

Historia 1: “Automatizando la Validación de CPLDs con IA Local”

(Principios: Invent and Simplify | Deliver Results | Ownership)

S – Situation

Durante la etapa crítica de validación del nuevo servidor Xeon, el proceso manual de validación de CPLDs tomaba más de 10 horas por build. Cada iteración dependía de scripts fragmentados y validaciones manuales, generando errores humanos y retrasos significativos justo antes del *Power On*.

T – Task

Como ingeniero senior de automatización, debía diseñar un sistema que redujera los tiempos de validación y permitiera detectar defectos pre-silicon con total trazabilidad, sin depender de herramientas propietarias externas.

A – Action

- Diseñé una arquitectura híbrida en **Python y C** sobre una **Raspberry Pi 5**, creando un sistema completamente autónomo para ejecutar más de **1 000 vectores JTAG por build**.
- Integré un **modelo de lenguaje local (SLM Phi-3.5 mini)** mediante **Ollama**, que actuaba como asistente de validación y corrección de código, permitiendo respuestas inmediatas sin conexión a la nube.
- Implementé **Azure CI/CD pipelines** y una suite de **unit tests con >90 % de cobertura**, asegurando calidad y reproducibilidad en cada release.
- Lideré la integración en laboratorio, coordinando con los equipos de firmware y validación para desplegar el sistema en solo **dos días** durante la ventana crítica de *Power On*.

R – Result

- **Reduje el ciclo de validación de 10 horas a 1 hora**, acelerando el *pattern release* en 90 %.
- **Detectamos 3 defectos críticos pre-silicon**, evitando fallas costosas en la etapa de bring-up.
- Logramos un **100 % de pass yield en muestras de ingeniería**.
- La solución fue adoptada como **referencia global** para validación inteligente con IA local, y estableció la base para futuras integraciones de agentes en flujos de hardware.

Historia 2: “El Primer Lanzamiento a Tiempo en una Década”

(Principios: Deliver Results | Ownership | Earn Trust | Dive Deep)

S – Situation

El programa de la **5.ª Generación Xeon** enfrentaba múltiples retrasos acumulados en desarrollo y pruebas. Históricamente, ninguna generación anterior había alcanzado el *time-to-market* prometido. Los cuellos de botella en automatización y comunicación entre equipos globales amenazaban con repetir el patrón.

T – Task

Como **Product Owner y Tech Lead**, debía coordinar equipos de diseño, validación y manufactura en tres países, mejorar la eficiencia de los flujos de automatización y asegurar un entregable a tiempo por primera vez en 10 años.

A – Action

- Introduje prácticas de **Agile/Scrum** en un entorno tradicionalmente rígido, implementando Jira y CA Agile Central para gestionar *backlogs* y *velocity metrics*.
- Definí KPIs claros para desarrollo y pruebas, identificando barreras de comunicación entre ingeniería de planta y laboratorio.
- Lideré la creación de un nuevo flujo de validación automatizado en Python y C#, reduciendo tiempos de testing en 25 % y aumentando eficiencia operativa en 15 %.
- Desarrollé una herramienta de analítica (“**Retest QBot**”) para eliminar retests innecesarios, conectada a bases de datos de manufactura vía MSAL/Azure API con autenticación segura.
- Fomenté una cultura de confianza entre equipos globales (Santa Clara, Austin y Costa Rica), promoviendo transparencia y aprendizaje continuo.

R – Result

- El equipo logró el **primer lanzamiento puntual en 10 años, acelerando la línea de tiempo en 6 semanas**.
 - Se obtuvieron **mejoras de 30 % en métricas de desarrollo, 20–40 % en rendimiento y >30 % en eficiencia energética** del producto.
 - Fui reconocido con el **Intel Achievement Award** por liderazgo y entrega excepcional de resultados.
 - La cultura de entregables a tiempo y colaboración fue adoptada como estándar regional para proyectos posteriores.
-