



Intel® FPGAs in Smart Factory – A Short Case Study

Authors Introduction

Dr. Muataz Hameed Al-Doori

Industry 4.0 and Intelligent
Automation Lead
Flex, Penang
Muataz.Aldoori@flex.com

Dr. Mark Jervis

Industrial and Automotive Architect
Intel® Corporation
Programmable Solutions Group
mark.jervis@intel.com

Mr. Takayuki Ikushima

Market Development Director, Industrial,
and Automotive Business Unit
Intel Corporation
Programmable Solutions Group
takayuki.ikushima@intel.com

Industrial manufacturing is undergoing its 4th revolution, so called Industry 4.0 (I4.0), with digitization and connectivity enabling the path beyond smart manufacturing towards intelligent factories defined by greater machine automation and greater agility; where data is transforming business. The Industrial Internet of Things (IIoT) is extending the benefits seen in the transformation of information technology (IT) to operational technology (OT); adding intelligence to manufacturing equipment, processes, and management. Smart manufacturing solutions use connected sensors and devices at the network edge to improve machine and human performance in real-time, and pass data to the on-premise or cloud server for deeper analysis and insights.

The benefits of this transformation include;

- Increased productivity and factory throughput
- Increased production quality
- Higher levels of reliability and uptime
- Improved predictive maintenance and model development
- New levels of flexibility, agility, and automation that improve system management

Intel has a legacy of working with industrial manufacturers from cloud to endpoints and a portfolio of products that help achieve workload consolidation, manage network traffic growth, and power the transformations that are making Industry 4.0 a reality.

This white paper explains an innovative approach Flex took to achieve Industry 4.0 transformation for its Surface-Mount Technology (SMT) line based on Intel® FPGAs.

Transforming the Production Line

The goal of the infrastructure transformation was to improve SMT line productivity, improve line flexibility, simplify the machine changeover process, reduce machine downtime, improve product quality, and raise utilization. The vision was to design a system element that was portable, scalable, and secure, giving each SMT machine intelligence that is robust, efficient, and accurate. System complexity would be lowered by consolidating new workloads and existing workloads currently done across many different hardware systems onto one configurable and customisable platform. This would require a platform capable of fast data computing with precise, real-time decision-making processes.

Originally, the SMT line consisted of legacy machines with traditional controllers, such as PLCs and industrial PCs running software machine handlers. These platforms were unable to fulfil the forward-looking requirements for dynamic and efficient decision making and fast computing. Furthermore, these machines were not fully networked, and not suitable to be connected to collect data and create insights from it to drive productivity and efficiency gains.

Table of Contents

Introduction	1
Transforming the Production Line ...	1
How the FPGA Overcomes the Barriers to Break-Through Performance	4
Conclusion	5

Two significant challenges confronted these existing platforms: the synthesis nature of the decision-making operation and data computing latency. The compound process of data gathering, filtering, and handling leads to a complex decision-making process. Also, signal synchronization, computing engine architecture, and limited computing power (operating frequency and throughput) in these platforms causes a marked latency in data processing. Flex sought to simplify the system architecture and data processing while improving the system’s real-time capabilities. The ability to consolidate and centralise compute for the SMT processes was considered a key factor. However, this itself brought further challenges. The first challenge is that consolidated real-time processing would require true parallelism. The second challenge is that together with connectivity came the need for strong cybersecurity.

One significant challenge in designing a system is managing ordering dependencies between multiple real-time computations. Dependences can arise in many ways in multiple contexts. For example, a dependency can occur between two processes at coarse level when one process waits for a message or signal to arrive from another process, such as sending width adjustment control bytes to conveyors after the Wi-Fi connectivity handshaking signal. Dependency

also arise in the fine-grain read-and-write operations (traditional Read-After-Write, Write-After-Write and Write-After-Read).

Managing dependencies is key, as dependencies can not only limit parallel performance, but also affect correctness. For example, a data dependence that crosses cybersecurity layers boundary creates a need to synchronize or communicate between the layers. Analyzing where data dependencies exist, gives an understanding of the consequences for parallelism and correctness even there is no prior mapping of what aspect of the computation caused the sequence relation in the first place.

Industrial architects have sought to overcome such problems using computing platforms such as industrial PCs, microprocessors and microcontrollers, or general-purpose graphic processors. While these platforms are accurate and very effective in certain tasks, they did not fulfil the requirements for deterministic low-latency compute with enough performance within the power consumption budget. Instead a platform was needed that had efficient concurrent and parallel architecture capable of hard real-time, high-performance compute; sufficiently low power, with I/O flexibility for the connectivity and the ability to integrate strong cyber-security to protect it.

