DOT/FAA/AR-08/32

Organización de Tráfico Aéreo
NextGen & Operaciones de Planeamiento
Oficina de Investigación y
Desarrollo Tecnológico
Washington, DC 20591

Manual de Gestión de Ingeniería de Requisitos

Junio de 2009

Informe Final

Departamento de Transporte de los EE.UU.
Administración Federal de Aviación

AVISO

Este documento se difunde bajo el patrocinio del Departamento de Transporte de los Estados Unidos con el fin de intercambiar información. El Gobierno de los Estados Unidos no asume ninguna responsabilidad por el contenido o el uso del mismo. El Gobierno de los Estados Unidos no respalda productos ni fabricantes. Los nombres comerciales o de fabricantes aparecen aquí únicamente porque se consideran esenciales para el objetivo de este informe. Este documento no constituye una política de certificación de la FAA. Consulte con su oficina local de certificación de aeronaves de la FAA sobre su uso.

Este informe está disponible en la página de informes técnicos de texto completo del Centro Técnico William J. Hughes de la Administración Federal de Aviación: actlibrary.tc.faa.gov en formato de documento portátil de Adobe Acrobat (PDF).

Página de documentación del informe técnico

1. Informe N°	2. Adhesión del Gobierno N°	3. Número de catálogo del destinatario.
DOT/FAA/AR-08/32		

4. Título y subtítulo	5. Fecha del informe
MANUAL DE GESTIÓN DE INGENIERÍA DE REQUISITOS	Junio de 2009
	6. Código de la organización ejecutora

7. Autor(es)	8. Informe de la organización ejecutante N.º
David L. Lempia y Steven P. Miller	

9. Nombre y dirección de la organización ejecutante	10. Unidad de Trabajo N° (TRAIS)
Compañía Rockwell Collins, Inc. 400 Collins Road NE Cedar Rapids, Iowa 52245	11. Contrato o subvención Núm. DTFACT-05-C-00004

12. Nombre y dirección de la agencia patrocinadora	13. Tipo de informe y período abarcado
Departamento de Transporte de los Estados Unidos Administración Federal de Aviación Organización de Tráfico Aéreo Planificación de operaciones y NextGen Oficina de Investigación y Desarrollo Tecnológico Washington, DC 20591	Informe final
	14. Código de la Agencia Patrocinadora AIR-120

15. Notas complementarias

El COTR de la División de Investigación y Desarrollo de Seguridad Aeronáutica y de Aeropuertos de la Administración Federal de Aviación fue Charles Kilgore.

16. Resumen

Este manual presenta un conjunto de prácticas recomendadas sobre cómo recopilar, redactar, validar y organizar requisitos. Intenta reunir las mejores ideas de varios enfoques, organizarlas en un todo coherente e ilustrarlas con ejemplos concretos que dejan en claro sus beneficios.

El manual está orientado al ámbito de los sistemas integrados en tiempo real y, específicamente, a la industria de la aviónica. Describe un conjunto de prácticas recomendadas en las que los conceptos básicos se pueden practicar de forma aislada, pero se refuerzan entre sí cuando se practican en conjunto. Estas prácticas permiten a los desarrolladores avanzar desde una descripción general inicial de alto nivel de un sistema hasta una descripción detallada de sus requisitos de comportamiento y rendimiento. Debido a la creciente importancia del software en los sistemas de aviónica, estas prácticas enfatizan las técnicas para facilitar la transición de los requisitos del sistema a los del software.

En todo el Manual se utilizan ejemplos concretos para aclarar los conceptos, pero existen muchos otros formatos que podrían utilizarse para lograr los mismos objetivos. Se espera que la mayoría de las organizaciones que deseen utilizar estas prácticas deseen modificarlas, tal vez de manera significativa, para integrarlas con sus procesos y herramientas existentes.

17. Palabras clave	18. Declaración de distribución
Requisitos, Ingeniería, Aviónica, Sistemas, Software	Este documento está disponible para el público de EE. UU. a través del Servicio Nacional de Información Técnica (NTIS) Springfield, Virginia 22161.

19. Clasificación de seguridad (de este informe)	20. Clasificación de seguridad (de esta página)	21. Número de páginas	22. Precio
Sin clasificar	Sin clasificar	146	

TABLA DE CONTENIDOS

LISTA DE FIGURAS

LISTA DE TABLAS

LISTA DE SIGLAS Y ABREVIATURAS

AP	Autopilot - Piloto Automático
AHS	Attitude Heading System - Sistema de Actitud y Rumbo
ARINC	Aeronautical Radio, Incorporated - Radio Aeronáutica, Incorporada
ARP	Aerospace Recommended Practice - Práctica Recomendada Aeroespacial
CoRE	Consortium Requirements Engineering - Ingeniería de Requisitos del Consorcio
CSA	Controlled Surface Actuators - Actuadores de Superficie Controlados
FCI	Flight Crew Interface - Interfaz de la Tripulación de Vuelo
FCS	Flight Control System - Sistema de Control de Vuelo
FD	Flight Director - Director de Vuelo
FG	Flight Guidance - Guía de Vuelo
FGS	Flight Guidance System - Sistema de Guía de Vuelo
FHA	Functional Hazard Assessment - Evaluación de Riesgos Funcionales
HMI	Human Machine Interface - Interfaz Hombre-Máquina
msec	millisecond - Milisegundo
N/A	Not Applicable - No Aplica
PFD	Primary Flight Display - Pantalla Principal de Vuelo
PSSA	Preliminary System Safety Assessment - Evaluación Preliminar de Seguridad del Sistema
REM	Requirements Engineering Management - Gestión de Ingeniería de Requisitos
RSML	Requirements State Machine Language - Lenguaje de Máquina de Estados para Requisitos
RSML-e	Requirements State Machine Language without events - Lenguaje de Máquina de Estados para Requisitos sin Eventos
SCR	Software Cost Reduction - Reducción de Costos de Software
SpecTRM	Specification Toolkit and Requirements Methodology - Kit de Herramientas de Especificación y Metodología de Requisitos

RESUMEN EJECUTIVO

Este manual presenta un conjunto de prácticas recomendadas sobre cómo recopilar, redactar, validar y organizar requisitos. Intenta reunir las mejores ideas de varios enfoques, organizarlas en un todo coherente e ilustrarlas con ejemplos concretos que dejan en claro sus beneficios.

El manual está orientado al ámbito de los sistemas integrados en tiempo real y, específicamente, a la industria de la aviónica. Describe un conjunto de prácticas recomendadas en las que los conceptos básicos se pueden practicar de forma aislada, pero se refuerzan entre sí cuando se practican en conjunto. Estas prácticas permiten a los desarrolladores avanzar desde una descripción general inicial de alto nivel de un sistema hasta una descripción detallada de sus requisitos de comportamiento y rendimiento. Debido a la creciente importancia del software en los sistemas de aviónica, estas prácticas enfatizan las técnicas para facilitar la transición de los requisitos del sistema a los del software.

En todo el Manual se utilizan ejemplos concretos para aclarar los conceptos, pero existen muchos otros formatos que podrían utilizarse para lograr los mismos objetivos. Se espera que la mayoría de las organizaciones que deseen utilizar estas prácticas deseen modificarlas, tal vez de manera significativa, para integrarlas con sus procesos y herramientas existentes.

1. INTRODUCCIÓN.

Esta investigación se llevó a cabo en respuesta a la convocatoria de Gestión de ingeniería de requisitos (REM) publicada por el Centro Técnico William J. Hughes de la Administración Federal de Aviación. El objetivo de esta tarea era determinar métodos que permitieran gestionar, controlar, integrar, verificar y validar con éxito los requisitos de sistemas y software que pueden desarrollar múltiples entidades.

1.1 OBJETO.

Este manual presenta un conjunto de prácticas recomendadas sobre cómo recopilar, redactar, validar y organizar requisitos. Intenta reunir las mejores ideas de varios enfoques, organizarlas en un todo coherente e ilustrarlas con ejemplos concretos que dejan en claro sus beneficios.

La literatura sobre ingeniería de requisitos es muy amplia y las prácticas varían ampliamente incluso dentro de una industria en particular. Por este motivo, el Manual está orientado al dominio de los sistemas integrados en tiempo real y, específicamente, a la industria de la aviónica. Debido a la creciente importancia del software en estos sistemas, se hace hincapié en las prácticas que facilitan la transición de los requisitos del sistema a los del software.

Las prácticas recomendadas de nivel principal se presentan aproximadamente en el orden en que se llevarían a cabo en un programa, pero no existe ningún requisito que obligue a respetar estrictamente este orden. Como sucede con la mayoría de los procesos, se espera una iteración significativa entre las diferentes actividades a medida que se perfeccionan los requisitos. En lugar de intentar especificar un proceso detallado, el Manual se centra en identificar qué información se necesita en una especificación de requisitos y en brindar recomendaciones sobre cómo se puede recopilar y organizar esa información.

Para concretar estas ideas, se utilizan dos ejemplos para ilustrar las prácticas recomendadas. El primero, un Termostato para un Isolette utilizado en una unidad de cuidados neonatales, proporciona un pequeño ejemplo de fácil comprensión que ilustra la aplicación de las prácticas recomendadas en un solo sistema. El segundo, un sistema de control de vuelo (FCS), un sistema de guía de vuelo (FGS) y un piloto automático (AP) muy simples, ilustra cómo se podrían asignar los requisitos de un sistema a sus subsistemas para facilitar el desarrollo por parte de múltiples subcontratistas. Ambos ejemplos tienen como objetivo ilustrar las prácticas recomendadas y no pretenden ser ejemplos completos de sistemas reales. Ambos ejemplos se mencionan a lo largo del análisis de las prácticas recomendadas y sus especificaciones se presentan en los apéndices.

Si bien los ejemplos parecen sugerir un estilo y un formato específicos, su verdadero propósito es ilustrar y aclarar las prácticas recomendadas y sus beneficios. Existen muchos formatos diferentes que podrían utilizarse para satisfacer los mismos objetivos, y la mayoría de las organizaciones que deseen utilizar alguna de las prácticas recomendadas deberán modificarlas, tal vez de manera significativa, para que se adapten a sus procesos y herramientas existentes.

1.2 ANTECEDENTES.

Las prácticas recomendadas se basan en los resultados de una encuesta de la industria y una búsqueda bibliográfica, ambas documentadas en la referencia 1. La encuesta de la industria proporcionó una útil descripción general del estado actual de la práctica e identificó muchos de los problemas y preocupaciones de los desarrolladores que trabajan en el dominio de la aviónica. La búsqueda bibliográfica identificó varias metodologías para REM[1] que se han aplicado con éxito a los sistemas de aviónica y abordan muchas de las preocupaciones planteadas en la encuesta. Gran parte del desafío en el desarrollo de este Manual consistió en encontrar formas para que las organizaciones integraran estas metodologías en sus prácticas existentes.

Las prácticas descritas en este Manual se basan en gran medida en la metodología de Reducción de Costos de Software (SCR) y el modelo de cuatro variables propuesto originalmente por Parnas y Madey [2] para especificar los requisitos del avión A-7E Corsair II de la Armada de los EE. UU. [3]. Más tarde, el Consorcio de Productividad de Software amplió estas ideas en la metodología de Ingeniería de Requisitos del Consorcio (CoRE) [4 y 5], que se utilizó para especificar los requisitos del avión C-130J [6]. Muchas de las prácticas recomendadas sobre cómo organizar los requisitos se basan en ideas desarrolladas originalmente con la metodología CoRE. La SCR también siguió evolucionando durante las últimas dos décadas. El Laboratorio de Investigación Naval desarrolló una serie de herramientas para la especificación y el análisis de modelos SCR [7].

Otra fuente importante de mejores prácticas fue el extenso trabajo realizado por Leveson y sus colegas en el desarrollo del lenguaje de máquina de estados de requisitos (RSML) [8] utilizado para especificar el sistema de alerta de tráfico y prevención de colisiones II [9]. Este enfoque se extendió posteriormente al RSML sin eventos (RSML-mi) [10] y se utiliza como base para el análisis formal de las especificaciones de requisitos [11 y 12]. Más recientemente, esta notación se extendió al Kit de herramientas de especificación y metodología de requisitos (SpecTRM) desarrollado por Safeware Engineering Corporation [13, 14 y 15]. Los modelos SpecTRM están integrados dentro de una especificación de intención más amplia que "proporciona un flujo continuo de fundamentos y razonamiento desde los objetivos de más alto nivel del sistema, a través del modelo SpecTRM hasta los materiales de implementación y capacitación del operador" [13, 14 y 16].

En las últimas dos décadas se ha llevado a cabo una gran cantidad de investigaciones para especificar los requisitos como casos de uso. En particular, los casos de uso parecen ser una técnica excelente para pasar de la descripción general inicial e informal del sistema a la especificación formal y detallada de los requisitos. El análisis de los conceptos operativos en este manual se basa en el uso de casos de uso textuales similares a los descritos en las referencias 17 a 21.

En la referencia 22 se puede encontrar una buena guía general sobre el proceso REM, que comparte muchas similitudes con el enfoque descrito en este manual. Ofrece muchas recomendaciones excelentes sobre el proceso REM, descripciones de lo que constituye un buen requisito y listas de verificación de ejemplo. Otra referencia conocida es la 23. El texto de Leveson sobre seguridad de sistemas y software también analiza el proceso REM en detalle [24].

1 N.d.T: Requirements Engineering Management (Gestión de ingeniería de requisitos)

Por último, el lector debe conocer una serie de normas relacionadas con el REM o relevantes para la práctica del REM en la industria de la aviónica, entre las que se incluyen:

- Guía IEEE para el desarrollo de especificaciones de requisitos del sistema (norma IEEE 1233) [25]

- IEEE Práctica Recomendada para Especificaciones de Requisitos de Software (Norma IEEE 830-1998) [26]

- Consideraciones de software en la certificación de sistemas y equipos aerotransportados (DO-178B) [27]

- Informe final para la aclaración de la DO-178B, "Consideraciones de software en la certificación de sistemas y equipos aerotransportados" (DO-248B) [28]

- Directrices para la garantía de la integridad del software de los sistemas de comunicación, navegación, vigilancia y gestión del tráfico aéreo (CNS/ATM) (DO-278) [29]

- Consideraciones de certificación para sistemas de aeronaves altamente integrados o complejos (ARP 4754) [30]

- Directrices y métodos para llevar a cabo el proceso de evaluación de la seguridad de los sistemas y equipos aerotransportados civiles, ARP 4761 [31]

2. PRÁCTICAS RECOMENDADAS.

¿Qué hace que una especificación de requisitos sea buena y otra mala? Parafraseando a David Parnas, una buena especificación de requisitos debería describir todo lo necesario para producir el sistema correcto, y nada más. Esto establece sucintamente el equilibrio que los requisitos deben lograr: especificar todo lo que se necesita del sistema que se va a construir, sin limitar excesivamente a los desarrolladores aventurándose en el diseño. Como se afirma a menudo, los requisitos deberían especificar lo que hará el sistema, no cómo lo hará.

Sin embargo, casi siempre se trata de una cantidad sorprendentemente grande de información. Se debe invertir un esfuerzo considerable en organizar los requisitos de modo que sean legibles, a menudo por audiencias con inquietudes y conocimientos muy diferentes, y que se puedan mantener ante cambios. Como resultado, una buena especificación de requisitos consiste en mucho más que una simple lista de enunciados obligatorios. Muchas de las prácticas recomendadas descritas en este Manual están dedicadas a establecer la estructura para garantizar que los requisitos sean completos, coherentes, claros y estén bien organizados.

Al mismo tiempo, el desarrollo de los requisitos es típicamente una progresión desde un estado en el que se sabe relativamente poco acerca del sistema a otro en el que se sabe mucho. Para ser eficaz, el proceso de ingeniería de requisitos debe progresar de manera similar desde prácticas informales al comienzo de la definición de los requisitos hasta prácticas más rigurosas a medida que se completan los requisitos.

En el caso de sistemas de gran tamaño, no suele ser práctico establecer los requisitos detallados del sistema independientemente de la arquitectura del mismo. En su lugar, se desarrollan requisitos de alto nivel, se completa el siguiente nivel de diseño y se desarrollan requisitos más detallados para cada componente. Este proceso continúa hasta que se alcanza el nivel de detalle necesario. En efecto, el proceso de especificación de requisitos se intercala con el desarrollo de la arquitectura del sistema.

Este manual describe las siguientes 11 prácticas recomendadas de nivel principal que permiten a los desarrolladores progresar desde una descripción general inicial de alto nivel del sistema hasta desarrollar una descripción detallada de sus requisitos de comportamiento y rendimiento.

1. Desarrollar la Descripción General del Sistema
2. Identificar los Límites del Sistema
3. Desarrollar los Conceptos Operativos
4. Identificar los Supuestos Ambientales
5. Desarrollar la Arquitectura Funcional
6. Revisar la Arquitectura para Cumplir con las Restricciones de Implementación
7. Identificar Modos del Sistema
8. Desarrollar los Requisitos Detallados de Comportamiento y Rendimiento
9. Definir los Requisitos del Software
10. Asignar Requisitos del Sistema a Subsistemas
11. Proporcionar Fundamentos

Estas 11 prácticas recomendadas de nivel principal se muestran y describen en detalle en las secciones 2.1 a 2.11.

Con excepción de las dos últimas prácticas recomendadas de nivel principal, estas prácticas recomendadas se presentan aproximadamente en el orden en que se llevarían a cabo en un programa de desarrollo de software, pero no es necesario respetar estrictamente este orden. Se espera que haya una iteración significativa entre estos pasos en cualquier desarrollo real. Si bien las prácticas recomendadas de nivel principal son aplicables a casi cualquier desarrollo, es posible que las organizaciones deseen adaptarlas para que se adapten mejor a sus prácticas existentes.

Dentro de cada una de las prácticas recomendadas de nivel principal se especifican prácticas recomendadas de nivel inferior más detalladas. Estas prácticas recomendadas de nivel inferior son más sensibles a cómo se implementa la práctica recomendada de nivel principal y dependen más de cómo se lleva a cabo la práctica recomendada de nivel principal. Es más probable que estas prácticas recomendadas de nivel inferior deban modificarse a medida que la práctica recomendada de nivel principal se adapta para que se ajuste a los métodos y herramientas existentes. Estas prácticas recomendadas de nivel inferior se muestran y describen en detalle en las secciones 2.1.1 a 2.11.7 y en las tablas correspondientes.

2.1 DESARROLLAR LA VISIÓN GENERAL DEL SISTEMA.

Consulte la tabla 1 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para desarrollar la descripción general del sistema.

Tabla 1. Desarrollo de la descripción general del sistema

Prácticas recomendadas para los niveles principal y secundario	
2.1	Desarrollar la Descripción General del Sistema: Desarrollar una descripción general del sistema que incluya una breve sinopsis, describa todos los contextos en los que se utilizará el sistema y enumere los objetivos, metas y limitaciones principales del sistema. Esto ayuda a definir el alcance del sistema mientras se desarrollan los requisitos y sirve como un medio para orientar rápidamente a un nuevo lector de los requisitos.
2.1.1	Desarrolle la descripción general del sistema al comienzo del proceso de ingeniería de requisitos y utilícela como introducción a la especificación de requisitos. Mantenga la descripción general en un nivel alto para que pueda usarse para orientar rápidamente a los nuevos lectores.
2.1.2	Proporcione una breve sinopsis textual del sistema como primera parte de la descripción general del sistema. La sinopsis debe nombrar el sistema, describir su propósito y resumir las capacidades del sistema.
2.1.3	Considere todo el ciclo de vida del sistema e identifique cada contexto distinto en el que se utilizará.
2.1.4	Utilice diagramas de contexto en la descripción general del sistema para proporcionar una representación gráfica de alto nivel del sistema, las entidades externas con las que interactúa y esas interacciones.
2.1.5	Para cada diagrama de contexto, proporcione una breve descripción de cada entidad externa y sus interacciones con el sistema.
2.1.6	Capturar un conjunto preliminar de objetivos del sistema al comienzo del proceso de ingeniería de requisitos para que puedan utilizarse para guiar el desarrollo de los requisitos.
2.1.7	Dependiendo del tamaño y la volatilidad de un proyecto, recopilar y mantener la información sobre cada objetivo del sistema necesaria para evaluar continuamente su importancia en relación con los demás objetivos.

2.1.1 Desarrollar una descripción general del sistema con anticipación.

La descripción general del sistema sirve como introducción a los requisitos del sistema. Aunque evolucionará continuamente a medida que se definan los requisitos, suele ser uno de los primeros artefactos creados. Su propósito es orientar rápidamente al lector respecto del sistema y el entorno en el que funcionará. No debe intentar describir completamente el sistema. Más bien, su intención es proporcionar al lector una visión general de lo que hace el sistema, cómo interactúa con su entorno y por qué es necesario. Como mínimo, debe incluir una breve sinopsis del sistema; una o más descripciones de contexto; una breve descripción de cada entidad externa en los diagramas de contexto; y un conjunto de objetivos, metas y limitaciones generales del sistema. También puede incluir otra información relevante, como antecedentes históricos, pero esto debe equilibrarse con la necesidad de orientar rápidamente a un nuevo lector. Se proporcionan ejemplos de descripciones generales del sistema en el apéndice A.1 para el Termostato Isolette, el apéndice B.1 para el FCS, el apéndice C.1 para el FGS y el apéndice D.1 para el AP.

Práctica recomendada 2.1.1: Desarrollar la descripción general del sistema en las primeras etapas del proceso de ingeniería de requisitos y utilizarla como introducción a la especificación de requisitos. Mantener la descripción general en un nivel alto para orientar rápidamente a los nuevos lectores.

2.1.2 Proporcionar una Sinopsis Del Sistema.

La descripción general debe comenzar con una sinopsis que proporcione una breve descripción narrativa del sistema. Lo ideal es que la sinopsis sea breve, clara y describa lo que hará el sistema sin implicar un diseño de sistema en particular. La sinopsis debe nombrar el sistema que se desarrollará, describir su propósito y resumir las principales capacidades que ofrece el sistema. Una versión preliminar de la sinopsis del Termostato Isolette establece lo siguiente:

> El sistema que se especifica es el Termostato de una Isolette. Una Isolette es una Incubadora para Bebés que proporciona temperatura, humedad y oxígeno controlados (si es necesario). Las Isolettes se utilizan ampliamente en las unidades de cuidados intensivos neonatales para el cuidado de Bebés prematuros.
>
> El termostato Isolette tiene como objetivo mantener la temperatura del aire de la unidad dentro de un Rango Deseado. Detecta la Temperatura Actual de la unidad y enciende y apaga la Fuente de Calor para calentar el aire según sea necesario. El sistema permite que la Enfermera establezca el Rango de Temperatura Deseado dentro de un rango seguro para los Bebés.

Al igual que con todas las descripciones generales del sistema, la sinopsis evolucionará a lo largo de la definición de los requisitos. Se proporcionan ejemplos de sinopsis completas del sistema al comienzo del apéndice A.1 para el termostato Isolette, el apéndice B.1 para el FCS, el apéndice C.1 para el FGS, y el apéndice D.1 para el AP.

Práctica recomendada 2.1.2: Proporcione una breve sinopsis textual del sistema como primera parte de la descripción general del sistema. La sinopsis debe nombrar el sistema, describir su propósito y resumir las capacidades del sistema.

2.1.3 Identificar Contextos del Sistema.

El propósito del contexto del sistema es describir, a un alto nivel, las entidades externas con las que interactúa el sistema y la naturaleza de esas interacciones. Nótese que puede haber más de un contexto en el que se utilizará el sistema durante todo su ciclo de vida. Por ejemplo, casi con certeza habrá un contexto operativo, que describe las entidades con las que interactúa el sistema durante el funcionamiento normal, pero también puede haber un contexto de prueba utilizado durante el desarrollo del sistema en el que el sistema interactúa con el equipo de prueba a través de diferentes interfaces, o puede haber un contexto de mantenimiento que se utiliza para diagnosticar y reparar el sistema en el campo. Cada contexto relevante para el sistema debe identificarse y describirse. Dado que diferentes personas estarán interesadas en diferentes contextos, generalmente es mejor describir cada contexto por separado que tratar de combinarlos en un solo contexto.

Práctica recomendada 2.1.3: Considerar todo el ciclo de vida del sistema e identificar cada contexto distinto en el que se utilizará.

2.1.4 Utilizar Diagramas de Contexto.

Para cada contexto, se deben identificar las entidades externas con las que interactúa el sistema y la naturaleza de esas interacciones. Una forma conveniente de hacerlo es mediante el uso de un diagrama de contexto que represente gráficamente cada entidad externa y su interacción con el sistema. En la figura A-1 se muestra un ejemplo de diagrama de contexto para el termostato Isolette. En la figura B-1 se muestra un diagrama de contexto similar para el FCS, en la figura C-1 para el FGS y en la figura D-1 para el AP. Tenga en cuenta que el sistema en sí se muestra en el diagrama de contexto como una caja negra sin estructura interna. También puede ser útil identificar los sistemas de nivel superior en los que está integrado el sistema (por ejemplo, el contexto del termostato incluye el contenido del sistema más amplio del Isolette).

Práctica recomendada 2.1.4: Utilice diagramas de contexto en la descripción general del sistema para proporcionar una representación gráfica de alto nivel del sistema, las entidades externas con las que interactúa y esas interacciones.

2.1.5 Describir Entidades Externas.

También se debe proporcionar una breve descripción de cada entidad externa y sus interacciones con el sistema. Dado que esto forma parte de la descripción general del sistema, estas descripciones deben ser breves y simples (se proporcionará información más detallada sobre las entidades y sus interacciones en secciones posteriores de la especificación aplicable). Se proporcionan ejemplos de las descripciones de las entidades externas y sus interacciones con el sistema en el apéndice A.1.1 para el termostato Isolette, el apéndice B.1.1 para el FCS, el apéndice C.1.1 para el FGS y el apéndice D.1.1 para el AP.

Práctica recomendada 2.1.5: Para cada diagrama de contexto, proporcione una breve descripción de cada entidad externa y sus interacciones con el sistema.

2.1.6 Capturar Objetivos Preliminares del Sistema.

Los objetivos son declaraciones informales de las necesidades de las partes interesadas (stakeholders en inglés) del sistema. No son requisitos, ya que no son verificables y no proporcionan suficiente información para construir el sistema [14]. Sin embargo, proporcionan una orientación importante sobre por qué se está construyendo el sistema y qué es importante para las partes interesadas. Durante las primeras fases del proyecto, pueden ser todo lo que esté disponible. Los objetivos con frecuencia entran en conflicto entre sí. Por ejemplo, un objetivo puede ser producir un sistema con una interfaz de operador sofisticada, mientras que otro objetivo es producir el sistema a un costo mínimo. Uno de los objetivos del proceso de ingeniería de requisitos es refinar los objetivos para convertirlos en requisitos verificables que resuelvan estos conflictos de la manera más eficaz posible.

Se debe recopilar un conjunto preliminar de objetivos del sistema al comienzo del proceso de ingeniería de requisitos para que se puedan utilizar como guía para la especificación de los

requisitos. Un lugar natural para presentar los objetivos del sistema es en la descripción general del sistema. Sin embargo, en un proyecto grande, los objetivos del sistema pueden ser lo suficientemente numerosos como para justificar su colocación en una sección propia o en otro documento.

Práctica recomendada 2.1.6: Capturar un conjunto preliminar de objetivos del sistema al comienzo del proceso de ingeniería de requisitos para que puedan utilizarse para guiar el desarrollo de los requisitos.

2.1.7 Mantener la Información de los Objetivos del Sistema.

Los objetivos suelen ayudar a explicar por qué existe un requisito en particular o por qué se enuncia de una manera particular y pueden citarse en la justificación (véase la sección 2.11) del requisito. Los conceptos operativos (véase la sección 2.3) también se remontan a los objetivos del sistema.

La gestión de los objetivos del sistema variará según el tamaño, la complejidad y la madurez del sistema que se esté desarrollando. En proyectos pequeños, los objetivos pueden entenderse bien. Se podrían enumerar los objetivos del sistema en una sola página y su gestión puede consistir únicamente en una revisión y actualización ocasionales. Este es el caso del ejemplo del termostato Isolette, que inicialmente solo tiene dos objetivos simples:

- G1—El Bebé debe mantenerse a una temperatura segura y confortable.
- G2—El costo de fabricación del termostato debe ser lo más bajo posible.

Sin embargo, en un proyecto de cualquier tamaño, y en particular en el caso de los nuevos sistemas de gran tamaño, los objetivos probablemente evolucionarán a lo largo del desarrollo del proyecto. Se identificarán nuevas necesidades, las prioridades cambiarán, los gerentes cambiarán, la comprensión del proyecto aumentará, todo lo cual dará como resultado un conjunto cambiante de objetivos del proyecto. La gestión de los objetivos en un proyecto de este tipo puede ser una tarea importante en sí misma. Para ello, se debe recopilar información que facilite el seguimiento del objetivo hasta su origen y la evaluación continua de la importancia de ese objetivo en relación con otros objetivos.

Las fuentes potenciales de los objetivos del sistema son numerosas e incluyen a los clientes, usuarios, organismos reguladores y desarrollos anteriores. Los objetivos pueden ser proporcionados directamente por estas organizaciones o pueden descubrirse a través de entrevistas o revisión de documentos proporcionados por las partes interesadas. Dependiendo del tamaño y la volatilidad de un proyecto, la información que puede ser conveniente recopilar sobre cada objetivo puede incluir:

- Origen: ¿De dónde surgió el objetivo (cliente, individuo, documento…)?
- Fecha de origen: ¿Cuándo se identificó el objetivo por primera vez?
- Autor: ¿Quién documentó el objetivo por primera vez?
- Prioridad: ¿Cuál es la importancia del objetivo en relación con otros objetivos?
- Partes interesadas: ¿Qué clientes o usuarios están más preocupados por este objetivo?
- Estabilidad: ¿Qué probabilidad hay de que este objetivo cambie?
- Fecha programada: ¿Cuál es la fecha en que está previsto implementar este objetivo?

En el caso de elementos como la prioridad y la estabilidad, cada organización probablemente desee proporcionar sus propios valores y definiciones permitidos. Si se espera que los objetivos del sistema cambien con frecuencia, se deben mantener bajo control de configuración para proporcionar un historial de su evolución.

Práctica recomendada 2.1.7: Dependiendo del tamaño y la volatilidad de un proyecto, recopilar y mantener la información sobre cada objetivo del sistema necesaria para evaluar continuamente su importancia en relación con los demás objetivos.

2.2 IDENTIFICAR EL LÍMITE DEL SISTEMA.

Consulte la tabla 2 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para identificar el límite del sistema.

Tabla 2. Identificar el límite del sistema

Prácticas recomendadas para los niveles principal y secundario
2.2 Identificar el Límite del Sistema: Desarrollar una definición clara de los límites entre el sistema y su entorno. Esto proporciona una comprensión sólida de lo que se encuentra dentro del sistema que se va a construir y lo que se encuentra dentro de un entorno más amplio. Esto se hace identificando un conjunto de variables en el entorno que el sistema supervisará y controlará.
2.2.1 Definir el límite del sistema en una etapa temprana del proceso de ingeniería de requisitos identificando un conjunto preliminar de variables monitoreadas y controladas.
2.2.2 Elegir variables ambientales que existan en el entorno independientemente del sistema a desarrollar.
2.2.3 Elija variables controladas que estén bajo el control directo del sistema que se está especificando.
2.2.4 Elija las variables monitoreadas que están siendo detectadas directamente por el sistema que se está especificando.
2.2.5 Asegúrese de que las variables monitoreadas y controladas sean lo más abstractas posible y no incluir detalles de implementación.
2.2.6 Evite incorporar detalles de la interfaz del operador en las variables monitoreadas y controladas. En su lugar, defina variables monitoreadas o controladas que describan la información que se debe transmitir independientemente de su formato de presentación.
2.2.7 Definir completamente todas las interfaces físicas del sistema, incluidas las definiciones de todas las entradas discretas, todos los mensajes, todos los campos de un mensaje y todos los protocolos seguidos.

Una de las actividades más importantes en la ingeniería de requisitos es definir claramente el límite entre el sistema y su entorno. Esto proporciona una comprensión sólida de lo que se encuentra dentro del sistema que se va a construir y lo que se encuentra dentro de un entorno más amplio. Cada sistema está integrado en el entorno en el que opera, y este entorno suele ser una

colección más grande de sistemas. Sin una definición clara del límite del sistema, es muy fácil escribir requisitos que se dupliquen o entren en conflicto con los que se definen en un nivel superior, o que se pasen por alto requisitos porque se supone que los proporciona el entorno. Esto es particularmente importante cuando un sistema está siendo desarrollado por múltiples entidades.
Una forma de definir el límite del sistema es ver el sistema como un componente que interactúa con su entorno a través de un conjunto de variables monitoreadas y controladas (figura 1) [1].

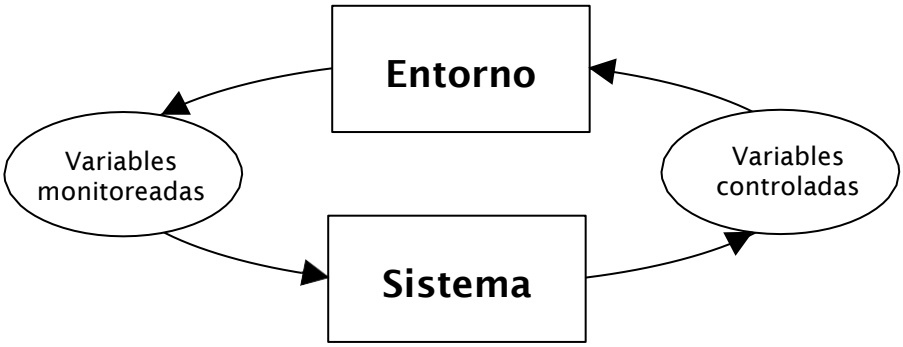


Figura 1. El sistema y su entorno

Las variables monitoreadas representan cantidades del entorno a las que responde el sistema, mientras que las variables controladas representan cantidades del entorno a las que el sistema afectará. Por ejemplo, los valores monitoreados pueden ser la altitud real de una aeronave y su velocidad aerodinámica, mientras que las variables controladas pueden ser la posición de una superficie de control, como un alerón, o el valor de altitud que se muestra en la pantalla de vuelo principal. Conceptualmente, las variables monitoreadas y controladas existen en el entorno fuera del sistema y seguirían existiendo incluso si el sistema fuera eliminado.[2] El objetivo del sistema es mantener una relación entre las variables monitoreadas y controladas que permita alcanzar los objetivos para los cuales fue creado. La definición de las variables monitoreadas y controladas y sus atributos definen los límites del sistema.

2.2.1 Identificar el Límite del Sistema con Anticipación.

La identificación de las variables controladas y monitoreadas debe iniciarse en las primeras etapas del proceso de ingeniería de requisitos, aunque no esté claro exactamente cuáles son. Tener una noción de los límites del sistema simplifica muchas de las actividades siguientes, y el proceso de identificación de las variables controladas y monitoreadas a menudo planteará preguntas adicionales. Como un ejemplo, en la tabla 3 se muestra un conjunto preliminar de variables controladas y monitoreadas para el termostato Isolette.

1 Este es el enfoque adoptado en las metodologías SCR [2 y 3] y CoRE [4 y 5]. Una noción similar se utiliza en RSML [8], donde se las denomina variables manipuladas y controladas.
2 Si bien las variables monitoreadas y controladas parecen similares a las entradas y salidas, los términos entradas y salidas se utilizan en este documento para referirse a los valores proporcionados y generados por el software en el sistema.

Tabla 3. Variables Monitoreadas y Controladas por el Termostato

Nombre	Tipo	Interpretación física
Temperatura Actual	Monitoreada	Temp. actual del aire en el interior del Isolette
Configuración del Operador		Ajustes del termostato proporcionado por el operador
Rango de Temp. Deseado		Rango deseado de temperatura del Isolette
Temp. Deseada Inferior	Monitoreada	Valor inferior del Rango de Temperatura Deseado
Temp. Deseada Superior	Monitoreada	Valor superior del Rango de Temperatura Deseado
Retroalimentación al Operadr		Información proporcionada al operador
Temperatura de Visualización	Controlada	Temperatura de visualización del Aire en la Isolette
Control de Calor	Controlada	Comando para encender o apagar la Fuente de Calor

Las variables monitoreadas y controladas de Configuración del operador y Retroalimentación del operador se agrupan en "agregados"[1] y utilizan los mismos nombres que se muestran en el diagrama de contexto de la figura A-1.[2] Si bien es un buen comienzo, esta lista de variables monitoreadas y controladas también plantea más preguntas. ¿Cuáles son los límites de los rangos de temperatura deseados? ¿La temperatura está en grados centígrados o Fahrenheit? Estas preguntas se pueden resolver en pasos posteriores. Al principio del proceso, es más importante simplemente identificar una lista preliminar de variables monitoreadas y controladas.

Práctica recomendada 2.2.1: Definir el límite del sistema en una etapa temprana del proceso de ingeniería de requisitos identificando un conjunto preliminar de variables monitoreadas y controladas.

2.2.2 Elegir Variables Ambientales.

Si bien lo ideal sería identificar correctamente las variables monitoreadas y controladas desde el principio, esto no es realista a menos que el sistema esté reemplazando a un sistema existente. Es probable que el límite del sistema no se comprenda por completo al comienzo del proyecto y es probable que cambie a medida que se traslada la funcionalidad entre sistemas durante el desarrollo. Sin embargo, es mejor tener una definición documentada incorrecta del límite del sistema que se pueda mejorar con el tiempo, que no tener ninguna definición.

Sin embargo, existen algunas reglas generales que pueden resultar útiles para identificar las variables monitoreadas y controladas. Estas deben existir en el entorno externo al sistema y deben existir independientemente del sistema en sí. Una heurística útil es preguntarse si la variable monitoreada o controlada seguiría existiendo incluso si se eliminara el sistema que se está definiendo.

Práctica recomendada 2.2.2: Elegir variables ambientales que existan en el entorno independientemente del sistema a desarrollar.

1 N.d.T.: "Agregado" proviene de "aggregates", y hace referencia a un conjunto de variables que están agrupadas de manera lógica, ya que tienen una relación entre sí. En este caso todas están relacionadas con la interfaz del operador.
2 Actualmente, el agregado de Retroalimentación al Operador solo tiene un único elemento. Se agregarán elementos de datos adicionales a medida que el ejemplo evolucione.

2.2.3 Elegir Variables Controladas.

Las variables controladas deben limitarse a aquellas que el sistema puede controlar directamente. Por ejemplo, en el termostato Isolette, la temperatura del aire no se selecciona como variable controlada, ya que el termostato solo puede afectarla indirectamente al encender la Fuente de Calor. En cambio, se elige como variable controlada el Control de Calor, que está bajo el control directo del termostato.

Práctica recomendada 2.2.3: Elegir variables controladas que estén bajo el control directo del sistema que se está especificando.

2.2.4 Elegir Variables Monitoreadas.

De manera similar, las variables monitoreadas deben corresponder a la cantidad física que detecta el sistema. Por ejemplo, al definir el comportamiento de un conjunto completo de aviónica, la altitud real de la aeronave podría ser una variable monitoreada adecuada. Al definir el comportamiento de un subsistema, la variable monitoreada adecuada puede ser una estimación de la altitud producida por otro subsistema. Elegir cuidadosamente las variables monitoreadas en el nivel correcto de abstracción para el sistema puede evitar muchas fuentes de confusión. Por ejemplo, si el propósito del sistema es determinar la altitud de la aeronave, puede necesitar obtener estimaciones de la altitud de varias fuentes. En esta situación, no tendría sentido elegir la altitud real de la aeronave como la variable monitoreada. En cambio, la altitud de cada fuente se identificaría como una variable monitoreada distinta y los requisitos del sistema definirían cómo combinar esas estimaciones en una estimación de la altitud real de la aeronave.

Práctica recomendada 2.2.4: Elija variables monitoreadas que estén siendo detectadas directamente por el sistema que se está especificando.

2.2.5 Asegúrese de que las Variables Ambientales sean Suficientemente Abstractas.

Al mismo tiempo, las variables monitoreadas y controladas deben ser lo más abstractas posible y no deben contener detalles sobre cómo se implementan. Por lo tanto, una variable monitoreada o controlada podría ser un número real con un dígito de precisión que va desde -100,0 a +50.000,0, pero no debe ser una “palabra 429” de la “Aeronautical Radio, Incorporated” (ARINC). Las variables monitoreadas y controladas no deben contener más detalles que los que el sistema necesita para realizar su función, suponiendo una interfaz perfecta con el mundo exterior.

Práctica recomendada 2.2.5: Garantizar que las variables monitoreadas y controladas sean lo más abstractas posible y no incluyan detalles de implementación.

