



# Verificación y Validación de Software en Instrumento de Medida

Gustavo Alessandrini\*, Alejandro Bertello\*

\*Instituto Nacional de Tecnología Industrial, Electrónica e Informática – Unidad Técnica Informática  
gusi@inti.gov.ar, bertello@inti.gov.ar

**Abstract** — Se describe el proceso de verificación y validación del software embebido (firmware), del instrumento de medida “Registrador Analizador de Calidad de Potencia PQ1000” de la empresa fabricante Ecamec SRL, para demostrar su conformidad con los requisitos esenciales de acuerdo con la norma IEC 61000-4-30, apartados 4.4 y 4.5 referidos al “método de agregación de medidas”. Este trabajo fue llevado a cabo en el Laboratorio de Software de la Unidad Técnica Informática de INTI Electrónica e Informática.

**Keywords** — Software embebido, Verificación, Validación, Analizador de calidad de potencia.

## I. INTRODUCCIÓN

Las actividades de validación y verificación (V&V) de especificaciones de software en el ámbito de la metrología, están cobrando mayor importancia debido al incremento de la cantidad de instrumentos de medida que tienen a las aplicaciones de software, ejecutándose en uno o más microcontroladores, como una de las principales piezas de su arquitectura.

La V&V es el conjunto de procesos de comprobación y análisis que aseguran que el software que se desarrolla está acorde a su especificación y cumple las necesidades de los clientes [1].

En el caso presentado, estas necesidades están establecidas por la norma IEC 61000-4-30, habiéndose realizando solamente la verificación y validación referida al “método de agregación de medida”, apartados 4.4 y 4.5, para un equipo clase A.

El objetivo del proceso de verificación y validación es el de asegurar que todos los requisitos del sistema en cuanto a la implementación de los mencionados apartados de la norma IEC 61000-4-30 han sido incluidos en el desarrollo del

software para el producto bajo ensayo, de una forma completa y correcta.

El alcance de la validación comprende los módulos de software involucrados en los procesos relacionados con el método de agregación presentado.

## II. EL ESTÁNDAR IEC 61000-4-30

El estándar IEC 61000-4-30 [2] especifica los métodos de medida de los parámetros de calidad de suministro de las redes de corriente alterna y la forma de interpretar los resultados obtenidos. Los parámetros de calidad de suministro que trata son: frecuencia, amplitud de tensión de alimentación, flicker, huecos de tensión y sobretensiones temporales de suministro, así como también las interrupciones de tensión, transitorios, desequilibrio, armónicos e interarmónicos de tensión y de corriente, las señales transmitidas en la red y las variaciones rápidas de tensión.

La norma define los métodos de medida pero no presenta ninguna especificación para su implementación, de forma tal que los requisitos funcionales de los instrumentos de medida quedan determinados por los márgenes de incertidumbre asignados a cada parámetro.

Se definen, dos clases de método de medida, el método clase A y el método clase B.

El método de clase A se aplica cuando se desean medidas precisas. Como ejemplo de aplicación de este método, el estándar indica que se aplica para verificación de conformidad, especificaciones contractuales, resolución de litigios, etc.

El método de clase B se aplica cuando no se requiere una gran precisión (por ejemplo en materia de estudios estadísticos).

### *Agregación de intervalos de tiempo de medida*

La norma IEC 61000-4-30 establece que las medidas obtenidas de cada uno de los parámetros de calidad de la energía eléctrica deben ser promediadas en el tiempo. Los intervalos temporales de agregación están definidos para cada uno de los dos métodos.

Para el método de clase A, los intervalos de tiempo de medida se adicionan en tres intervalos diferentes:

- Intervalos de 3 segundos
- Intervalos de 10 minutos
- Intervalos de 2 horas.

### *Algoritmo de agregación de medidas*

La agregación de medidas se calcula con la raíz cuadrada de la media aritmética de los cuadrados de los valores de entrada, definiéndose 3 categorías:

- Agregación de periodos
- Agregación de periodos en tiempo de reloj
- Agregación en tiempo de reloj.

## III. FORMULACIÓN DE REQUISITOS

Se establecieron 14 requisitos para asegurar que lo establecido en la norma, esté presente en la implementación del software de forma completa y correcta. A su vez, estos requisitos sirven como punto de partida para la elaboración de los casos de prueba para verificar el cumplimiento de lo solicitado [3]. Estos requisitos cubren lo enunciado en la norma referida, en los siguientes apartados:

- 4.4. Agregación de medida sobre intervalos de tiempo
- 4.5. Algoritmo de agregación de medidas
  - 4.5.1. Integración de las mediciones
  - 4.5.2. Agregación de periodos (150/180 cycle aggregation)
  - 4.5.3. Agregación de periodos en tiempo de reloj
  - 4.5.4. Agregación en tiempos de reloj (2 hour aggregation)

## IV. GENERACIÓN DE CASOS DE PRUEBA

Para la cobertura de todos los requisitos definidos, se establecieron casos de prueba, divididos en 2 categorías:

- Casos de prueba de verificación
- Casos de prueba de validación

En los casos de prueba de verificación se utilizaron técnicas de *inspección de código* en las funciones involucradas en la validación del ensayo, y *pruebas de integración* para verificar el comportamiento en conjunto de los componentes bajo prueba en un entorno sin interacción real con sistemas externos. En estas pruebas de integración, se excitó al equipo con una señal senoidal simulada por el firmware, utilizando una versión de programa que incluía directivas de compilación condicional para obtener este comportamiento. Esta versión

permitía, además, la generación de eventos de sincronismo también generados por el mismo programa para poder colocar estos en los tiempos de muestreo de interés para el análisis del ensayo. Las respuestas a las excitaciones introducidas en las distintas pruebas, se observaron a través de salidas (o puntos de prueba internos) previstas para tal fin en el equipo.