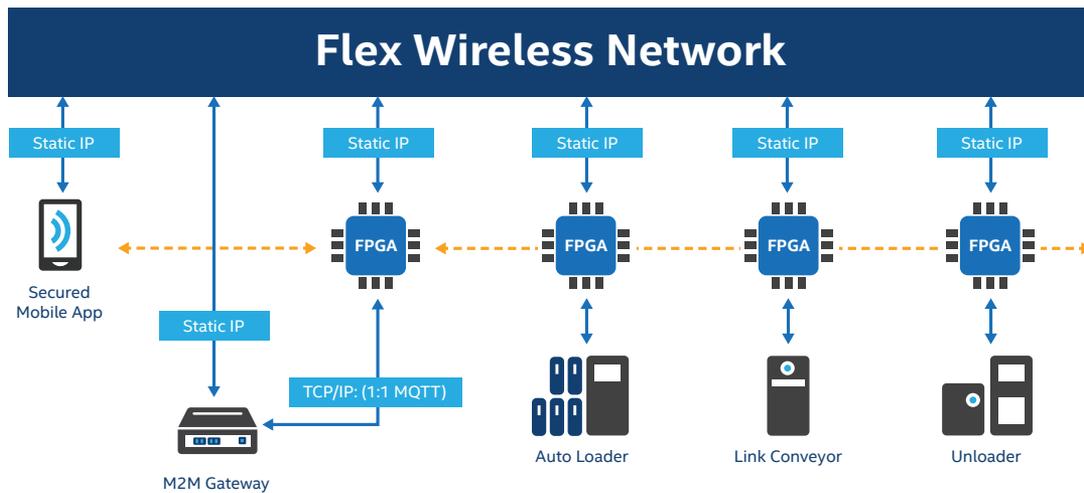


Figure 1. Two Layers of XANDERs Connectivity

Innovative Approach

Flex’s solution was to partner with Intel to develop an intelligent “machine brain” (“XANDER”) on their FPGA platform. In the systems architecture, two layers of XANDERs are integrated in the SMT line in a decentralized manner. Each end-point machine (e.g. auto-loader, link conveyor, unloader, etc.) has a corresponding XANDER that communicates automatically east-west with others in the layer and north with an SMT-line master as required. See Figure 1. The organization is flexible and decision-making largely localised.

This architecture reduces the communication bandwidth with the master FPGA system, freeing processing space for higher level connectivity with other SMT lines in production at the

master level. There are dozens of XANDERs installed in Flex SMT lines to manipulate and collect sensor data in real time securely. All have wireless connectivity with the line gateway connected to the Flex internal network. Figure 2 and Figure 3 depict the top-level design and architectural hierarchy of the XANDER.

Parallelism is a key factor in amalgamating efficiency and performance in embedded design as is clear in the XANDER top-level architecture. This combines two Reconfigurable Embedded Processor (REP) computing engines, four filter lines, embedded memories, address generators, an input unit manager, emulator, and clock generator as well as a Wi-Fi controller and other control subsystems all on a single FPGA chip. These main design units are outlined as follow:

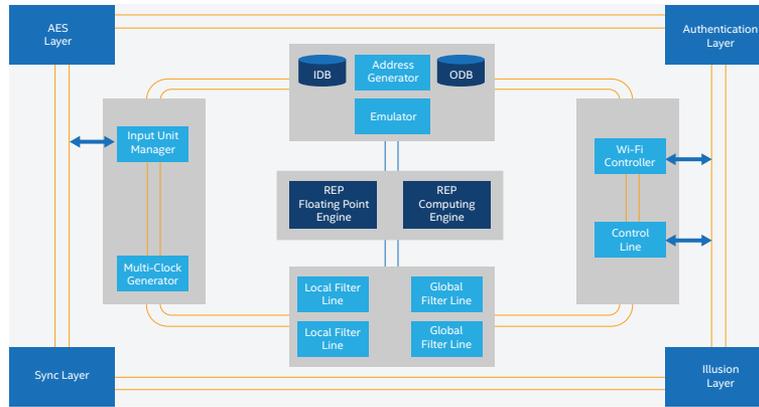


Figure 2. Top-Level Design of XANDER

Input Unit Manager: The FPGA has enough on-chip memory to buffer the incoming high-bandwidth data transmissions, which allows for efficient embedded processing. This accelerates computation compared to using off-chip memory. Dual port memories are configured to allow concurrent read/write operations to pre-process a wide range of incoming data formats received from different machines built by different manufacturers across the SMT line. The input can be ASCII data, HEX data, decimal all with different packet sizes and formats. This also must be cleaned to filter out data unnecessary for production operation while using minimal memory, then passed on in a standardized format ready for processing.