2.2.6 Evite los Detalles de Presentación en las Variables Ambientales.

Se debe tener especial cuidado para evitar incorporar detalles de presentación en variables monitoreadas y controladas que son parte de la interfaz del operador. Por ejemplo, una pantalla podría indicar una advertencia a un piloto mostrando una altitud en amarillo y una situación peligrosa en rojo intermitente. En esta situación, en realidad hay dos variables controladas, la altitud (que sería un número) y el estado (normal, advertencia o peligro). La presentación de las variables controladas al piloto se aborda de manera más apropiada como parte del diseño

detallado de la interfaz hombre- máquina (HMI).

Hay una serie de otros atributos que se deben recopilar para cada variable monitoreada y controlada, como el tipo, el rango, la precisión y el estado. Sin embargo, es más apropiado considerar estos atributos como parte de los supuestos ambientales de los que depende el sistema y se analizan en la sección 2.4.

La definición de las variables monitoreadas y controladas y sus atributos se pueden presentar en una sola ubicación en la especificación de requisitos, similar a un diccionario de datos, o se pueden agrupar con la entidad externa con la que están asociadas.[1] En el apéndice A.3 se muestra un ejemplo completo de asociación de las variables monitoreadas y controladas con su entidad externa para el termostato Isolette. Se proporcionan ejemplos similares en el apéndice B.3 para el FCS, en el apéndice C.3 para el FGS y en el apéndice D.3 para el AP.

Práctica recomendada 2.2.6: Evitar incorporar detalles de la interfaz del operador en las variables monitoreadas y controladas. En su lugar, definir variables monitoreadas o controladas que describan la información que se debe transmitir independientemente de su formato de presentación.

2.2.7 Definir Todas las Interfaces Físicas.

Finalmente, el límite del sistema debe extenderse a una definición completa de las interfaces físicas del sistema. Esto debe identificar todas las entradas y salidas discretas, todos los mensajes, todos los campos de un mensaje y los protocolos utilizados para recibir y enviar mensajes. Si la interfaz se adhiere a una interfaz o protocolo estándar, se debe citar el estándar.

Hooks y Farry [22] recomiendan hacer esto lo antes posible. Dado que los desajustes entre interfaces son una fuente común de error, esto es razonable. Sin embargo, existe un peligro al definir las interfaces físicas sin identificar también las variables monitoreadas y controladas. Las variables monitoreadas y controladas están, por definición, en un nivel de abstracción que es estable y es poco probable que cambie a menos que cambie el dominio del problema. En contraste, las interfaces físicas definen los medios por los cuales el sistema detecta y controla estas cantidades y están en un nivel de abstracción mucho más bajo. Si las interfaces físicas se utilizan directamente como las variables monitoreadas y controladas, la especificación de requisitos se reduce inmediatamente al mismo nivel de abstracción. Esto hace que la especificación dependa demasiado de los detalles de implementación y sea menos robusta y reutilizable.

Por estas razones, tanto las variables monitoreadas y controladas como las interfaces físicas deben definirse. Si las interfaces físicas se conocen al inicio del proyecto, se pueden utilizar para ayudar a identificar las variables monitoreadas y controladas, siempre que se tenga cuidado para identificar las verdaderas magnitudes ambientales que el sistema monitoreará y controlará y para definirlas en el nivel adecuado de abstracción. Si las interfaces físicas no se conocen al inicio del proyecto, serán mucho más fáciles de definir si se han identificado las variables monitoreadas y controladas y se ha definido el comportamiento completo del sistema.

1 Otra alternativa es utilizar herramientas para generar múltiples vistas de las variables monitoreadas y controladas.

Práctica recomendada 2.2.7: Definir completamente todas las interfaces físicas del sistema, incluidas las definiciones de todas las entradas discretas, todos los mensajes, todos los campos de un mensaje y todos los protocolos seguidos.

2.3 DESARROLLAR LOS CONCEPTOS OPERACIONALES.

Consulte la tabla 4 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para desarrollar conceptos operativos.

Tabla 4. Desarrollar los conceptos operativos

Prácticas recomendadas para los niveles principal y secundario
2.3 Desarrollar los conceptos operacionales: Para todos los contextos en los que se utilizará el sistema, defina una visión de caja negra de cómo interactuará el sistema con su entorno. Esto incluye la identificación de las funciones que los operadores u otros sistemas esperan que proporcione el sistema, los órdenes en los que se pueden invocar esas funciones, los valores que se pueden proporcionar como entradas y la información que se necesita del sistema como retroalimentación. Los casos de uso son una forma popular de hacer esto.
2.3.1 Documente primero el comportamiento nominal del sistema en “día soleado”. Más adelante, como extensión de este comportamiento nominal, aborde las fallas y excepciones.
2.3.2 Incluir casos de uso que describan cómo se utiliza el sistema de interés dentro del contexto más amplio de su entorno operativo.
2.3.3 Utilice el objetivo de cada caso de uso como su título.
2.3.4 Rastrear cada caso de uso hasta los objetivos del sistema que ayuda a satisfacer.
2.3.5 Identificar el actor principal que inicia cada caso de uso, las precondiciones que deben cumplirse antes de que comience el caso de uso y las poscondiciones que deben cumplirse cuando finaliza el caso de uso.
2.3.6 Asegúrese de que cada caso de uso describa un diálogo entre el actor principal, el sistema de interés y los demás actores.
2.3.7 Vincular cada paso de un caso de uso a cualquier función del sistema que requiera.
2.3.8 Consolidar acciones que se repiten en varios casos de uso en un único caso de uso que pueda llamarse desde múltiples ubicaciones.
2.3.9 Describir situaciones excepcionales o formas en las que un caso de uso puede no cumplir su objetivo o condiciones posteriores mediante el uso de casos de excepción.
2.3.10 Describa formas alternativas en las que un caso de uso puede cumplir su objetivo y sus poscondiciones mediante el uso de cursos alternativos.

Tabla 4. Desarrollo de los conceptos operativos (continuación)

Prácticas recomendadas para los niveles principal y secundario
2.3.11 En un caso de uso, utilice los nombres de entidades externas para los actores y los nombres de variables monitoreadas y controladas en la condición previa, la condición posterior y los pasos del caso de uso.
2.3.12 Evite especificar detalles de la interfaz del operador en los conceptos operativos. En su lugar, mencione las capacidades del sistema que el operador puede invocar.
2.3.13 Actualizar el límite del sistema con cualquier nueva variable monitoreada y controlada identificada durante el desarrollo de los casos de uso.
2.3.14 A partir de los casos de uso, ensamble un conjunto preliminar de funciones que proporcionará el sistema.

Los conceptos operativos son guiones o escenarios que describen cómo se utilizará el sistema [22]. Son un paso útil en la progresión desde la descripción general del sistema hasta los requisitos detallados. Los conceptos operativos consideran el sistema como una caja negra y describen cómo interactúa con sus operadores y otros sistemas en su entorno. Ayudan a identificar qué funciones esperan los operadores que realice el sistema, los valores que los operadores pueden proporcionar como entradas y la información que los operadores necesitan como retroalimentación. También ayudan a identificar la secuencia en la que los operadores pueden invocar las funciones del sistema durante el funcionamiento normal y la respuesta de los operadores cuando el sistema se comporta de manera diferente a la esperada.

Lo ideal es que los conceptos operativos se redacten en un estilo natural e intuitivo que todas las partes interesadas puedan comprender. Esto ayuda a generar consenso entre las diferentes partes interesadas y a identificar funciones del sistema que pueden haberse pasado por alto. Dado que los conceptos operativos se centran en cómo interactúan los operadores y otros sistemas con el sistema, a menudo revelarán problemas de interfaz del operador, en particular en términos de qué información se necesita del operador, cuándo pueden invocar funciones los operadores y qué información necesitan los operadores del sistema.

Al mismo tiempo, los conceptos operativos deben evitar definir detalles específicos de la HMI, ya que esto limitará la aplicabilidad de los requisitos a esa interfaz. En muchos sistemas de aviónica, la HMI se ha vuelto tan compleja que varios accidentes se han atribuido a un diseño deficiente de la interfaz del operador [24 y 32 a 37]. Por esta razón, el diseño de la HMI se considera a menudo una actividad integrada que se lleva a cabo en paralelo con el desarrollo del propio sistema [16 y 24].

Por ejemplo, en el termostato Isolette, el operador debe proporcionar el Rango de Temperatura Deseado, pero ¿debería introducirse mediante un teclado o mediante la configuración de punteros en un dial? ¿La alarma debería ser un timbre o una luz intermitente? Si bien estas decisiones son importantes, no es necesario que se resuelvan en la etapa de conceptos operativos. Desafortunadamente, una discusión completa del diseño de la HMI es un tema complejo que está más allá del alcance de este Manual, y las prácticas presentadas en este documento se limitan a identificar y documentar solo los conceptos operativos de alto nivel. Se remite al lector a las referencias 24, 33, 38 y 39 para obtener orientación adicional en el diseño de la HMI.

Los casos de uso son una forma popular de identificar y documentar las interacciones entre un sistema, sus operadores y otros sistemas. La literatura sobre casos de uso es amplia y cubre toda la gama, desde los requisitos de alto nivel hasta el diseño detallado. Algunas de las referencias más conocidas incluyen las referencias 17, 18 y 19. Si bien las convenciones para el formato de los casos de uso son similares, existen numerosos estilos que introducen distintos grados de rigor y formalidad. Sin embargo, los casos de uso parecen ser especialmente apropiados para su uso temprano en el proceso REM, ya que ayudan a comprender y describir cómo los operadores y otros sistemas interactuarán con el sistema. Uno de sus atractivos es que se pueden utilizar de manera relativamente informal para obtener una mejor comprensión de las funciones del sistema que necesitan los operadores. En efecto, proporcionan una validación temprana del comportamiento del sistema.

2.3.1 Documentar el Comportamiento del Sistema en Días Soleados.

En la terminología de los casos de uso, los operadores, el sistema de interés y otros sistemas que interactúan con el sistema se denominan actores. Cada caso de uso describe un diálogo de solicitudes y acciones entre los actores y el sistema para lograr algún objetivo. El actor que inicia el caso de uso se denomina actor principal. Una condición previa identifica las condiciones que deben ser verdaderas antes de que se inicie el caso de uso, y una condición posterior identifica las condiciones que deben ser verdaderas cuando el caso de uso se completa con éxito. El escenario de éxito principal describe un diálogo de día soleado en el que nada sale mal. Si hay más de una forma de lograr el mismo objetivo, se pueden definir cursos alternativos. Las formas en que un caso de uso puede no cumplir su objetivo o sus condiciones posteriores se manejan a través de casos de excepción. El caso de uso Funcionamiento Normal del Isolette[1], que se muestra en la figura 2, ilustra muchos de estos conceptos.

Este caso de uso describe el funcionamiento del termostato Isolette en días soleados sin tener en cuenta las condiciones de falla o las excepciones. Al principio del proceso, es aceptable simplemente identificar la funcionalidad nominal requerida del sistema. Luego, las condiciones de falla y las excepciones se abordan como una extensión de este comportamiento nominal. En particular, esto se puede hacer durante la evaluación de seguridad del sistema y el diseño funcional del sistema.

Práctica recomendada 2.3.1: documentar primero el comportamiento nominal del sistema en “días soleados”. Más adelante, como extensión de este comportamiento nominal, abordar las fallas y excepciones.

1 Este caso de uso difiere del caso de uso Funcionamiento Normal del Isolette del apéndice A.2.1. El ejemplo del apéndice incluye cambios realizados en secciones posteriores para manejar las restricciones de implementación y los casos de excepción.

Caso de uso A.2.1: Funcionamiento normal del Isolette

Este caso de uso describe el funcionamiento normal de la Isolette por parte de la Enfermera.

Objetivos del sistema relacionados: G1

Actor principal: Enfermera

Condición previa:

El Bebé está listo para ser colocado en la Incubadora.
La Incubadora y el termostato están apagados.

Postcondición:

Se retira al Bebé de la Incubadora.
La Incubadora y el termostato se apagan.

Principal escenario de éxito:

1. La Enfermera enciende la Isolette.
2. Isolette enciende el termostato.
3. El termostato se inicializa y entra en su modo de funcionamiento normal. [Función del Sistema A.5.1.2]
4. La Enfermera configura la Incubadora según las necesidades del Bebé. [UC A.2.2]
5. La Enfermera espera hasta que la temperatura actual se encuentre dentro del rango de temperatura deseado. [Función del sistema A.5.1.1]
6. La Enfermera coloca al Bebé en la Incubadora.
7. La Isolette mantiene la Temperatura Deseada. [UC A.2.3]
8. La Enfermera confirma que la temperatura actual se encuentra dentro del rango de temperatura deseado durante las rondas. [Función del sistema A.5.1.1]
9. La Enfermera retira al Bebé.
10. La Enfermera apaga la Isolette.
11. La Isolette apaga el termostato.

UC = Caso de uso

Figura 2. Ejemplo de caso de uso

2.3.2 Incluir Cómo se Utiliza el Sistema en su Entorno Operativo.

Tenga en cuenta que el caso de uso del día soleado describe el funcionamiento del termostato al explicar cómo se utiliza dentro del contexto más amplio del funcionamiento normal de la Isolette. Este es otro beneficio importante de los casos de uso, ya que describen el funcionamiento del sistema dentro de su entorno en un formato muy simple. Por ejemplo, este caso de uso deja en claro, en el escenario de éxito principal, que la Isolette enciende y apaga automáticamente el termostato, es decir, la Enfermera no tiene que encender y apagar también el termostato. Sería difícil transmitir esto en un caso de uso que se centrara solo en la interfaz con el termostato.

Práctica recomendada 2.3.2: Incluir casos de uso que describan cómo se utiliza el sistema de interés dentro del contexto más amplio de su entorno operativo.

2.3.3 Utilice el Objetivo del Caso de Uso como Título.

Tenga en cuenta también que el título del caso de uso describe el objetivo del caso de uso, es decir, define el funcionamiento normal de Isolette. Esto facilita la búsqueda y la referencia del caso de uso deseado.

Práctica recomendada 2.3.3: Utilice el objetivo de cada caso de uso como título.

2.3.4 Rastrear Cada Caso de Uso hasta los Objetivos del Sistema.

El caso de uso se remonta al objetivo G1 del termostato. El seguimiento de los casos de uso hasta los objetivos ayuda a satisfacer y garantizar que el caso de uso describa un comportamiento necesario. También facilita el mantenimiento si los objetivos del sistema o el caso de uso cambian.

Práctica recomendada 2.3.4: Rastrear cada caso de uso hasta los objetivos del sistema que ayuda a satisfacer.

2.3.5 Identificar Actor Principal, Precondiciones y Poscondiciones.

El actor principal que inicia este caso de uso es la Enfermera. La condición previa es que el Bebé esté listo para ser colocado en la Incubadora y que la Incubadora y el termostato estén apagados. La condición previa identifica las condiciones que deben ser verdaderas al inicio del caso de uso. Normalmente, la condición posterior hace explícitos los cambios que el caso de uso provocará en el entorno. Dado que este caso de uso describe un ciclo completo de uso normal, la condición posterior es la misma que la condición previa.

Práctica recomendada 2.3.5: Identificar en cada caso de uso el actor principal que inicia cada caso de uso, las precondiciones que deben cumplirse antes de que comience el caso de uso y las postcondiciones que deben cumplirse cuando finaliza el caso de uso.

2.3.6 Asegúrese de que Cada Caso de Uso Describa un Diálogo.

El escenario de éxito principal describe las interacciones entre la Enfermera y el termostato cuando todo va según lo planeado. En general, los casos de uso deben describir un diálogo entre el actor principal, el sistema de interés y los demás actores en el que cada línea describe la acción de un participante diferente. Si un participante domina el diálogo, probablemente se deba revisar el caso de uso.

Práctica recomendada 2.3.6: Garantizar que cada caso de uso describa un diálogo entre el actor principal, el sistema de interés y los demás actores.

2.3.7 Vincular los Pasos del Caso de Uso a las Funciones del Sistema.

Una de las ventajas de los casos de uso es que ayudan a identificar las funciones que proporcionará el sistema. Por ejemplo, en el paso 5 de la figura 2, la Enfermera confirma que la Temperatura Actual está dentro del Rango de Temperatura Deseado antes de colocar al Bebé en la Incubadora. Esto revela que el operador espera que el sistema muestre la Temperatura Actual. Esto se indica vinculando esa acción a la Interfaz del Regulador de Gestión de Funciones del Sistema (A.5.1.1) que proporciona esta capacidad. Esto se hace para que, si se debe cambiar una Función del Sistema, sea fácil encontrar y revisar todas las formas en que se utiliza la función. También permite que un lector del caso de uso busque la Función del Sistema para examinar más de cerca la interacción entre el caso de uso y el sistema.

Práctica recomendada 2.3.7: Vincular cada paso de un caso de uso con cualquier función del sistema que se requiera.

2.3.8 Consolidar Acciones Repetidas en un Único Caso de Uso.

En el paso 7, el caso de uso llama a otro caso de uso, Mantener la Temperatura Deseada (UC A.2.3), que describe cómo el termostato mantendrá la Temperatura Actual dentro del Rango de Temperatura Deseado. Dividir los casos de uso de esta manera permite que las acciones que se utilizan en varios lugares se consoliden en un solo caso de uso y luego se reutilicen. Esto también hace que los casos de uso sean más compactos y comprensibles.

Práctica recomendada 2.3.8: Consolidar acciones que se repiten en varios casos de uso en un único caso de uso que pueda llamarse desde múltiples ubicaciones.

2.3.9 Describir Situaciones Excepcionales como Casos de Excepción.

Los casos de excepción se utilizan para describir el comportamiento del sistema y los actores cuando se produce una excepción al comportamiento normal de un día soleado. En las secciones 2.6.4 a 2.6.6 se presenta un análisis completo de los casos de excepción y, en el caso de uso completo de Operación Normal de la Isolette en el apéndice A.2.1, se proporciona una ilustración de un caso de excepción. Este caso de uso se amplió para incluir cambios realizados para adaptarse a las restricciones de implementación y para manejar excepciones. En el paso 5 del Escenario de Éxito Principal, el caso de uso hace referencia al caso de excepción de Incapacidad para Mantener la Temperatura Deseada en el apéndice A.2.6. El caso de excepción A.2.6 describe cómo responde la Enfermera a una situación en la que la Temperatura Actual en la Incubadora no se encuentra en el Rango de Temperatura Deseado. Esto se identifica como un caso de excepción porque normalmente no ocurre durante el escenario de éxito principal y porque impediría que se satisfaga la condición posterior del caso de uso. En todos los demás aspectos, se trata como cualquier otro caso de uso.

Práctica recomendada 2.3.9: Describir situaciones excepcionales o formas en las que un caso de uso puede no cumplir su objetivo o condiciones posteriores mediante el uso de casos de excepción.

2.3.10 Describir Formas Alternativas de Satisfacer las Poscondiciones como Cursos Alternativos.

Si existen otras secuencias que se pueden tomar de manera rutinaria para cumplir con las condiciones posteriores, estas se pueden identificar como cursos alternativos. Aunque el ejemplo

actual es tan simple que no contiene cursos alternativos, se puede encontrar un ejemplo de un curso alternativo en el caso de excepción de Falla al Mantener la Temperatura Deseada que se muestra en el apéndice A.2.6. En el paso 1 del Escenario de Éxito Principal, la Enfermera intenta corregir el problema de una Isolette que no puede mantener la Temperatura Actual dentro del Rango de Temperatura Deseado, por ejemplo, cerrando la puerta de la Isolette. La situación en la que la Enfermera no puede corregir el problema se describe en el curso alternativo 1, en el que la Enfermera obtiene y configura una nueva Isolette. Esto también ilustra cómo los cursos alternativos o los casos de excepción se pueden presentar como adiciones al caso de uso principal, en lugar de casos de uso completamente separados.

Práctica recomendada 2.3.10: Describir formas alternativas en las que un caso de uso puede cumplir su objetivo y sus poscondiciones mediante el uso de cursos alternativos.

2.3.11 Utilizar Nombres de Entidades Externas o Variables Ambientales.

Cabe señalar que una gran proporción de las palabras reales en el caso de uso se refieren a las entidades externas identificadas en la sección 2.1, o a las variables monitoreadas y controladas identificadas en la sección 2.2. Esto es de esperarse, ya que los casos de uso tratan el sistema como una caja negra sin estructura interna. Las variables monitoreadas y controladas proporcionan una forma natural de describir cómo las entidades externas (los actores) interactúan con el sistema. Hacer esto ayuda a mantener la coherencia con el resto de la especificación de requisitos.

Práctica recomendada 2.3.11: En un caso de uso, utilice los nombres de entidades externas para los actores y los nombres de variables monitoreadas y controladas en la condición previa, la condición posterior y los pasos del caso de uso.

2.3.12 Evitar los Detalles de la Interfaz del Operador.

Tenga en cuenta que los casos de uso de la figura 2 y el apéndice A.2 evitan incorporar detalles de la interfaz del operador cuando se invocan las capacidades del sistema. Por ejemplo, el caso de uso Configurar la Isolette del apéndice A.2.2 no especifica si la Enfermera ingresa los rangos de temperatura configurando punteros en un dial o presionando teclas en un teclado. Si bien los detalles de la interfaz física del operador son extremadamente importantes, esas decisiones deben tomarse como parte del diseño de la HMI. Es demasiado pronto para comprometerse con interfaces físicas específicas durante el desarrollo de los conceptos operativos. Evitar los detalles de la interfaz física también hace que los casos de uso sean más generales y aplicables a una gama más amplia de sistemas.

Práctica recomendada 2.3.12: Evite especificar detalles de la interfaz del operador en los conceptos operativos. En su lugar, indique las capacidades del sistema que el operador debe invocar.

2.3.13 Actualizar el Límite del Sistema.

Cuando se utilizan de esta manera, los casos de uso son una excelente manera de obtener una validación temprana de los requisitos y documentar cómo los operadores y otros sistemas (los actores) interactúan con el sistema que se está desarrollando. Su uso puede ayudar a identificar inconsistencias y descuidos y proporciona una visión más profunda de cómo se utilizará el

sistema. También ayudan a identificar entidades externas pasadas por alto y variables monitoreadas y controladas. Por ejemplo, el paso 3 del Escenario de Éxito Principal de la figura 2 analiza la inicialización del termostato y su entrada en funcionamiento normal. Esto implica que la Enfermera puede determinar cuándo el termostato entra en funcionamiento normal para poder configurarlo, lo que a su vez implica la necesidad de una variable controlada adicional para mostrar el estado del termostato. La lista actualizada de variables monitoreadas y controladas se muestra en la tabla 5.

Tabla 5. Variables controladas y monitoreadas del termostato revisadas.

Nombre	Tipo	Interpretación física
Temperatura Actual	Monitoreada	Temp. actual del aire en el interior del Isolette
Configuración del Operador		Ajustes del termostato proporcionados por el operador
Rango de Temp. Deseado		Rango deseado de temperatura del Isolette
Temp. Deseada Inferior	Monitoreada	Valor inferior del Rango de Temperatura Deseado
Temp. Deseada Superior	Monitoreada	Valor superior del Rango de Temperatura Deseado
Retroalimentación al Operadr		Información proporcionada al operador
Temperatura de Visualización	Controlada	Temperatura del aire mostrada del Isolette
Estado del Termostato	Controlada	Estado operativo actual del termostato
Control de Calor	Controlada	Comando para encender o apagar la Fuente de Calor

Si se desea, se puede recopilar información adicional en los casos de uso. Algunos ejemplos incluyen el alcance del caso de uso, los actores secundarios y los objetivos de cada actor en el caso de uso. El estilo presentado aquí se basa en gran medida en el que se encuentra en la referencia 17. También se puede encontrar más información en las referencias 18 y 19. La totalidad de los casos de uso para el termostato Isolette se muestran en el apéndice A.2. Estos casos de uso se ampliaron, por lo que los casos de uso de días soleados A.2.1 a A.2.3 incluyen referencias a los casos de uso de excepción A.2.4 a A.2.6. También hacen referencia a capacidades que aún no se han analizado, como la activación de la alarma. Los casos de uso de ejemplo del FCS se muestran en el apéndice B.2.

Práctica recomendada 2.3.13: Actualizar el límite del sistema con cualquier nueva variable monitoreada y controlada identificada durante el desarrollo de los casos de uso.

2.3.14 Ensamblar un Conjunto Preliminar de Funciones del Sistema.

Al crear los casos de uso, se debe elaborar una lista preliminar de funciones del sistema. Estas se utilizarán como entrada para la actividad descrita en la sección 2.5. Esta lista preliminar de funciones del sistema para el termostato Isolette se muestra en la tabla 6.

Tabla 6. Conjunto preliminar de funciones del termostato Isolette

Encender y apagar el termostato	Indicar el estado del termostato
Establecer la Temperatura Deseada	Encender y apagar la Fuente de Calor
Mostrar la Temperatura Actual	

Práctica recomendada 2.3.14: A partir de los casos de uso, ensamblar un conjunto preliminar de funciones que proporcionará el sistema.

2.4 IDENTIFICAR LOS SUPUESTOS AMBIENTALES.

Consulte la tabla 7 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para identificar supuestos ambientales.

Tabla 7. Identificar los Supuestos Ambientales

Prácticas recomendadas para los niveles principal y secundario
2.4 Identificar los Supuestos Ambientales: Cada sistema hace suposiciones específicas sobre el entorno en el que operará. Algunas de estas suposiciones no son más que los tipos, rangos y unidades de las entradas que aceptará y las salidas que producirá. A menudo, el comportamiento correcto del sistema depende de suposiciones más complejas sobre su entorno. En realidad, se trata de requisitos que el sistema impone a su entorno. La identificación de las suposiciones ambientales de un sistema es esencial para el mantenimiento y para permitir su reutilización. La falta de identificación de las suposiciones ambientales y el mal uso posterior del sistema es una causa común de falla del sistema.
2.4.1 Definir el tipo, rango, precisión y unidades requeridas para todas las variables monitoreadas y controladas como parte de los supuestos ambientales del sistema.
2.4.2 Proporcionar una justificación que documente por qué se incluyen los supuestos ambientales.
2.4.3 Organizar los supuestos ambientales junto con la entidad externa que restringen de modo que sea fácil identificar todas las obligaciones impuestas a cada entidad externa.
2.4.4 Si un supuesto ambiental define una relación entre varias entidades externas, defina una entidad externa responsable de garantizar que se cumpla el supuesto y asocie el supuesto con esa entidad.
2.4.5 Defina un atributo de estado para cada variable monitoreada. Cada valor de la variable de estado debe corresponder a un comportamiento diferente del sistema. El estado inicial de la variable monitoreada debe garantizar que la variable monitoreada no se utilice hasta que se detecte al menos una vez.

Los supuestos ambientales son los supuestos sobre el entorno del que depende un sistema para su correcto funcionamiento. Se pueden especificar como una relación matemática entre las variables controladas y las monitoreadas.[1] Esta relación puede ser tan simple como los tipos, rangos y unidades de las variables monitoreadas y controladas (el sistema asume que la altitud nunca será menor a 100 pies bajo el nivel del mar o mayor a 50,000 pies sobre el nivel del mar), o tan compleja como un mapeo completo de las variables controladas a las variables monitoreadas que describe el comportamiento completo del ambiente.

1 En las metodologías SCR [2 y 3] y CoRE [4 y 5] esto se denomina relación NAT (natural).

Identificar las suposiciones que el sistema hace sobre su entorno es una parte tan importante del REM como especificar el comportamiento requerido del sistema. Hooks y Farry citan hechos o suposiciones incorrectas como la forma más común de errores de requisitos [22]. La falta de identificación y documentación de las suposiciones ambientales también han sido la causa de varias fallas dramáticas del sistema. En algunos casos, la falla ocurrió porque un subsistema desarrollado por un equipo no cumplía con las suposiciones hechas por otro equipo. Un primer paso esencial para prevenir tales fallas es identificar y documentar estas suposiciones.

Identificar los supuestos ambientales también es esencial para la reutilización de un componente. En varios casos, se han producido fallos dramáticos porque un sistema existente se reutilizó en un entorno diferente y los desarrolladores desconocían los supuestos que habían hecho los desarrolladores originales. Documentar los supuestos ambientales es un requisito necesario para la reutilización de componentes.

2.4.1 Definir el Tipo, el Rango, la Precisión y las Unidades.

Si bien es deseable que un sistema dependa de la menor cantidad posible de supuestos ambientales, no es posible diseñar un sistema que no haga algunos supuestos ambientales. Ningún sistema puede aceptar una gama infinita de entradas y, en algún momento, se deben seleccionar al menos los tipos y rangos de las entradas y salidas. Estos supuestos deben documentarse junto con la justificación de los valores seleccionados (véase la sección 2.11). A menudo, hay otros supuestos más complejos que también deben documentarse.

Por ejemplo, antes de construir el termostato Isolette del apéndice A, es necesario definir los tipos, rangos, precisión y unidades de las variables monitoreadas y controladas. Algunas de las suposiciones ambientales para la variable monitoreada Temperatura Actual se muestran en la tabla 8.

Tabla 8. Supuestos ambientales para la variable de Temperatura Actual monitoreada

Nombre	Tipo	Rango	Unidades	Interpretación física
Temperatura Actual	Real	[68.0..110.0]	°F	Temp actual del aire en el interior del Isolette

En ellos se establece que la Temperatura Actual proporcionada por el Sensor de Temperatura se supone que es un número real entre 68,0 y 110,0, con una precisión de al menos 0,1 que expresa una temperatura en grados Fahrenheit. En efecto, esto vuelve a imponer requisitos al Sensor de Temperatura y establece un contrato con él. Por ejemplo, si el Sensor de Temperatura proporciona la temperatura en grados Celsius, es poco probable que el termostato realice su función correctamente. Por supuesto, este acuerdo se puede cambiar y la Temperatura Actual podría proporcionarse en grados Celsius y el termostato podría hacer una conversión trivial a grados Fahrenheit. El punto importante es que las suposiciones ambientales estén claramente documentadas para que se conozcan todas las obligaciones y luego se actúe en consecuencia.

Práctica recomendada 2.4.1: Definir el tipo, rango, precisión y unidades requeridas para todas las variables monitoreadas y controladas como parte de los supuestos ambientales del sistema.

2.4.2 Justificar las Suposiciones.

Sin embargo, incluso para algo tan simple como la Temperatura Actual, surgen varias preguntas de inmediato. ¿Por qué la temperatura se especifica en grados Fahrenheit en lugar de grados Celsius? ¿Por qué el rango es de 68,0° a 110,0°F? ¿Por qué se especifica una precisión de 0,1°F? Esta información debe proporcionarse documentando la justificación de cada elección (consulte la sección 2.11). Para la Temperatura Actual, estas incluyen:

- La Temperatura Actual se proporcionará al Termostato en grados Fahrenheit.

  Justificación: Coherencia con la Interfaz del Operador de supuestos ambientales (EA-OI)-1 (Todas las temperaturas se mostrarán en grados Fahrenheit).

- La Temperatura Actual se detectará con una precisión de ±0,1 °F.

  Justificación: Es necesaria una precisión y exactitud de 0,1 °F para garantizar que el Termostato pueda encender y apagar la Fuente de Calor con la suficiente rapidez para mantener el Rango de Temperatura Deseado.

- La Temperatura Actual cubrirá el rango de al menos 68,0° a 103,0°F.

  Justificación: Este es el rango de funcionamiento especificado del Isolette. El extremo inferior de este rango es útil para monitorear un Isolette que se está calentando hasta el Rango de Temperatura Deseado. El extremo superior se establece 1° por encima de la Temperatura Deseada Superior para garantizar que se detecte la Temperatura Actual en todo el Rango de Temperatura Deseado.[1]

La justificación (sección 2.11) proporciona una base para debatir si se pueden modificar los supuestos ambientales. También proporciona información valiosa para los desarrolladores y para el mantenimiento futuro.

Práctica recomendada 2.4.2: Proporcionar una justificación que documente por qué se incluyen los supuestos ambientales.

2.4.3 Organizar Supuestos que Restringen una Sola Entidad.

Una forma útil de organizar los supuestos ambientales es presentarlos con una descripción más detallada de la entidad externa que restringen. Esto facilita la revisión de todas las obligaciones que recaen sobre cada entidad externa. En la especificación completa de requisitos del termostato Isolette, se crea una sección que describe cada entidad externa y enumera los supuestos ambientales que se hacen sobre ella (consulte el apéndice A.3).

Práctica recomendada 2.4.3: Organizar los supuestos ambientales junto con la entidad externa que restringen de modo que sea fácil identificar todas las obligaciones impuestas a cada entidad externa.

[1] En la especificación final, el rango superior se establece en 106,0 °F para superar la Temperatura de Alarma Superior.

2.4.4 Organizar Supuestos que Restringen Varias Entidades.

Algunas suposiciones ambientales definen relaciones más complejas entre varias variables ambientales. Por ejemplo, algunas suposiciones sobre el Rango de Temperatura Deseado incluyen:

- La Temperatura Mínima Deseada siempre será ≥ 97 °F.

  Justificación: Exponer al Bebé a temperaturas inferiores a 97 °F puede provocar una pérdida excesiva de calor y una caída de la frecuencia cardíaca secundaria a acidosis metabólica.

- La Temperatura Mínima Deseada siempre será menor o igual a la Temperatura Deseada Superior menos 1 °F.

  Justificación: si la Temperatura Mínima Deseada es mayor o igual que la Temperatura Deseada Superior, no queda claro si la Fuente de Calor debe estar encendida o apagada. Esto puede provocar un funcionamiento cíclico excesivo de la Fuente de Calor.

- La Temperatura Deseada Superior siempre será ≤ 100°F.

  Justificación: Exponer al Bebé a temperaturas superiores a 100 °F puede dar lugar a un diagnóstico incorrecto de fiebre, lo que requiere una evaluación agresiva (cultivo de sangre y punción lumbar) y un tratamiento para la infección.

La razonabilidad de estas suposiciones depende de cómo se implemente la interfaz del operador. Si la interfaz consiste en un dial con indicadores que solo se pueden ajustar en incrementos de 1°, lo que garantiza mediante su construcción mecánica que se cumplan las suposiciones, pueden ser muy razonables. Si los rangos de temperatura se ingresan mediante un teclado digital, la interfaz física puede no garantizar que se cumplan las suposiciones. En ese caso, será necesario reforzar los requisitos del sistema para garantizar que el Termostato funcione correctamente.

Lo importante es que los supuestos ambientales de los que depende el sistema estén claramente documentados y puedan comprobarse. Cuando se combinan con los requisitos del sistema, forman efectivamente un contrato que permite que los componentes se desarrollen de forma independiente. Los supuestos ambientales definen las obligaciones que debe cumplir el entorno, incluidos otros sistemas con los que interactúa el sistema. Si se cumplen, entonces el sistema en desarrollo debe satisfacer sus requisitos. En el desarrollo de software, los supuestos ambientales a menudo se denominan precondiciones y los requisitos, poscondiciones.

Como se mencionó anteriormente, las suposiciones ambientales también proporcionan documentación importante para el mantenimiento y la reutilización posteriores. Si el sistema se utilizará en un entorno diferente, se pueden verificar las suposiciones ambientales para asegurarse de que aún se cumplan. Por ejemplo, si la interfaz mecánica del operador del termostato Isolette se reemplazará por una interfaz digital moderna (que puede no garantizar todas las suposiciones ambientales realizadas por el termostato), debería ser sencillo para los desarrolladores encontrar y revisar primero las suposiciones ambientales relevantes.

Otras fuentes comunes de supuestos ambientales relacionan las variables monitoreadas con las variables controladas (en efecto, un modelo parcial de la planta que se está controlando) o limitan la velocidad a la que pueden cambiar los valores monitoreados. Por ejemplo, el Termostato de la Isolette depende de que la Temperatura Actual en la Isolette no cambie demasiado rápido para que el Termostato pueda encender y apagar la Fuente de Calor lo suficientemente rápido para mantener la Temperatura Actual dentro del Rango de Temperatura Deseado. Estas suposiciones ambientales se expresan de la siguiente manera:

- Cuando la Fuente de Calor está encendida y el Isolette está correctamente cerrado, la Temperatura Actual aumentará a una velocidad de no más de 1 °F por minuto.

- Cuando la Fuente de Calor está apagada y el Isolette está correctamente cerrado, la Temperatura Actual disminuirá a una velocidad de no más de 1 °F por minuto.

Sin embargo, estas suposiciones plantean el problema de con qué entidad externa deben asociarse. En realidad, no son suposiciones sobre el Sensor de Temperatura, ni sobre la Fuente de Calor, sino que abarcan ambas entidades. Una heurística útil es que deben agruparse con la entidad que es responsable de garantizar que se cumplan. Esta sería la propia Isolette, que incluye el Sensor de Temperatura, la Fuente de Calor y el Termostato. Por este motivo, la especificación de requisitos debe incluir una entidad externa para la Isolette que contenga estas suposiciones ambientales. Se puede encontrar un ejemplo de esto en el apéndice A.3.1.

Práctica recomendada 2.4.4: Si un supuesto ambiental define una relación entre varias entidades externas, defina una entidad externa responsable de garantizar que se cumpla el supuesto y asocie el supuesto con esa entidad.

2.4.5 Definir un Atributo de Estado para cada Variable Monitoreada.

Una forma más sutil de suposiciones ambientales está relacionada con las variables monitoreadas y el nivel de confianza que se puede depositar en su valor. Las variables monitoreadas deben ser detectadas por el sistema, y el comportamiento del sistema puede ser diferente cuando no es capaz de detectar el valor de una variable monitoreada. Por ejemplo, algunas variables monitoreadas no deben ser confiables después del encendido hasta que se detectan por primera vez, es decir, el valor de la variable monitoreada es desconocido. En otros casos, la variable monitoreada no debe ser confiable si su valor no se ha actualizado recientemente, es decir, el valor de la variable monitoreada está desactualizado. Se han rastreado varios accidentes graves a sistemas que dependían del valor de variables monitoreadas desconocidas o desactualizadas [24].

Una forma de manejar esto es asociar un atributo de estado con cada variable monitoreada. Los valores posibles del atributo de estado deben estar en correspondencia uno a uno con los diferentes comportamientos del sistema especificado. Por ejemplo, si el sistema se comporta de una manera cuando se puede confiar en la variable monitoreada y de otra manera cuando no se puede confiar, su estado solo necesita tomar dos valores: válido o inválido. Si el sistema se comporta de una manera cuando se puede confiar en la variable monitoreada, de otra manera cuando es desconocida y de una tercera manera cuando la variable monitoreada se vuelve obsoleta, entonces debe haber tres valores del atributo de estado: válido, desconocido y obsoleto. El estado inicial de la variable monitoreada debe establecerse en un valor que indique que no se puede confiar en ella hasta que se detecte al menos una vez.

Por supuesto, en la etapa actual, puede que no haya suficiente información para determinar cuáles serán todos los valores del estado en el sistema final. En ese caso, se puede especificar un rango inicial de válidos e inválidos y revisarlo a medida que evolucionen los requisitos.

Práctica recomendada 2.4.5: Defina un atributo de estado para cada variable monitoreada. Cada valor de la variable de estado debe corresponder a un comportamiento diferente del sistema. El estado inicial de la variable monitoreada debe garantizar que no se utilice hasta que se detecte al menos una vez.

2.4.6 Resumen.

En resumen, los supuestos ambientales deben identificar todos los comportamientos ambientales de los que depende el sistema para funcionar correctamente. Este es un paso básico para permitir que los componentes se desarrollen de forma independiente. Los supuestos ambientales deben estar asociados con la entidad externa responsable de garantizar que se cumplan. Esto facilita la identificación de los supuestos que se violarían si el sistema se utilizara en un entorno diferente. Por otro lado, si los supuestos ambientales se pueden debilitar o eliminar fortaleciendo el comportamiento del sistema que se está desarrollando, es mejor que depender de los supuestos ambientales. Dicho de otra manera, un sistema robusto tendrá menos dependencias del entorno que un sistema frágil. Sin embargo, todo sistema realizable siempre tendrá algunos supuestos ambientales, incluso si consisten en nada más que los tipos, rangos y unidades de las variables monitoreadas y controladas.

2.5 DESARROLLAR LA ARQUITECTURA FUNCIONAL.

Consulte la tabla 9 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para desarrollar la Arquitectura Funcional.

