Advancing Functional Safety Using Silicon-Proven Programmable Logic IP

by Keith Guidry and Scott Winning, Sawblade Ventures, LLC

Introduction

The world of electronics is growing at an astronomical rate. With the advancement of Artificial Intelligence (AI), electronic devices are becoming more intelligent than many of its users. Electronics are in every part of our lives – our healthcare components, communication devices, interactive entertainment, social devices. Automobiles evolving toward automated transportation have many distributed networked computers in every vehicle. Future electronics will expand this complexity by unknowable magnitudes with Internet of Things (IoT) bringing unpredictable safety related vectors.

Security measures to verify proper boot-up, proper configuration, authorized use, and data protection offer no assurance the data passing through the electronics is proper beyond the mechanics of 'trust'. This means a trustworthy chip can operate in a verifiable way with no outward indicated problem while improper software running through the chip creates potential risk to property and lives.

Description

The traditional definition of functional safety is intended to provide freedom from the risk of damage to people and property by uncontrolled automated processes. By modern definition, any improper operation for any reason should result in a predetermined failure mode actively rendering the controlled machine safe and not a threat of damage to people or property.

Traditional regimes were faced with foreseeable modes of improper operation. This is no longer sufficient due diligence. Increasingly distributed and interoperable components force chip designers and software creators to reach for higher levels of functional awareness beyond the designed-for potential "accident".

More effective methods are critical to discover unknown failure modes built into the design but missed in pre and post production testing.

Sawblade Ventures (SbV) offers tools and components able to address this pressing and growing change in reliability and safety. Instrumentation injected into the Hardware Description Language (HDL) of microelectronic design can be employed as an effective means of overwatch throughout the chip's lifetime.

SbV components can monitor and modify chip behavior under any operation for any set of granular circuit function. Targeted hardware overwatch is the only reliable means of protecting from unpredictable improper operation.

State of the Industry

The integration of electronic components into automated machines must be designed per specification to be qualified as operationally trustworthy. Specifications guiding engineering discipline for functional safety are fundamental design practice guidelines. The criteria for safe functional design in industrial electronic systems (IEC 61508) [1] are reinforced by vertical disciplines for critical operating systems such as automobiles (ISO 26262) [2], railway equipment (EN 50128) [3], medical devices and equipment (IEC 62304) [4], and nuclear facilities (IEC 61513) [5].

These guidelines depend on acquired engineering experience in each field to create safe systems. Compliance with these and improved guidelines does not guarantee functional safety. Such compliance only gives an assurance that reasonable step-wise development is followed to achieve a planned result.

More importantly, the use of these specifications cannot address intentionally injected error by adversarial operation that security systems fail to identify and stop.

The most publicized historical example for this kind of unpredictable functional safety danger was revealed in 2010 when the world learned of the Stuxnet worm. Surreptitiously inserted firmware values allowed machinery motor controllers to exceed safe limits. The damage from Stuxnet was made more severe because the microelectronics controlling the motors appeared to be working properly. Nothing was "wrong" other than the never-exceed limits were replaced and the motors revved to destructive speed profiles while machine operators watched displays that showed everything was working properly.

SbV has documented how its solutions could have seen the Stuxnet worm at work and applied autonomous pro-active counter-measures to prevent the damage.

Traditional Solutions

While modern methods have evolved in attempts to safeguard functional designs, system complexity has slowed the builder's ability to resolve new methods.

Built-in-self-test (BIST) [6] hardware designs rely on engineering skill acting on pre-production test analysis to anticipate unsafe states in the microelectronic operating blocks.

Dual-redundant processors (lock-step) [7] run the same software and sample the same input signals to look for processor error.

Triple-redundant data storage registers (TMR) [8] store three versions of the same data. Error is discovered if one of the processes is outvoted by the other two, correcting the error per the vote result.

Error Correcting Code (ECC) and Error Detection and Correction (EDAC) [9] are programmatic error resolution methods.

SbV instrumentation can build on these known safeguards, augmenting them to be more robust and more informative.

SbV augmentation and advancement of these basic approaches are needed to begin resolving issues beyond the reach of these legacy solutions.

