system, for which only a probability of success can be given
4) Deterministic components are necessary, but not sufficient for the determinism of the whole chain
5) Determinism does not imply strict cyclic operation, although this is the easiest way to implement it
6) Determinism requires limiting the production rate to the throughput that the weakest link in the chain can process
7) The current specifications of HSR (IEC 62439-3) in conjunction with the precise clock synchronization (IEC 62439-3 Annex C and IEC/IEEE 61850-9-3) allow deterministic process bus operation with a fixed, application-dependent cycle
8) Deterministic HSR is implemented with the standard IEEE 802.3 technology
9) Deterministic HSR is backward compatible, but to benefit from deterministic operation, all devices must implement it
10) Deterministic HSR as a side effect provides optimization of the bandwidth for sporadic messages
11) The implementation along HSR and the PTP clock only costs a modest fraction of the FPGA.