Tabla 9. Desarrollar la Arquitectura Funcional

Prácticas recomendadas para los niveles principal y secundario
2.5 Desarrollar la Arquitectura Funcional: Para mejorar la legibilidad de los requisitos y hacerlos robustos frente a los cambios, los requisitos se organizan en funciones que están relacionadas de manera lógica con dependencias mínimas entre ellas. Para permitir que los requisitos se adapten a sistemas grandes, las funciones se dividen en funciones más pequeñas.
2.5.1 Organizar las funciones del sistema en grupos que estén estrechamente relacionados y que probablemente cambien juntos.
2.5.2 Utilice diagramas de flujo de datos para representar gráficamente las funciones del sistema y sus dependencias.

Tabla 9. Desarrollo de la Arquitectura Funcional (Continuación)

Prácticas recomendadas para los niveles principal y secundario
2.5.3 Minimizar las dependencias entre funciones garantizando que la información compartida entre funciones represente conceptos estables y de alto nivel del dominio del problema que tengan pocas probabilidades de cambiar. Llevar las dependencias volátiles lo más abajo posible en la jerarquía de funciones.
2.5.4 Definir el tipo, rango, precisión y unidades de todas las variables internas introducidas para especificar dependencias de datos entre funciones.
2.5.5 Organizar especificaciones de requisitos grandes en múltiples niveles anidando funciones y dependencias de datos.
2.5.6 Si se proporcionan requisitos de alto nivel, defina únicamente lo que se puede indicar en ese nivel de la jerarquía. No utilice términos definidos en niveles inferiores de la jerarquía.
2.5.7 Si se proporcionan requisitos de alto nivel para funciones, evite incorporar una justificación en el requisito en un intento de especificar detalles que se especificarían de manera más apropiada en una función de nivel inferior.

Por ejemplo, por muy pequeño que sea el Termostato Isolette, toda la especificación de requisitos se puede escribir en unas pocas páginas. En el caso de los sistemas reales, no es así. Por ejemplo, cuando se aplicó la metodología CoRE a la aviónica del avión C-130J, se monitorizaron y controlaron más de 1600 variables [6]. Para que sea utilizable, una especificación de requisitos para un sistema de cualquier tamaño debe estar organizada de alguna manera.

La mejor organización puede depender de cómo se utilizarán los requisitos. Una estructura puede mejorar la legibilidad de los requisitos, mientras que otra puede facilitar la definición de una familia de productos. Estos objetivos pueden entrar en conflicto entre sí: la organización para respaldar una familia de productos puede producir una especificación que sea más difícil de entender que una diseñada para un solo miembro de la familia. Las herramientas automatizadas pueden ayudar en este sentido, produciendo diferentes vistas de la misma especificación subyacente según sea necesario.

2.5.1 Organizar las Funciones del Sistema en Grupos Relacionados.

La Arquitectura Funcional se desarrolla mediante un proceso que identifica recursivamente las funciones que debe proporcionar el sistema, agrupando las funciones que están relacionadas lógicamente y que es probable que cambien juntas. Lo ideal sería utilizar esta estructura para organizar los requisitos detallados del sistema de manera que sean legibles y robustos ante los cambios.[1] En realidad, otros factores, como el manejo de los requisitos de seguridad del sistema o el manejo de las restricciones de implementación, generalmente requieren que se modifique esta estructura. Estas cuestiones se explican en la sección 2.6, que describe un enfoque iterativo para modificar la arquitectura funcional para manejar los requisitos de seguridad y las restricciones de implementación. Sin embargo, el punto de partida para esa actividad se produce en este paso.

[1] Esto es muy similar a los principios utilizados para organizar los requisitos en la metodología CoRE [4 y 5].

Otra función importante de la arquitectura funcional es que permite la trazabilidad de los requisitos dentro de la especificación. Es decir, dado que los requisitos se agrupan recursivamente por función, la estructura del documento traza automáticamente los requisitos de nivel inferior hasta los de nivel superior.

Este proceso comienza con el examen del conjunto preliminar de Funciones del Sistema desarrollado en la sección 2.3 y la agrupación de aquellas que están estrechamente relacionadas en las funciones principales del sistema. La descomposición de cada función principal en el siguiente nivel de funciones se realiza de manera natural a medida que se perfeccionan los requisitos.

Como ejemplo, considere la lista preliminar de Funciones del Sistema (tabla 6) desarrollada en la sección 2.3 para el Termostato Isolette. Las funciones para encender y apagar el Termostato, para indicar el Estado del Termostato, establecer el Rango de Temperatura Deseado, mostrar la Temperatura Actual e indicar si el Termostato ha fallado están estrechamente relacionadas con la interfaz del operador y es probable que todas se vean afectadas si se cambia la interfaz física del operador. Estas funciones se pueden agrupar en una función más amplia llamada Administrar Interfaz del Operador. La función preliminar restante para encender y apagar la Fuente de Calor es relativamente independiente de las demás y se puede asignar a una función llamada Administrar Fuente de Calor.

Práctica recomendada 2.5.1: Organizar las funciones del sistema en grupos que estén estrechamente relacionados y que probablemente cambien juntos.

2.5.2 Utilice Diagramas de Flujo de Datos para Representar Funciones del Sistema.

Una forma sencilla de representar las funciones del sistema y sus dependencias entre sí es mediante un diagrama de dependencia. El diagrama de dependencia superior para el Termostato Isolette se muestra en la figura 3.[1]

El diagrama de dependencia de la figura 3 abre la caja negra del sistema del diagrama de contexto de la figura A-1 para revelar las principales funciones del sistema del termostato y las dependencias entre ellas. Cada Función del Sistema que se acaba de describir se muestra en el diagrama, junto con una Función del Sistema adicional (Administrar Modo del Termostato) introducida para administrar los modos principales del sistema.[2]

En la figura 3, las variables monitoreadas se muestran como flechas que no se originan en una función (Temperatura Actual y Configuración del Operador). Las variables controladas se muestran como flechas que no terminan en una función (Control de Calor y Retroalimentación del Operador), y las dependencias de datos entre funciones se muestran como flechas que se originan y terminan en funciones (Modo, Rango Deseado y Encendido/Apagado del Termostato).

[1] El diagrama de dependencia de la figura 3 difiere de la figura A-2 del apéndice A, porque esta última está modificada para acomodar las restricciones de implementación.

[2] La mayoría de los sistemas funcionan en más de un modo, y es útil definir el comportamiento del sistema por separado para cada modo. Esto se analiza con más detalle en la sección 2.7. Las entradas a la función de modo de gestión del termostato se agregarán más adelante cuando se consideren las fallas internas y de entrada.

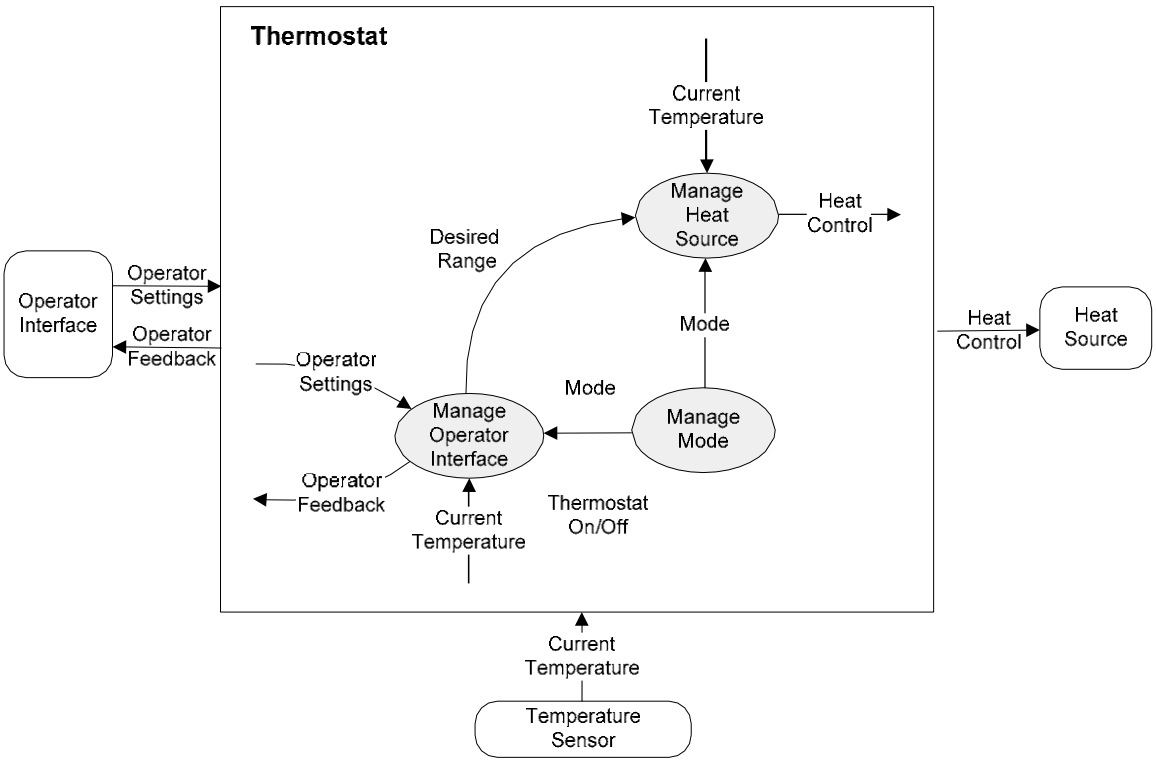


Figura 3. Diagrama de Dependencia del Termostato

Estrictamente hablando, las dependencias entre funciones se ejecutan en la dirección opuesta a la que se muestra en la figura 3. Por ejemplo, la especificación de la Función Gestión de Fuente de Calor depende de la definición de la variable interna de Rango Deseado especificada en la Función de Gestión de Interfaz de Operador y la flecha debe apuntar desde la Función Gestión de Fuente de Calor a la Función de Gestión de Interfaz de Operador. Sin embargo, dibujar los diagramas de dependencia con las flechas apuntando en la dirección de la dependencia no es intuitivo y entra en conflicto con el uso ampliamente aceptado de los diagramas de flujo de datos en los que las flechas indican la dirección del flujo de datos. En la práctica, dibujar las flechas como se muestra es mucho más aceptado y el impacto práctico en la especificación es menor, aunque técnicamente, no se mueven datos en una especificación de requisitos.

Práctica recomendada 2.5.2: Utilizar diagramas de flujo de datos para representar gráficamente las funciones del sistema y sus dependencias.

2.5.3 Minimizar las Dependencias Entre Funciones.

Las flechas que fluyen hacia una función indican todas las definiciones que esa función necesita, mientras que las flechas que fluyen hacia afuera de una función muestran todos los valores definidos por esa función. Para que la especificación de requisitos sea robusta frente a los cambios, la etiqueta de cada flecha debe representar un concepto estable y de alto nivel del dominio del problema que es poco probable que cambie. Las dependencias que es probable que cambien están ocultas dentro de las funciones. Esto crea cortafuegos que ayudan a evitar que los cambios se propaguen por toda la especificación. Idealmente, cuanto más volátil sea una dependencia, más abajo debería estar en la jerarquía de funciones.

Práctica recomendada 2.5.3: Minimizar las dependencias entre funciones asegurándose de que la información compartida entre funciones represente conceptos estables y de alto nivel del dominio del problema que tengan pocas probabilidades de cambiar. Llevar las dependencias volátiles lo más abajo posible en la jerarquía de funciones.

2.5.4 Definir Variables Internas.

Organizar los requisitos de esta manera también puede facilitar la reutilización de los mismos, ya que las áreas con mayor probabilidad de cambiar entre productos tienen más probabilidades de estar localizadas. Para un enfoque más sistemático y planificado de la reutilización, se puede introducir un análisis de variabilidad y puntos en común en el proceso general para intentar identificar qué parte de los requisitos es estable y qué partes es probable que cambien [40]. Luego, se puede realizar el análisis de funcionalidad con esa información adicional como entrada al proceso.

Las definiciones de las variables internas de un diagrama de dependencia se pueden proporcionar en una única ubicación con el diagrama. En la tabla A-1 se muestra un ejemplo para el Termostato Isolette. Alternativamente, las variables internas se pueden definir dentro de la especificación de la función en la que se origina la variable, de manera similar a la forma en que se definen las variables monitoreadas y controladas en las entidades externas. Por ejemplo, la variable interna Rango de Alarma para el Termostato Isolette se definiría en la Función de Gestión de Interfaz del Operador y la variable interna Modo se definiría en la Función Gestión de Modo del Termostato.

Práctica recomendada 2.5.4: Definir el tipo, rango, precisión y unidades de todas las variables internas introducidas para especificar dependencias de datos entre funciones.

2.5.5 Funciones de Anidamiento y Dependencias de Datos para Especificaciones Grandes.

A medida que los sistemas se hacen más grandes, un único nivel de Funciones del Sistema puede no ser suficiente. Para manejar esto, las funciones se pueden anidar dentro de otras funciones, lo que permite crear especificaciones arbitrariamente grandes. Esto se mostrará en las figuras 7, 8 y 9 en la sección 2.6. De manera similar, las dependencias (flechas) (ver sección 2.5.2) pueden representar agregados de variables para permitir que se representen colecciones arbitrariamente grandes de variables monitoreadas, controladas e intermedias.

La descomposición recursiva de funciones en subfunciones de esta manera proporciona un marco natural para organizar los requisitos en el que se agrupan los requisitos relacionados lógicamente y las dependencias entre grupos de requisitos se encapsulan en las variables internas. En la metodología CoRE original [4, 5 y 6], e incluso en el análisis estructurado mediante diagramas de flujo de datos [41], los requisitos detallados del sistema se presentaban en las funciones de nivel más bajo. De manera similar, la sección 2.8 analiza cómo se especifican los requisitos detallados de comportamiento y rendimiento del sistema en el nivel más bajo de la arquitectura funcional.

Práctica recomendada 2.5.5: Organizar especificaciones de requisitos grandes en múltiples niveles anidando funciones y dependencias de datos.

2.5.6 Proporcionar Requisitos de Alto Nivel que Sean Realmente de Alto Nivel.

Los diagramas de dependencia y las definiciones de las variables internas definen implícitamente los requisitos de alto nivel del sistema para cada función que no se encuentre en el nivel más bajo (es decir, cada función no terminal). Si se desea, también se pueden definir requisitos explícitos de alto nivel para cada función no terminal. Sin embargo, se debe tener cuidado de no extraer detalles de las funciones de nivel inferior para las funciones de alto nivel en un esfuerzo por ser exhaustivos y precisos. Hacer esto anula todo el propósito del desarrollo de la arquitectura funcional, que es proporcionar un marco para organizar una especificación compleja.

La figura 4 muestra un ejemplo de requisitos de alto nivel para la Función del Termostato. Estos simplemente establecen que el Termostato establecerá el valor de las variables controladas de Control de Calor y Retroalimentación del Operador, pero no definen cómo se establecerán. Como se muestra en el diagrama de contexto de la figura A-1, la Función del Termostato solo establece dos variables controladas (agregadas), el Control de Calor y la Retroalimentación del Operador. En este nivel de abstracción, todo lo que se sabe es que los valores son establecidos por la Función del Termostato y el rango de esos valores (ya que sus valores permitidos se definen como parte de los supuestos ambientales).

REQ-1	La Función Termostato establecerá el valor del Control de Calor. Justificación: Una función principal del Termostato es encender y apagar el Control de Calor para mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado.
REQ-2	La Función Termostato establecerá el valor de la Retroalimentación del Operador. Justificación: El Termostato proporciona información al operador a través de la Interfaz del Operador, que la utilizará para informar al Operador sobre el estado general del Termostato.

Figura 4. Requisitos de alto nivel para la Función del Termostato

Los detalles precisos de cómo se establecerán esas variables se proporcionan en los niveles inferiores de la jerarquía de funciones. Los intentos de proporcionar más información en el nivel más alto introducen ambigüedad o duplican la información definida en los niveles inferiores. Por ejemplo, a primera vista, podría parecer que REQ-1 podría enunciarse mejor así:

- REQ-1—El Termostato deberá establecer el valor del Control de Calor para mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado.

Existen dos problemas con esto. Primero, solo es Verdadero cuando el Termostato está en su modo de operación NORMAL. Durante el modo INIT o FAILED, el Control de Calor se establece en el valor de apagado. Desafortunadamente, los modos del sistema no se definen hasta el siguiente nivel inferior de refinamiento, por lo que especificar con precisión cómo se debe configurar el Control de Calor requiere el uso de términos y definiciones que no están disponibles en este nivel.

Práctica recomendada 2.5.6: Si se proporcionan requisitos de alto nivel, defina únicamente lo que se puede indicar en ese nivel de la jerarquía. No utilice términos definidos en niveles inferiores de la jerarquía.

2.5.7 No Incorporar Fundamentos en los Requisitos.

En segundo lugar, confunde lo que hará el sistema con el motivo por el que lo hará; es decir, está mezclando la lógica con el requisito en sí (véase la sección 2.11). La especificación completa de cómo se debe configurar el Control de Calor es un algoritmo sorprendentemente complejo de su Modo Actual, la Temperatura Actual, el Rango de Temperatura Deseado y su estado anterior (véase el apéndice A.5.1.3 para una especificación completa de este algoritmo). Mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado es realmente una explicación de por qué existe este requisito (su lógica). Tenga en cuenta que en la figura 4, esta información se proporciona para ayudar al lector a entender por qué se incluye el requisito, pero se proporciona como lógica, no como un requisito.

Todos los ejemplos de los apéndices A a D incluyen requisitos textuales de alto nivel para las funciones que no se encuentran en el nivel más bajo de descomposición. Si bien la justificación que se proporciona con estos ejemplos es útil para comprender los ejemplos, los requisitos de alto nivel en sí mismos básicamente repiten la información de los diagramas de dependencia y no son estrictamente necesarios.

Práctica recomendada 2.5.7: Si se proporcionan requisitos de alto nivel para funciones, evite incorporar fundamentos en el requisito en un intento de especificar detalles que se especificarían de manera más apropiada en una función de nivel inferior.

2.6 REVISAR LA ARQUITECTURA PARA CUMPLIR CON LAS RESTRICCIONES DE IMPLEMENTACIÓN.

Consulte la tabla 10 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para revisar la arquitectura a fin de cumplir con las restricciones de implementación.

Tabla 10. Revisar la Arquitectura para Cumplir con las Restricciones de Implementación

Prácticas recomendadas para los niveles principal y secundario
2.6 Revisar la Arquitectura para Cumplir con las Restricciones de Implementación: La organización que se produce a través del análisis funcional es una arquitectura lógica que puede no tener en cuenta restricciones adicionales, como la necesidad de satisfacer requisitos de seguridad del sistema, integrarse con sistemas heredados o cumplir con las restricciones de implementación impuestas por una plataforma en particular. Esta práctica describe un proceso iterativo que comienza con la arquitectura funcional desarrollada previamente y conduce a una arquitectura que aborda estas preocupaciones. Esta arquitectura se utiliza luego como marco para organizar los requisitos detallados.
2.6.1 Si las restricciones de implementación no pueden satisfacerse con la arquitectura funcional ideal que se desarrolla durante el análisis funcional, modifique la arquitectura funcional según sea necesario y utilice la arquitectura final del sistema como marco para organizar los requisitos detallados.

Tabla 10. Revisión de la Arq. para Cumplir con las Restricciones de Implementación (cont.)

Prácticas recomendadas para los niveles principal y secundario
2.6.2 Al modificar la arquitectura funcional para adaptarse a las restricciones de implementación, mantenga la arquitectura final del sistema lo más cercana posible a la arquitectura funcional ideal.
2.6.3 Revise la descripción general del sistema para reflejar cualquier cambio en la forma en que el sistema interactúa con su entorno, cualquier funcionalidad nueva agregada al sistema para satisfacer las restricciones de implementación o cualquier cambio en los objetivos del sistema.
2.6.4 Revisar los conceptos operativos para reflejar cualquier cambio en cómo los operadores u otros sistemas interactúan con la arquitectura del sistema revisada.
2.6.5 Revise los casos de uso para identificar los pasos en los que podrían ocurrir excepciones al comportamiento nominal. Desarrolle casos de excepción para identificar cómo se manejará cada excepción.
2.6.6 Si una excepción sólo puede ocurrir en unos pocos puntos, vincule esos pasos al caso de excepción. Si la excepción puede ocurrir en casi cualquier momento, utilice la condición previa del caso de excepción para identificar cuándo ocurre el caso de excepción.
2.6.7 Revisar el límite del sistema para reflejar cualquier cambio en las variables monitoreadas y controladas.
2.6.8 Identificar y documentar cualquier supuesto ambiental nuevo o modificado para la arquitectura funcional revisada.
2.6.9 Revisar los diagramas de dependencia para mostrar la arquitectura funcional revisada.
2.6.10 Revisar cualquier requisito de alto nivel afectado por los cambios en la arquitectura funcional revisada.

Lo ideal sería que la arquitectura funcional desarrollada en la sección 2.5 se pudiera utilizar como la estructura de la especificación de requisitos con los requisitos detallados de comportamiento y rendimiento definidos para cada función. Desafortunadamente, los componentes fallan con consecuencias para la seguridad, los nuevos sistemas deben integrarse con los sistemas heredados existentes y las restricciones de implementación afectan la arquitectura del sistema. Muchas de estas preocupaciones se abordan en el diseño de la arquitectura del sistema con la consecuencia de que la arquitectura final del sistema puede no corresponderse directamente con la arquitectura lógica desarrollada durante el análisis funcional. En lugar de intentar hacer una correspondencia continua entre la arquitectura funcional ideal y la arquitectura final del sistema, es más práctico revisar la arquitectura funcional para tener en cuenta las restricciones más importantes.

2.6.1 Modificar la Arquitectura para Cumplir con las Restricciones de Implementación.

El objetivo de esta práctica recomendada es introducir un proceso iterativo que parte de la arquitectura funcional ideal desarrollada en la sección 2.5 y conduce a una arquitectura funcional que describe la arquitectura final del sistema. Esta arquitectura final puede utilizarse como marco para organizar los requisitos detallados.

Gran parte de la motivación para desarrollar esta arquitectura es proporcionar un marco natural para especificar los requisitos funcionales y de rendimiento detallados del sistema. En efecto, los diagramas de esta sección pueden considerarse una tabla gráfica de contenidos para la especificación detallada de los requisitos. La especificación de los requisitos detallados se describe en la sección 2.8. Sin embargo, antes de especificar los requisitos detallados, es necesario analizar los modos del sistema, como se muestra en la sección 2.7.

Práctica recomendada 2.6.1: Si las restricciones de implementación no pueden satisfacerse con la arquitectura funcional ideal que se desarrolla durante el análisis funcional, modifique la arquitectura funcional según sea necesario y utilice la arquitectura final del sistema como marco para organizar los requisitos detallados.

2.6.2 Mantener la Arquitectura Final del Sistema Cerca de la Arquitectura Funcional Ideal.

Por otra parte, la arquitectura funcional original se desarrolló mediante el análisis del dominio del problema sin tener en cuenta las restricciones de implementación. En general, es menos probable que cambie el dominio del problema que las restricciones impuestas por la implementación. Por este motivo, cuanto más parecida sea la arquitectura final a la arquitectura funcional original, más estable será. Por este motivo, también es deseable minimizar, tanto como sea posible, las diferencias entre la arquitectura funcional ideal y la arquitectura final.

Práctica recomendada 2.6.2: Al modificar la arquitectura funcional para adaptarse a las restricciones de implementación, mantenga la arquitectura final del sistema lo más cercana posible a la arquitectura funcional ideal.

2.6.3 Revisar la Descripción General del Sistema.

En el caso de los sistemas críticos para la seguridad, el proceso de modificación de la arquitectura funcional para adaptarse a las limitaciones de implementación suele estar impulsado por la necesidad de alcanzar niveles muy altos de confiabilidad que dicta el proceso de seguridad del sistema [24]. En el caso de los sistemas de aviónica comercial, este proceso se describe en ARP 4761 [31]. Uno de los primeros pasos de este proceso es la Evaluación de Riesgos Funcionales[1] (FHA) del sistema, que identifica los riesgos de alto nivel del sistema. La FHA se utiliza durante la Evaluación Preliminar de Seguridad del Sistema[2] (PSSA) para determinar si el sistema podría contribuir a la materialización de alguno de estos riesgos. En caso afirmativo, la PSSA establece requisitos de seguridad derivados del diseño del sistema.

Por ejemplo, la FHA para el sistema Isolette se completó antes de la definición de requisitos del Termostato Isolette e identificó el siguiente peligro relevante:

> H1. Exposición prolongada del lactante a calor o frío peligrosos
>
> Clasificación: catastrófica
> Probabilidad: $<10^{-9}$ por hora de operación

[1] N.d.T: Functional Hazard Assessment, en inglés.
[2] N.d.T: Preliminary System Safety Assessmen, en inglés.

El sistema Isolette PSSA (no el Termostato en sí) identifica varias formas en las que este peligro podría materializarse.

- El Termostato podría fallar, y encender o apagar la Fuente de Calor durante demasiado tiempo.
- El Sensor de Temperatura podría proporcionar una temperatura incorrecta al Termostato.
- La Interfaz del Operador podría proporcionar un Rango de Temperatura Deseado incorrecto al Termostato.
- La Fuente de Calor podría fallar, ya sea por permanecer encendida o apagada durante demasiado tiempo o por no proporcionar suficiente calor para mantener el Rango de Temperatura Deseado.

El árbol de fallas derivado durante la PSSA del sistema Isolette se muestra en la figura 5. Dado que cada Función del Sistema podría causar el peligro H1, a cada función se le asigna una probabilidad de falla de menos de 2 x 10^{-10} por hora de funcionamiento.

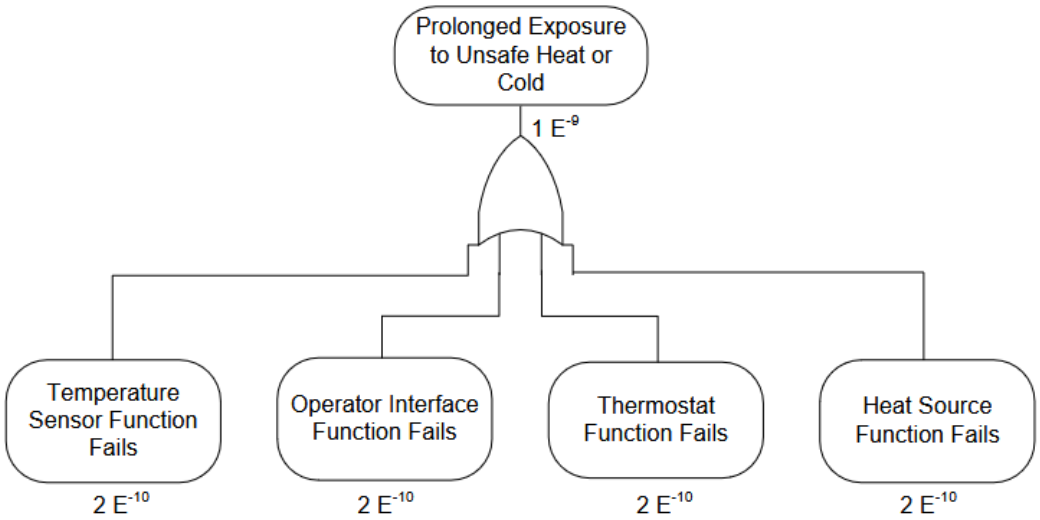


Figura 5. Árbol de fallas inicial de Isolette

Desarrollar componentes individuales que logren este nivel de confiabilidad sería muy costoso. Incluso diseñar un Termostato que proporcione este nivel de confiabilidad contradiría el objetivo G2 de producir el Termostato con un costo de fabricación mínimo. Una solución menos costosa es agregar un monitor que active una alarma si la Temperatura Actual en la Incubadora cae por debajo o sube por encima de un nivel seguro. Otra PSSA muestra que la combinación de dicha alarma y el monitoreo normal del Bebé por parte de la Enfermera protegería contra una Función Termostato fallida y una Función de Administración de Fuente de Calor fallida (pero no contra una Función de Sensor de Temperatura Engañosa o una Función de Interfaz de Operador engañosa) y solo requeriría que el Termostato, la Fuente de Calor y el Monitor tengan una probabilidad de falla de menos de 10^{-5} por hora de funcionamiento.[1] El árbol de fallas revisado derivado durante esta PSSA se muestra en la figura 6.

1 Tenga en cuenta que los requisitos de confiabilidad para el sensor de temperatura y la interfaz del operador son en realidad para el dispositivo y su ruta de comunicación al termostato, es decir, son para la entrega de las salidas del dispositivo al termostato.

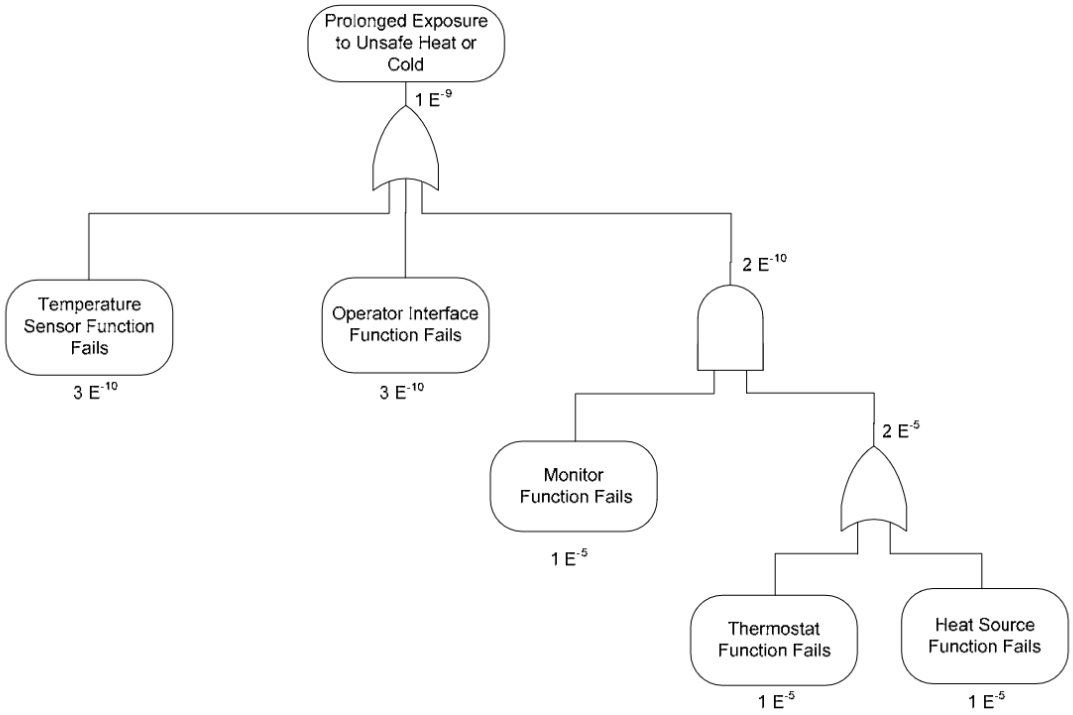


Figura 6. Árbol de Fallas de Isolette Revisado

Esta solución aún requiere una Función de Sensor de Temperatura y una Función de Interfaz de Operador altamente confiables. Sin embargo, para simplificar este ejemplo, se supone que esto se puede lograr, tal vez mediante el uso de un diseño pasivo ante fallas utilizando dos o más sensores de temperatura y una interfaz de operador mecánica simple, pero altamente confiable. El resto del ejemplo se centrará en las funciones de Termostato y Monitor. La PSSA conduce a los siguientes requisitos de seguridad derivados para estos componentes:

- El Isolette incluirá una Función Termostato independiente que mantiene la Temperatura Actual dentro del Rango de Temperatura Deseado dentro del Isolette.

  Fundamento: La Enfermera establecerá el Rango de Temperatura Deseado en el rango ideal según el peso y la salud del Bebé. El Termostato debe mantener la Temperatura Actual dentro de este rango en condiciones normales de funcionamiento.

  Probabilidad de fallo permitida: $< 10^{-5}$ por hora.

- El Isolette incluirá una Función de Monitor independiente que activa una Alarma en 5 segundos siempre que

  - La Temperatura Actual dentro de la Isolette cae por debajo o aumenta por encima del Rango de Temperatura de Alarma.

  - La Temperatura Actual está marcada como no válida.

  Justificación: La Enfermera establecerá el Rango de Temperatura de Alarma en función del peso y la salud del Bebé. El Bebé debe ser retirado de la Incubadora dentro de los 15 segundos posteriores a que la Temperatura Actual caiga por debajo o suba por encima del Rango de Temperatura de Alarma. Con el monitoreo normal proporcionado por la

Enfermera, esto se puede lograr en 10 segundos, dejando 5 segundos para que el sistema active la Alarma. Es deseable activar la Alarma en menos tiempo.

Probabilidad de fallo permitida: $<10^{-5}$ por hora.

Para cumplir con estos requisitos y minimizar los costos de fabricación, los diseñadores de Isolette propusieron un diseño en el que la función de monitor se implementa dentro del propio termostato. Después de extender la PSSA a este diseño, esto fue aceptable, siempre que se mantenga la independencia del monitor.[1] Para evitar confusiones, la Función Termostato pasó a denominarse Función de Regulador, y ahora se considera que el Termostato es la combinación de las Funciones de Regulador y Monitor. Esto dio lugar a un diagrama de dependencia de nivel superior para el Termostato, como se muestra en la figura 7.

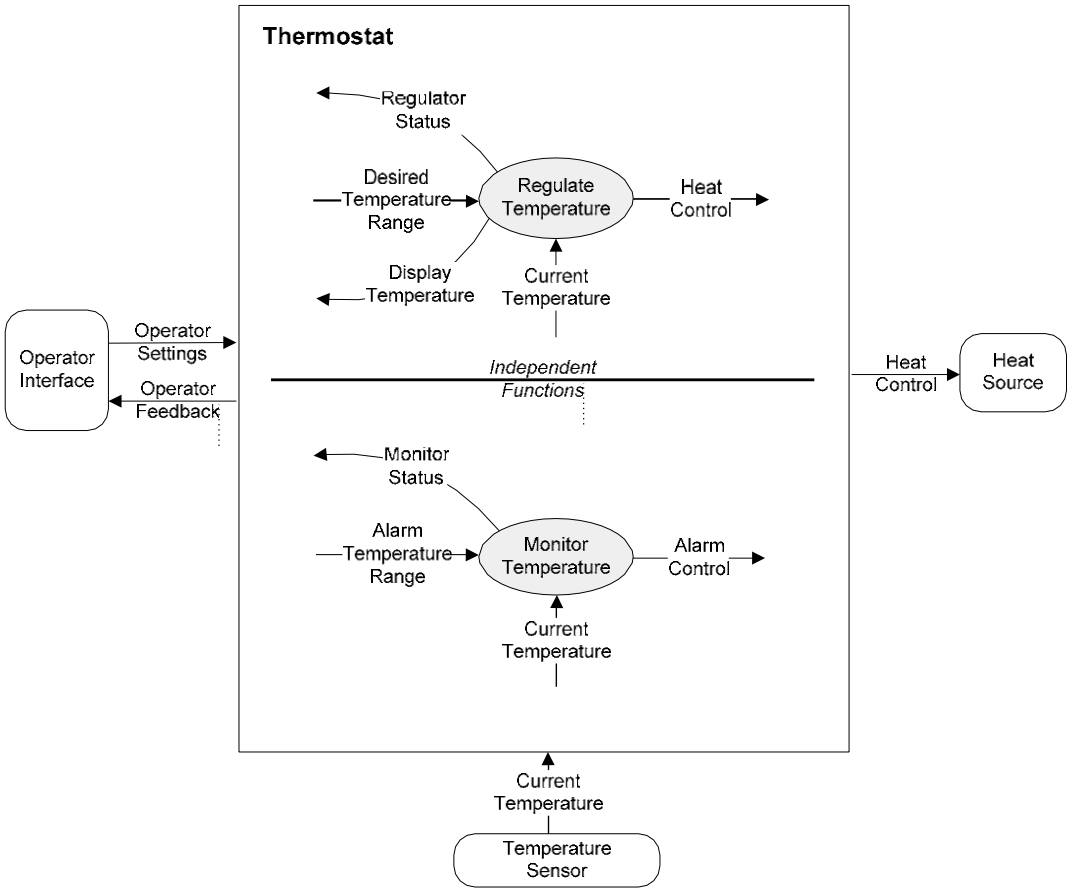


Figura 7. Diagrama de dependencia del termostato revisado

En la figura 7, la funcionalidad del Termostato se asignó a dos funciones, Regular la Temperatura y Monitorear la Temperatura, cuyas implementaciones deben ser independientes (es decir, cada función debe fallar independientemente de la otra). La función Regular la Temperatura controla la Fuente de Calor y garantiza que la Temperatura Actual se mantenga dentro del Rango de Temperatura Deseado.
El Monitor de Temperatura activa la Alarma en la Interfaz del Operador siempre que la

1 Esto puede requerir fuentes de alimentación independientes o un monitor adicional en la fuente de alimentación del Isolette.

Temperatura Actual cae por debajo o aumenta por encima del Rango de Temperatura de Alarma.

Al revisar las variables monitoreadas y controladas para estas funciones, es evidente que se ha modificado el límite del sistema. Hay una nueva variable monitoreada, Rango de Temperatura de Alarma, proporcionada por la Interfaz del Operador. Además, el Estado del Termostato ha cambiado de nombre a Estado del Regulador y se ha definido una nueva variable controlada, Estado del Monitor. Esto es necesario ya que el estado general del Termostato ahora está determinado por dos funciones independientes. Además, tenga en cuenta que la variable controlada Temperatura de Visualización se proporciona a la Interfaz del Operador solo por la Función Regulación de Temperatura, ya que su valor no puede contribuir a ninguna condición peligrosa y puede proporcionarse por una sola fuente.

Estos cambios requieren que se revisen la descripción general del sistema, los límites del sistema, los conceptos operativos y las suposiciones ambientales. En la descripción general del sistema, se debe agregar a la sinopsis la funcionalidad adicional de configurar el Rango de Temperatura de Alarma y activar una Alarma. Además, se debe agregar a la lista de objetivos del sistema un nuevo objetivo para advertir a la Enfermera si el Bebé tiene demasiado frío o demasiado calor. Estas incorporaciones se analizan en el apéndice A.1.

Práctica recomendada 2.6.3: Revisar la descripción general del sistema para reflejar cualquier cambio en la forma en que el sistema interactúa con su entorno, cualquier funcionalidad nueva agregada al sistema para satisfacer las restricciones de implementación o cualquier cambio en los objetivos del sistema.

2.6.4 Revisar los Conceptos Operativos.

Si se modificó la interacción con otros sistemas u operadores de sistemas para cumplir con las restricciones de implementación, también se deben actualizar los conceptos operativos.

Práctica recomendada 2.6.4: Revisar los conceptos operativos para reflejar cualquier cambio en cómo los operadores u otros sistemas interactúan con la arquitectura del sistema revisada.

2.6.5 Desarrollar Casos de Excepción.

Dado que la PSSA inició la consideración de cómo se deben manejar las fallas, este también es un momento apropiado para volver atrás y ampliar los casos de uso con casos de excepción. A medida que se revisan los casos de uso y se agregan nuevas funcionalidades, se deben identificar los pasos en los que podrían ocurrir excepciones al comportamiento nominal (de día soleado). Se deben definir los casos de excepción, describiendo cómo se manejará cada excepción.

Práctica recomendada 2.6.5: Revise los casos de uso para identificar los pasos en los que podrían ocurrir excepciones al comportamiento nominal. Desarrolle casos de excepción para identificar cómo se manejará cada excepción.

2.6.6 Vincular Casos de Excepción a Casos de Uso.

Si la excepción solo puede ocurrir en algunos puntos, se debe crear un vínculo desde esos pasos en los casos de uso hasta el caso de excepción. Si la excepción puede ocurrir casi en cualquier momento (como una falla del sistema), no es razonable crear un vínculo desde cada paso del caso de uso. En su lugar, la condición previa para el caso de excepción debe dejar en claro cuándo puede ocurrir la excepción.

A medida que se identifican los casos de excepción, se debe considerar si podrían contribuir o no a un peligro del sistema identificado por la FHA. Se proporcionan ejemplos de casos de excepción y sus vínculos en el apéndice A.2 para el Termostato Isolette y en el apéndice B.2 para el FCS.

Práctica recomendada 2.6.6: Si una excepción solo puede ocurrir en algunos puntos, vincule esos pasos con el caso de excepción. Si la excepción puede ocurrir casi en cualquier punto, utilice la condición previa del caso de excepción para identificar cuándo ocurre el caso de excepción.

2.6.7 Revisar el Límite del Sistema.

Si la arquitectura funcional revisada introdujo nuevas variables monitoreadas y controladas, es necesario actualizar la definición del límite del sistema. Para el Termostato Isolette, se agregaron la variable monitoreada Rango de Temperatura de Alarma y la variable controlada Control de Alarma, y la variable controlada Estado del Termostato se reemplazó por las variables controladas Estado del Regulador y Estado del Monitor.