The modern chip/code combination presents a risk posture that is so overwhelming, most industries today run on what the Defense Advanced Research Program Agency (DARPA) System Security Integrated Through Hardware and Firmware (SSITH) [10] initiative calls a "Patch and Pray" regime.

Sawblade Resolution

Chips must be made more self-aware. They must become autonomous arbiters of their own operating states and response profiles independent of or at least in concert with software. SbV patented integrated hardware solutions offer a way to outfit any circuit within simple or complex designs with an instrumentation fabric to monitor and modify circuit behavior. (Figure 1)

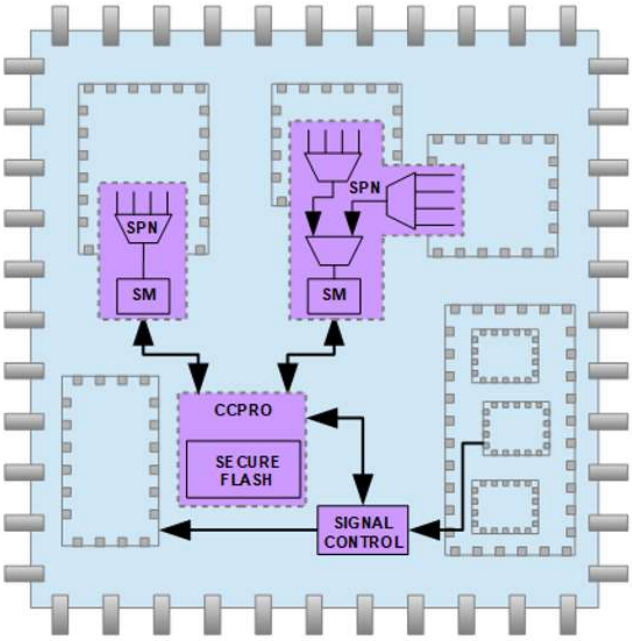

Figure 1: SbV Fabric Installed on Typical Host Circuit

SbV's segregated formal approach provides a more flexible, reprogrammable means of adapting chip operation to code intent. The ability to autonomously or cooperatively resolve granular internal chip anomalies are required if automated and intelligent systems of the future are to adequately inform system overwatch.

SbV's reprogrammable hardware instrumentation embedded deep inside the microelectronic system of an automated machine's hardware has the benefit of observing activities not seen by software. At the same time, such instrumentation can enforce rules and profiles that cannot be altered by defective or malicious software. The implanted hardware is not visible to the software application running through the chip or the operating system supporting the application with machine-level instructions. An embedded network of instruments can detect and deter attacks that software cannot touch or affect.

As the microelectronics builder is only responsible for shipping a functionally accurate product and as the software builder is only responsible for shipping error free code, the system builder who inherits chips and codes becomes responsible for attempting to protect the resulting system against unpredictable action and operation. The system builder should be given a way of watching chip operation modes and methods within the system design.

Sawblade EDA (Electronic Design Automation)
SbV's tools and intellectual property embed hardware monitors and active controls (components) deep in the silicon of any chip-based system. Figure 1 shows a view of this host/component relationship. The host circuit design is outfitted with parasitic automation capable of monitoring signals and injecting control signals from an operationally segregated set of components.

SbV tool and component properties for design, verification and operation offer a formal, segregated engineering domain separate from but seamless within all other traditional EDA tool workflows. Domain segregation means functional safety teams may create, modify, extend, and delete overwatch capabilities as needed during host circuit design phases without interfering with the ongoing design process.

SbV on-chip networks work to inform and control distributed components as a unified fabric. Multiple fabrics may be selectively bridged across separate networks to provide multi-purpose sampling, qualifying, controlling and supervisory structures. These give a beneficially parasitic augmentation to the original host circuit design as needed per functional safety scenario. Fabrics may also be networked across chip and system boundaries to augment existing safety and security functions. This approach reinforces existing functional safety regimes, giving added insight into machine states in real-time at silicon speed across multiple time domains.