Los casos de prueba de validación, se generaron para cubrir los siguientes requisitos:

- Agregación de medida sobre intervalos de tiempo. Para instrumentos clase A, para una red de distribución eléctrica de 50 Hz, para:
  - Tensión de suministro
  - Armónicas
  - Inter-armónica
- Agregación sobre intervalos adicionales
  - Intervalos de 150 ciclos
  - Intervalos de 10 minutos
  - Intervalos de 2 horas
  - Otros intervalos propuestos por el fabricante
- Algoritmo de agregación de medidas
- Agregación de periodos (agregación de 150 ciclos)
- Agregación de periodos en tiempo de reloj
- Agregación de periodos en tiempo de reloj (agregación de 2 horas)

Los métodos de verificación y validación aplicados fueron los siguientes:

- Identificación del software
- Revisión del diseño del software
- Revisión de la documentación del software
- Inspección de código
- Comprobación funcional
- Simulación de las señales de entrada
- Operación en situación real
- Operación con instrumento calibrador

## V. IDENTIFICACIÓN DE LA VERSIÓN DE FIRMWARE VALIDADO

La versión de firmware validada de acuerdo con lo presentado en el plan, fue identificada mediante una firma digital, utilizando el mecanismo de doble clave, para garantizar que el firmware instalado en los equipos producidos corresponda a la versión ensayada.

Finalizadas las pruebas, se realizaron los siguientes pasos:

- a) Se generó el programa ejecutable de la versión ensayada, en la forma que será instalada definitivamente en el producto.
- b) Se colocó a la versión del programa generado, una nomenclatura única que permite al fabricante identificarla en su sistema de control de versiones.

- c) El archivo ejecutable generado en el paso a), fue grabado en la memoria de programa del microcontrolador del equipo ensayado, y se verificó su correcto funcionamiento.
- d) Se hizo una copia del archivo ejecutable creado y se lo sometió al proceso de firmado. Esta operación se realizó con una aplicación de firmado que utiliza como algoritmo de firma el RSA – especificado en el estándar ANSI x9.31 [4] – La aplicación de firmado utiliza para la implementación del mencionado algoritmo, la biblioteca de funciones openssl [5].
- e) Para comprobar que el proceso de firmado no alteró el archivo original, se efectuó una comparación binaria entre el archivo original y la copia firmada

## VI. RESULTADOS

Se han validado sin encontrar fallos los siguientes requisitos de la norma IEC 61000-4-30 ítems 4.4 y 4.5:

- Agregación de medida sobre intervalos de tiempo (solamente método para instrumentos clase A, para una red de distribución eléctrica de 50 Hz)
  - Tensión de suministro
  - Armónicas
  - Inter-armónica
- Agregación sobre intervalos adicionales
  - Intervalos de 150 ciclos
  - Intervalos de 10 minutos
  - Intervalos de 2 horas
  - Otros intervalos propuestos por el fabricante
- Algoritmo de agregación de medidas
- Agregación de períodos (agregación de 150 ciclos)
- Agregación de períodos en tiempo de reloj
- Agregación de períodos en tiempo de reloj (agregación de 2 horas)

Se ha realizado satisfactoriamente la verificación por inspección de código y análisis estático, de los módulos y funciones involucrados en la agregación de medida sobre intervalos de tiempo para desbalance (instrumentos clase A, para una red de distribución eléctrica de 50 Hz).

Para cada función inspeccionada se pudo observar el criterio utilizado por los desarrolladores para resolver las distintas situaciones, tales como: nivel de anidamiento, completitud de las condiciones, estructura jerárquica de llamados, comprobación de las operaciones de cálculo, variables, lectura y escritura en memoria, uso de comentarios en el código, etc.

## VII. CONCLUSIÓN

El software del instrumento de medida “Registrador Analizador de Calidad de Potencia PQ1000” cumplió con los

requisitos esenciales de los puntos 4.4 y 4.5 de la norma IEC 61000-4-30.

El protocolo desarrollado para el ensayo mencionado en este trabajo, sirvió de plataforma para la realización de otros ensayos, por la experiencia adquirida en la formulación de casos de prueba y el desarrollo de una metodología (plantillas, descripciones, formularios) como así también en el uso de herramientas de depuración y pruebas (análisis estático) y documentación.

Se pudo comprobar que el esquema de firma electrónica aplicado para garantizar la autenticidad del firmware bajo ensayo resultó adecuado para este tipo de trabajos.

## RECONOCIMIENTOS

Agradecemos a la empresa Ecamec SRL la conformidad para la publicación de este trabajo.

## REFERENCIAS

- [1] IEEE Standard for Software Verification and Validation. IEEE Std 1012™-2004 (Revision of IEEE Std 1012-1998).
- [2] IEC Standard 61000-4-30. IEC 61000: Electromagnetic Compatibility (EMC), Part 4: Testing and measurement techniques, Section 30: Power quality measurement methods. 2008.
- [3] Testing Embedded Software. Bart Broekman and Edwin Notenboom. An imprint of Pearson Education.
- [4] ANSI X9.31-1998, Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA).
- [5] The open source toolkit for SSL/TLS [www.openssl.org](http://www.openssl.org)