Práctica recomendada 2.6.7: Revisar el límite del sistema para reflejar cualquier cambio en las variables monitoreadas y controladas.

2.6.8 Documentar Cambios en los Supuestos Ambientales.

Es necesario identificar y documentar los nuevos supuestos ambientales. En el caso del Termostato Isolette, es necesario documentar los puntos finales del Rango de Temperatura de Alarma, junto con la justificación de sus valores. Esto es particularmente importante, ya que define los límites bajo los cuales opera el Monitor y está estrechamente vinculado al riesgo H1 del sistema. A medida que se agregan más variables monitoreadas y controladas, se debe tener cuidado para garantizar que se identifiquen y documenten todos los supuestos ambientales. Estos pueden ser bastante sutiles. Por ejemplo, ahora hay varias relaciones más que la interfaz del operador debe mantener entre el Rango de Temperatura de Alarma y el Rango de Temperatura Deseado. Estas se documentan en el apéndice A.3.

Práctica recomendada 2.6.8: Identificar y documentar cualquier supuesto ambiental nuevo o modificado para la arquitectura funcional revisada.

2.6.9 Revisar Diagramas de Dependencia.

Por último, el diagrama de dependencia de la figura 3 debe reemplazarse por el diagrama de dependencia del Termostato revisado de la figura 7 y los diagramas de dependencia creados para las Funciones de Regulación de Temperatura y Control de Temperatura. El diagrama de dependencia de la Función Regulación de Temperatura se muestra en la figura 8.

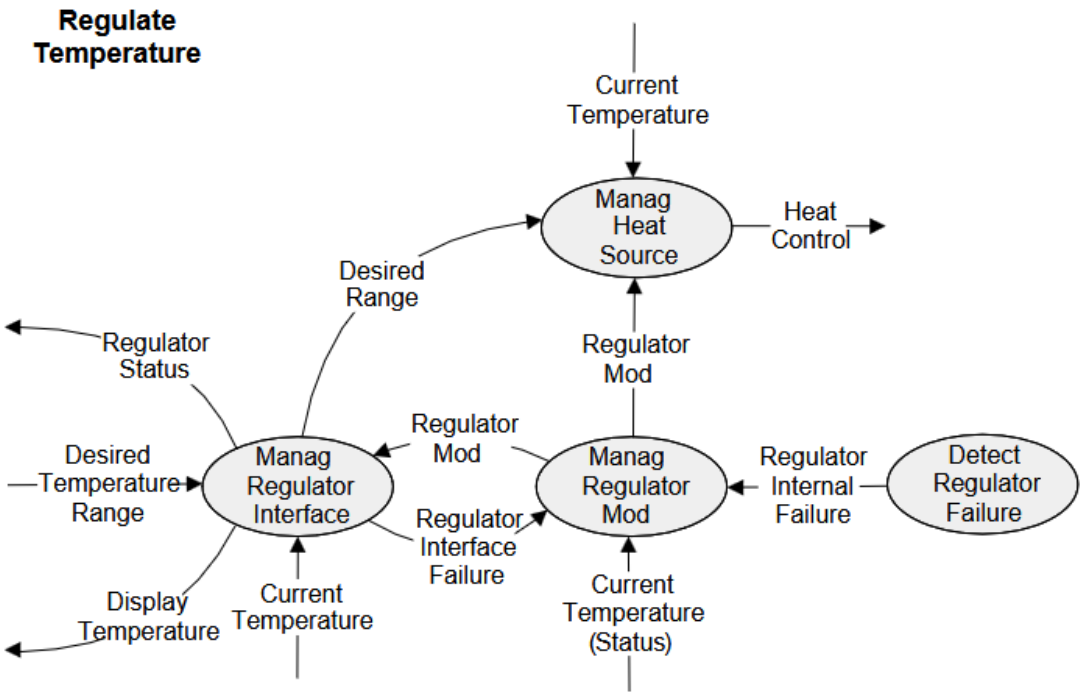


Figura 8. Diagrama de Dependencia de la Temperatura Regulada

En muchos sentidos, este diagrama de dependencia se parece al diagrama de dependencia del Termostato original que se muestra en la figura 3. Esto no es sorprendente, ya que cubre la misma funcionalidad básica. Un cambio que se realizó fue dividir los datos agregados de Configuración del Operador y Retroalimentación al Operador en sus valores de componentes de Rango de Temperatura Deseado, Estado del Regulador y Temperatura de Visualización. Esto hace que el diagrama sea un poco más fácil de leer y facilita la garantía de que las entradas y salidas del diagrama coincidan con las de su diagrama principal.

Además, se ha añadido una nueva función, Detectar Fallo del Regulador, que se encarga de detectar fallos internos mediante pruebas automáticas u otras comprobaciones. También se han añadido las variables internas de Fallo Interno del Regulador y Fallo de la Interfaz del Regulador, junto con el atributo Estado de la variable monitorizada Temperatura Actual. Estas dependencias se han añadido para especificar cómo gestionar los fallos en la detección de las variables monitorizadas o los fallos internos de la Función Regulación de Temperatura. Su uso se explicará con más detalle en la sección 2.7.

El diagrama de dependencia de la Función Monitor de Temperatura se muestra en la figura 9. Es muy similar a la Función Regulación de Temperatura que se muestra en la figura 8.

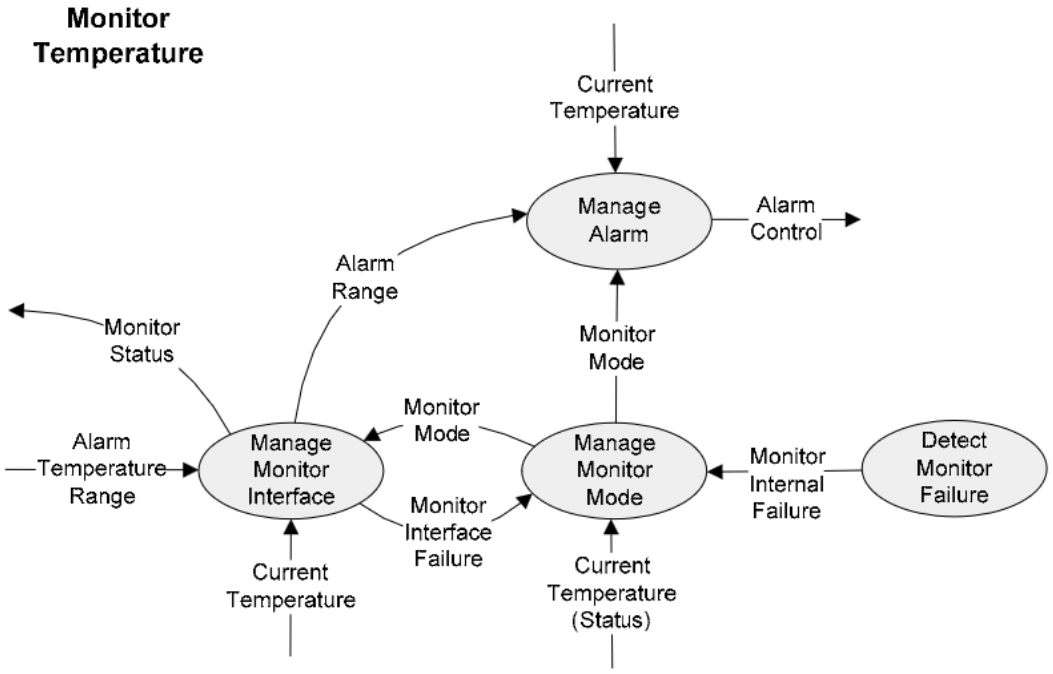


Figura 9. Diagrama de Dependencia de la Temperatura del Monitor

Los diagramas de las figuras 7, 8 y 9 muestran cómo se relacionan entre sí las funciones del Termostato Isolette, tanto a través de la jerarquía de funciones como de sus dependencias de datos. Esta estructura se derivó de la arquitectura lógica original desarrollada durante el análisis funcional (figura 3) en el curso del manejo de los requisitos de seguridad derivados. Otras restricciones, como la necesidad de integración con un sistema heredado o de cumplir con restricciones de implementación específicas, podrían tener un efecto similar en la arquitectura del sistema de interés.

Práctica recomendada 2.6.9: Revisar los diagramas de dependencia para mostrar la arquitectura funcional revisada.

2.6.10 Revisar los Requisitos de Alto Nivel.

Por último, todos los requisitos de alto nivel se deben actualizar si el cambio en la arquitectura funcional del sistema así lo requiere.

Práctica recomendada 2.6.10: Revisar cualquier requisito de alto nivel afectado por los cambios en la arquitectura funcional revisada.

2.7 IDENTIFICAR LOS MODOS DEL SISTEMA.

Consulte la tabla 11 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para identificar los modos del sistema.

Tabla 11. Identificar los modos del sistema

Prácticas recomendadas para los niveles principal y secundario
2.7 Identificar los Modos del Sistema: Los modos definen comportamientos inconexos del sistema que son visibles para sus operadores o para otros sistemas. Los requisitos detallados de comportamiento y rendimiento del sistema suelen ser diferentes para los distintos modos del sistema. La identificación de los modos del sistema es un paso útil que simplifica la especificación detallada de los requisitos de comportamiento y rendimiento.
2.7.1 Identifique los principales modos del sistema antes de definir los requisitos detallados del sistema.
2.7.2 Definir cómo se permite que el sistema realice la transición entre modos.
2.7.3 Introduzca modos únicamente para identificar las discontinuidades visibles externamente en el comportamiento del sistema. No defina modos que no puedan inferirse a partir del comportamiento visible externamente del sistema.

Leveson define los modos como comportamientos distintos del sistema [35]. Por ejemplo, un sistema puede responder a un estímulo, como presionar un botón de una manera durante el encendido del sistema, de otra manera durante una prueba automática y de otra manera durante el funcionamiento normal. Estos comportamientos son tres modos del sistema. Los modos del sistema pueden ser muy simples o muy complejos. Pueden ser relevantes para todo el sistema o solo para una parte del sistema. Sin embargo, siempre se introducen modos para manejar discontinuidades en el comportamiento del sistema.

La identificación de los modos principales del sistema resulta útil al redactar los requisitos detallados del sistema, como se explica en la sección 2.8. Sin embargo, en este punto, puede que no esté claro si se identificaron todos los modos del sistema o si todos los modos enumerados son realmente necesarios. La utilidad de los modos propuestos se hará evidente durante la especificación detallada de los requisitos del sistema, y la necesidad de modos adicionales se hará más evidente. Al igual que con muchos de los pasos anteriores, los modos del sistema definidos en este punto sirven principalmente como punto de partida para el trabajo que sigue.

Dado que los modos del sistema están tan estrechamente relacionados con el comportamiento del sistema visible externamente, un diseño deficiente del modo del sistema puede generar confusión entre los modos, lo que puede hacer que el operador no sepa en qué modo se encuentra el sistema. Este es un problema de seguridad importante que se ha visto implicado en varios accidentes de aviación [32 y 34].

Algunas de las posibles fuentes de confusión entre modos son la falta de retroalimentación adecuada a los operadores, errores en la interpretación o introducción de información en diferentes modos, comportamiento inconsistente del sistema en diferentes modos, límites de autoridad del operador diferentes en diferentes modos, transiciones de modo silenciosas (sin previo aviso) y efectos secundarios no deseados de las transiciones de modo. Desarrollar sistemas en los que los modos se indiquen claramente a los operadores y el comportamiento del sistema sea consistente, fácil de anticipar y de entender para los operadores es una tarea desafiante que está más allá del alcance de este Manual. Se proporciona más información en las referencias 33, 35, 36 y 37.

2.7.1 Identificar los Principales Modos del Sistema.

Un modo del sistema puede o no mostrarse explícitamente a un usuario del sistema, pero, por definición, es visible, ya que el sistema responderá de manera diferente a los estímulos cuando se encuentre en diferentes modos. En este sentido, los modos son una parte visible externamente del comportamiento del sistema que debe especificarse en los requisitos del sistema. La identificación de los modos del sistema también simplifica la especificación de los requisitos del sistema al permitir que la relación entre las variables monitoreadas y controladas se desglose en partes más pequeñas para cada modo del sistema. Por estas razones, es útil identificar los principales modos operativos del sistema antes de comenzar a escribir los requisitos funcionales detallados.

Práctica recomendada 2.7.1: Identificar los principales modos del sistema antes de definir los requisitos detallados del sistema.

2.7.2 Definir Cómo el Sistema Realiza la Transición Entre Modos.

A modo de ejemplo, los modos del sistema de Función Regulación de Temperatura del Termostato Isolette se muestran en la figura 10.

El sistema se inicia en el modo INIT cuando se enciende y permanece en este modo hasta que el Estado del Regulador (que se muestra en la tabla 12) se vuelve válido. Esto ocurre cuando el Regulador ha completado su secuencia de inicialización y ha pasado todas las pruebas automáticas (es decir, la Falla Interna del Regulador, como se muestra en la figura 8, es falsa) y ha detectado con éxito todas sus variables monitoreadas (la Falla de la Interfaz del Regulador, como se muestra en la figura 8, es falsa y el atributo de estado de Temperatura Actual es válido).

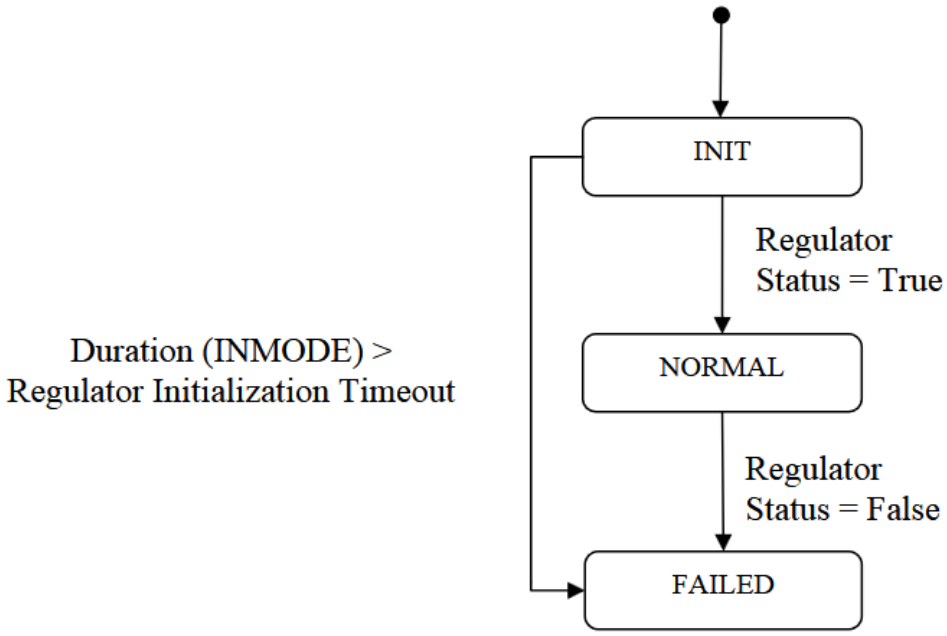


Figura 10. Modos de Función Regulación de Temperatura

Tabla 12. Definición de la Condición de Regulador

Nombre	Tipo	Definición
Estado del Regulador	Booleano	NOT (Falla de la Interfaz del Regulador O Falla Interna del Regulador) AND Estado de Temperatura Actual = Válido

En este punto, ingresa al modo de funcionamiento NORMAL, donde generalmente permanece hasta que se apaga el sistema. Si el sistema no completa su inicialización dentro de un período de tiempo de espera especificado, o si el Estado del Regulador se vuelve Falso mientras está en modo NORMAL, la función ingresa al modo FALLA, donde permanece hasta que se apaga y se enciende. Se define un conjunto similar de modos para la Función de Monitoreo de Temperatura.

Práctica recomendada 2.7.2: Definir cómo se permite que el sistema realice la transición entre modos.

2.7.3 Introducir Modos para Discontinuidades Visibles Externamente.

Si bien los diagramas de transición de estados, como el que se muestra en la figura 10, son una forma popular de especificar los modos del sistema [42], los diagramas demasiado complejos pueden ser una indicación de que se están incluyendo decisiones de diseño que deberían quedar fuera de la especificación de requisitos. Los modos deberían definirse solo para identificar las discontinuidades visibles externamente en el comportamiento del sistema. Esto permite que los requisitos se escriban de la siguiente manera:

- "Si el sistema está en modo de inicialización, la variable controlada se establecerá en …"
- "Si el sistema está en modo de falla, la variable controlada se establecerá en …"

Esto también hace que sea más fácil determinar si los requisitos son completos y consistentes (consulte la sección 2.8 para ver más ejemplos).

Práctica recomendada 2.7.3: Introduzca modos únicamente para identificar discontinuidades visibles externamente en el comportamiento del sistema. No defina modos que no puedan inferirse a partir del comportamiento visible externamente del sistema.

2.8 DESARROLLAR LOS REQUISITOS DETALLADOS DE COMPORTAMIENTO Y RENDIMIENTO.

Consulte la tabla 13 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para desarrollar requisitos detallados de comportamiento y rendimiento.

En las secciones anteriores se ha analizado cómo crear la descripción general del sistema, definir los límites del sistema, desarrollar los conceptos operativos, definir los supuestos ambientales, comenzar el análisis funcional, modificar el análisis funcional para adaptarlo a las restricciones de implementación e identificar los modos principales del sistema. Una vez completadas estas actividades de forma parcial o total, se pueden especificar los requisitos detallados de comportamiento y rendimiento.

Tabla 13. Desarrollar Requisitos Detallados de Comportamiento y Rendimiento

Práctica recomendada para los niveles principal y secundario
2.8 Desarrollar Requisitos Detallados de Comportamiento y Rendimiento: Los requisitos de comportamiento y rendimiento definen cómo el sistema debe cambiar las variables controladas en respuesta a los cambios en las variables monitoreadas. Esto incluye especificar el valor asignado a la variable controlada para cada estado y entrada del sistema, la tolerancia permitida sobre este valor y las características de rendimiento, como la latencia permitida. Esta práctica proporciona pautas sobre cómo producir un conjunto completo y coherente de requisitos detallados de comportamiento y rendimiento del sistema.
2.8.1 Utilice los nombres de las variables monitoreadas y controladas, los modos del sistema y las variables internas al escribir los requisitos detallados del sistema.
2.8.2 Para cada requisito, especifique los modos del sistema y las condiciones bajo las cuales se aplicará el requisito, seguido del cambio en la variable afectada.
2.8.3 Asegúrese de que los requisitos detallados estén completos, es decir, que se asigne un valor ideal a cada variable controlada y variable interna para cada estado del sistema. Utilice un valor NO ESPECIFICADO cuando no exista una asignación significativa para un estado del sistema.
2.8.4 Asegúrese de que los requisitos detallados sean consistentes, es decir, que solo se asigne un valor ideal a cada variable controlada y a cada variable interna para cada estado posible del sistema.
2.8.5 Asegúrese de que no haya dos requisitos detallados que se dupliquen entre sí, es decir, especifique el mismo resultado para modos y condiciones superpuestos.
2.8.6 Presentar los requisitos detallados del sistema en la función que produce la variable que se especifica. Esto crea una organización en la que la definición de cada variable se puede rastrear directamente hasta su función principal.
2.8.7 Definir la latencia aceptable para cada variable controlada junto con la justificación de su valor como parte de los requisitos detallados del sistema.
2.8.8 Definir la tolerancia aceptable para cada variable controlada numéricamente.
2.8.9 No definir latencia y tolerancia para variables internas.

Los requisitos detallados definen qué comportamiento impondrá el sistema a su entorno. Estos pueden especificarse como otra relación matemática entre las variables monitoreadas y controladas.[1] Esta relación define cómo cambiarán las variables controladas en respuesta a los cambios en las variables monitoreadas. Debido al tamaño y la complejidad de la mayoría de los sistemas, esto se hace definiendo primero, en el nivel más bajo de la arquitectura funcional, cuál sería el valor de cada variable controlada e interna para un sistema perfecto. En la metodología CoRE, esto se conoce como la función de valor ideal para una variable [5]. Esto establece una cadena completa de asignaciones que definen cómo cambiarán las variables controladas (en el caso ideal) en respuesta a los cambios en las variables monitoreadas. Sin embargo, esto suele ser demasiado restrictivo. Para la mayoría de los sistemas, un rango de valores cercanos al valor ideal es aceptable. A continuación, para capturar esto, se especifica una tolerancia sobre el valor ideal para cada variable controlada. Esta tolerancia puede ser una constante simple o una función arbitrariamente compleja del estado del sistema. Finalmente, se deben especificar los aspectos de

[1] En las metodologías SCR [2 y 3] y CoRE [4 y 5], esto se denomina relación de requisitos (REQ).

rendimiento del sistema. Esto se hace especificando una latencia, es decir, un período de tiempo, en el cual cada variable controlada debe completar su respuesta. Nuevamente, la latencia puede ser una constante simple o una función arbitrariamente compleja del estado del sistema.[1]

2.8.1 Especificar el Comportamiento de Cada Variable Controlada.

El siguiente ejemplo ilustra cómo hacer esto utilizando las instrucciones tradicionales "debe" para los requisitos. Por lo general, es más simple comenzar definiendo el comportamiento de cada variable controlada. Por ejemplo, parte de la función de valor ideal para la variable controlada Control de Calor está dada por los siguientes requisitos:

1. Si el Modo Regulador es NORMAL y la Temperatura Actual es menor que la Temperatura Mínima Deseada, el Control de Calor se configurará en Encendido.

2. Si el Modo Regulador es NORMAL y la Temperatura Actual es mayor que la Temperatura Deseada Superior, el Control de Calor se configurará en Apagado.

Vale la pena destacar varios aspectos sobre estos dos requisitos. Los nombres de las variables monitoreadas y controladas, los modos, las variables internas y sus valores se utilizan como vocabulario de los requisitos. Esto ayuda a vincular los requisitos con el marco que se creó para su definición en los pasos anteriores.

Práctica recomendada 2.8.1: Utilice los nombres de las variables monitoreadas y controladas, los modos del sistema y las variables internas al escribir los requisitos detallados del sistema.

2.8.2 Especificar el Requisito como una Condición y un Valor Asignado.

Con respecto a los dos requisitos previstos en la sección 2.8.1, observe también cómo cada requisito se divide en una condición bajo la cual el requisito mantiene una asignación de un valor a la variable controlada. La condición bajo la cual se cumple el requisito se divide a su vez en los modos del sistema (por ejemplo, el modo del regulador es NORMAL) y otras condiciones basadas en las variables monitoreadas y las variables internas (por ejemplo, la Temperatura Actual es menor que la Temperatura Mínima Deseada). Este patrón es muy común cuando se especifican los requisitos como declaraciones obligatorias.

Práctica recomendada 2.8.2: Para cada requisito, especifique los modos del sistema y las condiciones bajo las cuales se aplicará el requisito, seguido del cambio en la variable afectada.

2.8.3 Asegúrese de que los Requisitos Detallados estén Completos.

Los dos requisitos previstos en la sección 2.8.1 son incompletos. Estos requisitos no dicen nada sobre el valor de Control de Calor cuando el Modo del regulador es INIT o FAILED o cuando la Temperatura Actual está entre la Temperatura Mínima Deseada y la Temperatura Deseada Superior.

1 La especificación de los requisitos como función de valor ideal, tolerancia y latencia se describen en la metodología CoRE. [4 y 5].

Lo ideal sería que el valor de cada variable controlada y de cada variable interna se especificara para todos los modos y condiciones posibles dentro de ese modo. El valor del Control de Calor se puede especificar completamente con tres requisitos adicionales:

3. Si el Modo regulador es INIT, el Control de Calor se configurará en Apagado.

   Justificación: un Regulador que se está inicializando no puede regular la Temperatura Actual y el Control de Calor debe apagarse.

4. Si el Modo del regulador es NORMAL y la Temperatura Actual es mayor o igual a la Temperatura Mínima Deseada y menor o igual a la Temperatura Deseada Superior, no se modificará el valor del Control de Calor.

   Justificación: cuando la Isolette se está calentando hacia la Temperatura Deseada Superior, la Fuente de Calor debe dejarse encendida hasta que se alcance dicha temperatura. De manera similar, si la Isolette se está enfriando hacia la Temperatura Deseada Inferior, la Fuente de Calor debe dejarse apagada hasta que se alcance dicha temperatura.

5. Si el Modo Regulador es FALLA, el Control de Calor se configurará en apagado.

   Justificación: En el Modo de Falla, el Regulador no puede regular la Temperatura Actual de la Isolette y el Control de Calor debe apagarse.

Este conjunto de cinco requisitos garantiza que se defina la función de valor ideal para la variable de Control de Calor para todos los estados posibles del sistema. Este proceso debe repetirse para cada variable interna y controlada. Esto dará como resultado una cadena de asignaciones que define cómo cambiarán idealmente las variables controladas en respuesta a los cambios en las variables monitoreadas.

A veces no es importante hacer una asignación a una variable en un modo o condición particular. Por ejemplo, cuando el modo del regulador es INIT o FAILED, la Temperatura Mostrada en la interfaz del operador puede no ser confiable. Sin embargo, es importante documentar esto ya que los usuarios de esta información (en este caso la interfaz del operador) necesitan saber que el valor de la Temperatura de Visualización no es significativo bajo esas condiciones.

Esto se puede hacer agregando un requisito de que el valor no se especifique, por ejemplo:

6. Si el Modo Regulador no es NORMAL, el valor de la Temperatura de Visualización es NO ESPECIFICADO.

   Justificación: En modos distintos a NORMAL, el valor de la Temperatura de Visualización no es importante y no se debe utilizar.

Práctica recomendada 2.8.3: Asegúrese de que los requisitos detallados estén completos, es decir, se asigne un valor ideal a cada variable controlada y variable interna para cada estado del sistema. Utilice un valor NO ESPECIFICADO cuando no exista una asignación significativa para un estado del sistema.

2.8.4 Asegúrese de que los Requisitos Detallados sean Coherentes.

También se debe tener cuidado de garantizar que los requisitos sean coherentes, es decir, que los requisitos no especifiquen dos valores para una variable para el mismo estado del sistema. Por ejemplo, si el primer requisito se hubiese escrito así

1. Si el Modo Regulador es NORMAL y la Temperatura Actual es menor o igual a la Temperatura Mínima Deseada, el Control de Calor se configurará en Encendido.

entraría en conflicto con el cuarto requisito, ya que el primer requisito especificaría que el Control de Calor toma el valor Encendido y el cuarto requisito especificaría que el Control de Calor conserva su valor anterior cuando la Temperatura Actual es mayor o igual a la Temperatura Mínima Deseada.

Práctica recomendada 2.8.4: Garantizar que los requisitos detallados sean consistentes, es decir, que solo se asigne un valor ideal a cada variable controlada y a cada variable interna para cada estado posible del sistema.

2.8.5 Asegúrese de que los Requisitos Detallados no se Dupliquen.

También se deben verificar los requisitos para garantizar que el mismo requisito no se establezca más de una vez, ya sea mediante duplicación directa o mediante requisitos en los que los modos y las condiciones se superponen.

Práctica recomendada 2.8.5: Garantizar que no haya dos requisitos detallados que se dupliquen entre sí, es decir, especificar el mismo resultado para modos y condiciones superpuestos.

2.8.6 Organizar los Requisitos.

Los requisitos para cada variable controlada y variable interna se presentan en la función que define esa variable. De esta manera, los diagramas de dependencia sirven como una tabla de contenidos visual para los requisitos detallados. Por ejemplo, los requisitos de valor ideal para la variable controlada de Control de Calor se proporcionan en la Función Gestión de Fuente de Calor del apéndice A.5.1.3.

Práctica recomendada 2.8.6: Presentar los requisitos detallados del sistema en la función que produce la variable que se especifica. Esto crea una organización en la que la definición de cada variable se puede rastrear directamente hasta su función principal.

2.8.7 Definir la Latencia Aceptable para Cada Variable Controlada.

Una vez especificada la función de valor ideal para cada variable controlada y variable interna, los requisitos se completan especificando la tolerancia y la latencia para cada variable controlada.

La latencia especifica el retraso máximo entre el momento en que una o más variables monitoreadas cambian de valor y el momento en que una variable controlada afectada debe cambiar su valor. Por ejemplo, si la latencia para la variable controlada Control de Calor se especifica como 6 segundos, entonces la variable controlada Control de Calor debe asumir su nuevo valor dentro de los 6 segundos posteriores a que la Temperatura Actual, o cualquier otra

variable monitoreada, cambie su valor. La latencia puede ser una constante única o puede ser una función arbitraria de las variables monitoreadas e internas. A menudo, muchas variables controladas compartirán la misma latencia, y se puede definir una constante o función única y asignarla a varias variables controladas. En cualquier caso, siempre se debe incluir una justificación para la latencia especificada.

Para la variable controlada de Control de Calor, se define una latencia constante de 6 segundos en la Función Gestión de Fuente de Calor del apéndice A.5.1.3. Tenga en cuenta que se proporciona una justificación de la latencia que se remonta a los supuestos ambientales (consulte la tabla 14).

Tabla 14. Comportamiento de Latencia Permitido de la Fuente de Calor

Nombre	Tipo	Valor	Unidades	Interpretación física
Latencia Permitida de la Fuente de Calor	Real	6.0	Segundo	El tiempo máximo durante el cual la Fuente de Calor debe estar encendida o apagada para garantizar un funcionamiento aceptable del sistema Isolette.
Justificación: Dado que un Isolette cerrado se calentará o enfriará a una velocidad máxima de 1 °F por minuto (EA-IS1 y EA-IS2), encender o apagar la Fuente de Calor dentro de los 6 segundos garantiza que la Temperatura Actual no cambiará en más de 0,1 °F, la precisión y resolución requeridas del Sensor de Temperatura (EA-TS2).				

Práctica recomendada 2.8.7: Definir la latencia aceptable para cada variable controlada junto con la justificación de su valor como parte de los requisitos detallados del sistema.

2.8.8 Definir la Tolerancia Aceptable para Cada Variable Controlada.

Cuando una variable controlada representa un valor numérico (en lugar de un valor Booleano o enumerado), también se debe especificar la tolerancia aceptable con respecto al valor ideal. Por ejemplo, un valor controlado que muestra la altitud de la aeronave debe incluir el delta por encima y por debajo de la altitud real que se puede tolerar. Para los valores booleanos y enumerados, la tolerancia no suele ser importante y no se debe indicar. Por ejemplo, dado que la variable controlada Control de Calor solo puede adoptar los valores de encendido y apagado, su tolerancia se especifica como no aplicable (N/A) (consulte el apéndice A.5.1.3).

Práctica recomendada 2.8.8: Definir la tolerancia aceptable para cada variable controlada numéricamente.

2.8.9 No Definir Latencia y Tolerancia para Variables Internas.

Tenga en cuenta que no se especifican tolerancias ni latencias para las variables internas, ya que el requisito es que la variable controlada cambie con la latencia y la tolerancia especificadas. No existen tales requisitos para las variables internas, que son simplemente ayudas que se utilizan para dividir la especificación de la función de valor ideal en partes manejables.

Práctica recomendada 2.8.9: No definir latencia y tolerancia para variables internas.

Los requisitos detallados del sistema para el termostato Isolette que utiliza declaraciones "deberá" se detallan en el apéndice A.5.

2.8.10 Formas Alternativas de Especificar Requisitos.

Existen otras formas de especificar los requisitos además de las instrucciones "debe". Un enfoque consiste en utilizar un modelo gráfico para definir la función de valor ideal y complementarla con las tolerancias y latencias para cada variable controlada. Una desventaja de este enfoque es que ya no resulta obvio qué constituye un requisito individual.

Otro enfoque consiste en utilizar tablas para especificar los requisitos. En la tabla 15 se muestra un ejemplo para especificar el valor ideal del Control de Calor.

En la tabla 15, el valor del Control de Calor se proporciona en la fila inferior como una función del Modo del Regulador y las condiciones actuales. Cada celda interior sin sombrear corresponde a un requisito y está etiquetada con los mismos identificadores que se utilizan en el apéndice A.5.1.3 para etiquetar las declaraciones de obligación. Tenga en cuenta que la especificación del valor del Control de Calor en la fila inferior puede ser una constante (por ejemplo, encendido o apagado) o una función del estado anterior y actual (por ejemplo, valor anterior). La principal ventaja de una presentación tabular de este tipo es que es más fácil confirmar que los requisitos están completos (se define un valor ideal para cada modo y estado del sistema) y son consistentes (solo se define un valor ideal para cada modo y estado del sistema).

Tabla 15. Especificación Tabular de Requisitos

Modo regulador	Condición		
INIT			REQ-MT1 SIEMPRE
NORMAL	REQ-MT2 Temperatura Actual < Temperatura Mínima Deseada	REQ-MT4 Temperatura Mínima Deseada ≤ Temperatura Actual ≤ Temperatura Deseada Superior	REQ-MT3 Temperatura Actual > Temperatura Deseada Superior
FALLA			REQ-MT5 SIEMPRE
Control de calor =	Encendido	Valor Anterior	Apagado

MT = Monitor de Temperatura

Se han propuesto muchos formatos diferentes para especificar funciones utilizando tablas. SCR [2 y 3] y CoRE [4 y 5] hacen un uso intensivo de tablas de condiciones (similares a la tabla 15), tablas de eventos y tablas de modos. RSML [8] y SpecTRM [15] utilizan un formato denominado tablas and/or. En la referencia 43 se describen otros formatos tabulares.

Sin embargo, la cuestión clave no es si los requisitos se especifican como declaraciones obligatorias o modelos gráficos o tablas, sino que los requisitos deben especificar la relación que el sistema mantendrá entre las variables monitoreadas y controladas, y deben enunciarse de una manera que sea completa, consistente, inequívoca y comprobable.

2.9 DEFINIR LOS REQUISITOS DEL SOFTWARE.

Consulte la tabla 16 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para definir los requisitos de software.

Tabla 16. Definir los requisitos del software

Prácticas recomendadas para los niveles principal y secundario
2.9 Definir los requisitos del software: Con una estructuración cuidadosa, los requisitos del software y su arquitectura pueden corresponderse directamente con los requisitos del sistema y su arquitectura. Esta práctica recomendada describe cómo definir los requisitos del software como una extensión directa de los requisitos del sistema.
2.9.1 Para cada entrada que el software debe leer, se debe proporcionar una descripción de todo lo que el desarrollador del software debe saber para acceder a la entrada e interpretarla correctamente. Esto puede incluir una descripción de la entrada, el formato de los datos, el rango de valores que puede asumir, su ubicación y cualquier protocolo que se deba seguir al acceder a ella.
2.9.2 Para cada entrada que el software debe leer, proporcionar una especificación de su precisión, donde la precisión se refiere a la cantidad en que su valor puede desviarse de su valor ideal.
2.9.3 Para cada entrada que el software debe leer, proporcionar una especificación de su latencia, donde la latencia se refiere al tiempo máximo que su valor puede demorarse respecto del valor real de la variable o variables monitoreadas que representa.
2.9.4 Para cada variable monitoreada, especifique cómo recrear una imagen de la variable monitoreada en el software a partir de las variables de entrada.
2.9.5 Para cada variable monitoreada, especifique cómo recrear su atributo de estado a partir de las variables de entrada.
2.9.6 Si se deben tomar decisiones de diseño al recrear una variable monitoreada en el software que puedan afectar el comportamiento del sistema visible externamente o la seguridad del sistema, marque esas decisiones como requisitos de software derivados que deben revisarse en el proceso de evaluación de seguridad.

Tabla 16. Definición de los requisitos del software (continuación)

Prácticas recomendadas para los niveles principal y secundario
2.9.7 Para cada salida que el software debe configurar, proporcione una descripción de todo lo que el desarrollador del software debe saber para acceder y configurar correctamente su valor. Esto puede incluir una descripción de la salida, el formato de los datos, el rango de valores que puede asumir, su ubicación y cualquier protocolo que se deba seguir para acceder a ella.
2.9.8 Para cada variable de salida que el software debe configurar, proporcione una especificación de su latencia, donde latencia se refiere al tiempo máximo permitido desde que se establece la variable de salida hasta que cambia el valor de la variable controlada.
2.9.9 Para cada salida que el software debe establecer, proporcionar una especificación de su precisión, donde la precisión se refiere a la cantidad en que la variable de control afectada puede divergir de su valor ideal.
2.9.10 Para cada variable controlada, especifique cómo establecer los valores de las variables de salida en función del valor de la imagen de la variable controlada en el software.
2.9.11 Para cada variable controlada, confirme que la latencia y la precisión especificadas en los requisitos del sistema se pueden cumplir dada la latencia y la precisión de las variables de entrada y salida y el tiempo de cálculo del software.

En algún momento, se hace necesario asignar los requisitos del sistema al hardware y al software. A medida que ha aumentado el uso de procesadores de propósito general, cada vez más requisitos del sistema se asignan al software. De hecho, los requisitos detallados del sistema y los requisitos del software a menudo se parecen notablemente. Sin embargo, normalmente hay suficientes diferencias como para que los requisitos del sistema y los requisitos del software se traten como dos especificaciones separadas. Este tratamiento se debe en gran medida a que las entradas y salidas del software no coinciden exactamente con las variables monitoreadas y controladas que forman la base de los requisitos del sistema. Esta sección describe un enfoque que proporciona una transición fluida de los requisitos del sistema a los requisitos del software en el que los requisitos del software se crean ampliando los requisitos del sistema.

La base del enfoque de esta sección es el modelo de cuatro variables desarrollado por Parnas y Madey [2] como parte de la metodología SCR. El modelo de cuatro variables aclara la relación entre los requisitos funcionales del sistema y del software, de modo que los requisitos del software se puedan especificar como una adición a los requisitos del sistema. En la figura 11 se muestra una descripción general del modelo de cuatro variables.

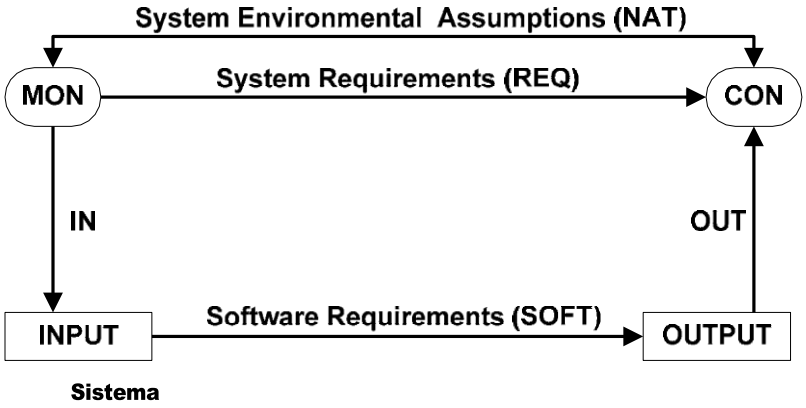


Figura 11. El modelo de cuatro variables

Las variables monitoreadas y controladas, los supuestos ambientales del sistema y los requisitos del sistema ya se han analizado en profundidad. Las variables monitoreadas (MON) consisten en las cantidades del entorno que el sistema monitoreará, y las variables controladas (CON) son las cantidades del entorno que el sistema controlará. Los supuestos ambientales (NAT) son las relaciones que mantiene el entorno del que depende el sistema, mientras que los requisitos del sistema (REQ) definen la relación que mantendrá el sistema entre las variables monitoreadas y controladas.

Las variables monitoreadas y controladas no pueden ser detectadas o controladas directamente por el software. En cambio, las variables monitoreadas deben ser proporcionadas al software por el sistema como variables de entrada (INPUT) que el software puede leer. De manera similar, el sistema debe traducir las variables de salida (OUTPUT) que el software puede escribir en cambios sobre las variables controladas en el entorno. Las variables monitoreadas y controladas están típicamente en un nivel más alto de abstracción que las variables de entrada y salida. Por ejemplo, mientras que una variable monitoreada puede ser la altitud de una aeronave informada por un altímetro de radar que es un entero entre -20 y 2500 pies, la variable de entrada correspondiente puede ser una palabra de bus ARINC 429 en la que la altitud está codificada como un entero de complemento a 2 de 17 bits ubicado en los bits 13 a 28 de la palabra de bus con el signo en el bit 29 y el punto decimal ubicado entre los bits 15 y 16 que representa un valor entre - 8192.0 y +8191.875.

La relación IN define la relación de cada variable de entrada con una o más variables monitoreadas. Esto consiste en la definición de la función de valor ideal de la variable de entrada, su latencia y su precisión. La función de valor ideal define el valor ideal de la variable de entrada como una función de una o más variables monitoreadas. La latencia especifica el tiempo máximo desde que se modifica una variable monitoreada (de la que depende la variable de entrada) hasta que la variable de entrada indica dicho cambio. La precisión especifica la cantidad en la que el valor real de la variable de entrada puede desviarse de su valor ideal.