Instrumentation and information structures built with SbV EDA tools and components may be reused

and repurposed from one design/test/operate phase to the next without impacting design criteria or performance of the underlying host. These structures can be removed before production or used in post-production information/control scenarios throughout the host circuit lifecycle. They offer a way to maintain verifiable functional safety information and action processes throughout the chip's life without altering the original host design or performance. When needed, they offer a way to interdict error and improper operation without complicating the original host design.

The resulting active information fabric allows reprogrammable profiles of whitelisting and blacklisting requirements for individually specified limit values. These hardware instruments can report from within the chip to inform system software of real-time at-speed operating conditions and assertions. They can also perform autonomous mitigation to supervise host circuits and other system controls to fail in a predictable safe state.

This method allows for maximum flexibility in creating non-invasive functional safety related scenarios in the design process where traditional BIST tools and techniques may introduce undesired interference within the design under test (DUT) circuit. SbV instrumentation operates independently from the processor circuit and any software running through the processor or in other sections of the system. The installed components are also hardware obfuscated by injection methods and by compounded fabric design.

The programmable monitors and controls are inserted during the design phase as components of a formal test plan. The monitor and control programs to be loaded into the instruments are also designed during the design phase, but new programs can be designed at any point in the product lifecycle to enhance functionality, improve coverage, or adapt to new and unexpected issues.

Fabric monitoring can extend into software by enabling a constant challenge response handshake to ensure continuous authority, and to exchange performance and behavior metrics as derived from within the hardware. Software can likewise command instrumented controls to alter circuit functionality and configurations in response to test parameters, real-world real-time faults and anomalies.

In this way, the functional testing regime can seamlessly extend the information quality spectrum from conceptual prototyping, formal design, formal qualification, and operational life adaptable to deal with unexpected and unpredictable findings.

SbV components can also retrace host circuitry, reconfiguring signal flow to repair and repurpose circuit wiring as desired on host or fabric circuits at speed. Routing instruments and installed spare traces can augment and modify any circuit function with signal personality that cannot be touched by software methods or hardware reverse-engineering tampering methods. In this way, functional safety regimes can proactively and selectively apply a wide range of pro-active hardware options to address unforeseeable operational requirements without resorting to a physical chip revision cycle.

Future-proofing is a significant requirement for modern functional safety analysis and operation. SbV's patented methods ensure the functional safety engineering design team has unique options to pro-actively meet challenges of any kind without having to return to the drawing board.

Sawblade Example

The fact that programmable active monitors can be repurposed at any time is crucial to scenario testing during DFMEA (Design Failure Mode and Effect Analysis) and FMEDA (Failure Modes Effects and Diagnostics Analysis) campaigns. Monitors can be programmed in-system to perform different

functions over time. By time-slicing functionality, a wider range of behaviors can be detected using the same resources.

Real versus ideal scenarios run on real-world in-operation hardware can be carried out avoiding the complexity, cost and inaccuracy associated with emulation or simulation. This real-time feedback from in-use circuits provides functional safety engineering with high fidelity metrics that cannot be achieved any other way.