Embedded Parallel Systolic Filters: The spatial compute structure of the FPGA is ideally suited to the implementation of pipelined filter substages. A controller orchestrates execution on these functional units in a sequence of pipelined and clocked steps, ensuring all steps to have scheduled time slots within an allowed jitter time. High computational efficiency can be achieved with concurrent compute pipelined through these functional units on every timestep, taking less cycles overall, while also using minimal FPGA resource area. Each synchronous pipeline has several computing stages, each performing discrete tasks with embedded registers storing partial results. Potential data and control conflicts can be overcome using a flag bit assigned to each stage to indicate its status; for example, either idle or still processing data.

Signal emulator: This is a standalone unit implemented within the XANDER that can emulate the signal and control words from various factory machines. With this functionality, XANDER can be unit-tested in isolation, to diagnose any system issues and provide assurance of performance and efficiency.

Control sub-system: This issues commands to other connected machines. Commands are complex control words sent to the machines to instruct them to perform various tasks and process operations, such as width adjustment, auto mode flow, enabling, and disabling. The command words generally consist of several ASCII bytes with different words size related to process functionality.

The main core consists of two REPs, which are separately responsible for floating-point and fixed-point task processing. These custom task-processing compute engines autonomously carry out common required tasks, such as calibrating the board width adjustment.

The REP engine comprises of a dual-issue computing unit and a set of pipelined functional units that are configurable to form many embedded custom processes. The REP consists of two stages of dual-issue task fetch and decode and load/store line, task and data memory and a timer. The task unit can fetch, decode, and issue two tasks per cycle to be executed by the pipelined functional units. These units make extensive use of threads and loops around small task sets to carry out local sub-tasks. With this architecture, a large compute bandwidth is achieved in relatively small memories.

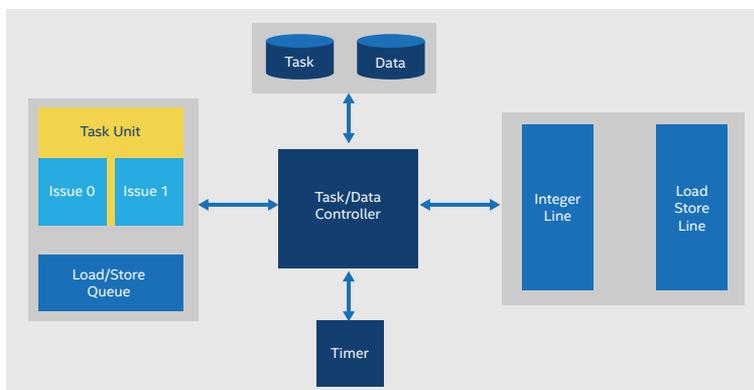


Figure 3. REP Architecture

These compute pipeline functional units can be reconfigured to implement many different tasks, making this scalable to any SMT line, regardless of the brand of the machine.

Alongside the immediate control functions, the FPGAs provide capacity to include analytics to assess and monitor each machine's condition, adding the further benefit of reducing downtime and improving productivity through predictive maintenance. The machine learning solution consists of both supervised and unsupervised approaches. Algorithms such as Extremely Randomized Trees (ERT) and Decision Trees (DT) are used together to provide forward prediction of machine health with a very high accuracy of 99.2%¹. To achieve this, large data sets, with over a million data points are captured and used for training and test in a ratio of 80% data for training and 20% for test.

The result of the analytics is a view of the state of the production line components over a range of timescales.

In the shorter-term timescales, the Machine-to-Machine (M2M) Intel Gateway uses edge analytics via the Machine Brain to understand immediate productivity, efficiency, and defect rates; and trigger actions such as stopping the conveyor upon detecting continuous defects, or stopping and calling an engineer if the pick-and-place (PnP) rate drops below 98%.

In the longer-term timescales, analytics run in software highlights anomalies, provides trends and predicts individual machine performance and health predicted forwards by a few weeks. This gives the plant manager the information to address machine maintenance ahead of issues developing.

The architecture of the system has also improved the update time for the SMT line. Previously updating analytics was a manual operation taking of the order of 20 minutes; for example, in updating the models used in product classification. It is now a push-button process that can be sequentially rolled out down the line in 2.5 minutes. Furthermore, the FPGA can store and switch between multiple models concurrently, meaning this can occur with no idle while the SMT line is in full production.