De manera similar, la relación OUT define la relación de cada variable controlada con una o más variables de salida. Esto incluye una definición de la función de valor ideal de la variable controlada, su latencia y su precisión. La función de valor ideal define el valor ideal de la variable controlada como una función de una o más variables de salida. La latencia especifica el tiempo máximo desde que se modifica una variable de salida (de la que depende la variable

controlada) hasta que la variable controlada indica ese cambio. La precisión especifica la cantidad en la que el valor real de la variable controlada puede desviarse de su valor ideal.

La especificación de las relaciones NAT, REQ, IN y OUT limita implícitamente el comportamiento permitido del software, que se muestra en la figura 11, como la relación SOFT, sin especificar el diseño del software. Si bien la relación SOFT define los requisitos reales para el software, la correspondencia entre los requisitos del sistema especificados en REQ y los requisitos del software en SOFT no es obvia. Los requisitos del sistema y los requisitos del software se encuentran en diferentes niveles de abstracción y tienen diferentes dominios y rangos. Una forma de abordar estas preocupaciones es extender los requisitos del software SOFT en tres partes, IN', REQ' y OUT' [44], como se muestra en la figura 12.

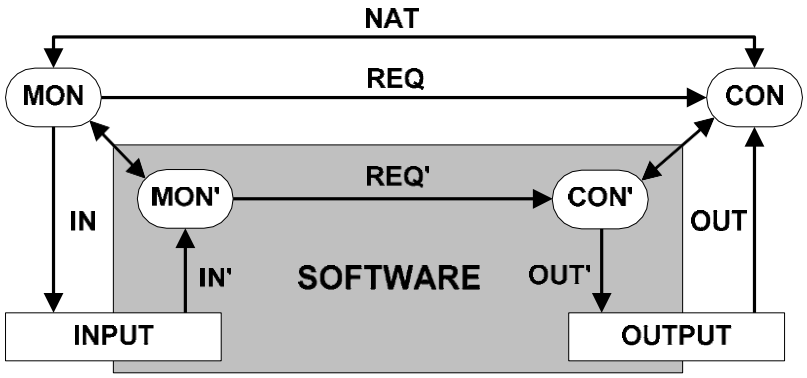


Figura 12. Requisitos de software ampliados

En la figura 12, IN' es la inversa de IN y define cómo recrear una imagen de las variables monitoreadas (MON') en el software a partir de las variables INPUT. De manera similar, OUT' es la inversa de OUT y define cómo cambiarán las variables OUTPUT a medida que se cambia una imagen de las variables controladas (CON') en el software. Así como REQ describe la relación que debe mantener el sistema entre las variables monitoreadas y controladas, REQ' describe la relación que debe mantener el software entre las imágenes de las variables monitoreadas y controladas. La relación SOFT que especifica los requisitos del software se reemplaza por IN', REQ' y OUT'.

La principal ventaja de esta extensión es que hace que la relación entre los requisitos del sistema y los requisitos del software sea directa y sencilla. La función de valor ideal definida en el requisito del sistema, REQ, se asigna directamente a una función idéntica en el requisito del software, REQ'. Como resultado, los requisitos del software en realidad consisten en una adición a los requisitos del sistema que define las variables INPUT y OUTPUT y las relaciones IN, OUT, IN' y OUT'. Otra ventaja es que, dado que IN' y OUT' se implementan en el software (puede ser útil pensar en ellos como la especificación para los controladores de hardware), proporcionan una separación útil de preocupaciones en los requisitos del software; REQ' cambiará a medida que cambien los requisitos del sistema, mientras que IN' y OUT' cambiarán a medida que cambie el hardware subyacente. Esto ayuda a aislar el software que implementa los requisitos del sistema de los cambios en la plataforma de hardware.

Cabe señalar que MON' y CON' no son las mismas variables de nivel de sistema representadas por MON y CON. Son las imágenes de las variables monitoreadas y controladas recreadas en el software. El hardware y el software introducen pequeñas diferencias en los valores, y se introducen latencias entre las imágenes y las cantidades reales en el entorno. Por ejemplo, el valor de la altitud de una aeronave recreada en el software siempre va a estar por detrás, y diferir un poco, de la altitud real de la aeronave. Estas diferencias deben tenerse en cuenta para garantizar que se cumplan las latencias y tolerancias permitidas especificadas para las variables controladas. Para mayor claridad en la siguiente discusión, se utilizará un signo de prima (') para distinguir una variable monitoreada o controlada específica recreada en el software (por ejemplo, velocidad aerodinámica') de la variable monitoreada o controlada real (velocidad aerodinámica). En la práctica, esta distinción suele quedar clara a partir del contexto.

2.9.1 Especificar las variables de entrada.

Para concretar estas ideas, considere el ejemplo del termostato Isolette del apéndice A. El primer paso fue definir las variables de entrada. El objetivo de esto es proporcionar al ingeniero de software toda la información necesaria para leer e interpretar correctamente las variables de entrada. La forma de hacerlo variará según el tipo de interfaz de hardware.

Por ejemplo, la variable monitoreada Temperatura Actual se especifica en el apéndice A.3.2 como un número real entre 68,0° y 105,0°F con una precisión de al menos 0,1°F. La información sobre esta variable monitoreada se proporciona al software del termostato mediante dos variables de entrada, “Curr Temp In” y “Curr Temp Status In”.

La variable de entrada Curr Temp In se define en la tabla 17 como un entero sin signo de 16 bits que representa la Temperatura Actual multiplicada por 256 (es decir, se utilizan 8 bits para representar fracciones de un grado). Este entero se encontrará en las dos palabras de memoria contiguas de 8 bits ubicadas en las direcciones X'0048' y X'0049'.

Tabla 17. Variable de entrada Curr Temp In

Descripción	Temperatura Actual de la isoleta en °F multiplicada por 256
Representación de datos	Entero sin signo de 16 bits
Valores	[0..65,535]
Ubicación	Palabras de memoria X'0048' y X'0049'
Valor ideal	Temperatura Actual multiplicada por 256
Estado latente	0,20 milisegundos
Exactitud	±20 (es decir, ±0,08 °F)

Práctica recomendada 2.9.1: Para cada entrada que el software debe leer, proporcione una descripción de todo lo que el desarrollador del software debe saber para acceder a la entrada e interpretarla correctamente. Esto puede incluir una descripción de la entrada, el formato de los datos, el rango de valores que puede asumir, su ubicación y cualquier protocolo que se deba seguir al acceder a ella.

2.9.2 Especificar la precisión de cada variable de entrada.

Se especifica que la precisión de Curr Temp In sea de al menos ±20, lo que equivale aproximadamente a ±0,08 °F. Esta es la desviación máxima introducida por el hardware con respecto al valor real de la variable monitoreada de Temperatura Actual, que se define como el valor producido por el Sensor de Temperatura.[1]

Práctica recomendada 2.9.2: Para cada entrada que el software debe leer, proporcionar una especificación de su precisión, donde la precisión se refiere a la cantidad en que su valor puede desviarse del valor ideal.

2.9.3 Especificar la latencia de cada variable de entrada.

De manera similar, la latencia especifica el tiempo máximo desde que se modifica la variable monitoreada hasta que la variable de entrada indica dicho cambio. Para Curr Temp In, se especifica que es de 0,20 milisegundos (ms). La precisión y la latencia son necesarias para determinar con qué precisión y rapidez el software debe completar su cálculo. Esto se analiza en la sección 2.9.11.

Práctica recomendada 2.9.3: Para cada entrada que el software debe leer, proporcionar una especificación de su latencia, donde la latencia se refiere al tiempo máximo que su valor puede retrasarse respecto del valor real de la variable o variables monitoreadas que representa.

2.9.4 Especificar IN' para cada variable monitoreada.

Tenga en cuenta que la información presentada en la tabla 17 describe tanto la variable de entrada Curr Temp In como la parte esencial de su relación IN, es decir, la función ideal que define cómo calcular la variable de entrada a partir de las variables monitoreadas, su latencia y su precisión.

La otra variable de entrada que se utiliza para proporcionar al software información sobre el valor de la Temperatura Actual es Curr Temp Status In, que especifica si el valor de Curr Temp In es válido o no. Su definición se muestra en la tabla 18. Si se cree que el valor de Curr Temp In es válido, se establece en 0. De lo contrario, se le asigna un valor distinto de 0. Esta información se presenta como un entero sin signo de 8 bits ubicado en la dirección de memoria X'0046'. Tenga en cuenta que su valor inicial es 255. Esto garantiza que Curr Temp In se trate como no válido hasta que se haya leído correctamente al menos una vez.[2] El valor ideal que describe cómo se relaciona el valor de la variable de entrada con las variables monitoreadas proporciona la información mínima necesaria para implementar los requisitos del sistema. Esto se podría ampliar asignando valores específicos a diferentes tipos de fallas, pero como los requisitos del sistema especifican el mismo comportamiento independientemente del tipo de falla, esta información no es necesaria.

1 Tenga en cuenta que el sensor de temperatura puede introducir su propia inexactitud y latencia al detectar la temperatura real en la Isolette.

2 Recuerde que el atributo de estado de todas las variables monitoreadas debe inicializarse como inválido.

Tabla 18. ENTRADA Variable Corriente Temp Estado En

Descripción	Indicación de si Curr Temp In es válido
Representación de datos	Entero sin signo de 8 bits
Valores	[0 .. 255]
Valor inicial	255
Ubicación	Palabra de memoria X'0046'
Valor ideal	No hay error al detectar Curr Temp In → 0, Error al detectar Curr Temp In → 1 a 255
Estado latente	0,20 mseg
Exactitud	N / A

El siguiente paso es definir la relación IN' que describe cómo recrear el valor de Temperatura Actual monitoreado en el software. La relación IN' puede ser una asignación simple de uno a uno de una variable de entrada a una variable monitoreada, o puede ser una asignación compleja de varias variables de entrada a una variable monitoreada. Por ejemplo, la relación IN' que describe cómo recrear el valor de Temperatura Actual' es una función simple de la variable de entrada Curr Temp In como se muestra en la tabla 19.

Tabla 19. Relación IN' para el valor de la Temperatura Actual

	Temperatura Actual en < 17,408	17,408 ≤ Temperatura Actual en ≤ 26.880	Temperatura Actual en > 26,880
Valor =	68.0	Temperatura Actual de entrada/256,0	110.0

Si el valor de Curr Temp In es menor que 17 408, el valor de Current Temperature' se establece en 68,0 °F. Si Curr Temp In es mayor o igual que 17 408 y menor o igual que 26 880, el valor de Current Temperature' se establece en Curr Temp In dividido por 256,0. Si el valor de Curr Temp In es mayor que 26 880, el valor de Current Temperature' se establece en 105,0 °F.

Práctica recomendada 2.9.4: Para cada variable monitoreada, especifique cómo recrear una imagen de la variable monitoreada en el software a partir de las variables de entrada.

2.9.5 Especificar el estado de cada variable monitoreada.

La saturación del valor calculado de la Temperatura Actual en 68,0° y 105,0°F se realiza porque la suposición ambiental EA-TS3 para el Sensor de Temperatura establece que la variable monitoreada de Temperatura Actual estará entre 68,0° y 105,0°F, y puede haber otras partes de la especificación que dependan de esta suposición. El hecho de que la variable de entrada Curr Temp In pueda contener físicamente un valor que (después de la conversión) varíe entre 0,0° y 256,0°F es un artefacto del tamaño de palabra estándar de la implementación en lugar de un requisito.

Como se explicó en la sección 2.4, cada variable monitoreada debe tener un atributo de estado asociado para indicar su estado, es decir, qué tan bien puede ser detectada por el sistema. La forma de configurar el atributo de estado de la variable monitoreada recreada debe especificarse exactamente de la misma manera que se especifica cómo configurar su valor. Como ejemplo, la especificación para configurar el atributo de estado de la variable monitoreada Temperatura Actual se muestra en la tabla 20. Aquí, el estado se configura como Inválido si la variable de entrada Estado de Temperatura Actual no es 0 o si el valor de Temperatura Actual es menor que 17,409 o mayor que 16,880.

Tabla 20. Relación IN' para el estado de la Temperatura Actual

		Estado de Temperatura Actual en = 0	
	Temperatura Actual Estado En ≠ 0	Temperatura Actual en < 17,408 O Temperatura Actual en > 26,880	17,408 ≤ Temperatura Actual en AND Temperatura Actual en ≤ 26,880
Estado =	Inválido	Inválido	Válido

Práctica recomendada 2.9.5: Para cada variable monitoreada, especifique cómo recrear su atributo de estado a partir de las variables de entrada.

2.9.6 Decisiones de diseño de banderas como requisitos derivados.

La elección de saturar el valor de la 'Temperatura Actual' es un ejemplo de un requisito derivado, como se define en RTCA DO-178B [27] y DO-248B [28], es decir, un requisito que no es directamente atribuible a un requisito de nivel superior. Esto se indica por el hecho de que se podrían realizar otras elecciones de diseño que afectarían el comportamiento del sistema visible externamente, sin dejar de cumplir con los requisitos del sistema. Por ejemplo, el rango que se muestra en la tabla 20 se podría ampliar a 17 000 a 27 000. Esto reduciría el riesgo de poner el termostato en un estado de falla debido al ruido en el valor de Curr Temp In, pero abre la posibilidad de operar con una temperatura peligrosamente alta. Las decisiones como estas que afectan el comportamiento del sistema visible externamente se deben marcar como un requisito de software derivado para garantizar que se proporcionen al proceso de evaluación de seguridad del sistema, como se indica en DO-178B.

Práctica recomendada 2.9.6: Si se deben tomar decisiones de diseño al recrear una variable monitoreada en el software que pueda afectar el comportamiento del sistema visible externamente o la seguridad del sistema, marcar esas decisiones como requisitos de software derivados que deben revisarse mediante el proceso de evaluación de seguridad.

2.9.7 Especificar las variables de salida.

La definición de las variables OUTPUT y las relaciones OUT y OUT' se realizan de forma muy similar, excepto que el valor ideal, la latencia y la precisión se especifican para la variable controlada, no para la variable de salida. Por ejemplo, la Fuente de Calor se puede encender y apagar mediante el termostato escribiendo en la variable de salida de control de calor. La definición de esta variable se muestra en la tabla 21.

Tabla 21. SALIDA Control de calor variable SALIDA

Descripción	Comando para encender o apagar la Fuente de Calor
Representación de datos	Entero sin signo de 8 bits
Valores	[0 .. 255]
Ubicación	Palabra de memoria X'0060'
Valor ideal	0 → control de calor = apagado 2 a 255 → control de calor = encendido
Estado latente	0,60 mseg
Exactitud	N / A

Esta variable de salida es un entero sin signo de 8 bits ubicado en la dirección de memoria X'0060'. Un valor de 0 apaga la Fuente de Calor, un valor de 2 a 255 la enciende.[1]

Práctica recomendada 2.9.7: Para cada salida que el software debe configurar, proporcione una descripción de todo lo que el desarrollador del software debe saber para acceder y configurar correctamente su valor. Esto puede incluir una descripción de la salida, el formato de los datos, el rango de valores que puede asumir, su ubicación y cualquier protocolo.

2.9.8 Especificar la latencia de cada variable de salida.

Al igual que con las variables de entrada, también se deben especificar la latencia y la precisión de las variables de salida. Sin embargo, en el caso de las variables de salida, la latencia especifica el retraso máximo posible entre el momento en que se establece la variable de salida y el momento en que cambia el valor de la variable controlada.

Práctica recomendada 2.9.8: Para cada variable de salida que el software debe configurar, proporcionar una especificación de su latencia, donde la latencia se refiere al tiempo máximo permitido desde que se configura la variable de salida hasta que cambia el valor de la variable controlada.

2.9.9 Especificar la precisión de cada variable de salida.

En el caso de las variables de salida, la precisión define cuánto puede desviarse la variable controlada del valor ideal determinado por la variable de salida. En el caso de una cantidad numérica, se debe establecer en un rango. En el caso de una variable de salida discreta, la precisión se debe establecer en N/D.

Al igual que con las variables de entrada, la especificación de esta variable de salida incluye la definición de la variable de salida en sí y la parte esencial de la relación OUT debe implementar los requisitos, es decir, la función ideal que define el valor de la variable controlada a partir de la variable de salida y la latencia y precisión de la variable controlada.

1 El valor 1 no se utiliza para garantizar que los comandos para encender o apagar el control de calor difieran en más de un bit.

Práctica recomendada 2.9.9: Para cada salida que el software debe establecer, proporcionar una especificación de su precisión, donde la precisión se refiere a la cantidad en que la variable de control afectada puede divergir de su valor ideal.

2.9.10 Especifique OUT' para cada variable controlada.

La relación OUT' se define para cada variable de salida. Esto define cómo configurar la variable de salida en función del valor actual de una o más imágenes de variables controladas en el software. Por ejemplo, el valor de la variable de salida Heat Control Out es una asignación directa de la imagen de la variable controlada Heat Control' mantenida en el software, como se muestra en la tabla 22. Tenga en cuenta que cuando la imagen de la variable controlada Heat Control' en el software está activada, el implementador puede elegir configurar el valor de Heat Control Out en cualquier valor entre 1 y 255, ya que todos tienen el mismo efecto.

Tabla 22. Relación OUT para el control del calor

	Control de calor' = Apagado	Control de calor' = Encendido
Valor =	0	[1..255]

Con la definición de las variables INPUT y OUTPUT y las relaciones IN' y OUT', la especificación de los requisitos del software está prácticamente completa. El desarrollador de software ahora sabe cómo recrear las imágenes de las variables monitoreadas en el software y cómo establecer las variables de salida en función de las imágenes de las variables controladas en el software. La otra información que el desarrollador de software necesita saber es cómo cambiar la imagen de la variable controlada en el software cuando cambia la imagen de la variable monitoreada en el software (es decir, REQ'). Sin embargo, la función de valor ideal para REQ' es idéntica a la función de valor ideal definida para los requisitos detallados del sistema (es decir, la relación REQ) definida en la sección 2.8.

Práctica recomendada 2.9.10: Para cada variable controlada, especifique cómo establecer los valores de las variables de salida en función del valor de la imagen de la variable controlada en el software.

2.9.11 Confirmar la latencia y precisión generales.

La única tarea pendiente es confirmar que la latencia y precisión generales especificadas para cada variable controlada se pueden cumplir dada la latencia y precisión de las variables de entrada y salida y el tiempo de cálculo del software. Por ejemplo, la latencia y precisión de la variable controlada de control de calor se especifica en 6 segundos y N/A en el apéndice A.5.1.3. La latencia introducida en la detección de la Temperatura Actual se especifica en la tabla 18 en 0,20 ms. La latencia en la configuración del valor de la variable controlada de control de calor se especifica en la tabla 21 en 0,60 ms. Por lo tanto, la latencia total en la detección de las variables monitoreadas y la configuración de las variables controladas es de 0,8 segundos, lo que indica que el software debe completar sus cálculos en 5,2 segundos. Dado que el control de calor es una variable discreta de dos valores, la confirmación de su precisión es inmediata.

La DO-178B [27] especifica que los requisitos de software deben organizarse en requisitos de software de alto nivel y de bajo nivel. Además, los requisitos de software de alto nivel deben cumplir con los requisitos del sistema (objetivo 1 de la tabla A-3 del apéndice A), y los requisitos de software de bajo nivel deben cumplir con los requisitos de software de alto nivel (objetivo 1 de la tabla A-4 del apéndice A). Estos se definen en la DO-178B como

> "Requisitos de alto nivel: Requisitos de software desarrollados a partir del análisis de los requisitos del sistema, los requisitos relacionados con la seguridad y la arquitectura del sistema.
>
> Requisitos de bajo nivel: Requisitos de software derivados de alto nivel requisitos, requisitos derivados y restricciones de diseño a partir de los cuales se puede implementar directamente el código fuente sin información adicional".

¿Cuáles de los requisitos analizados hasta ahora son requisitos de software de alto nivel, cuáles son requisitos de software de bajo nivel y cómo se puede demostrar la conformidad de los requisitos de software con los requisitos del sistema? Estas relaciones se aclaran en la figura 13.

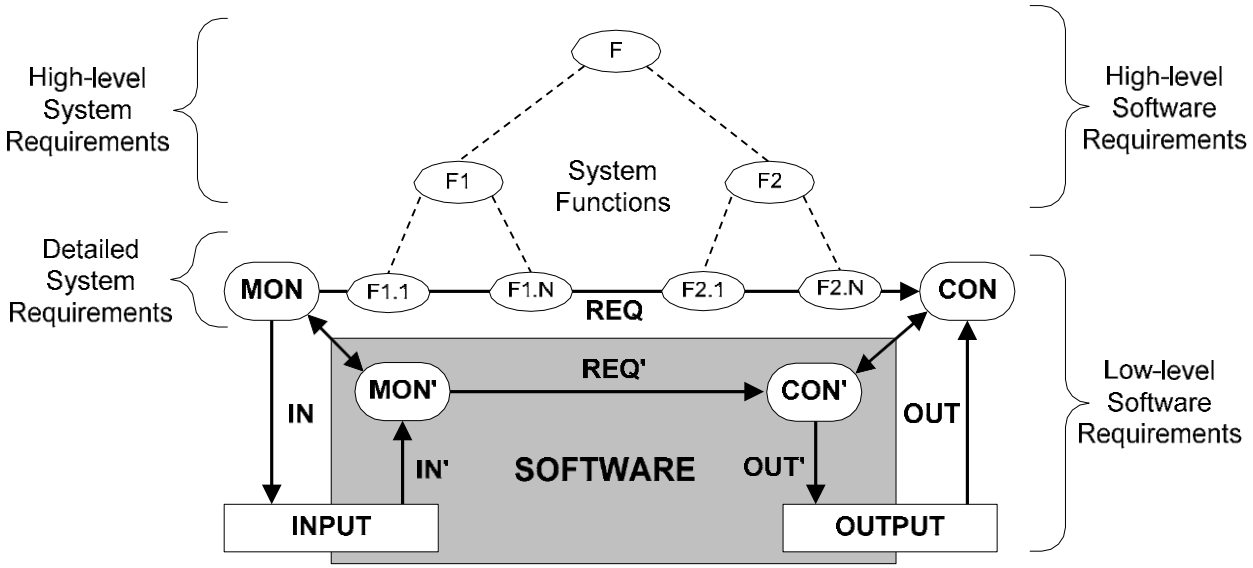


Figura 13. Requisitos de software de alto y bajo nivel

En la figura 13, la estructura de funciones del sistema anidadas se muestra sobre el modelo de cuatro variables. Las funciones del sistema de nivel más bajo (F1.1…F2.N) definen el comportamiento detallado del sistema.[1] requisitos, como se explica en la sección 2.8. En otras palabras, cada función del sistema de bajo nivel define su parte de la relación REQ entre las variables monitoreadas y controladas. Las funciones del sistema de nivel superior (F, F1 y F2) definen los requisitos del sistema de alto nivel, como se explica en la sección 2.6.

1 La latencia y la tolerancia definidas para cada variable controlada son requisitos de extremo a extremo y es más apropiado considerarlas como requisitos de alto nivel.

Los requisitos de software de bajo nivel, es decir, aquellos a partir de los cuales se puede implementar directamente el código fuente sin más información, incluyen IN', REQ' (cuya parte de comportamiento es idéntica a REQ) y OUT'. En otras palabras, los requisitos de software de bajo nivel consisten en los requisitos detallados del sistema junto con las definiciones de IN' y OUT'.

Dado que los requisitos de software de comportamiento son idénticos a los requisitos de sistema de comportamiento, la opción obvia para los requisitos de software de alto nivel son los requisitos de sistema que no son también requisitos de software de bajo nivel, es decir, los diagramas de dependencia, las definiciones de variables internas y cualquier requisito de sistema de alto nivel, como se analiza en la sección 2.5. Esta es una consecuencia natural de implementar los requisitos de sistema directamente en el software.

Por ejemplo, para el termostato Isolette, los requisitos de software de alto nivel serían los definidos para la Función Termostato (A.5), la Función Regulación de Temperatura (A.5.1) y la Función Monitor de Temperatura (A.5.2). Los requisitos de software de bajo nivel serían los definidos para todas las funciones de nivel más bajo, como la función de gestión de interfaz del regulador (A.5.1.1) y todas las relaciones IN' y OUT'.

Práctica recomendada 2.9.11: Para cada variable controlada, confirme que la latencia y la precisión especificadas en los requisitos del sistema se pueden cumplir dada la latencia y la precisión de las variables de entrada y salida y el tiempo de cálculo del software.

2.10 ASIGNAR REQUISITOS DEL SISTEMA A LOS SUBSISTEMAS.

Consulte la tabla 23 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para asignar requisitos del sistema a los subsistemas.

Tabla 23. Asignar requisitos del sistema a subsistemas

Prácticas recomendadas para los niveles principal y secundario
2.10 Asignar requisitos del sistema a los subsistemas: En el caso de sistemas de gran tamaño, los requisitos deben asignarse a subsistemas que pueden ser desarrollados de forma independiente por subcontratistas. Esta práctica describe cómo se puede hacer esto de una manera coherente con las demás prácticas recomendadas.
2.10.1 Completar lo suficiente la arquitectura funcional del sistema padre (teniendo en cuenta restricciones de implementación) para poder identificar funciones que probablemente sean candidatas a ser asignadas a subsistemas.
2.10.2 Al asignar una función del sistema a un subsistema separado, duplique la función de alto nivel. Requisitos de nivel de la función en el subsistema. Esto proporciona una trazabilidad directa entre los requisitos del subsistema superior y los requisitos de la función del sistema.
2.10.3 Desarrollar una descripción general del sistema para cada especificación del subsistema. Esto proporciona una esta es una visión de alto nivel importante del sistema que necesita el subcontratista y que ayuda a aclarar el alcance del subsistema y su relación con el sistema.

Tabla 23. Asignación de requisitos del sistema a subsistemas (continuación)

Prácticas recomendadas para los niveles principal y secundario
2.10.4 Identificar las variables monitoreadas y controladas para el subsistema que se comparten con el sistema principal. Asegúrese de que su definición en la especificación del subsistema sea coherente con su definición en la especificación del sistema. No incluya variables monitoreadas y controladas de la especificación del sistema que no sean utilizadas por el subsistema.
2.10.5 Crear nuevas variables monitoreadas y controladas para las variables internas del sistema que se exponen mediante la asignación de funciones del sistema al subsistema. Asegúrese de que sean coherentes con las definiciones de las variables internas en la especificación del sistema.
2.10.6 Especificar los conceptos operativos del subsistema para proporcionar a los desarrolladores una comprensión de cómo se utilizará el sistema y cómo interactuará con su entorno.
2.10.7 Identifique cualquier suposición ambiental que el subsistema comparte con su padre Sistema. Documente estos supuestos ambientales en la especificación del subsistema y asegúrese de que sean coherentes con los del sistema principal.
2.10.8 Identificar los supuestos ambientales asociados con las nuevas variables monitoreadas y controladas. Asegurarse de que sean coherentes con las variables internas definido en el sistema padre.
2.10.9 Complete la especificación de requisitos del subsistema • completando la descomposición funcional. • modificando la arquitectura funcional para tener en cuenta los requisitos de seguridad o las restricciones de implementación. • identificando los modos del subsistema. • desarrollando los requisitos detallados de comportamiento y rendimiento para el subsistema. • desarrollando los requisitos de software si el subsistema se implementará en software
2.10.10 Asegurar las latencias y tolerancias especificadas para las variables controladas en el los subsistemas son consistentes con las latencias y tolerancias de extremo a extremo especificadas para el sistema general.

En las secciones anteriores se analizó cómo desarrollar y especificar los requisitos para un sistema único definiendo las relaciones que el sistema mantendrá entre sus variables monitoreadas y controladas. En la sección 2.9 se analizó cómo realizar una transición fluida de los requisitos del sistema a los requisitos del software. Sin embargo, para manejar la complejidad de los sistemas grandes, es una práctica estándar dividir el sistema en subsistemas que se pueden desarrollar de forma independiente y asignar los requisitos del sistema a estos subsistemas [24 y 45]. En esta sección se analiza cómo se puede hacer esto dentro del marco descrito en este documento.

2.10.1 Identificar Funciones Del Subsistema.

Consideremos la situación que se muestra en la figura 14. En ella, el contratista que especifica los requisitos para el sistema 1 (contratista 1) identificó las variables monitoreadas y controladas y completó la descomposición funcional mediante las funciones F1.1… F1.N, F2 y F3.1… F3.N. En este punto (o quizás incluso antes), el contratista decide separar las funciones F1 y F3 como subsistemas separados para que sean desarrollados por subcontratistas independientes.

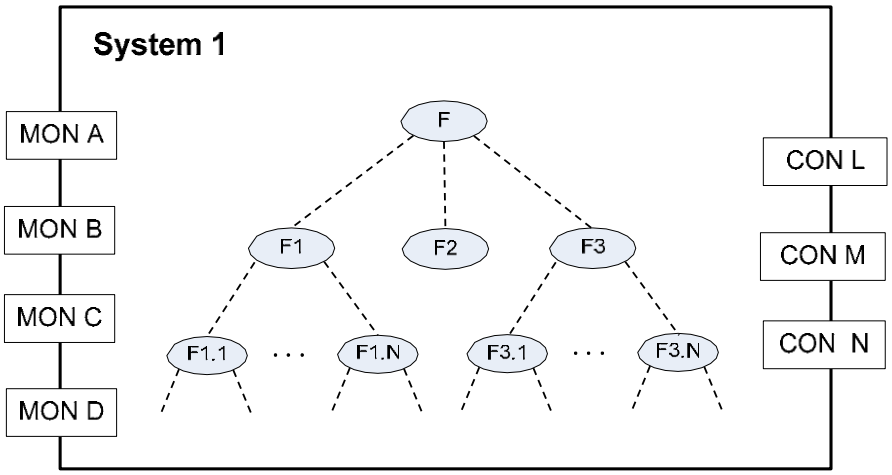


Figura 14. Descomposición Funcional Del Sistema 1

En la figura 15 se muestra una descripción general del enfoque para realizar esto. En este caso, la función F1 y sus funciones secundarias se trasladaron a la especificación del sistema 1.1, y la función F3 y sus funciones secundarias se trasladaron a la especificación del sistema 1.3. El objetivo es crear toda la especificación de requisitos del sistema 1 para que la utilice el contratista 1, una especificación de requisitos para el sistema 1.1 para que la utilicen tanto el contratista 1 como el subcontratista 1.1, y una especificación de requisitos para el sistema 1.3 para que la utilicen tanto el contratista 1 como el subcontratista 1.3.

El contratista 1 completará y mantendrá los requisitos del sistema 1. Contiene una especificación para todo el sistema que el contratista puede no desear compartir con el subcontratista 1.1 y el subcontratista 1.3. Sin embargo, la especificación para el sistema 1 no contendrá todos los requisitos detallados para las funciones F1 y F3.[1] En cambio, los requisitos detallados para la función F1 serán desarrollados y perfeccionados de manera cooperativa por el contratista 1 y el subcontratista 1.1. De manera similar, los requisitos detallados para la función F3 serán desarrollados de manera cooperativa por el contratista 1 y el subcontratista 1.3.

[1] La especificación para el sistema 1 puede contener los requisitos detallados para la función F2 que no fueron asignados a un subsistema separado.

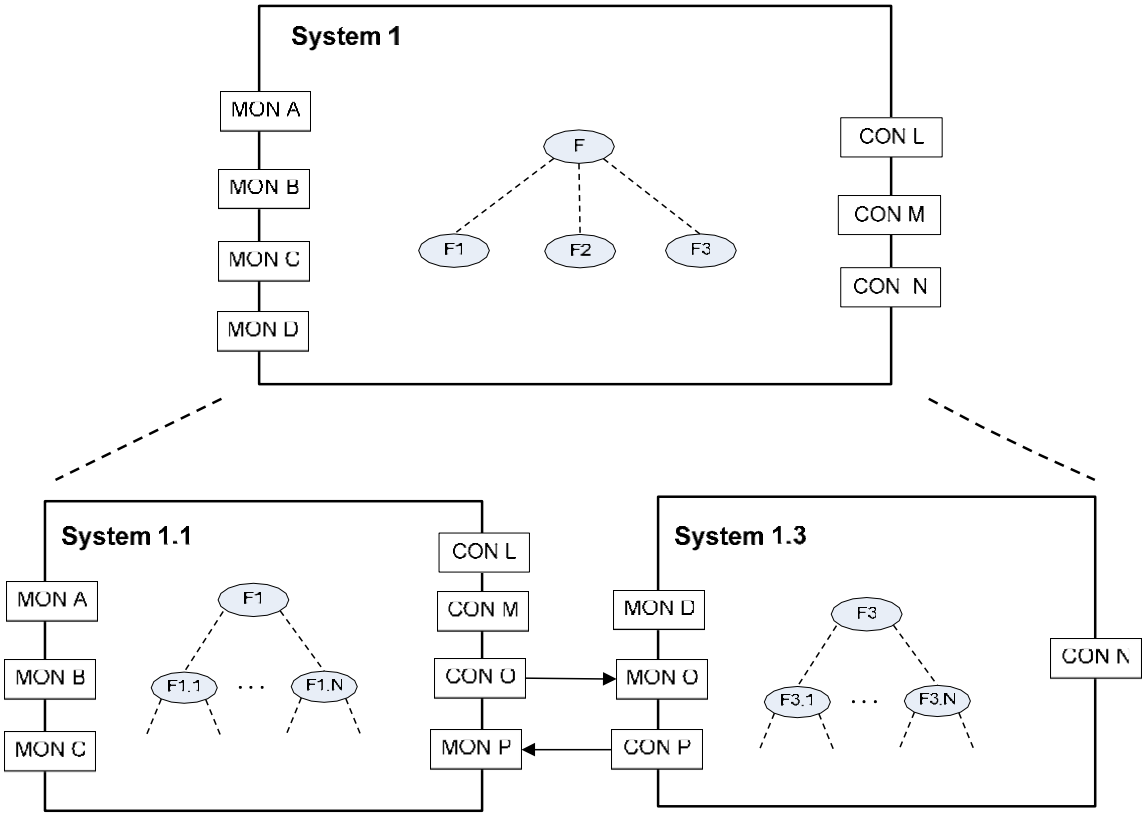


Figura 15. Descomposición Del Sistema 1 En Subsistemas

La división del trabajo entre el contratista y el subcontratista en el desarrollo de los requisitos para cada subsistema probablemente dependerá de la confianza que el contratista tenga en el subcontratista. Si el contratista tiene una gran confianza en el subcontratista, el contratista puede proporcionar sólo la especificación inicial de los requisitos del subsistema y permitir que el subcontratista haga la mayor parte del trabajo de perfeccionarla para obtener una especificación completa de los requisitos. Si el contratista no tiene una gran confianza en el subcontratista, el contratista puede optar por hacer todo el perfeccionamiento y proporcionar al subcontratista una especificación completa de los requisitos. Por lo general, la especificación del subsistema no se completará cuando se firme el contrato legal real, pero la asignación de responsabilidades para completar la especificación de los requisitos del subsistema se especificará en el contrato.

El propósito de la especificación de requisitos del subsistema es proporcionar una especificación común que el contratista y el subcontratista compartan. Sin embargo, puede que no sea necesario que el subcontratista comparta toda la información con el contratista. Puede haber detalles, como algoritmos, o incluso algunos requisitos detallados, que el subcontratista considere de su propiedad. Esto puede dar lugar a una situación en la que el subcontratista mantenga una parte de la especificación del subsistema por separado.

Práctica recomendada 2.10.1: Completar lo suficiente del sistema de arquitectura funcional principal (teniendo en cuenta las restricciones de implementación) para poder identificar funciones que probablemente sean candidatas para ser asignadas a subsistemas.

2.10.2 Duplicar Funciones Superpuestas Del Sistema Al Subsistema.

¿Qué información debe incluirse en la especificación de requisitos inicial para el subsistema? Como se muestra en la figura 15, la función F1 y sus funciones secundarias se trasladaron al subsistema 1.1, y la función F3 y sus funciones secundarias se trasladaron al subsistema 1.3. Esto implica que la jerarquía de funciones para F1 y F3, y cualquier requisito desarrollado para esa jerarquía, ahora forman parte de la especificación del subsistema 1.1 y el subsistema 1.3. Esto garantiza que no se descarte ningún trabajo realizado por el contratista. Si se planificó la descomposición en subsistemas, es posible que el contratista no haya desarrollado ninguna función secundaria para F1 y F3.

Además, la figura 15 también muestra que se conserva una copia de las funciones F1 y F3 (pero no de sus funciones hijas) junto con sus requisitos de alto nivel en la especificación para el sistema 1. La duplicación de los requisitos de alto nivel para F1 y F3 en la especificación del sistema 1 se realiza deliberadamente para proporcionar trazabilidad y continuidad entre el sistema 1 y el subsistema 1.1. Cada requisito de alto nivel de los subsistemas 1.1 y 1.3 se relaciona directamente con un requisito idéntico en el sistema 1.

En este punto, la especificación de requisitos para los subsistemas 1.1 y 1.3 consiste en poco más que las funciones copiadas del sistema 1 y sus requisitos de alto nivel. Como se discutió en las secciones 2.1 a 2.9, una especificación de requisitos completa consiste en mucho más que una lista de instrucciones. El siguiente paso es completar la información necesaria para hacer que las especificaciones de los subsistemas sean una especificación de requisitos completa por derecho propio. Esto consiste en agregar una descripción general del sistema, variables monitoreadas y controladas, conceptos operativos y supuestos ambientales a cada especificación de subsistema. Una vez hecho esto, la descomposición funcional de los subsistemas puede continuar hasta que se especifiquen los requisitos detallados de comportamiento y rendimiento.

Un ejemplo de esto se muestra en la figura 16 y los apéndices B, C y D. En la figura 16, una especificación para un FCS simple (apéndice B) se descompone en una función de guía de vuelo (FG) y una función AP. Estas funciones del FCS se dividen en una especificación de subsistema para un FGS simple (apéndice C) y un sistema AP simple (apéndice D).

La especificación del FCS general (apéndice B) contiene las funciones FG y AP. La descripción general del sistema del FCS se puede encontrar en el apéndice B.1. Como se muestra en su diagrama de contexto (figura B-1 en el apéndice B), monitorea la actitud de la aeronave desde el sistema de actitud y rumbo (AHS) y las entradas del piloto desde la interfaz de tripulación de vuelo (FCI). Establece los valores de las variables controladas de guía del director de vuelo (FD), FCS fallido y estado AP proporcionadas a las pantallas de vuelo
principales (PFD), así como la variable controlada de comandos de actuador proporcionada a los actuadores de la superficie de control.

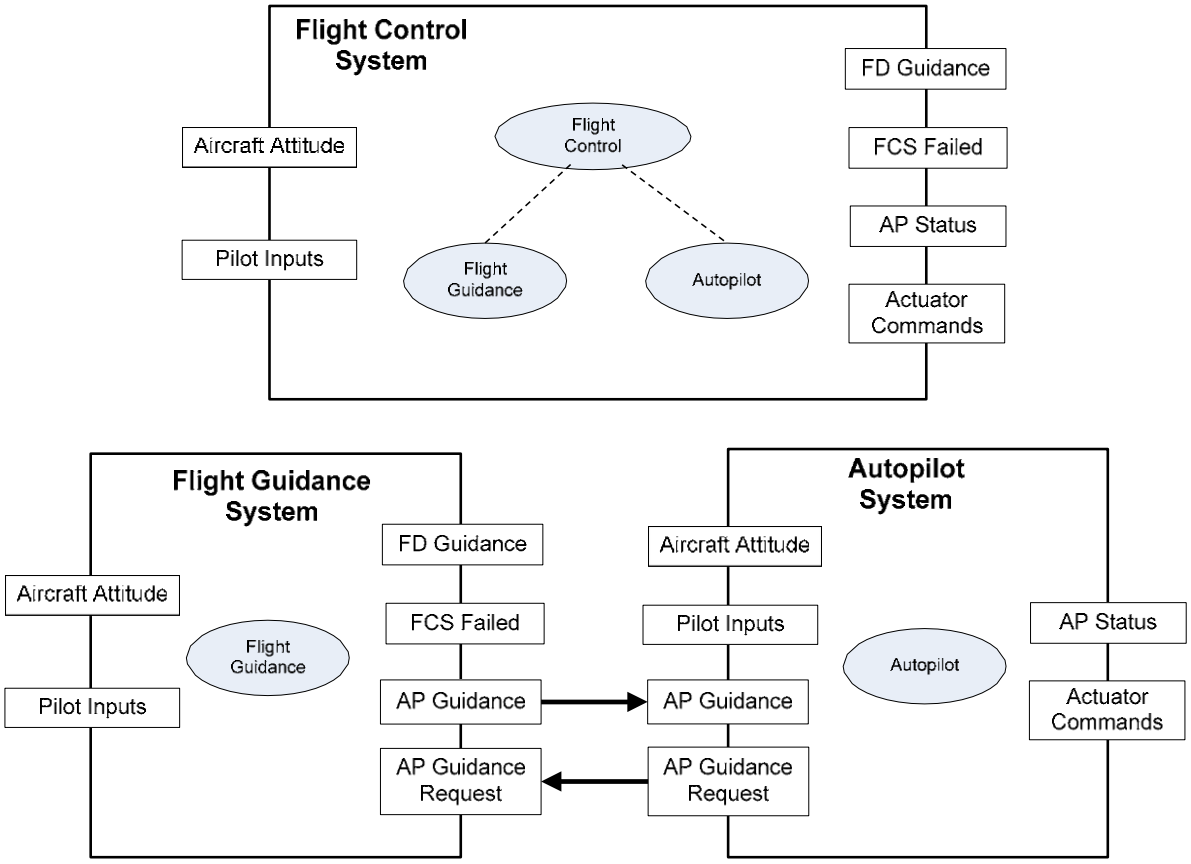


Figura 16. Asignación De Requisitos De FCS A Subsistemas

Los conceptos operativos del FCS se especifican como casos de uso en el apéndice B.2. Las entidades externas con las que interactúa el FCS se definen, junto con sus variables monitoreadas y controladas y sus supuestos ambientales, en el apéndice B.3. Las funciones del sistema para el FCS, junto con sus requisitos de alto nivel, se definen en el apéndice B.4. En particular, los requisitos de alto nivel para la función FG se especifican en el apéndice B.4.1, y los requisitos de alto nivel para la función AP se especifican en el apéndice B.4.2.