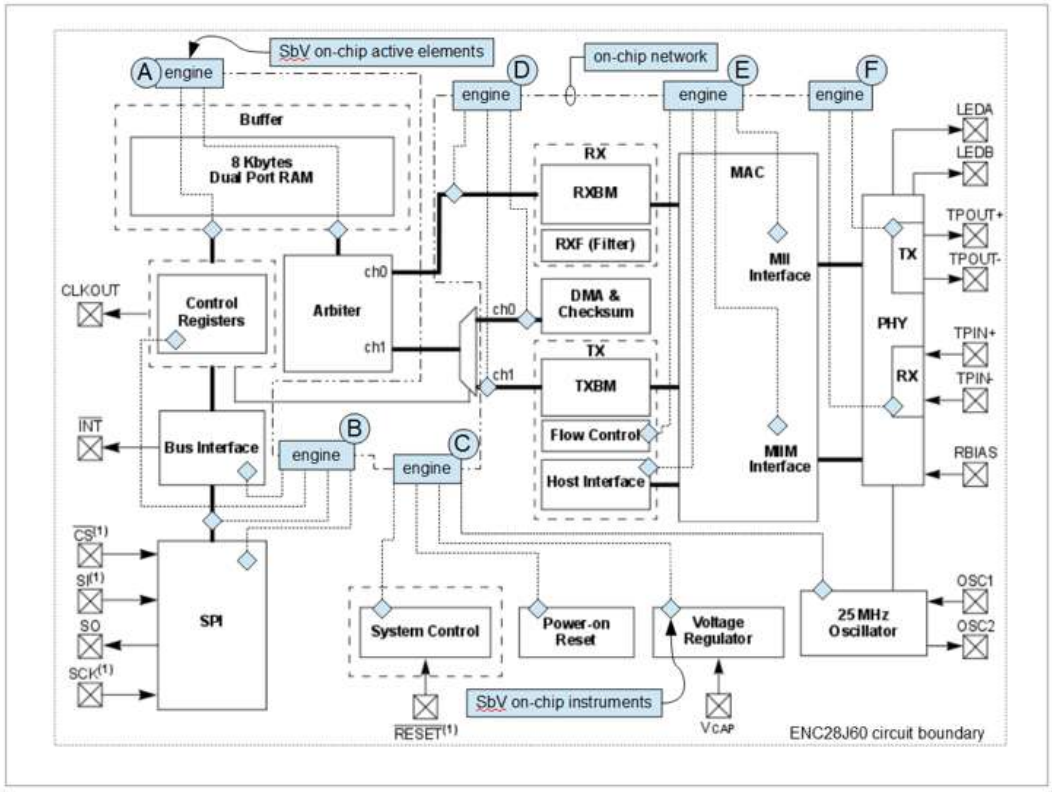

Figure 2: SbV Instrumentation Fabric Installed in Ethernet Controller Circuit

As low and high speed serial communication paths within modern operate-by-wire machinery are a necessary but complicating link in distributed functional safety concerns, Figure 2 shows an ENC28J20 ethernet controller selected at random with a hypothetical SbV fabric.

Figure 3 offers a reference legend for this example.

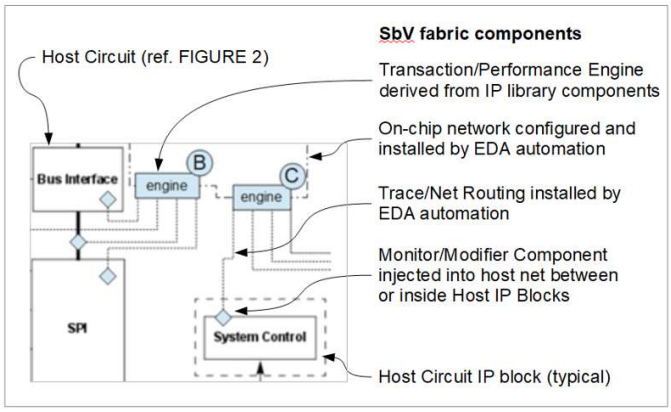

Figure 3: Legend Reference for Figure 2

SbV transaction and performance engines derived from finite state machines, trace routers and dynamic monitor/modifier signal attachments are used to actively sample signal nets of the host circuit. These host nets may be located inside the functional blocks or between functional blocks.

SbV engines can communicate with each other using SbV on-chip networks to act as a single fabric. Multiple networks may be installed to create segregated instrumentation fabrics.

Network bridges can connect multiple fabrics which can be used in various combinations.

Each engine can be outfitted with independent memory (not shown) to capture signals and to provide signal data for stimulus. Memory may be volatile or non-volatile as needed to provide state persistence during powered or between
powered instances.

In this example at Figure 2, engine A is sampling the data buses entering and exiting the host circuit's random access memory (RAM). Engine A can test data passing across each bus in conjunction with signal assertions derived by any of the other engines across the on-chip SbV network. In a monitor-only mode, the RAM bus signals can be used to qualify control scenarios within any other engines. In an active mode, each signal tap can be configured as a dynamic wrap which is injected into the host HDL to provide a means of stimulating a dynamic signal to replace the host's existing operational signal. In this way, the control and data buses in and out of the RAM may be monitored for proper data transfer and content as well as providing a means for the SbV fabric to change data content as needed at speed.