How the FPGA Overcomes the Barriers to Break-Through Performance

Flex chose to implement each Machine Brain on a Cyclone® IV FPGA. This method was capable of meeting the tough compute and operating requirements, including low power, with proven reliability and long lifetime, and with efficient cost.

FPGAs provide a truly parallel simultaneous computing architecture on which to achieve the low latency and deterministic control functionality required for high-speed production lines like SMT. The critical key performance indicators in choosing Cyclone IV FPGA were data ingest, synchronization and processing capability. The platform had to be capable of handling a massive amount of parallel data processing; across the input unit manager, the filtering modules, the reconfigurable embedded processor, the data buffering, the control subsystem, the Wi-Fi controller - all computing reliably in microseconds and with device and communication integrity protected. These could all be

implemented on the Cyclone IV FPGA while also balancing cost and power.

As mentioned above, both coarse and fine granularity of the system components are key in managing dependency constraints. FPGAs offer a deterministic platform with extremely low jitter, and with a cycle-by-cycle execution that allows precise control of data. On this, Flex could implement the required architectural sub-modules and tie the computation to the carefully parameterized hardware's resources to meet the design's specific needs.

Cybersecurity is another significant concern in connected smart manufacturing. Security attacks are a real credible threat, as demonstrated by attacks such as Stuxnet, Triton, and Industroyer. Another reason for selecting FPGAs is that they are very difficult to attack remotely, with no operating system, fewer data vulnerabilities, and fewer published attacks. No system is 100% secure so defenses must be agile to respond to new threats and patch any discovered weaknesses. FPGAs are programmable at the hardware level offering agility down to the hardware-level customization. In addition, cryptography, security, and adjacent algorithms are often feed-forward mathematically heavy computations, which are very well-suited to the FPGA spatial compute architecture with its array of digital signal processing (DSP) blocks logic and embedded memories.

The implementation of XANDER on Cyclone IV FPGAs provides comprehensive cybersecurity consisting of a defense-in-depth approach with multiple layers of protection (as shown in the 4 layers in the corners of Figure 2); including an Advanced Encryption Standard (AES) encryption/decryption layer, an authentication layer, and last-line intrusion detection, prevention, and mitigation mechanisms in an 'illusion' layer.

The AES layer uses a 256 bit key for encryption, a strength generally accepted to be suitable to 2031 and beyond. However, one advantage of the FPGA is that if this were proved not to be the case, this can be updated in the firmware. We can take further advantage of the FPGAs low-level hardware customization to further hinder attacks; for example, obfuscating by slicing and distributing keys over multiple embedded memory location. The Authentication Code layer is responsible for generating handshaking codes for all communications and data transmissions. In the Flex system these codes are generated by a custom one-way mathematical function using embedded differential equations and no transmission can be done without these code verifications.

Handshaking verifies authorization and verifies data has been received successfully. The last layer in the defense (called the Illusion Layer in the Flex system) observes for anomalies, including handshaking response failures. On suspicious activity, it ensures no intruder can gain access to data or further access the system by zeroizing all data (in this case overwriting with new dummy data) and locking all the I/O (with the exception for two pins for system salvage and cleaning by the designer).

The FPGA architecture is well-suited to solve all these design challenges. It contains programmable logic elements as well as configurable static random-access memory (SRAM), high-speed input/output (I/O) pins and routing interconnect. These computing elements are distributed inside the chip to perform parallel and simultaneous computational tasks. It provides capability for users to design custom processor, which is tailored to manage unique data flow required.

Cyclone IV FPGAs were a perfect fit for Flex's XANDER due to its well-balanced low-power logic fabric and I/O performance with long lifecycle and high reliability for industrial use cases.

Conclusion

This collaboration project with Flex gives useful insights as to how production efficiency improvement promised through Industry 4.0 transformation can be realised with Intel FPGAs. The FPGA's truly parallel real-time compute enabled the consolidation of SMT processes, fulfilling the requirement for effective and efficient decision making and fast processing requirements. The flexible I/O and cybersecurity allowed these to be connected securely in a multi-layered decentralized architecture.

Intel IoT is revolutionizing intelligence in machines, buildings, supply chains, factories, and electrical grids helping the industrial world respond to new challenges in the age of big data, security breaches, and IT/OT convergence. Across the factory, Intel helps to unleash the potential of data by transforming it into real-time insights that can increase uptime, improve quality, and invigorate revenue.



¹ Based on internal Flex testing.

Intel technologies may require enabled hardware, software or service activation.

No product or component can be absolutely secure.

Your costs and results may vary.

Intel does not control or audit third-party data. You should consult other sources to evaluate accuracy.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. *Other names and brands may be claimed as the property of others.