Los requisitos de alto nivel para la función FG se repiten como los requisitos más importantes del subsistema FGS en el apéndice C.4. De manera similar, los requisitos de alto nivel para la función AP se repiten como los requisitos más importantes del subsistema de piloto automático en el apéndice D.4. Esta duplicación conecta los requisitos de las especificaciones del subsistema con la especificación del sistema original.

Práctica recomendada 2.10.2: Al asignar una función del sistema a un subsistema independiente, duplique los requisitos de alto nivel de la función en el subsistema. Esto proporciona una trazabilidad directa entre los requisitos del subsistema superior y los requisitos de la función del sistema.

2.10.3 Desarrollar Una Descripción General Del Sistema Para Cada Subsistema.

El siguiente paso en el desarrollo de la especificación del subsistema FGS es completar la descripción general del sistema. En el apéndice C.1 se incluye una nueva descripción general que describe el propósito, el contexto y los objetivos del subsistema FGS. Esta información proporciona la visión de alto nivel que necesita el subcontratista del subsistema.

Práctica recomendada 2.10.3: Desarrollar una descripción general del sistema para cada especificación de subsistema. Esto proporciona una importante visión general del sistema que necesita el subcontratista y ayuda a aclarar el alcance del subsistema y su relación con el sistema.

2.10.4 Identificar Las Variables Monitoreadas Y Controladas Del Subsistema.

Tal como se describe en la sección 2.2, el siguiente paso consiste en definir el límite del subsistema mediante la identificación de las variables monitoreadas y controladas del subsistema. Tenga en cuenta que el subsistema FGS comparte muchas de las mismas interfaces externas con el FCS. Ambos sistemas interactúan con el FCI, el AHS y el PFD, y las definiciones de las variables monitoreadas y controladas para estas entidades externas deben ser coherentes con las de la especificación del FCS. Al mismo tiempo, existen diferencias que deben tenerse en cuenta (consulte la figura 16). Por ejemplo, el FGS no utiliza el campo AP de activación de la variable monitoreada de entradas del piloto y no establece la variable controlada de estado de AP proporcionada al PFD. Tampoco establece las variables controladas de comandos del actuador proporcionadas a los actuadores de la superficie de control.

Práctica recomendada 2.10.4: Identificar las variables monitoreadas y controladas para el subsistema que se comparten con el sistema principal. Asegurarse de que su definición en la especificación del subsistema sea coherente con su definición en la especificación del sistema. No incluir variables monitoreadas y controladas de la especificación del sistema que no sean utilizadas por el subsistema.

2.10.5 Crear Nuevas Variables Monitoreadas Y Controladas.

La mayor diferencia entre el límite del FCS y el límite del subsistema FGS es que el FCS interactúa directamente con los actuadores de superficie de control, mientras que el FGS interactúa directamente con el subsistema AP. Como resultado, el FGS tiene una nueva variable controlada (Guía AP) que el FGS proporciona al subsistema AP y una nueva variable monitoreada (Solicitud de Guía AP) que el subsistema AP proporciona al subsistema FGS. Estas corresponden a las variables internas que se muestran en el diagrama de dependencia para el FCS (figura B-2 en el apéndice B). La asignación del sistema FCS en dos subsistemas separados expone estas variables internas, que ahora se tratan como variables monitoreadas y controladas de los subsistemas FGS y AP.

Las variables monitoreadas y controladas para el subsistema FGS se definen junto con la definición de sus entidades externas en el apéndice C.3.

Práctica recomendada 2.10.5: Crear nuevas variables monitoreadas y controladas para las variables internas del sistema que se exponen mediante la asignación de funciones del sistema al subsistema. Asegúrese de que sean coherentes con las definiciones de las variables internas en la especificación del sistema.

2.10.6 Especificar Los Conceptos Operativos Del Subsistema.

Como se describe en la sección 2.3, el siguiente paso es definir los conceptos operativos para el nuevo subsistema. Esto describiría cómo el subsistema interactuará con sus operadores y otros subsistemas. Al igual que con la descripción general del sistema, esto proporciona un contexto importante para el El subcontratista y el subcontratista pueden utilizar el subsistema para identificar qué funciones debe proporcionar, qué información se necesita para invocar esas funciones y qué información debe proporcionar el sistema a sus operadores y a otros sistemas. Una forma de hacerlo es desarrollar casos de uso para el subsistema. También se pueden utilizar otros enfoques definidos por el contratista y el subcontratista, cuyo alcance y nivel de rigor dependen de la familiaridad del subcontratista con el funcionamiento del subsistema.

En los apéndices C.2 y D.2 se proporcionan marcadores de posición para los conceptos operativos de los sistemas FGS y AP, pero no se proporcionan ejemplos, ya que previamente se desarrollaron varios ejemplos de casos de uso.

Práctica recomendada 2.10.6: Especificar los conceptos operativos del subsistema para proporcionar a los desarrolladores una comprensión de cómo se utilizará el sistema y cómo interactuará con su entorno.

2.10.7 Identificar Los Supuestos Ambientales Del Subsistema Compartidos Con El Sistema Principal.

El siguiente paso consiste en identificar los supuestos ambientales para el nuevo subsistema. Si el sistema y el subsistema comparten varias de las mismas interfaces con entidades externas, muchos de los supuestos ambientales se encontrarán en la especificación del sistema. Esto es particularmente cierto en lo que respecta a los tipos, rangos y unidades de las variables monitoreadas y controladas.

Práctica recomendada 2.10.7: Identificar los supuestos ambientales que el subsistema comparte con su sistema principal. Documentar estos supuestos ambientales en la especificación del subsistema y asegurarse de que sean coherentes con los del sistema principal.

2.10.8 Identificar Supuestos Ambientales De Las Nuevas Variables Monitoreadas Y Controladas.

Por supuesto, será necesario definir los tipos, rangos y unidades de las nuevas variables monitoreadas y controladas. Estos deben ser coherentes con las definiciones de las variables internas que se encuentran en la especificación original. También se debe hacer un esfuerzo para asegurar que se identifiquen los supuestos ambientales más complejos que relacionan las nuevas variables monitoreadas y controladas con las otras variables monitoreadas y controladas del subsistema. Es posible que estos supuestos no se hayan identificado en el sistema original, ya que las relaciones entre las variables internas y las variables monitoreadas y controladas suelen capturarse implícitamente como parte de los requisitos detallados de comportamiento y rendimiento.

Práctica recomendada 2.10.8: Identificar los supuestos ambientales asociados con las nuevas variables monitoreadas y controladas. Asegurarse de que sean coherentes con las variables internas definidas en el sistema original.

2.10.9 Completar La Especificación De Requisitos Del Subsistema.

En este punto, la versión inicial de la especificación de requisitos del subsistema está completa. Contiene requisitos de alto nivel que pueden rastrearse directamente hasta la función del sistema de la que se derivaron. También contiene una descripción general del sistema, variables monitoreadas y controladas, conceptos operativos y supuestos ambientales que son consistentes con el sistema original. En el apéndice C se proporciona un ejemplo de una especificación inicial de este tipo para el FCS. El ejemplo para el sistema AP se proporciona en el apéndice D.

Por supuesto, la especificación de los requisitos del subsistema aún no está completa. En este punto, el contratista y el subcontratista están listos para continuar el desarrollo de los requisitos del subsistema hasta que se especifiquen los requisitos detallados de comportamiento y rendimiento. Como se explicó en la sección 2.10.1, esta tarea la realizan tanto el contratista como el subcontratista, aunque la división del trabajo entre ellos probablemente se basará en la confianza del contratista en el subcontratista.

La primera tarea de este proceso es completar la descomposición funcional del subsistema, como se explica en la sección 2.5. Una vez identificadas las principales funciones del subsistema, se debe reanudar el PSSA para el subsistema, como se explica en la sección 2.6, para identificar los requisitos de seguridad del sistema derivados. La arquitectura funcional del subsistema se debe modificar para tener en cuenta estos requisitos de seguridad u otras restricciones de implementación. Se deben identificar los principales modos del subsistema. Como se explica en la sección 2.7, se deben desarrollar los requisitos detallados de comportamiento y rendimiento, como se explica en la sección 2.8, y si el subsistema se implementará en software, se deben desarrollar los requisitos de software, como se explica en la sección 2.9.

Práctica recomendada 2.10.9: Completar la especificación de requisitos del subsistema

• completar la descomposición funcional.

• modificar la arquitectura funcional para tener en cuenta los requisitos de seguridad o las limitaciones de implementación.

• identificar los modos del subsistema.

• desarrollar los requisitos detallados de comportamiento y rendimiento para el subsistema.

• desarrollar los requisitos de software si el subsistema se implementará en software

2.10.10 Asegúrese De Que Las Latencias Y Las Tolerancias Sean Consistentes.

El desarrollo de los requisitos detallados de comportamiento y rendimiento para cada subsistema incluirá la especificación de la latencia y la tolerancia para cada variable controlada en el subsistema. Estas también deben revisarse para garantizar que sean coherentes con la especificación del sistema. La suma de las latencias para las variables controladas en los subsistemas debe ser al menos menor que la latencia de extremo a extremo especificada para la variable controlada correspondiente en el sistema principal.[1] Por ejemplo, en la figura 16, la

1 Si la arquitectura física que implementa el sistema introduce latencias adicionales (por ejemplo, si hay un retraso en el bus) introducidas entre dos subsistemas), estas latencias deben incluirse en esta comprobación.

suma de las latencias especificadas para la variable controlada FGS AP Guidance y la variable controlada AP Actuator Commands debe ser menor que la latencia de extremo a extremo especificada para la variable controlada FCS Actuator Commands. De manera similar, la composición de las tolerancias especificadas para las variables controladas del subsistema debe verificarse para garantizar que sean coherentes con la tolerancia de extremo a extremo especificada para la variable controlada correspondiente en el sistema principal.

Práctica recomendada 2.10.10: Garantizar que las latencias y tolerancias especificadas para las variables controladas en los subsistemas sean coherentes con las latencias y tolerancias de extremo a extremo especificadas para el sistema general.

2.11 PROPORCIONAR JUSTIFICACIÓN.

Consulte la Tabla 24 para obtener una descripción de las prácticas recomendadas de nivel principal y subnivel para proporcionar justificación.

Cuadro 24. Proporcionar Justificación

Prácticas recomendadas para los niveles principal y secundario
2.11 Proporcionar Justificación: En cada práctica recomendada, se alienta a los desarrolladores de requisitos a que proporcionen comentarios adicionales o justificaciones sobre por qué existe el requisito o la suposición ambiental o por qué se especifica un valor o rango en particular. Esta práctica analiza dónde y cómo se deben proporcionar las justificaciones en una especificación de requisitos.
2.11.1 Proporcionar una justificación durante todas las fases de desarrollo de requisitos para explicar por qué un existe un requisito o por qué se especifican valores específicos.
2.11.2 No utilice la justificación para especificar lo que hará el sistema, es decir, como una fuente alternativa de requisitos. Si la justificación es esencial para el comportamiento requerido del sistema, Debe indicarse como requisito.
2.11.3 Proporcionar una justificación para cada requisito que no tenga una razón obvia de por qué existe.
2.11.4 Proporcionar una justificación para cada supuesto ambiental que no tenga una razón obvia por la cual se incluye el supuesto.
2.11.5 Proporcionar una justificación para cada valor o rango en un requisito o supuesto, explicando por qué se especificó ese valor o rango en particular.
2.11.6 Mantenga la justificación breve y pertinente a la afirmación que se está explicando. Resuma la relevancia de los documentos largos y cite el documento en la justificación.
2.11.7 Captar la justificación del autor original de la afirmación que se está explicando lo antes posible.

Los requisitos documentan lo que hará un sistema. Los documentos de diseño, cómo lo hará el sistema. La justificación documenta por qué existe un requisito o por qué está escrito de la forma en que está. La justificación debe incluirse siempre que haya algo en el requisito que pueda no ser obvio para el lector o que pueda ayudar al lector a comprender por qué existe el requisito.

2.11.1 Proporcionar Una Justificación Para Explicar Por Qué Existe Un Requisito.

La justificación debe incluirse en todas las fases de definición de los requisitos. La justificación puede consistir en texto libre, referencias a estudios comerciales, otros documentos, artículos o citas de libros. Los objetivos del sistema (sección 2.1.3) suelen ser una fuente útil de justificación.

Hooks y Farry sostienen que proporcionar una justificación es la forma más eficaz de reducir el coste y mejorar la calidad de los requisitos [22]. La justificación puede reducir la cantidad de tiempo necesario para comprender un requisito al proporcionar información de fondo sobre por qué existe el requisito. Dado que la mayoría de las especificaciones de requisitos se leen varias veces, invertir el esfuerzo para proporcionar la justificación debería ahorrar tiempo y dinero a largo plazo. La justificación también puede ayudar a los lectores a evitar hacer suposiciones incorrectas, cuyo coste aumenta rápidamente a medida que el desarrollo avanza hacia la implementación. La justificación también puede reducir el coste de mantenimiento al documentar por qué se seleccionaron ciertas opciones o valores. También puede hacer que el desarrollo de variaciones del mismo producto sea más barato al proporcionar información sobre el impacto de cambiar los requisitos y las suposiciones. Por último, la justificación puede reducir el coste de la formación de nuevos empleados al proporcionarles información de fondo sobre por qué el sistema se comporta de la forma en que lo hace.

Proporcionar una justificación también puede mejorar la calidad y reducir el costo de crear los requisitos. Encontrar la justificación de un requisito o una suposición incorrecta puede ser difícil. Obligar al especificador a pensar en por qué el requisito es necesario o por qué se está haciendo la suposición a menudo mejorará la calidad del requisito. Incluso puede conducir a la eliminación del requisito cuando el autor se da cuenta de que no es necesario o que en realidad es un detalle de implementación. Esto elimina restricciones adicionales para los desarrolladores y el costo de verificar un requisito innecesario. La justificación también hace que sea más fácil mantener los requisitos centrados en lo que hará el sistema al proporcionar un lugar para la información de fondo.

Práctica recomendada 2.11.1: Proporcionar justificación durante todas las fases del desarrollo de requisitos para explicar por qué existe un requisito o por qué se especifican valores específicos.

2.11.2 Evite Especificar Requisitos En La Justificación.

Al mismo tiempo, la justificación no debe documentar lo que el sistema debe hacer, es decir, no debe incluir requisitos ni utilizarse como excusa para redactar requisitos deficientes. La justificación no es vinculante desde el punto de vista contractual y no necesita implementarse. Si algo incluido en la justificación es esencial para el comportamiento requerido del sistema, debe especificarse como un requisito.

Práctica recomendada 2.11.2: No utilice la justificación para especificar lo que hará el sistema, es decir, como una fuente alternativa de requisitos. Si la justificación es esencial para el comportamiento requerido del sistema, debe indicarse como un requisito.

2.11.3 Proporcionar Justificación Cuando El Motivo De Un Requisito No Es Obvio.

A continuación, se presentan algunos ejemplos de uso de la justificación. Como se mencionó anteriormente, se debe proporcionar una justificación siempre que el motivo de la existencia de un requisito no sea obvio. Un buen ejemplo de esto se puede encontrar en el ejemplo del termostato Isolette.

- REQ-MHS-4 Si el Modo Regulador es NORMAL y la Temperatura Actual es mayor o igual a la Temperatura Deseada Inferior y menor o igual a la Temperatura Deseada Superior, no se deberá cambiar el valor del Control de Calor.

  Justificación: cuando la Isolette se está calentando hacia la Temperatura Deseada superior, la Fuente de Calor debe dejarse encendida hasta que se alcance dicha temperatura. De manera similar, si la Isolette se está enfriando hacia la Temperatura Deseada inferior, la Fuente de Calor debe dejarse apagada hasta que se alcance dicha temperatura.

No resulta obvio de inmediato por qué la Fuente de Calor debe permanecer inalterada en este rango. Sin embargo, la justificación hace que el razonamiento detrás de esto sea mucho más claro. Hooks y Farry recomiendan capturar la justificación de cada requisito [22] para mejorar la calidad de todos los requisitos. Como mínimo, se debe proporcionar una justificación para cada requisito para el cual no sea obvia la razón por la que existe el requisito.

Práctica recomendada 2.11.3: Proporcionar una justificación para cada requisito que no tenga una razón obvia de por qué existe.

2.11.4 Proporcionar Justificación Para Los Supuestos Ambientales.

De manera similar, también se debe justificar cada hipótesis ambiental de la que depende el sistema. Por ejemplo, una de las hipótesis ambientales del termostato Isolette establece lo siguiente:

- EA-IS-1: Cuando la Fuente de Calor está encendida y el Isolette está correctamente cerrado, la Temperatura Actual aumentará a una velocidad de no más de 1 °F por minuto.

  Justificación: si la Temperatura Actual puede aumentar a una velocidad de más de 1 °F por minuto, es posible que el termostato no pueda apagar la Fuente de Calor lo suficientemente rápido para mantener el Rango de Temperatura Deseado a menos que se reduzca la latencia permitida especificada para el control de calor.

No resulta inmediatamente obvio a partir de la suposición en sí por qué se incluyó. Resulta que el termostato enciende y apaga el control de calor de la Fuente de Calor, y la latencia permitida especificada en esa variable controlada se calcula en función de esta suposición. La justificación ayuda a aclarar esto.

Práctica recomendada 2.11.4: Proporcionar una justificación para cada supuesto ambiental que no tenga una razón obvia por la cual se incluye el supuesto.

2.11.5 Proporcionar Justificación Para Los Valores Y Rangos.

También se debe incluir la justificación cada vez que se ingresa un número o un rango en la especificación. Esto ayuda al desarrollador a comprender por qué un valor en particular es o no importante. Esto puede resultar invaluable durante el desarrollo y el mantenimiento. Por ejemplo, en el termostato Isolette, la justificación de una de las suposiciones ambientales establece lo siguiente:

- EA-OI-3 La Temperatura de Alarma Inferior siempre será ≥93 °F.

  Justificación: La exposición a temperaturas inferiores a 93 °F provocará hipotermia, que puede causar la muerte en pocos minutos en Bebés prematuros gravemente enfermos.

La lógica deja claro por qué no se debería cambiar este número a 90 °F, incluso si esto pudiera permitir el uso de una interfaz de operador más comúnmente disponible y menos costosa.

Práctica recomendada 2.11.5: Proporcionar una justificación para cada valor o rango en un requisito o supuesto explicando por qué se especificó ese valor o rango en particular.

2.11.6 Mantenga La Justificación Breve Y Pertinente.

Al mismo tiempo, la mejor justificación es breve y concisa. En lugar de copiar un estudio comercial que justifique un requisito en particular, resuma la relación del estudio comercial con el requisito y cite el estudio comercial en la justificación.

Práctica recomendada 2.11.6: Mantenga la justificación breve y pertinente a la afirmación que se está explicando. Resuma la relevancia de los documentos largos y cite el documento en la justificación.

2.11.7 Capturar La Justificación Lo Antes Posible.

La justificación debe recopilarse junto con el desarrollo del requisito o la suposición que explica. Esto garantiza que el autor capte la justificación mientras piensa en ella. También ayuda a garantizar que haya tiempo para que otros revisen y cuestionen la justificación. En el ejemplo anterior, que relaciona la latencia del control de calor con el aumento máximo de temperatura en la Isolette, es mucho más simple registrar la justificación cuando se calcula la latencia que intentar rediseñar el razonamiento más tarde.

Práctica recomendada 2.11.7: Capturar la justificación del autor original de la declaración que se está explicando lo antes posible.

3. RESUMEN.

La gestión de requisitos es una de las actividades más importantes en el desarrollo de sistemas digitales. En la referencia 46, Fred Brooks plantea el problema de forma sucinta:

> "La parte más difícil de construir un sistema de software es decidir exactamente qué construir. Ninguna otra parte del trabajo conceptual es tan difícil como establecer los requisitos técnicos detallados... Ninguna otra parte del trabajo perjudica tanto el sistema resultante si se hace mal. Ninguna otra parte es tan difícil de rectificar posteriormente."

Si bien a lo largo de los años se han desarrollado numerosas metodologías para REM, los resultados de una encuesta de la industria, descrita en la referencia 1, indican que las mejores de estas prácticas rara vez se utilizan, si es que se utilizan, y los desarrolladores de sistemas digitales tienen muchas preguntas sobre cómo recopilar, documentar y organizar los requisitos de manera eficaz.

Este manual intenta abordar esta situación reuniendo las mejores ideas de varios enfoques, organizándolas en un todo coherente e ilustrándolas con ejemplos concretos que dejan en claro sus beneficios. Describe 11 prácticas recomendadas de nivel principal que permiten a un desarrollador avanzar desde enfoques informales al comienzo de la definición de requisitos hasta métodos más rigurosos a medida que los requisitos se acercan a su finalización. Estas prácticas están dirigidas al dominio de los sistemas integrados en tiempo real y, específicamente, a la industria de la aviónica. Debido a la creciente importancia del software en estos sistemas, el énfasis en los conceptos facilita la transición de los requisitos del sistema a los del software.

Si bien las prácticas recomendadas se presentan aproximadamente en el orden en que se llevarían a cabo, no existe ningún requisito que obligue a respetar estrictamente este orden. Como sucede con la mayoría de los procesos, es de esperar que se produzcan iteraciones significativas entre las distintas actividades a medida que se perfeccionan los requisitos.

Cada práctica recomendada de nivel principal se explica en el Manual seleccionando un enfoque particular para su implementación e ilustrando ese enfoque con un ejemplo práctico. Si bien los ejemplos y las prácticas recomendadas de nivel inferior pueden parecer sugerir un estilo y formato particular, su propósito real es ilustrar y aclarar las prácticas de nivel principal y sus beneficios. Se espera que la mayoría de las organizaciones que deseen incorporar cualquiera de las prácticas recomendadas deseen modificarlas, tal vez significativamente, para usarlas en su entorno.

Para utilizar este Manual, una organización debe identificar qué prácticas de nivel principal no se están utilizando de manera efectiva dentro de su organización y luego determinar cómo y en qué orden desea incorporarlas. Si el enfoque detallado descrito en el Manual satisface sus necesidades (por ejemplo, casos de uso para definir los conceptos operativos), es posible que desee implementar la práctica como se describe en este Manual. Si se necesitan más detalles o una mayor adaptación, las referencias citadas para esa práctica son una excelente fuente de información adicional. Cabe señalar que varios de los enfoques citados ya están adaptados en este Manual para complementar las otras prácticas recomendadas.

Como se ilustra en este Manual, un buen conjunto de requisitos consiste en mucho más que una simple lista de enunciados de obligaciones y no es fácil de producir. Sin embargo, se ha demostrado que invertir tiempo y esfuerzo al comienzo de un proyecto para producir buenos requisitos reduce en última instancia los costos y mejora la calidad del producto final [22-24 y 47]. La incorporación de las prácticas de nivel principal recomendadas en este Manual ayudará a proporcionar la estructura necesaria para producir requisitos que sean completos, coherentes, claros, fáciles de mantener y bien organizados.

4. REFERENCIAS.

1. Lempia, D. y Miller, S., "Informe de hallazgos de gestión de ingeniería de requisitos", informe de la FAA DOT/FAA/AR-08/34, mayo de 2009.

2. Parnas, D. y Madey, J., "Documentación funcional para ingeniería de sistemas informáticos (versión 2)", Informe técnico CRL 237, Universidad McMaster, Hamilton, Ontario, septiembre de 1991.

3. Van Schouwen, A., "El modelo de requisitos A-7: reexamen para sistemas en tiempo real y una aplicación a sistemas de monitoreo", Informe técnico 90-276, Queens University, Hamilton, Ontario, 1990.

4. Faulk, S., Brackett, J., Ward, P. y Kirby, J., Jr., "El método CoRE para requisitos en tiempo real", *Programas informáticos IEEE*, Vol. 9, No. 5, septiembre de 1992, págs. 22-33.

5. Faulk, S., Finneran, L., Kirby, J. y Moini, A., "Consortium Requirements Engineering Guidebook", Informe técnico SPC-92060-CMS, Software Productivity Consortium, 2214 Rock Hill Road, Herndon, Virginia, diciembre de 1993.

6. Faulk, S., Finneran, L., Kirby, J., Shah, S. y Sutton, J., "Experiencia en la aplicación del método CoRE a los requisitos de software del Lockheed C-130J",*Actas de la Novena Conferencia Anual sobre Seguridad Informática*Gaithersburg, Maryland, junio de 1994, págs. 3-8.

7. Heitmeyer, C., Kirby, J. y Labaw, B., "Verificación automatizada de la consistencia de la especificación de requisitos",*Transacciones ACM sobre ingeniería de software y metodología*(TOSEM), Vol. 5, No. 3, julio de 1996, págs. 231-261.

8. Leveson, N., Heimdahl, M., Hildreth, H. y Reese, J., "Especificaciones de requisitos para sistemas de control de procesos",*Transacciones IEEE sobre ingeniería de software*, Vol. 20, No. 9, septiembre de 1994, págs. 684-707.

9. Leveson, N., Heimdahl, M., Hildreth, H. y Reese, J., "Cambio de especificación de requisitos del sistema de prevención de colisiones (CAS) TCAS II 6.00", Administración Federal de Aviación, Departamento de Transporte de los EE. UU., marzo de 1993.

10. Thompson, J., Heimdahl, M. y Miller, S., "Prototipado basado en especificaciones para sistemas integrados",*Actas del Séptimo Simposio ACM SIGSOFT sobre los Fundamentos de la ingeniería de software*, LNCS 1687, septiembre de 1999, págs. 163-179.

11. Leveson, N., Heimdahl, M. y Reese, J., “Diseño de lenguajes de especificación para sistemas de control de procesos: lecciones aprendidas y pasos hacia el futuro”,Actas del Séptimo Simposio ACM SIGSOFT sobre los Fundamentos de la Ingeniería de Software,LNCS 1687, septiembre de 1999, págs. 127-145.

12. Miller, S., Tribble, A., Whalen, M. y Heimdahl, M., “Proporcionar los bienes”, Revista internacional sobre herramientas de software para la transferencia de tecnología (STTT), febrero de 2006.

13. Howard, J. y Anderson, P., “El riesgo de seguridad de los requisitos incompletos”, Actas del 20ElConferencia Internacional sobre Seguridad del Sistema(ISSC 2002), Denver, Colorado, agosto de 2002.

14. Lee, G., Howard, J. y Anderson, P., “Especificación de requisitos críticos de seguridad utilizando SpecTRM”,Actas de la 2.ª reunión del Grupo de trabajo sobre seguridad de sistemas de software de EE. UU., febrero de 2002.

15. Leveson, N., Reese, J. y Heimdahl, M., “SpecTRM: un sistema CAD para automatización digital”,Actas del 17ElConferencia sobre sistemas de aviónica digital (DASC98), Seattle, Washington, noviembre de 1998.

16. Leveson, N., “Especificaciones de intención: un enfoque para construir especificaciones centradas en el ser humano”,Transacciones IEEE sobre ingeniería de software, Vol. 26, No. 1, enero de 2000, págs. 15-35.

17. Cockburn, A.,Cómo escribir casos de uso efectivos, Addison-Wesley, Boston, Massachusetts, 2001.

18. Alexander, I. y Zink, T., “Introducción a la ingeniería de sistemas con casos de uso”, Ingeniería informática y de control del IEEE, Diciembre de 2002.

19. Leffingwell, D. y Widrig, D.,Gestión de requisitos de software, Addison-Wesley, Reading, Massachusetts, 2003.

20. Booch, G., Rumbaugh, J. y Jacobson, I.,Guía del usuario del lenguaje de modelado unificado, Addison Wesley, Reading, Massachusetts, 1999.

21. Fowler, M.,UML destilado,Una breve guía del lenguaje de modelado de objetos estándar, Tercera edición, Addison Wesley, Reading, Massachusetts, septiembre de 2003.

22. Hooks, I. y Farry, K., “Productos centrados en el cliente: creación de productos exitosos mediante la gestión inteligente de requisitos”, AMACOM American Management Association, Nueva York, Nueva York, 2001.

23. Davis, A.,Requisitos de software (revisados): objeto, funciones y estados, Prentice-Hall, Englewood Cliffs, Nueva Jersey, 1993.

24. Leveson, N.,Safeware: Seguridad del sistema y computadoras, Addison-Wesley Publishing Company, Reading, Massachusetts, 1995.

25. “Guía IEEE para el desarrollo de especificaciones de requisitos del sistema”, IEEE Std 1233, Instituto de Ingenieros Eléctricos y Electrónicos, Nueva York, Nueva York, diciembre de 1998.

26. “IEEE Recommended Practice for Software Requirements Specification”, IEEE Std 830-1998, Instituto de Ingenieros Eléctricos y Electrónicos, Nueva York, Nueva York, Junio de 1998.

27. “Consideraciones de software en la certificación de sistemas y equipos aerotransportados”, DO-178B, RTCA, Washington, DC, 1 de diciembre de 1992.

28. “Informe final para la aclaración de la DO-178B, “Consideraciones de software en la certificación de sistemas y equipos aerotransportados”, DO-248B, RTCA, Washington, DC, 12 de octubre de 2001.

29. “Directrices para la comunicación, navegación, vigilancia y gestión del tránsito aéreo (CNS/ATM) Garantía de integridad del software de sistemas”, DO-278, 5 de marzo de 2002.

30. “Consideraciones de certificación para sistemas de aeronaves altamente integrados o complejos”, ARP 4754, SAE International, noviembre de 1996.

31. “Directrices y métodos para llevar a cabo el proceso de evaluación de seguridad en sistemas y equipos aerotransportados civiles”, ARP 4761, SAE International, diciembre de 1996.

32. Hughes, D. y Dornheim, M., “Informe especial sobre cabinas automatizadas, partes I y II”, Semana de la aviación y la tecnología espacial, 30 de enero al 6 de febrero de 1995.

33. Billings, C.,Automatización de la aviación: la búsqueda de un enfoque centrado en el ser humano, Lawrence Erlbaum Associates, Inc., Mahwah, Nueva Jersey, 1997.

34. Equipo de Seguridad de la Aviación Comercial, “Informe final sobre el JSAT de pérdida de control: resultados y análisis”, P. Russell y J. Pardee, copresidentes, 15 de diciembre de 2000.

35. Leveson, N., et al., “Análisis de especificaciones de software para potencial de confusión de modos”, Actas de un taller sobre error humano y desarrollo de sistemasGlasgow, Escocia, marzo de 1997, págs. 132-146.

36. Leveson N., “Diseño de automatización para reducir los errores del operador”,Actas de la Conferencia IEEE sobre sistemas, hombre y cibernética, Octubre de 1997.

37. Sarter, N., Woods, D. y Billings, C., “Sorpresas de autómatas”,Manual de factores humanos/ ergonomía, 2Dakota del Norteed., G. Salvendy, ed., John Wiley & Sons, Nueva York, Nueva York, 1997.

38. Woodson, W., Tilman, P. y Tilman, B.,Manual de diseño de factores humanos, segunda edición, Compañías McGraw-Hill, 1992.

39. Boff, K. y Lincoln, J.,Compendio de datos de ingeniería: percepción y desempeño humanos, Base de la Fuerza Aérea Wright-Patterson, Laboratorio de Investigación Médica Aeroespacial Harry G. Armstrong, Ohio, 1998.

40. Coplien, J., Hoffman, D. y Weiss, D., "Común y variabilidad en la ingeniería de software", Programas informáticos IEEE, Vol. 20, No. 6, noviembre de 1998.

41. De Marco, T., "Análisis estructurado y especificación del sistema", Yourdon Press, Nueva York, 1979.

42. Harel, D., "Diagramas de estados: un formalismo visual para sistemas complejos",Ciencia de la programación informática, Vol. 8, 1987, págs. 231.

43. Hoffman, D. y Weiss, D., "Fundamentos de software: documentos recopilados por David L. Parnas", Addison-Wesley Professional, Boston, Massachusetts, 2001.

44. Miller, S. y Tribble, A., "Extensión del modelo de cuatro variables para cerrar la brecha entre el sistema y el software", 20ElConferencia sobre sistemas de aviónica digital (DASC01), Daytona Beach, Florida, del 14 al 18 de octubre de 2001.

45. Stevens, R., Jackson, B. y Arnold, S., "Ingeniería de sistemas: cómo afrontar la complejidad" Prentice-Hall, Londres, 1998.

46. Brooks, F., "No hay bala de plata: esencia y accidentes de la ingeniería de software",Computadora IEEE, Abril de 1987.

47. Boehm, B., "Economía de la ingeniería de software", Prentice-Hall, Englewood Cliffs, Nueva Jersey, 1981.

APÉNDICE A — EJEMPLO DE TERMOSTATO ISOLETTE

Este apéndice contiene un ejemplo de especificación de requisitos para el Termostato Isolette analizado en la sección 3. El formato presentado es un ejemplo de cómo se podrían implementar las mejores prácticas de la sección 2. Hay muchos otros formatos que serían igualmente efectivos.

A.1. DESCRIPCIÓN GENERAL DEL SISTEMA.

El sistema que se especifica es el Termostato de una Isolette.[1] Una Incubadora para Bebés es una Incubadora que proporciona temperatura, humedad y oxígeno controlados (si es necesario). Las Incubadoras se utilizan ampliamente en las unidades de cuidados intensivos neonatales para el cuidado de Bebés prematuros.

El propósito del Termostato de la Isolette es mantener la temperatura del aire de la Isolette dentro de un Rango Deseado. Detecta la Temperatura Actual de la Isolette y enciende y apaga la Fuente de Calor para calentar el aire según sea necesario. Si la temperatura cae demasiado por debajo o sube demasiado por encima del Rango de Temperatura Deseado, activa una alarma para alertar a la Enfermera. El sistema permite a la Enfermera establecer el Rango de Temperatura Deseado y establecer el Rango de Temperatura de Alarma fuera del Rango de Temperatura Deseado para el cual se debe activar la alarma.

A.1.1 CONTEXTO DEL SISTEMA.

El contexto operacional del Termostato Isolette se muestra en la figura A-1.

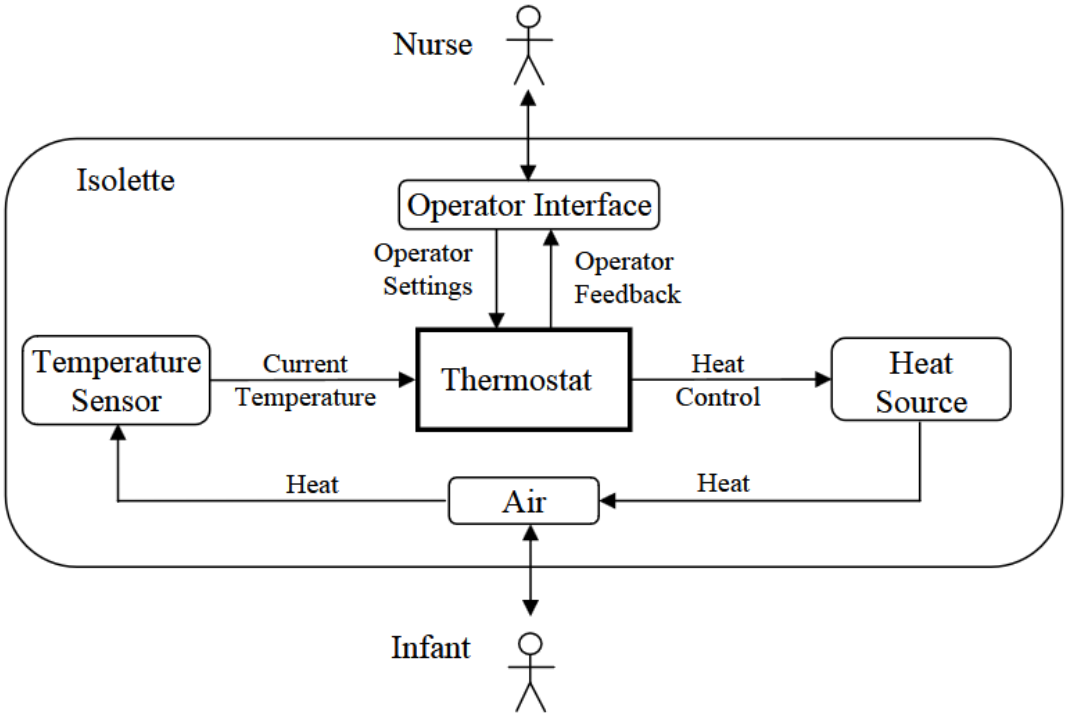


Figura A-1. Diagrama de Contexto del Termostato Isolette

1 Para simplificar este ejemplo, la Interfaz del Operador se trata como una entidad externa fuera del Termostato.

El Termostato interactúa directamente con tres entidades que forman parte de la Isolette:

- El Sensor de Temperatura proporciona la Temperatura Actual del aire en la Isolette al Termostato.

- La Fuente de Calor calienta el aire en la Isolette. Se enciende y se apaga mediante el Control de Calor.

- La Interfaz del Operador proporciona las Configuraciones del Operador para el Termostato y recibe Retroalimentación al Operador desde el Termostato.

El Termostato también interactúa indirectamente con otras entidades fuera de la Isolette:

- La Enfermera que utiliza la Interfaz del Operador para ingresar a la Configuración del Operador y ver la Retroalimentación al Operador.

- El Aire en la Isolette.

- El Bebé que se coloca en la Isolette y se calienta con el Aire.

A.1.2 OBJETIVOS DEL SISTEMA.

Los objetivos de alto nivel (G) del sistema son:

- G1—El Bebé debe mantenerse a una temperatura segura y confortable.
- G2—Se debe advertir a la Enfermera si el Bebé tiene demasiado calor o demasiado frío.
- G3—El costo de fabricación del Termostato debe ser lo más bajo posible.

A.2. CONCEPTOS DE OPERACIÓN/ALES.

Los siguientes casos de uso y excepción describen cómo interactúan los operadores con el Isolette y el termostato. En la tabla A-1 se ofrece un resumen de los casos de uso y excepción.

Tabla A-1. Resumen de Casos de Uso y Excepciones del Termostato Isolette

ID	Actores Primarios	Título y Descripción
A.2.1	Enfermera	Funcionamiento Normal de Isolette Describe el funcionamiento normal de la Isolette por parte de la Enfermera.
A.2.2	Enfermera	Configurar la Isolette Describe cómo la Enfermera configura la Incubadora y el Termostato para el Bebé.
A.2.3	Termostato	Mantener la Temperatura Deseada Describe cómo el Termostato enciende y apaga la Fuente de Calor para mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado.

Tabla A-1. Resumen de Casos de Uso y Excepciones del Termostato Isolette (cont.)