Engine B is installed here to monitor control registers, bus interface and serial peripheral interface (SPI) signals. These may likewise monitor nets within each functionality block as well as monitoring/modifying individual control lines and bused signals between blocks. Anomalies in circuit operations per sampled assertions can trigger other engines to activate measures while maintaining a per-block segregation of signals and active elements.

Engine C is watching chip-wide resource elements such as reset actions, power states and clock speed. This method allows host chip operation to be dynamically controlled by stopping/stepping clock operations and, if required, destroying the chip using overclocking and power manipulation. This is an important capability for highly sensitive equipment falling into improper hands.

Engine D is tasked with overwatch of the bus values passing through the transmit/receive controls. Data packets may be stored in engine memory for use in monitoring specific data pattern assertions and for dynamic injection per sensed asserted conditions.

Engine E may independently oversee control of incoming and outgoing ethernet packets from within the various flow control, host interface and medium access control (MAC) functions. This can include secret hardware values used to replace the ethernet module's MAC serial number to achieve various communications using the same ethernet port without disrupting the priority ethernet use.

Engine F is able to watch transmitted and received data streams in conjunction with all other networked engines to ensure data streams and controller functionality are working in proper context per profile.

This fabric method allows comprehensive overwatch of ethernet packet content delivery using an augmentation of each set of legacy functions. Anomalies may be dealt with by direct injection of dynamic signals replacing erroneous command and data as found at speed. The fabric can reinforce and extend a legacy circuit into a robust self-aware functional device ready for the future unknown.

Should host circuits require reconfiguration, repair and/or repurposing, SbV fabric automation can install spare trace elements and reconfiguration engines to alter the host circuit wiring as desired. This may be applied to individual command and data traces as well as buses of any size.

Figure 4 illustrates this reconfigurable digital instrumentation (ReDI) capability using formal

automation tooling in GUI form. Command line interface (CLI) scripting may also be used to achieve these reconfiguration scenarios programmatically on-the-fly to alter the circuit capabilities without resorting to a redesign of the host circuit to repurpose functionality throughout the host circuit's lifecycle.