ID	Actores Primarios	Título y Descripción
A.2.4	Termostato	Falla al mantener una Temperatura Segura Describe cómo responden el Termostato y la Enfermera cuando la Incubadora no puede mantener la Temperatura Actual dentro del Rango de Temperatura de Alarma.
A.2.5	Termostato	Respuesta ante fallas del Termostato Describe cómo responden el Termostato y la Enfermera cuando el Termostato detecta una falla interna.
A.2.6	Enfermera	Falla al mantener la Temperatura Deseada Describe cómo la Enfermera se ocupa de la Incubadora cuando no puede mantener la Temperatura Actual dentro del Rango de Temperatura Deseado, pero puede mantener la Temperatura Actual dentro del Rango de Temperatura de Alarma.

Los actores y sus principales objetivos se muestran en la tabla A-2.

Tabla A-2. Principales Actores y Objetivos del Termostato Isolette

Actor	Objetivos primarios del actor
Enfermera	Proporcionar al Bebé los cuidados de enfermería adecuados, incluyendo mantenerlo abrigado.
Infante	Estar cómodo y saludable
Isolette	Contener al Bebé y mantener la Temperatura Actual dentro del Rango de Temperatura Deseado
Termostato	Mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado

A.2.1 CASO DE USO: FUNCIONAMIENTO NORMAL DE LA ISOLETTE.

Este caso de uso describe el funcionamiento normal de la Isolette por parte de la Enfermera.

- Objetivos del Sistema relacionados: G1 y G2
- Actor Principal: Enfermera
- Condición previa:
    - El Bebé está listo para ser colocado en la Incubadora.
    - La Isolette y el Termostato están apagados

- Postcondición:

    - El Bebé es retirado de la Incubadora
    - La Isolette y elTermostato están apagados

- Escenario principal de éxito:

    1. La Enfermera enciende la Isolette

    2. Isolette enciende el Termostato

    3. El Termostato se inicializa y entra en su modo de funcionamiento normal (caso de excepción 1) (A.2.5, A.5.1.2 y A.5.2.2)

    4. La Enfermera configura la Isolette según las necesidades del Bebé (A.2.2)

    5. La Enfermera espera hasta que la Temperatura Actual esté dentro del Rango de Temperatura Deseado (A.2.6 y A.5.1.1)

    6. La Enfermera coloca al Bebé en la Incubadora

    7. Isolette mantiene la Temperatura Deseada (A.2.3)

    8. La Enfermera confirma que la Temperatura Actual está dentro del Rango de Temperatura Deseado durante las rondas (A.2.6 y A.5.1.1)

    9. La Enfermera retira al Bebé

    10. La Enfermera apaga la Isolette

    11. Isolette apaga el Termostato

- Caso de excepción 1:

    1. La Alarma se activa porque la Temperatura Actual está fuera del Rango de Temperatura de Alarma (A.5.2.3)

    2. La Enfermera ignora la alarma[1]

    3. Continúe con el Escenario de Éxito Principal, paso 4.

1 Para simplificar, no se especifica la función para apagar la Alarma. Como ejercicio, el lector podría considerar qué cambios serían necesarios para agregar esta capacidad al ejemplo.

A.2.2 CASO DE USO: CONFIGURAR LA ISOLETTE.

Este caso de uso describe cómo la Enfermera configura la Incubadora y el Termostato para el Bebé.

- Objetivos del sistema relacionados: G1 y G2
- Actor principal: Enfermera
- Condición previa: la Isolette y el Termostato están encendidos
- Postcondición:
  - El Rango de Temperatura Deseado se establece según las necesidades del Bebé.
  - El Rango de Temperatura de Alarma se configura según las necesidades del Bebé.
  - La Temperatura Actual en la Isolette está dentro del Rango de Temperatura Deseado
- Escenario principal de éxito:
  1. La Enfermera establece el Rango de Temperatura de Alarma para el Bebé (A.5.2.1)
  2. La Enfermera establece el Rango de Temperatura Deseado para el Bebé (A.5.1.1)
  3. El Termostato mantiene el Rango de Temperatura Deseado (A.2.3)

A.2.3 CASO DE USO: MANTENER LA TEMPERATURA DESEADA.

Este caso de uso describe cómo el Termostato enciende y apaga la Fuente de Calor para mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado.

- Objetivos del sistema relacionados: G1
- Actor principal: Termostato
- Condición previa: la Isolette y el Termostato están encendidos
- Postcondición:
  - La Isolette y el Termostato están encendidos
  - La Temperatura Actual está dentro del Rango de Temperatura Deseado
- Escenario principal de éxito:
  1. La Temperatura Actual cae por debajo del Rango de Temperatura Deseado
  2. El Termostato enciende la Fuente de Calor para calentar la Isolette (A.5.1.3)
  3. La Temperatura Actual aumenta por encima del Rango de Temperatura Deseado
  4. El Termostato apaga la Fuente de Calor para enfriar la Isolette (A.5.1.3)
  5. Repita los pasos 1 a 4

A.2.4 CASO DE EXCEPCIÓN: INCUMPLIMIENTO DE MANTENER UNA TEMPERATURA SEGURA.

Este caso de excepción describe cómo responden el Termostato y la Enfermera cuando la Incubadora no puede mantener la Temperatura Actual dentro del Rango de Temperatura de Alarma.

- Objetivos del sistema relacionados: G2

- Actor principal: Termostato

- Condición previa:

  - La Isolette y el Termostato están encendidos
  - La Temperatura Actual está dentro del Rango de Temperatura de Alarma
  - La Alarma está apagada

- Postcondición:

  - La Isolette y el Termostato están encendidos
  - La Temperatura Actual está dentro del Rango de Temperatura Deseado
  - La Alarma está apagada

- Escenario principal de éxito:

  1. La Temperatura Actual cae por debajo o aumenta por encima del Rango de Temperatura de Alarma.
  2. El Termostato activa la alarma (A.5.2.3)
  3. La Enfermera responde a la alarma y ve que la Temperatura de Visualización está dentro del Rango de Temperatura de Alarma (A.5.1.1)
  4. La Enfermera retira al Bebé de la Incubadora
  5. La Enfermera corrige el problema, por ejemplo, cerrando una puerta abierta (curso alternativo 1)
  6. La Enfermera espera hasta que la Temperatura de Visualización esté dentro del Rango de Temperatura Deseado (A.2.6 y A.5.1.1)
  7. La Enfermera vuelve a colocar al Bebé en la Incubadora.

- Curso alternativo 1:

  1. La Enfermera no puede corregir el problema.
  2. La Enfermera adquiere otra Isolette.
  3. La Enfermera inicia el funcionamiento normal de la nueva Isolette (A.2.1)

A.2.5 CASO DE EXCEPCIÓN: RESPONDER ANTE FALLA DEL TERMOSTATO.

Este caso de excepción describe cómo responden el Termostato y la Enfermera cuando el Termostato detecta una falla interna.

- Objetivos del Sistema relacionados: G2

- Actor principal: Termostato

- Condición previa:

  - La Isolette y el Termostato están encendidos
  - El estado del Termostato está activado
  - La Alarma está apagada

- Postcondición:

  - La Isolette y el Termostato están encendidos
  - La Temperatura Actual está dentro del Rango de Temperatura Deseado
  - La alarma está apagada

- Escenario principal de éxito:

  1. El Termostato detecta una falla interna (A.5.1.4 y A.5.2.4)
  2. El Termostato ingresa al modo FALLA (A.5.1.2 y A.5.2.2)
  3. El Termostato establece su Estado Mostrado en fallado (A.5.1.1 y A.5.2.1)
  4. El Termostato activa la Alarma
  5. La Enfermera responde a la Alarma y ve que el Termostato está averiado.
  6. La Enfermera retira al Bebé de la Incubadora.
  7. La Enfermera obtiene otra Isolette
  8. La Enfermera inicia el funcionamiento normal de la nueva Isolette (A.2.1)

A.2.6 CASO DE EXCEPCIÓN: FALLA AL MANTENER LA TEMPERATURA DESEADA.

Este caso de excepción describe cómo la Enfermera maneja una Incubadora que no puede mantener la Temperatura Actual dentro del Rango de Temperatura Deseado, pero puede mantener la Temperatura Actual dentro del Rango de Temperatura de Alarma.

- Objetivos del sistema relacionados: G1

- Actor principal: Enfermera

- Condición previa:

  - La Isolette y el Termostato están encendidos
  - La Temperatura Actual no está dentro del Rango de Temperatura Deseado
  - La Temperatura Actual está dentro del Rango de Temperatura de Alarma

- Postcondición:

  - La Isolette y el Termostato están encendidos
  - La Temperatura Actual está dentro del Rango de Temperatura Deseado

- Escenario principal de éxito:

    1. La Enfermera intenta corregir el problema, por ejemplo, cerrando una puerta abierta.
    2. La Enfermera espera hasta que la Temperatura Actual de la Incubadora esté dentro del Rango de Temperatura Deseado (curso alternativo 1) (A.5.1.1)
    3. Retornar al caso de uso que llamó a este caso de excepción.

- Curso alternativo 1:

    1. La Temperatura de Visualización no ingresa al Rango de Temperatura Deseado (A.5.1.1)
    2. La Enfermera retira al Bebé de la Incubadora
    3. La Enfermera obtiene otra Isolette
    4. La Enfermera inicia el funcionamiento normal de la nueva Isolette (A.2.1)
    5. Retornar al caso de uso que llamó a este caso de excepción.

A.3. ENTIDADES EXTERNAS.

Las siguientes secciones describen las entidades externas con las que interactúa directamente el Termostato: el Sensor de Temperatura, la Interfaz del Operador y la Fuente de Calor. Se enumeran las variables monitoreadas y controladas asociadas con cada entidad, junto con las suposiciones ambientales realizadas sobre la entidad. También se define una entidad externa Isolette para especificar suposiciones ambientales que abarcan más de una entidad externa.

A.3.1 ISOLETTE.

Una Isolette es una incubadora para bebés que proporciona temperatura, humedad y oxígeno controlados (si es necesario). Incluye el Termostato, el Sensor de Temperatura, la Interfaz del Operador y la Fuente de Calor. El Termostato hace las siguientes suposiciones ambientales sobre la Incubadora.

- EA-IS-1: Cuando la Fuente de Calor está encendida y el Isolette está correctamente cerrado, la Temperatura Actual aumentará a una velocidad de no más de 1 °F por minuto.

    Justificación: si la Temperatura Actual puede aumentar a una velocidad de más de 1 °F por minuto, es posible que el Termostato no pueda apagar la Fuente de Calor lo suficientemente rápido para mantener el Rango de Temperatura Deseado a menos que se reduzca la latencia permitida especificada para el Control de Calor.

- EA-IS-2: Cuando la Fuente de Calor está apagada y el Isolette está correctamente cerrado, la Temperatura Actual disminuirá a una velocidad de no más de 1 °F por minuto.

    Justificación: Si la Temperatura Actual puede disminuir a una velocidad de más de 1 °F por minuto, es posible que el Termostato no pueda encender la Fuente de Calor lo suficientemente rápido para mantener el Rango de Temperatura Deseado a menos que se reduzca la latencia permitida especificada para el Control de Calor.

A.3.2 SENSOR DE TEMPERATURA.

El Sensor de Temperatura proporciona la Temperatura Actual del aire en el Isolette al Termostato. Las variables monitoreadas se muestran en la tabla A-3.

Tabla A-3. Variables monitoreadas por el Termostato para el Sensor de Temperatura

Nombre	Tipo	Rango	Unidades	Interpretación física
Temperatura Actual	Real	[68.0..105.0]	°F	Temperatura Actual del aire en el interior de Isolette
	Estado	•Inválido, Válido		

• denota valor inicial

Se hacen los siguientes supuestos ambientales:

- EA-TS-1: La Temperatura Actual se proporcionará al Termostato en grados Fahrenheit

  Justificación: coherencia con los supuestos ambientales Interfaz del Operador EA-OI-1

- EA-TS-2: La Temperatura Actual se detectará con una precisión de ±0,1 °F.

  Justificación: Es necesaria una precisión de 0,1 °F para garantizar que el Termostato pueda encender y apagar la Fuente de Calor con la suficiente rapidez para mantener el Rango de Temperatura Deseado.

- EA-TS-3: La Temperatura Actual cubrirá el rango de al menos 68,0 °F a 105,0 °F.

  Justificación: Este es el rango de funcionamiento especificado del Isolette. El extremo inferior de este rango es útil para monitorear un Isolette que se está calentando hasta el Rango de Temperatura Deseado. El extremo superior es mayor que la Temperatura de Alarma Superior para garantizar que la Temperatura Actual se detecte en todo el Rango de Temperatura de Alarma.

A.3.3 FUENTE DE CALOR.

La Fuente de Calor calienta el aire en la Isolette. Se activa y desactiva modificando el valor de la variable controlada de Control de Calor. Las variables controladas se muestran en la tabla A-4. No se realizan suposiciones ambientales.

Tabla A-4. Variables controladas por Termostato para Fuente de Calor

Nombre	Tipo	Rango	Unidades	Interpretación física
Control de Calor	Enumerado	Apagado, encendido		Comando para encender y apagar la Fuente de Calor

A.3.4 INTERFAZ DEL OPERADOR.

La Interfaz del Operador proporciona los ajustes del operador para el Termostato y recibe Retroalimentación al Operador desde el Termostato. Las suposiciones ambientales asociadas con la Interfaz del Operador son bastante estrictas, lo que simplifica la Función de Gestión de la Interfaz del Operador. Si la entidad externa de la Interfaz del Operador no cumpliera con estas suposiciones, sería necesario reforzar la Función de Gestión de la Interfaz del Operador para garantizar la coherencia de las entradas al Termostato. Las variables monitoreadas y controladas se muestran en las tablas A-5 y A-6, respectivamente.

Tabla A-5. Variables monitoreadas por el Termostato para la Interfaz del Operador

Nombre	Tipo	Rango	Udad	Interpretación física
Configuración del Operador				Ajustes del Termostato proporcionados por el operador
Rango de Temperatura Deseado				Rango deseado de temperatura de Isolette
Temperatura Deseada Inferior	Entero	[97..99]	°F	Valor inferior del Rango de Temperatura Deseado
	Estado	●Inválido, Válido		
Temperatura Deseada Superior	Entero	[98..100]	°F	Valor superior del Rango de Temperatura Deseado
	Estado	●Inválido, Válido		
Rango de Temperatura de Alarma				Activar Alarma cuando esté fuera de este rango
Temperatura de Alarma Inferior	Entero	[93..98]	°F	Valor inferior del Rango de Temperatura de Alarma
	Estado	●Inválido, Válido		
Temperatura de Alarma Superior	Entero	[99..103]	°F	Valor superior del Rango de Temperatura de Alarma
	Estado	●Inválido, Válido		

● denota valor inicial

Tabla A-6. Variables controladas por Termostato para la Interfaz del Operador

Nombre	Tipo	Rango	Unidades	Interpretación física
Retroalimentación al Operador				Información proporcionada al operador
Estado del Regulador	Enumerado	Inicializando, Encendido, Falla		Estado de la Función Regulador del Termostato
Estado del Monitor	Enumerado	Inicializando, Encendido, Falla		Estado de la Función Monitorización del Termostato
Temperatura de Visualización	Entero	[68..105]	°F	Temperatura de Visualización de Isolette
Alarma	Enumerado	Apagado, encendido		Comando para activar o desactivar la alarma

Se hacen los siguientes supuestos ambientales:

- EA-OI-1: Todas las temperaturas se ingresarán y se mostrarán en grados Fahrenheit.

  Justificación: Minimizar la complejidad de este ejemplo. Un sistema real probablemente admitiría Celsius o tal vez tanto Fahrenheit como Celsius

- EA4-OI-2: Los operadores establecerán y mostrarán todas las temperaturas en incrementos de 1 °F.

  Justificación: Los estudios de marketing han demostrado que los clientes prefieren establecer las temperaturas en incrementos de 1 grado. Una resolución de 1 °F es suficiente para ser coherente con los requisitos funcionales y de rendimiento especificados en el resto del documento.

- EA-OI-3: La Temperatura de Alarma Inferior siempre será ≥93 °F.

  Justificación: La exposición a temperaturas inferiores a 93 °F provocará hipotermia, que puede causar la muerte en pocos minutos en Bebés prematuros gravemente enfermos.

- EA-OI-4: La Temperatura de Alarma Inferior siempre será menor o igual a la Temperatura Deseada Inferior menos 1 °F.

  Justificación: si la Temperatura de Alarma Inferior es mayor o igual a la Temperatura Deseada Inferior, la alarma podría activarse mientras la Temperatura Actual todavía esté dentro del Rango de Temperatura Deseado.

- EA-OI-5: La Temperatura Deseada Inferior siempre será ≥97 °F.

  Justificación: Exponer al Bebé a temperaturas inferiores a 97 °F puede provocar una pérdida excesiva de calor y una caída de la frecuencia cardíaca secundaria a acidosis metabólica.

- EA-OI-6: La Temperatura Deseada Inferior siempre será menor o igual a la Temperatura Deseada Superior menos 1 °F.

  Justificación: si la Temperatura Deseada Inferior es mayor o igual que la Temperatura Deseada Superior, no queda claro si la Fuente de Calor debe estar encendida o apagada. Esto puede provocar un funcionamiento cíclico excesivo de la Fuente de Calor.

- EA-OI-7: La Temperatura Deseada Superior siempre será ≤100°F.

  Justificación: Exponer al Bebé a temperaturas superiores a 100 °F puede dar lugar a un diagnóstico incorrecto de fiebre que dé lugar a una evaluación agresiva (cultivo de sangre y punción lumbar) y al tratamiento de la infección.

- EA-OI-8: La Temperatura de Alarma Superior siempre será mayor o igual a la Temperatura Deseada Superior de 1 °F.

  Justificación: si la Temperatura de Alarma Superior es menor o igual a la Temperatura Deseada Superior, la alarma podría activarse mientras la Temperatura Actual todavía esté dentro del Rango de Temperatura Deseado.

- EA-OI-9: La Temperatura de Alarma Superior siempre será ≤103 °F.

Justificación: La exposición a temperaturas superiores a 103 °F provocará hipertermia, que puede provocar arritmias cardíacas y convulsiones febriles en pocos minutos.

- EA-OI-9: La Temperatura de Visualización cubrirá el rango de al menos 68,0° a 105,0°F.

  Justificación: Este es el rango de funcionamiento especificado del Isolette. El extremo inferior de este rango es útil para monitorear un Isolette que se está calentando hasta el Rango de Temperatura Deseado. El extremo superior está configurado para ser mayor que el valor máximo de la Temperatura de Alarma Superior.

A.4. REQUISITOS DE SEGURIDAD.

Los siguientes peligros relevantes se identificaron mediante el proceso de evaluación de seguridad:

- H1: Exposición prolongada del Bebé a calor o frío peligrosos
  Clasificación: catastrófica
  Probabilidad: $<10^{-9}$ por hora de operación

Para garantizar que la probabilidad de peligro H1 sea 10^{-9} por cada hora de operación, se imponen al Termostato Isolette los siguientes requisitos de seguridad derivados:

- SR-1: El Isolette deberá incluir una función reguladora independiente que mantenga la Temperatura Actual adentro de la Isolette dentro del Rango de Temperatura Deseado.

  Justificación: La Enfermera establecerá el Rango de Temperatura Deseado en el rango ideal según el peso y la salud del Bebé. El regulador debe mantener la Temperatura Actual dentro de este rango en condiciones normales de funcionamiento.

  Probabilidad de fallo permitida: $<10^{-5}$ por hora

- SR-2: El Isolette deberá incluir una función de monitor independiente que active una alarma en un máximo de 5 segundos siempre que

  - La Temperatura Actual cae por debajo o aumenta por encima del Rango de Temperatura de Alarma.
  - La Temperatura Actual o el Rango de Temperatura de Alarma están marcados como no válidos.
  - Se ha detectado un fallo interno en la función de monitorización.

  Justificación: La Enfermera establecerá el Rango de Temperatura de Alarma en función del peso y la salud del Bebé. El Bebé debe ser retirado de la Incubadora dentro de los 15 segundos posteriores a que la Temperatura Actual caiga por debajo o suba por encima de este rango. Con el control normal proporcionado por la Enfermera, esto se puede lograr en 10 segundos, lo que deja 5 segundos para que el sistema active la alarma. Es deseable activar la alarma en menos tiempo.

  Si la Temperatura Actual o el Rango de Temperatura de Alarma proporcionados a la función de monitorización se marcan como no válidos o si se detecta una falla interna en la función de monitorización, no se debe confiar en que la función de monitorización funcione correctamente.

Probabilidad de fallo permitida: $<10^{-5}$ por hora.

A.5. FUNCIONAMIENTO DEL SISTEMA DE TERMOSTATO.

El Termostato realiza dos funciones lógicamente independientes. La primera regula la Temperatura Actual en la Isolette para que se mantenga dentro del Rango de Temperatura Deseado. La segunda monitorea la Temperatura Actual en la Isolette y activa una alarma si cae por debajo o sube por encima del Rango de Temperatura de Alarma.

Los requisitos de alto nivel para la Función Termostato son los siguientes:

- REQ-TH-1: El Termostato deberá fijar el valor del Control de Calor.

  Justificación: Una función principal del Termostato es encender y apagar el Control de Calor para mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado, lo cual es requerido por SR-1.

- REQ-TH-2: La Función Termostato deberá establecer el valor del Estado del Regulador.

  Justificación: SR-1 requiere que el Termostato proporcione una función de regulación independiente. El Termostato proporciona el estado de esta función a la Interfaz del Operador. La Interfaz del Operador utilizará el Estado del Regulador y el Estado del Monitor para informar el estado general del Termostato, que es requerido por SR-1.

- REQ-TH-3: El Termostato deberá fijar el valor de la Temperatura de Visualización.

  Justificación: La Temperatura Actual se muestra en la Interfaz del Operador para proporcionar a los operadores un medio adicional para confirmar que el Isolette mantiene la temperatura correctamente. El Termostato proporciona este valor a la Interfaz del Operador como Temperatura de Visualización.

- REQ-TH-4: El Termostato deberá fijar el valor del Control de Alarma.

  Justificación: Una función principal del Termostato es activar la Alarma si el Isolette no puede mantener la Temperatura Actual dentro del Rango de Temperatura de Alarma, lo cual es requerido por SR-2.

- REQ-TH-5: El Termostato deberá establecer el valor del Estado del Monitor.

  Justificación: SR-2 requiere que el Termostato proporcione una función de monitor independiente. El estado de esta función debe proporcionarse a la Interfaz del Operador, que la utilizará junto con el estado de la función del regulador para informar el estado general del Termostato.

La función del Termostato se divide en subfunciones como se muestra en la figura A-2.

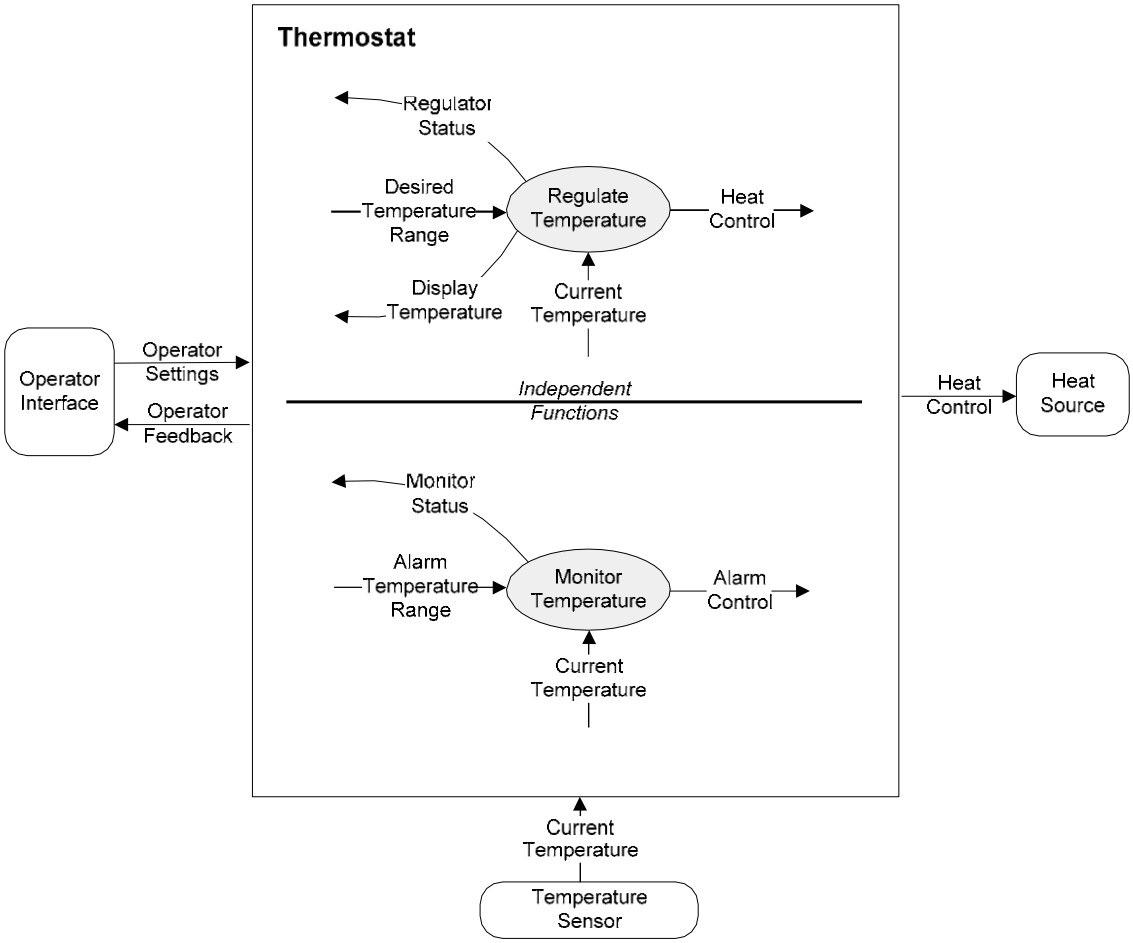


Figura A-2. Diagrama de Dependencia del Termostato

A.5.1 FUNCIÓN REGULACIÓN DE TEMPERATURA.

La Función Regulación de Temperatura compara la Temperatura Actual del Sensor de Temperatura con el Rango de Temperatura Deseado proporcionado por la Interfaz del Operador y enciende o apaga la Fuente de Calor para mantener la Temperatura Actual dentro del Rango de Temperatura Deseado. También proporciona la Temperatura de Visualización y el Estado del Regulador a la Interfaz del Operador.

Los requisitos de alto nivel para la Función Regulación de Temperatura son los siguientes:

- REQ-RT-1: La Función Regulación de Temperatura deberá establecer el valor del Control de Calor.

  Justificación: La función principal de la Función Regulación de Temperatura es encender y apagar el Control de Calor para mantener la Temperatura Actual en la Isolette dentro del Rango de Temperatura Deseado, como lo requiere SR-1.

- REQ-RT-2: La Función Regulación de Temperatura debe establecer el valor del Estado del Regulador.

  Justificación: El estado de la Función Regulación de Temperatura se proporciona a la Interfaz del Operador para que pueda utilizar el estado de las Funciones Regulación de Temperatura y Monitor de Temperatura para informar el estado general del Termostato, como lo exige SR-1.

- REQ-RT-3: La Función Regulación de Temperatura establecerá el valor de la Temperatura de Visualización.

  Justificación: La Temperatura Actual de la Isolette se muestra en la Interfaz del Operador para proporcionar a los operadores un medio adicional para confirmar que la Isolette mantiene la temperatura correctamente. Este valor lo proporciona la Función Regulación de Temperatura a la Interfaz del Operador como Temperatura de Visualización.

La Función Regulación de Temperatura se divide en subfunciones en la figura A-3.

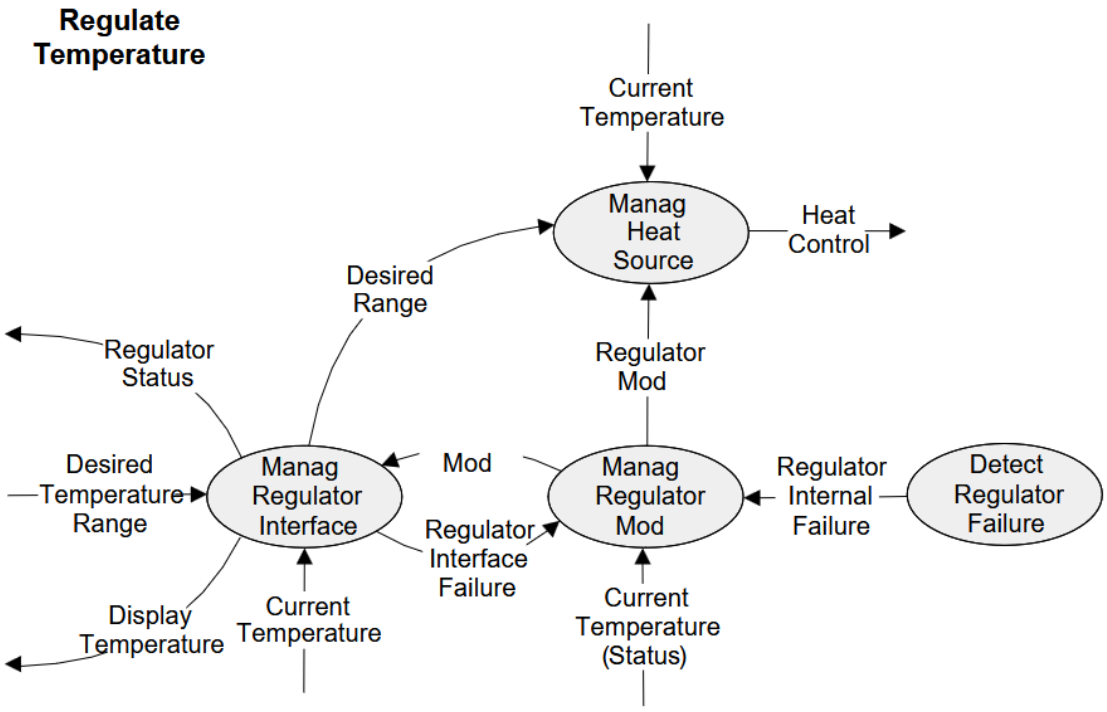


Figura A-3. Diagrama de Dependencia de Regulación de Temperatura

Las variables internas para la Función Regulación de Temperatura se muestran en la tabla A-7.

Tabla A-7. Variables internas del Regulador de Temperatura

Nombre	Tipo	Rango	Udad	Interpretación física
Rango deseado				Rango deseado de temperatura de Isolette
Temperatura Deseada Inferior	Entero	[96..101]	°F	Valor inferior del Rango Deseado
Temperatura Deseada Superior	Entero	[97..102]	°F	Valor superior del Rango Deseado
Falla de la Interfaz del Regulador	Booleano	Falso, Verdadero		Indica una falla de la interfaz de operador
Falla Interna del Regulador	Booleano	Falso, Verdadero		Indica un fallo interno
Modo del Regulador	Enumerado	Iniciando		Inicialización después del encendido
		NORMAL		Modo normal de funcionamiento
		FALLA		Fallo interno detectado

A.5.1.1 Función Gestión de la Interfaz del Operador.

La Función Gestión de la Interfaz del Operador define la interacción con la entidad externa de la Interfaz del Operador. Esto incluye la obtención del Rango Deseado, la notificación del estado de la Función Regulación de Temperatura y la notificación de la Temperatura de Visualización. Las constantes se muestran en la tabla A-8.

Tabla A-8. Constantes de la Función Gestión de la Interfaz del Operador

Nombre	Tipo	Valor	Udad	Interpretación física
Tiempo de Respuesta al Operador Máximo	Real	0,5	seg	El tiempo que un operador tolerará entre una solicitud del operador o un cambio en el estado del Termostato y la respuesta visible
Justificación: Un estudio comercial ha demostrado que este retraso no debería ser superior a 0,5 segundos.				

Los requisitos para la variable controlada Estado del Regulador son los siguientes:

- REQ-MRI-1: Si el Modo del Regulador es INIT, el Estado del Regulador se establecerá en Init.

- REQ-MRI-2: Si el Modo del Regulador es NORMAL, el Estado del Regulador se establecerá en Encendido.

- REQ-MRI-3: Si el Modo Regulador es FALLA, el Estado del Regulador se establecerá en Fallido.

  Latencia: < Tiempo de Respuesta al Operador Máximo
  Tolerancia: N/D

Los requisitos para la variable controlada Temperatura de Visualización son los siguientes:

- REQ-MRI-4: Si el Modo del Regulador es NORMAL, la Temperatura de Visualización se establecerá en el valor de la Temperatura Actual redondeado al entero más cercano.

  Justificación: Mostrar el valor redondeado de la Temperatura Actual proporciona la visualización más precisa de la Temperatura Actual posible utilizando una pantalla de números enteros.
  Cuando se combina con la precisión del Sensor de Temperatura (EA-TS-2), la Temperatura de Visualización debe estar dentro de los 0,6 °F del valor real.

- REQ-MRI-5: Si el Modo del Regulador no es NORMAL, el valor de la Temperatura de Visualización no está especificado.

  Justificación: En modos distintos a NORMAL, el valor de la Temperatura de Visualización no es significativo y no se debe utilizar.

  Latencia: < Tiempo de Respuesta al Operador Máximo
  Tolerancia: ±0,6 °F

Los requisitos para la variable interna Falla de la Interfaz del Regulador son los siguientes:

- REQ-MRI-6: Si el atributo Estado de la Temperatura Deseada Inferior o de la Temperatura Deseada Superior es Inválido, la Falla de la Interfaz del Regulador se establecerá en Verdadero.

- REQ-MRI-7: Si el atributo Estado de la Temperatura Deseada Inferior y la Temperatura Deseada Superior es Válido, la Falla de la Interfaz del Regulador se establecerá en Falso.

  Justificación: La variable interna Falla de la Interfaz del Regulador indica si se han producido errores en la detección de las variables monitoreadas de la Interfaz del Operador que necesita la Función Regulación de Temperatura. Tenga en cuenta que su valor inicial en el encendido siempre será Verdadero, ya que el estado de la Temperatura Deseada Inferior y la Temperatura Deseada Superior son inicialmente Inválidos.

Los requisitos para la variable interna Rango Deseado son los siguientes:

- REQ-MRI-8: Si la Falla de la Interfaz del Regulador es Falsa, el Rango Deseado se establecerá en el Rango de Temperatura Deseado.

- REQ-MRI-9: Si la Falla de la Interfaz del Regulador es Verdadera, el Rango Deseado NO ESPECIFICADO.

  Justificación: el Rango Deseado solo tiene sentido cuando no hay una Falla de la Interfaz del Regulador. Si la hay, no se debe utilizar su valor y se puede configurar en cualquier valor.

A.5.1.2 Función Gestión del Modo del Regulador.

La Función Gestión del Modo del Regulador determina el modo de la Función Regulación de Temperatura. Las constantes y definiciones se muestran en las tablas A-9 y A-10, respectivamente.

Tabla A-9. Constantes de la Función Gestión del Modo del Regulador

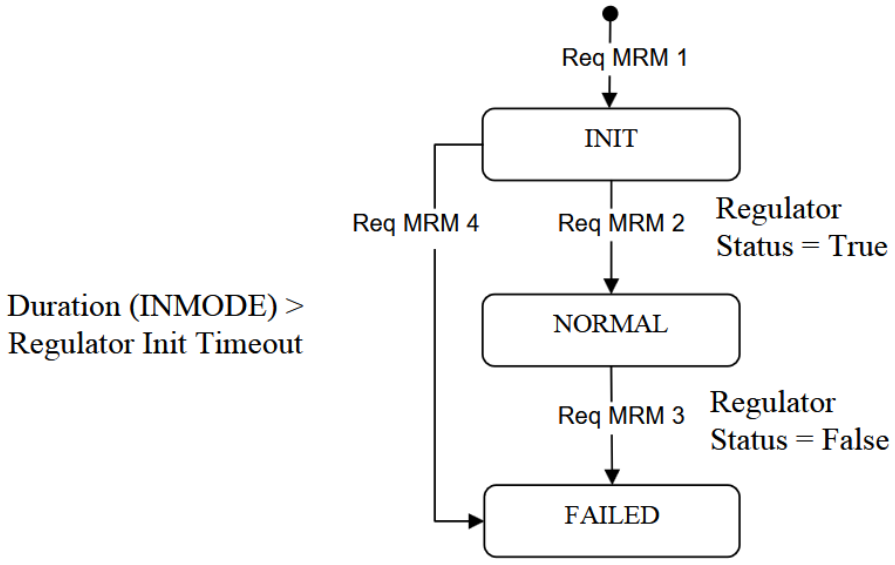
Nombre	Tipo	Valor	Udad	Interpretación física
Tiempo de espera de inicialización del Regulador	Real	1.0	seg	El tiempo permitido para la inicialización de la Función Regulación de Temperatura antes de declarar la falla
Justificación: Un estudio comercial ha demostrado que los usuarios se impacientan si el Termostato requiere más de un segundo para inicializarse.				

Tabla A-10. Definiciones de la Función Gestión del Modo del Regulador

Nombre	Tipo	Definición
Estado del Regulador	Booleano	NOT (Falla de la Interfaz del Regulador OR Falla Interna del Regulador) AND Temperatura Actual. Estado = Válido

Los requisitos para la variable interna del Modo Regulador son los siguientes:

- Los modos y transiciones de la Función Gestión del Modo del Regulador se especifican en el diagrama de transición de estados que se muestra en la figura A-4. Cada transición es un requisito independiente y se le asigna un identificador único (por ejemplo, Req MRM 1). Se supone que todas las transiciones ocurren en un tiempo insignificante.



MRM = Modo de gestión del regulador

Figura A-4. Diagrama de Transición del Modo de Regulación de Temperatura

Justificación: (Req MRM 3 y Req MRM 4) Una vez que el regulador ha fallado, la única forma de que vuelva a funcionar normalmente es apagándolo y encendiéndolo. Esto garantiza que los operadores estén al tanto de cualquier falla transitoria que pueda estar experimentando el regulador.

A.5.1.3 Función Gestión de Fuente de Calor.

La Función Gestión de Fuente de Calor activa y desactiva la Fuente de Calor para mantener la Temperatura Actual de la Isolette dentro del Rango de Temperatura Deseado. Las constantes se muestran en la tabla A-11.

Tabla A-11. Constantes de la Función Gestión de Fuente de Calor

Nombre	Tipo	Valor	Udad	Interpretación física
Latencia permitida de la Fuente de Calor	Real	6.0	seg	El tiempo máximo durante el cual la Fuente de Calor debe estar encendida o apagada para garantizar un funcionamiento aceptable del sistema Isolette
Justificación: Dado que un Isolette cerrado se calentará o enfriará a una velocidad máxima de 1 °F por minuto (EA-IS1 y EA-IS2), encender o apagar la Fuente de Calor dentro de los 6 segundos garantiza que la Temperatura Actual no habrá cambiado en más de 0,1 °F, la precisión y resolución requeridas del Sensor de Temperatura (EA-TS2).				

Los requisitos para la variable controlada de Control de Calor son los siguientes:

- REQ-MHS-1: Si el Modo del Regulador es INIT, el Control de Calor se configurará en Apagado.

  Justificación: Un regulador que se está inicializando no puede regular la Temperatura Actual del Isolette y el Control de Calor debe apagarse.

- REQ-MHS-2: Si el Modo del Regulador es NORMAL y la Temperatura Actual es menor que la Temperatura Deseada Inferior, el Control de Calor se configurará en Encendido.

- REQ-MHS-3: Si el Modo del Regulador es NORMAL y la Temperatura Actual es mayor que la Temperatura Deseada Superior, el Control de Calor se configurará en Apagado.

- REQ-MHS-4: Si el Modo del Regulador es NORMAL y la Temperatura Actual es mayor o igual a la Temperatura Deseada Inferior y menor o igual a la Temperatura Deseada Superior, no se deberá cambiar el valor del Control de Calor.

  Justificación: cuando la Isolette se está calentando hacia la Temperatura Deseada Superior, la Fuente de Calor debe dejarse encendida hasta que se alcance dicha temperatura. De manera similar, si la Isolette se está enfriando hacia la Temperatura Deseada Inferior, la Fuente de Calor debe dejarse apagada hasta que se alcance dicha temperatura.

- REQ-MHS-5: Si el Modo del Regulador es FALLA, el Control de Calor se configurará en apagado.

  Justificación: En el modo de falla, el regulador no puede regular la Temperatura Actual de la Isolette y el Control de Calor debe apagarse.

  Latencia: < Latencia permitida de la Fuente de Calor
  Tolerancia: N/D

A.5.1.4 Función Detección de Falla del Regulador.

La Función Detección de Falla del Regulador identifica fallas internas (por ejemplo, una falla en la verificación de memoria) en la Función Regulación de Temperatura. Esta define una única variable interna con valor booleano, Falla Interna del Regulador, que se establece en Verdadero si se detecta una falla interna.