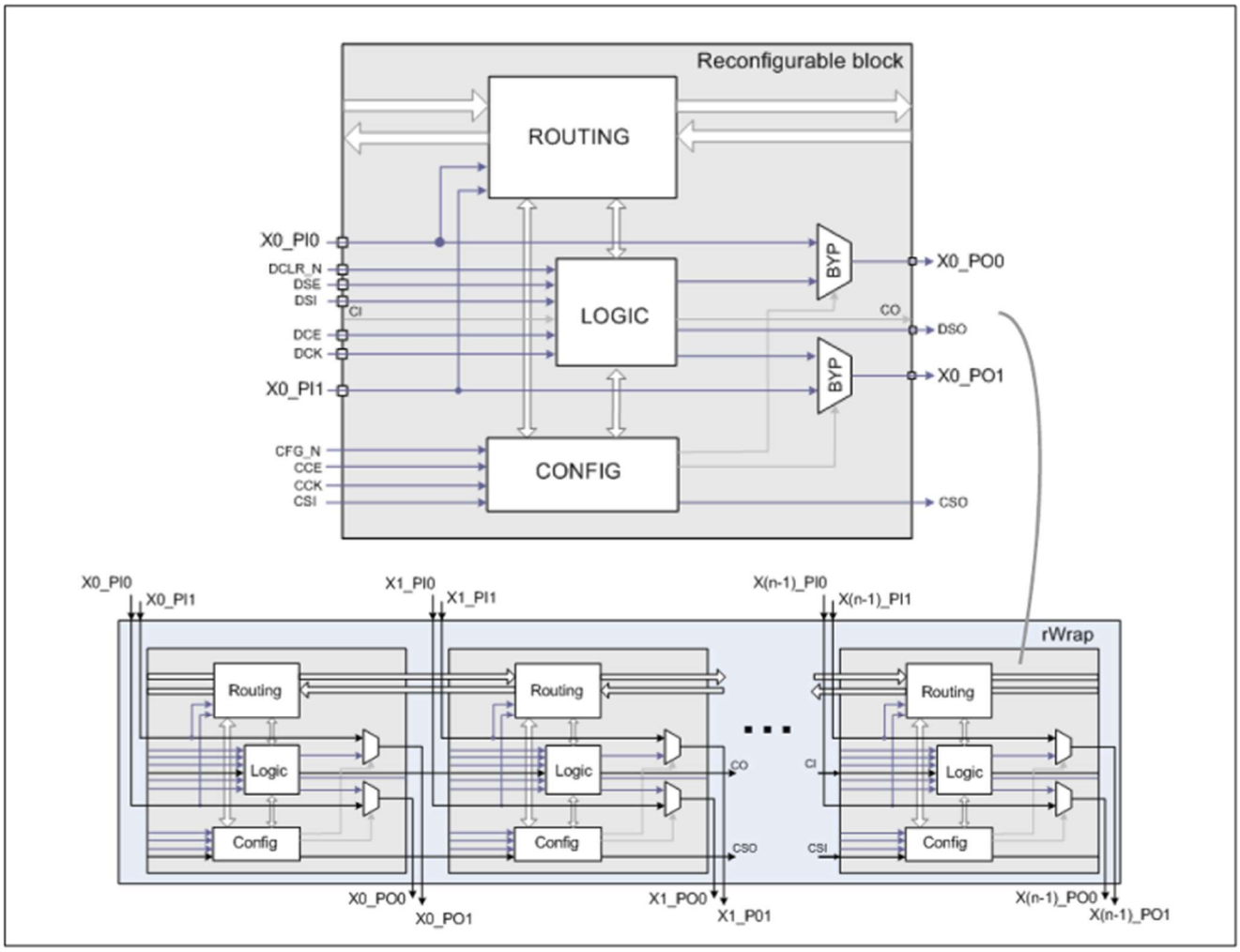

Figure 4: SbV GUI representation for circuit routing reconfiguration

These unique patented capabilities create multiple ways to monitor and modify the host circuit without breaking circuit design boundaries, compromising or complicating host circuit design and performance.

Summary

Engineering discipline alone cannot be expected to cover the range of uncertainty created by the expanding use of interconnected and interoperable distributed electronic controls. Traditional solutions are increasingly insufficient when faced with the functional safety unknowns arising from system complexity and vulnerable attack points. Instrumentation can be applied to oversee the host chip operations, providing system builders and operators a way to analyze functional processes under real-world operations. SbV tools can offer considerable breathing room for engineering disciplines challenged by the semiconductor industry's rapid change.

References

[1] IEC 61508 http://www.iec.ch/functionalsafety/explained/
[2] ISO 26262 https://www.iso.org/standard/43464.html
[3] EN 50128
http://profs.etsmtl.ca/claporte/English/Enseignement/CMU_SQA/Notes/Normes/Standard_IEC_EN_50128_Software_for_Railway_control.pdf
[4] IEC 62304 https://www.iso.org/obp/ui/#iso:std:iec:62304:ed-1:v1:en
[5] IEC 61513 https://iecetech.org/issue/2017-06/Improving-safety-and-reliability-in-process-industry-plants
https://webstore.iec.ch/publication/5532
[6] BIST http://eecs.ceas.uc.edu/~jonewb/BIST2.pdf
[7] Lockstep processing https://en.wikipedia.org/wiki/Lockstep_(computing)
[8] Triple mode redundancy (TMR) https://en.wikipedia.org/wiki/Triple_modular_redundancy
[9] ECC & EDAC https://en.wikipedia.org/wiki/Error_detection_and_correction
[10] SSITH http://mil-embedded.com/news/security-at-the-hardware-level-is-the-goal-of-darpas-ssith-program/

Contact:
Keith Guidry CTO – keith@sawbladeventures.com
Sawblade Ventures, LLC
6001 William Cannon Drive
Ste #203A
Austin, Texas 78749