Los requisitos para la variable de Falla Interna del Regulador son específicos de la implementación y no se pueden especificar hasta que se elija una plataforma de implementación.

A.5.2 FUNCIÓN MONITOR DE TEMPERATURA.

La Función Monitor de Temperatura compara la Temperatura Actual del Sensor de Temperatura con el Rango de Temperatura de Alarma proporcionado por la Interfaz del Operador y activa o desactiva el Control de Alarma para alertar a la Enfermera si la Temperatura Actual cae por debajo o aumenta por encima del rango seguro. También proporciona el Estado del Monitor a la Interfaz del Operador.

Los requisitos de alto nivel para la Función Monitor de Temperatura son los siguientes:

- REQ-MT-1: La Función Control de Temperatura debe establecer el valor del Control de Alarma.

  Justificación: La función principal de la Función Control de Temperatura es generar una alarma si la Isolette no puede mantener la Temperatura Actual dentro del Rango de Temperatura de Alarma, como lo exige el requisito de seguridad SR-2.

- REQ-MT-2: La Función Monitor de Temperatura deberá establecer el valor del Estado del Monitor.

  Justificación: El requisito de seguridad SR-2 exige que el Termostato proporcione una función de monitor independiente. El estado de esta función debe proporcionarse a la Interfaz del Operador, que la utilizará junto con el estado de la Función Regulación de Temperatura para informar el estado general del Termostato, como lo exige el requisito de seguridad SR-2.

La Función Monitor de Temperatura se divide en subfunciones como se muestra en la figura A-5.

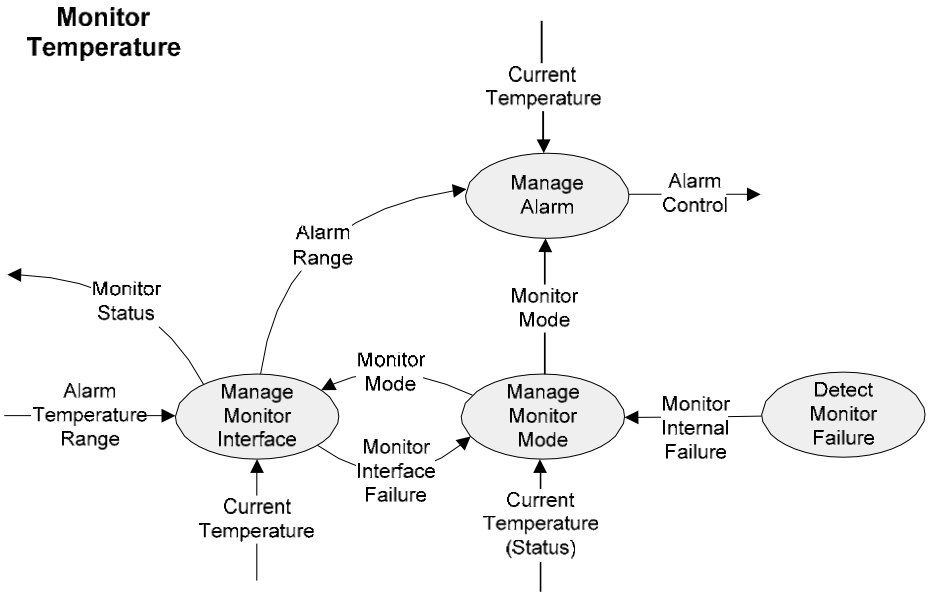


Figura A-5. Diagrama de Dependencia de la Monitor de Temperatura

Las variables internas del Monitor de Temperatura se muestran en la tabla A-12.

Tabla A-12. Variables internas del Monitor de Temperatura

Nombre	Tipo	Rango	Udad	Interpretación física
Rango de Alarma				Rango seguro de temperatura de Isolette
Temperatura de Alarma Inferior	Entero	[96..101]	°F	Valor inferior del rango de alarma
Temperatura de Alarma Superior	Entero	[97..102]	°F	Valor superior del rango de alarma
Falla de la Interfaz del Monitor	Booleano	Falso, Verdadero		Indica una falla de la interfaz del Operador
Falla Interna del Monitor	Booleano	Falso, Verdadero		Indica un fallo interno
Modo del Monitor	Enumerado	INIT		Inicialización después del encendido
		NORMAL		Modo normal de funcionamiento
		FALLA		Fallo interno detectado

A.5.2.1 Función Gestión de la Interfaz del Monitor.

La Función Gestión de la Interfaz del Monitor define la interacción con la entidad externa de la Interfaz del Operador. Esto incluye la obtención del Rango de Alarma e informar el estado de la Función Monitor de Temperatura. Las constantes se muestran en la tabla A-13.

Tabla A-13. Constantes de la Función Gestión de la Interfaz del Monitor

Nombre	Tipo	Valor	Udad	Interpretación física
Tiempo Máximo de Respuesta al Operador	Real	0,5	seg	El tiempo que un operador tolerará entre una solicitud del operador o un cambio en el estado del Termostato y la respuesta visible
Justificación: Un estudio comercial ha demostrado que este retraso no debería ser superior a 0,5 segundos.				

Los requisitos para la variable controlada Estado del Monitor son los siguientes:

- REQ-MMI-1: Si el Modo de Gestión de la Interfaz del Monitor es INIT, el Estado del Monitor se establecerá en Init.
- REQ-MMI-2: Si el Modo de Gestión de la Interfaz del Monitor es NORMAL, el Estado del Monitor se establecerá en Encendido.
- REQ-MMI-3: Si el Modo de Gestión de la Interfaz del Monitor es FALLA, el Estado del Monitor se establecerá en Falla.

  Latencia: < Tiempo Máximo de Respuesta al Operador
  Tolerancia: N/D

Los requisitos para la variable interna de Falla de la Interfaz del Monitor son los siguientes:

- REQ-MMI-4: Si el atributo Estado de la Temperatura de Alarma Inferior o de la Temperatura de Alarma Superior es Inválido, la variable Falla de la Interfaz del Monitor se establecerá en Verdadero.
- REQ-MMI-5: Si el atributo Estado de la Temperatura de Alarma Inferior y la Temperatura de Alarma Superior es Válido, la variable Falla de la Interfaz del Monitor se establecerá en Falso.

  Justificación: La variable interna de Falla de la Interfaz del Monitor indica si se han producido errores en la detección de las variables monitoreadas de la Interfaz del Operador que necesita la Función Gestión de Temperatura. Tenga en cuenta que su valor inicial en el encendido siempre será Verdadero, ya que el atributo Estado de la Temperatura de Alarma Inferior y la Temperatura de Alarma Superior inicialmente serán Inválidos.

Los requisitos para la variable interna de Rango de Alarma son los siguientes:

- REQ-MMI-6: Si la Falla de la Interfaz del Monitor es Falsa, la variable Rango de Alarma se establecerá en el Rango de Temperatura Deseado.
- REQ-MMI-7: Si la Falla de la Interfaz del Monitor es Verdadera, la variable de Rango de Alarma es NO ESPECIFICADA.

Justificación: La variable Rango de Alarma solo tiene sentido cuando no hay una Falla en la Interfaz del Monitor. Si la hay, no se debe utilizar su valor y se puede configurar en cualquier valor.

A.5.2.2 Función Gestión del Modo del Monitor.

La Función Gestión del Modo del Monitor determina el modo de la Función Monitor de Temperatura. Las constantes y definiciones se muestran en las tablas A-14 y A-15, respectivamente.

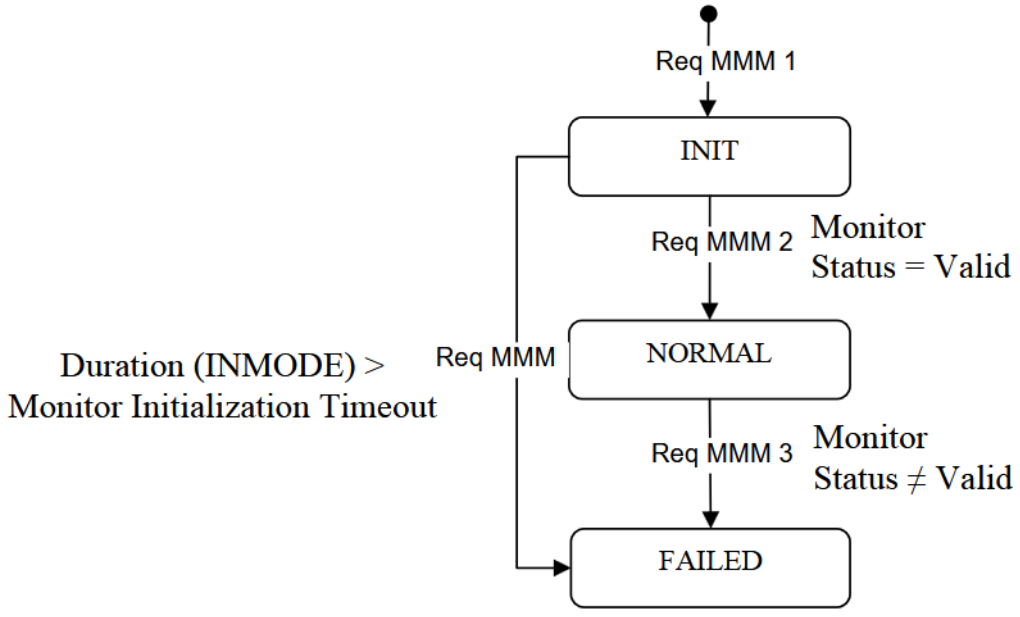
Tabla A-14. Constantes de la Función Gestión del Modo del Monitor

Nombre	Tipo	Valor	Udad	Interpretación física
Tiempo de Espera de Inicialización del Monitor	Real	1.0	seg	El tiempo permitido para la inicialización de la Función Monitor de Temperatura antes de declarar la falla.
Justificación: Un estudio comercial ha demostrado que los usuarios se impacientan si el Termostato requiere más de un segundo para inicializarse.				

Tabla A-15. Definiciones de la Función Gestión del Modo del Monitor

Nombre	Tipo	Definición
Estado del Monitor	Booleano	NOT (Falla de la Interfaz del Monitor OR Falla Interna del Monitor) AND Temperatura Actual. Estado = Válido

Los modos y transiciones de la Función Gestión del Modo del Monitor se especifican en el diagrama de transición de estados que se muestra en la figura A-6. Cada transición es un requisito independiente y se le asigna un identificador único (por ejemplo, Req MMM 1). Se supone que todas las transiciones ocurren en un tiempo insignificante.



MMM = Gestión del Modo del Monitor

Figura A-6. Diagrama de Transición del Modo del Monitor de Temperatura

Justificación: (Req. MMM 3 y Req. MMM 4) Una vez que el monitor ha fallado, la única forma de que vuelva a funcionar normalmente es apagándolo y encendiéndolo. Esto garantiza que los operadores estén al tanto de cualquier falla transitoria que pueda estar experimentando el monitor.

A.5.2.3 Función Gestión de Alarma.

La Función Gestión de Alarma activa el Control de Alarma cuando la Temperatura Actual de la Isolette cae por debajo o aumenta por encima del Rango de Temperatura de Alarma.

Los requisitos para la variable controlada de Control de Alarma son los siguientes:

- REQ-MA-1: Si el Modo del Monitor es INIT, el Control de Alarma se establecerá en Apagado.

  Justificación: Un monitor que se está inicializando no debe activar la alarma a menos que entre en el modo FALLA.

- REQ-MA-2: Si el Modo del Monitor es NORMAL y la Temperatura Actual es menor que la Temperatura de Alarma Inferior o mayor que la Temperatura de Alarma Superior, el Control de Alarma se configurará en Encendido.

- REQ-MA-3: Si el Modo del Monitor es NORMAL y la Temperatura Actual es mayor o igual a la Temperatura de Alarma Inferior y menor que la Temperatura de Alarma Inferior +0.5°, o la Temperatura Actual es mayor que la Temperatura de Alarma Superior -0.5° y menor o igual a la Temperatura de Alarma Superior, el valor del Control de Alarma no se modificará.

  Justificación: Esto proporciona una histéresis que evita alarmas transitorias, consulte la figura A-7.

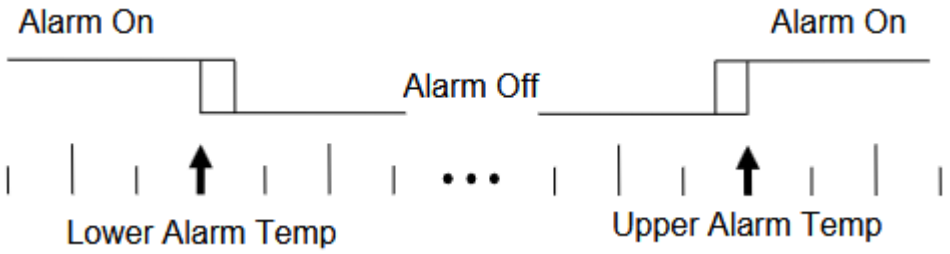


Figura A-7. Histéresis de Alarma Transitoria

- REQ-MA-4: Si el Modo del Monitor es NORMAL y el valor de la Temperatura Actual es mayor o igual a la Temperatura de Alarma Inferior +0,5° y menor o igual a la Temperatura de Alarma Superior -0,5°, el Control de Alarma se configurará en Apagado.

  Justificación: Esto apaga la alarma en el mismo momento en que la Temperatura de Visualización muestra un valor mayor que la Temperatura de Alarma Inferior y menor que la Temperatura de Alarma Superior.

- REQ-MA-5: Si el Modo del Monitor es FALLA, el Control de Alarma se establecerá en Encendido.

  Justificación: un monitor averiado no puede monitorear la Temperatura Actual de la Isolette y la alarma debe estar activada.

  Latencia: <5 segundos
  Tolerancia: N/A

  Justificación: Requerido por SR-2.

A.5.2.4 Función Detección de Fallas del Monitor.

La Función Detección de Falla del Monitor identifica fallas internas (por ejemplo, una falla en la verificación de memoria) en la Función Monitor de Temperatura. Define una única variable interna con valor booleano, Falla Interna del Monitor, que se establece en Verdadero si se detecta una falla interna.

Los requisitos para la variable Falla Interna del Monitor son específicos de la implementación y no se pueden especificar hasta que se elija una plataforma de implementación.

APÉNDICE B—EJEMPLO DE SISTEMA DE CONTROL DE VUELO

Este apéndice contiene una especificación de alto nivel para un sistema de control de vuelo (FCS) simplificado. El propósito de este ejemplo es ilustrar cómo se podría asignar una especificación a subsistemas separados, como se explicó en la sección 2.10. Por este motivo, el ejemplo solo se descompone en las funciones de guía de vuelo (FG) y piloto automático (AP). Estas funciones se desarrollan posteriormente como especificaciones de subsistemas separados, que se muestran en los apéndices C y D. Sin embargo, no se incluyen requisitos detallados de comportamiento y rendimiento.

B.1. DESCRIPCIÓN GENERAL DEL SISTEMA .

El sistema que se especifica es una parte de un FCS. El FCS compara la actitud medida de la aeronave con una actitud de referencia y genera comandos de guía del director de vuelo (FD) que se muestran como señales visibles, es decir, el FD, en las pantallas de vuelo primarias (PFD) izquierda y derecha. El piloto o copiloto puede volar manualmente la aeronave para seguir el FD y lograr la actitud de referencia. El piloto o copiloto puede borrar el FD de los PFD, volver a encender el FD y sincronizar la actitud de referencia con la actitud actual de la aeronave.

El FCS también proporciona una función AP que el piloto o el copiloto pueden solicitar. Cuando se activa la función AP, el FCS genera comandos de actuador que indicarán a las superficies de control de la aeronave que la lleven a la actitud de referencia. Mientras la función AP está activada, el piloto o el copiloto pueden iniciar la dirección del volante de control, lo que suspende la función AP y permite que el piloto o el copiloto lleven manualmente la aeronave a una nueva actitud y luego reanuden la función AP utilizando la nueva actitud como actitud de referencia.

Además del FD, los PFD también muestran si la función AP está activada, si la función AP ha fallado y si el FCS ha fallado.

B.1.1 CONTEXTO DEL SISTEMA .

El contexto operativo del FCS se muestra en la figura B-1.

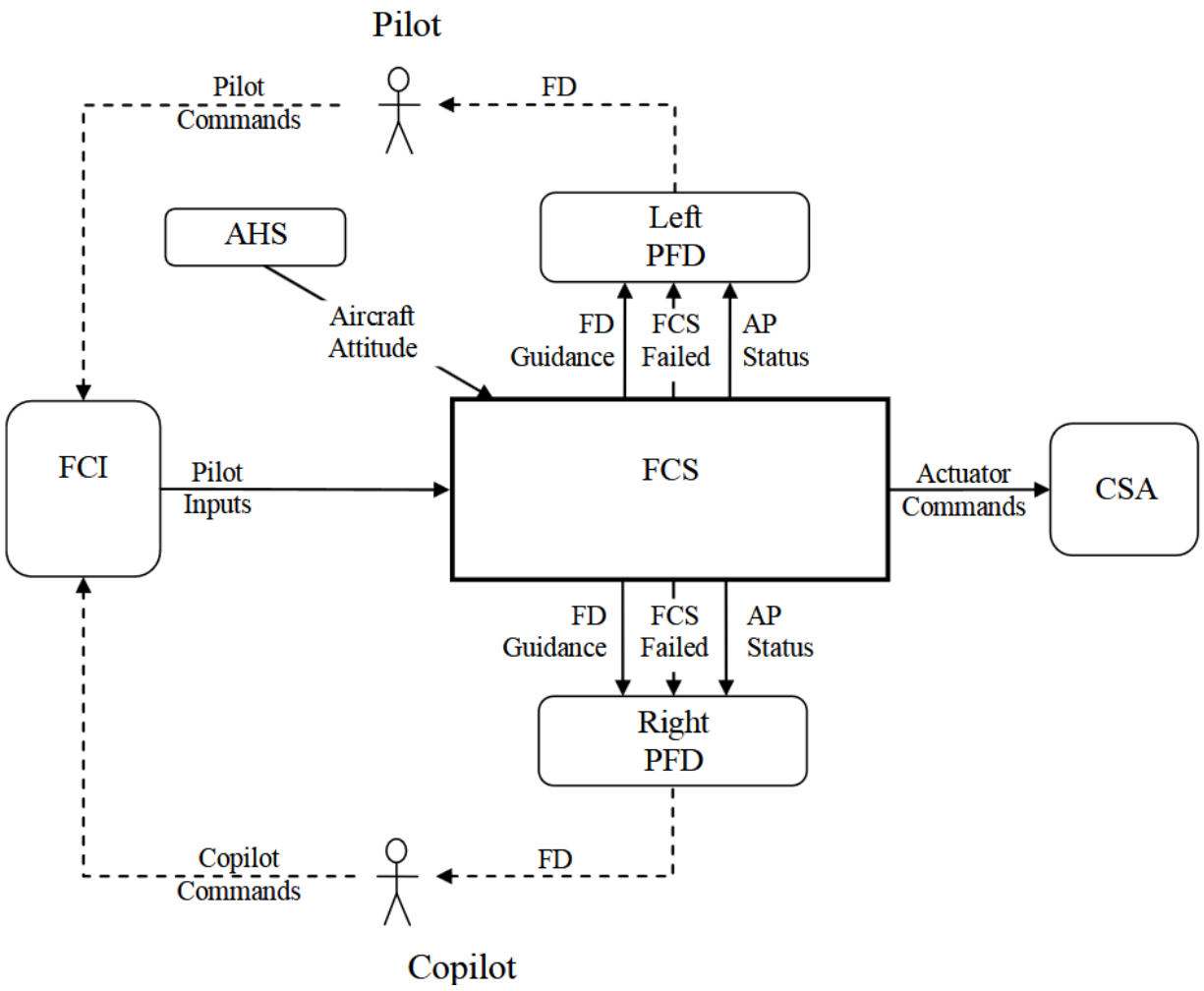


Figura B-1. Diagrama de contexto del sistema de control de vuelo

El FCS interactúa directamente con

- la interfaz de tripulación de vuelo (FCI) que suministra las entradas proporcionadas por el piloto y el copiloto, como las solicitudes de visualización FD, sincronización FD y activación de AP.
- el sistema de altitud y rumbo (AHS) que proporciona la actitud actual de la aeronave.
- los PFD derecho e izquierdo que muestran la guía FD, el estado del AP y la indicación de falla del FCS.
- los actuadores de superficie de control (CSA) que posicionan las superficies de control de la aeronave según los comandos del actuador.

El FCS también interactúa indirectamente con

- el piloto y el copiloto, quienes visualizan el FD, el estado del AP, la indicación de FCS fallido en el PFD y proporcionan las entradas del piloto a través del FCI.

B.1.2 OBJETIVOS DEL SISTEMA .

Los objetivos de alto nivel de este sistema son los siguientes:

- G1—Brindar orientación al piloto y al copiloto para permitirles volar manualmente la aeronave a una actitud deseada.
- G2: hace volar automáticamente la aeronave para mantener la actitud deseada cuando se activa el AP
- G3—Garantizar que el FCS no ordene maniobras inseguras de la aeronave
- G4—Asegurarse de que las maniobras de la aeronave que puedan causar incomodidad a los pasajeros no sean ordenadas por el FCS
- G5—Mantener el costo de fabricación del FCS lo más bajo posible
- G6—Mantener el costo de operación del FCS lo más bajo posible

B.2. CONCEPTOS OPERACIONALES .

Los casos de uso en las tablas B-1 a B-5 describen cómo los operadores (el piloto y el copiloto) interactúan con el FCS.

Consulte la tabla B-1 para el caso de uso 1: el piloto o copiloto activa y vuela el FD.

Tabla B-1. Activación y vuelo del FD

Objetivo	Describe el uso normal del FCS por parte del piloto o copiloto cuando vuela manualmente la aeronave para seguir las barras de comando del FD.		
Actor principal		Objetivos relacionados	
Piloto o copiloto		G1, G3 y G4	
Condición previa		Postcondición	
Ambos FD están borrados La función AP está desactivada		Se muestran ambos FD La función AP está desactivada	
Escenario de éxito principal (caso de uso 1)		Excepción	Función
1. El piloto o copiloto solicita la visualización del FD.1			
2. FCS sincroniza la actitud de referencia con la actitud actual de la aeronave		CE 1	B.4.1
3. Cada PFD muestra su FD en la actitud de referencia			
4. El piloto o copiloto vuela la aeronave hasta la actitud deseada.			
5. El piloto o copiloto solicita la sincronización de referencias2			
6. FCS sincroniza la actitud de referencia con la actitud actual de la aeronave		CE 2	B.4.1
7. Cada PFD posiciona gradualmente su FD en la nueva Actitud de Referencia			
8. El piloto o copiloto vuela manualmente la aeronave para seguir el FD			
Caso de excepción CE 1			
1. El FD no se muestra debido a la actitud insegura de la aeronave			
2. Cada PFD anuncia una actitud insegura para alertar al piloto y al copiloto.			
3. El piloto o copiloto vuela la aeronave a una actitud segura.			
4. Regrese al paso 1 del escenario de éxito principal			
Caso de excepción CE 2			
1. El FCS no logra sincronizarse debido a una actitud insegura			B.4.1
2. Cada PFD anuncia una actitud insegura para alertar al piloto y al copiloto.			
3. El piloto o copiloto vuela la aeronave hasta una actitud segura.			
4. Regrese al paso 4 del escenario de éxito principal			

1Por ejemplo, presionando el interruptor FD en el FCP
2Por ejemplo, presionando el botón SYNC en el yugo de control

Consulte la tabla B-2 para el caso de uso 2: el piloto o copiloto borra el FD.

Tabla B-2. Limpieza del FD

Objetivo	Describe cómo el piloto o copiloto apaga el FD		
Actor principal		Objetivos relacionados	
Piloto o copiloto		G1, G2 y G3	
Condición previa		Postcondición	
Se muestran ambos FD		Ambos FD están borrados La función AP está desactivada	
Escenario de éxito principal (caso de uso 2)		Excepción	Función
1. El piloto o copiloto solicita autorización para el vuelo FD1			B.4.1
2. Cada PFD borra su FD		CE 1	
Caso de excepción 1 (CE 1)			
1. El FD no se borra porque la función AP está activada			B.4.2
2. El piloto o copiloto solicita la desconexión de la función AP2			
3. AP se desvincula			B.4.2
4. Regrese al paso 1 del escenario de éxito principal			

1Por ejemplo, presionando el interruptor FD en el FCP
2Por ejemplo, presionando el interruptor AP en el FCP

Consulte la tabla B-3 para el caso de uso 3: El piloto o copiloto activa la función AP.

Tabla B-3. Activación de la función AP

Objetivo	Describe la activación de la función AP por parte del piloto o copiloto cuando se muestra el FD		
Actor principal		Objetivos relacionados	
Piloto o copiloto		G2 y G3	
Condición previa		Postcondición	
Se muestran ambos FD La función AP está desactivada		Se muestran ambos FD. La función AP está activada.	
Escenario de éxito principal (caso de uso 3)		Excepción	Función
1. El piloto o copiloto vuela la aeronave hasta la actitud deseada.			
2. El piloto o copiloto solicita la participación de AP*			
3. FCS sincroniza la actitud de referencia con la actitud de la aeronave		CE 1	B.4.1
4. Cada PFD posiciona su FD en la nueva Actitud de Referencia			
5. FCS activa la función AP		CE 2	B.4.2
6. Cada PFD anuncia la activación del AP al piloto y al copiloto.			
7. La función AP genera comandos de actuador para mantener la aeronave en la actitud de referencia			B.4.2
Caso de excepción 1 (CE 1)			
1. El FCS no logra sincronizarse debido a una actitud insegura			B.4.1
2. Cada PFD anuncia una actitud insegura para alertar al piloto y al copiloto.			
3. El piloto o copiloto vuela la aeronave a una actitud segura.			
4. Regrese al paso 2 del escenario de éxito principal			
Caso de excepción 2 (CE 2)			
1. La función AP no se activa debido a una falla de la función AP			B.4.2
2. Cada PFD anuncia la falla del AP para alertar al piloto y al copiloto.			
3. El piloto o copiloto vuela el avión manualmente.			

* Por ejemplo, presionando el interruptor AP en el FCP

Consulte la tabla B-4 para el caso de uso 4: El piloto o copiloto inicia la dirección del volante de control.

Tabla B-4. Inicio de la dirección del volante de control

Objetivo	Describe cómo el piloto o copiloto vuela la aeronave hacia una nueva actitud cuando se activa el AP.	
Actor principal	Objetivos relacionados	
Piloto o copiloto	G2, G3 y G4	
Condición previa	Postcondición	
Se muestran ambos FD. La función AP está activada.	Se muestran ambos FD. La función AP está activada.	
Escenario de éxito principal (caso de uso 4)	Excepción	Función
El piloto o copiloto solicita el control de la dirección del volante1		
La función AP se desacopla2de CSA		B.4.2
Cada PFD anuncia que la función AP está desacoplada		
FCS inicia la sincronización continua de la actitud de referencia con la actitud de la aeronave		B.4.1
El piloto o copiloto vuela la aeronave a la actitud deseada.	CE 1	
El piloto o copiloto cancela el control de dirección del volante3		
FCS detiene la sincronización de la actitud de referencia		B.4.1
La función AP se acopla a CSA		B.4.2
Cada PFD anuncia que la función AP está acoplada		
La función AP dirige gradualmente la aeronave hacia una nueva actitud de referencia		
Caso de excepción 1 (CE 1)		
La función AP se desactiva debido a la actitud insegura de la aeronave		B.4.2
Cada PFD anuncia una actitud insegura para alertar al piloto y al copiloto.		
FCS detiene la sincronización de la actitud de referencia		B.4.1
El piloto o copiloto vuela la aeronave a una actitud segura.		
FCS reanuda la sincronización de la Actitud de Referencia		B.4.1
El piloto o copiloto cancela el control de dirección del volante		
FCS detiene la sincronización de la actitud de referencia		B.4.1

1Por ejemplo, manteniendo presionado el botón SYNC en el yugo de control 2Desconecta temporalmente el AP del CSA sin desconectar el AP 3Por ejemplo, soltando el botón SYNC en el yugo de control

Consulte la tabla B-5 para el caso de uso 5: el piloto o copiloto desactiva la función AP.

Tabla B-5. Desactivación de la función AP

Objetivo	Describe cómo el piloto o copiloto desactiva la función AP		
Actor principal		Objetivos relacionados	
Piloto o copiloto		G2	
Condición previa		Postcondición	
Se muestran ambos FD. La función AP está activada.		Se muestran ambos FD La función AP está desactivada	
Escenario de éxito principal (caso de uso 5)		Excepción	Función
El piloto o copiloto solicita la desconexión de la función AP*			B.4.2
FCS desactiva la función AP		EC1	
Cada PFD anuncia la desconexión de la función AP			

* Por ejemplo, presionando el interruptor AP en el FCP

B.3. ENTIDADES EXTERNAS .

Las siguientes secciones describen las entidades externas con las que el FCS interactúa directamente: el FCI, el AHS, el CSA y el PFD izquierdo y derecho. Se enumeran las variables monitoreadas y controladas asociadas con cada entidad junto con los supuestos ambientales realizados sobre la entidad.

B.3.1 INTERFAZ CON LA TRIPULACIÓN DE VUELO .

El FCI proporciona los datos de entrada del piloto y del copiloto que afectan el comportamiento del FCS. Las variables monitoreadas se muestran en la tabla B-6. No se realizan suposiciones ambientales.

Tabla B-6. Variables monitoreadas del sistema de control de vuelo para FCI

Nombre	Tipo	Rango	Unidades	Interpretación física
Entradas del piloto				Comandos proporcionados por el piloto o copiloto
Mostrar FD	Booleano	Falso, Verdadero		Comando para mostrar o borrar el FD al Piloto y Copiloto
	Estado	● Inválido, Válido		
Sincronizar FD	Booleano	Falso, Verdadero		Comando para establecer la actitud de referencia a la actitud actual de la aeronave
	Estado	● Inválido, Válido		
Involucrar a AP	Booleano	Falso, Verdadero		Comando para activar o desactivar la Función AP
	Estado	● Inválido, Válido		

● denota valor inicial

B.3.2 SISTEMA DE RUMBO DE ACTITUD .

El AHS proporciona la actitud actual de la aeronave al FCS. Las variables monitoreadas se muestran en la tabla B-7.

Tabla B-7. Variables monitoreadas del sistema de control de vuelo para el sistema de rumbo y altitud

Nombre	Tipo	Rango	Unidades	Interpretación física
Actitud de la aeronave				Actitud actual de la aeronave
Balanceo de aeronave	Real	[-180.0..179.9]	Grados	Ángulo de balanceo actual de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
	Estado	● Inválido, Válido		
Aeronave Paso	Real	[-180.0..179.9]	Grados	Ángulo de inclinación actual de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
	Estado	● Inválido, Válido		

● denota valor inicial

Se hacen los siguientes supuestos ambientales:

- EA-AHS-1: El balanceo de la aeronave oscilará entre -180,0° y +179,9°, inclusive.

Justificación: El AHS proporcionará un balanceo real de la aeronave, que puede adoptar cualquier valor en todo el rango de movimiento de la aeronave.

- EA-AHS-2: El balanceo de la aeronave se detectará con una precisión de ±0,1°.

Justificación: Esta precisión es necesaria para garantizar que la actitud de la aeronave mostrada se mueva suavemente en el PFD y para garantizar que el FCS calcule los comandos del actuador y los comandos de guía del FD con la precisión necesaria.

- EA-AHS-3: El ángulo de inclinación de la aeronave oscilará entre -180,0° y +179,9°, inclusive.

Justificación: El AHS proporcionará el paso real de la aeronave, que puede adoptar cualquier valor en todo el rango de movimiento de la aeronave.

- EA-AHS-4: La inclinación de la aeronave se detectará con una precisión de ±0,1°.

Justificación: Esta precisión es necesaria para garantizar que la actitud de la aeronave mostrada se mueva suavemente en el PFD y para garantizar que el FCS calcule los comandos del actuador y los comandos de guía del FD con la precisión necesaria.

B.3.3 ACTUADORES DE SUPERFICIE DE CONTROL .

El CSA posiciona las superficies de control de la aeronave en función de los comandos de actuador generados por el FCS para mantener la aeronave en la actitud de referencia. Las variables controladas se muestran en la tabla B-8.

Tabla B-8. Variables controladas del sistema de control de vuelo para actuadores de superficie controlados

Nombre	Tipo	Rango	Unidades	Interpretación física
Comandos del actuador				Tasas de actuadores comandados
Rollo Solenoide	Real	[-20.0..20.0]	∘ superficie/ segundo	Velocidad comandada del actuador de balanceo: · 0 → sin cambios en las superficies de control · -X → ala izquierda abajo. • +X → ala derecha hacia abajo
Paso Solenoide	Real	[-20.0..20.0]	∘ superficie/ segundo	Velocidad comandada del actuador de paso: · 0 → sin cambios en las superficies de control · -X → nariz hacia arriba • +X → nariz hacia abajo
AP En	Booleano	Falso, Verdadero		Indicación de si la función AP está activada y se utilizarán los comandos del actuador

Se hacen los siguientes supuestos ambientales:

- EA-CSA-1: La velocidad del actuador de alabeo oscilará entre -20,0 y +20,0° de superficie/segundo, ambos inclusive.

Fundamento: El fabricante de la aeronave lo especifica como el rango necesario para controlar adecuadamente la aeronave.

- EA-CSA-2: La velocidad del actuador de alabeo se establecerá con una resolución de 0,1° de superficie/segundo.

Fundamento: El análisis de capacidad de control muestra que es necesaria una resolución de 0,1° de superficie/segundo para mantener el control de la aeronave.

- EA-CSA-3: La velocidad del actuador de cabeceo oscilará entre -20,0 y +20,0° de superficie/segundo, ambos inclusive.

Fundamento: El fabricante de la aeronave lo especifica como el rango necesario para controlar adecuadamente la aeronave.

- EA-CSA-4: La velocidad del actuador de cabeceo se establecerá con una resolución de 0,1° de superficie/segundo.

Justificación: El análisis de controlabilidad muestra que una resolución de 0,1° superficie/segundo es necesaria para mantener el control de la aeronave.

B.3.4 B.3.4 PANTALLA DE VUELO PRIMARIA .

El PFD izquierdo y derecho muestra el FD, la indicación de falla del FCS y el estado del AP. Las variables controladas se muestran en la tabla B-9.

Tabla B-9. Variables controladas del sistema de control de vuelo para PFD

Nombre	Tipo	Rango	Unidades	Interpretación física
Orientación del Departamento de Finanzas				Comandos de guía para FD
Guía de balanceo	Real	[-45.0..45.0]	Grados	Ángulo de giro deseado de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
Guía de lanzamiento	Real	[-45.0..45.0]	Grados	Ángulo de inclinación deseado de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
FD encendido	Booleano	Falso, Verdadero		Indicación si se debe visualizar FD: · Falso → No mostrar FD · Verdadero → Mostrar FD
FCS falló	Booleano	Falso, Verdadero		Indicación si el FCS falla: · Falso → FCS está funcionando · Verdadero → El FCS ha fallado
Estado de AP	Enumerado	Fallido, Apagado, Encendido		Estado de la función AP: · Falló → La función AP ha fallado · Apagado → La función AP está desactivada · En → La función AP está activada

Se hacen los siguientes supuestos ambientales:

- EA-PFD-1: La guía de alabeo oscilará entre -45,0° y +45,0°, ambos inclusive.

Fundamento: Especificado por el fabricante de la aeronave como el rango máximo para la guía de alabeo.

- EA-PFD-2: La guía de alabeo se establecerá en décimas de grado.

Fundamento: Esta resolución es necesaria para lograr un movimiento suave del FD durante la dirección del volante de control.

- EA-PFD-3: La guía de cabeceo oscilará entre -45,0° y +45,0°, ambos inclusive.

Fundamento: Especificado por el fabricante de la aeronave como el rango máximo para la guía de cabeceo.

- EA-PFD-4: La guía de cabeceo se establecerá en décimas de grado.

Fundamento: Esta resolución es necesaria para lograr un movimiento suave del FD durante la dirección del volante de control.

B.4. FUNCIONES DEL SISTEMA DE CONTROL DE VUELO .

En esta sección se describen las principales funciones proporcionadas por el FCS.

Los requisitos de alto nivel para la función FCS son:

- REQ-FCS-1: El FCS deberá generar los comandos de guía FD.

Fundamento: Esta es una función principal del FCS. El PFD utiliza los comandos de guía del FD para posicionar al FD y mostrarle al piloto y al copiloto cómo volar la aeronave hasta la actitud de referencia.

- REQ-FCS-2: El FCS deberá establecer la actitud de referencia.

Justificación: El FCS establece la actitud de referencia cuando el piloto o copiloto solicita al FCS que sincronice la actitud de referencia con la actitud actual o que active la función AP.

- REQ-FCS-3: El FCS deberá establecer el estado FCS Failed para indicar si la función FCS ha fallado.

Fundamento: Si el FCS ha fallado, se debe dar de baja el FD y notificar al piloto y al copiloto que la función FCS ha fallado. Esto suele indicarlo el PFD.

- REQ-FCS-4: El FCS deberá generar los comandos del actuador.

Fundamento: Esta es una función principal del FCS. El CSA utiliza los comandos del actuador para hacer volar la aeronave a la actitud de referencia cuando la función AP está activada.

- REQ-FCS-5: El FCS deberá establecer el estado del AP para indicar el estado actual de la función del AP.

Justificación: El estado actual de la función AP debe mostrarse al piloto y al copiloto para garantizar que sepan cuándo la función AP está controlando la aeronave.

Consulte la figura B-2 para conocer las funciones del FCS.

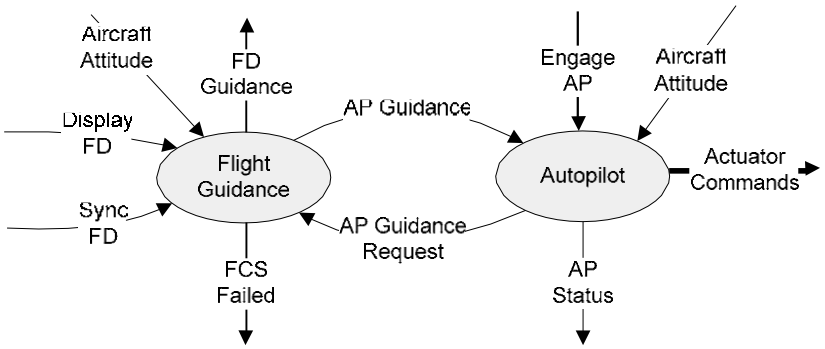


Figura B-2. Diagrama de funciones del sistema de control de vuelo

Para las variables internas del FCS consulte la tabla B-10.

Tabla B-10. Variables internas del sistema de control de vuelo

Nombre	Tipo	Rango	Unidades	Interpretación física
Orientación AP				Comandos de orientación para AP
Guía de balanceo	Real	[-45.0..45.0]	Grados	Ángulo de giro deseado de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
Guía de lanzamiento	Real	[-45.0..45.0]	Grados	Ángulo de inclinación deseado de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
Orientación válida	Booleano	Falso, Verdadero		Indica si la guía AP es válida y se puede utilizar
Solicitud de orientación de AP	Booleano	Falso, Verdadero		Solicitud de orientación AP válida

B.4.1 FUNCIÓN DE GUÍA DE VUELO .

La función FG compara la actitud de la aeronave medida con una actitud de referencia y genera comandos de guía FD que se muestran como señales visibles en los comandos de guía FD y AP que utiliza la función AP.

Los requisitos de alto nivel para la función FG son:

- REQ-FG-1: La función FG deberá generar los comandos de guía FD.

Fundamento: Una función principal del FG es proporcionar los comandos de guía del FD para la función FCS.

- REQ-FG-2: La función FG deberá establecer la actitud de referencia.

Fundamento: La función FG establece la actitud de referencia cuando se le solicita sincronizar la actitud de referencia con la actitud actual (ya sea por el piloto, el copiloto o la función AP).

- REQ-FG-3: La función FG deberá generar los comandos de guía AP.

Fundamento: Una función principal de la función FG es proporcionar los comandos de guía AP para la función FCS.

- REQ-FG-4: La función FG deberá establecer la indicación de FCS fallido.

Fundamento: La función FCS se divide entre la función FG y la función AP. El estado de la función AP se indica de forma independiente al piloto y al copiloto en el PFD. Como resultado, el estado de la función FCS (FCS fallido) lo establece la función FG.

B.4.2 FUNCIÓN DE PILOTO AUTOMÁTICO .

La función AP genera comandos de actuador a partir de los comandos de guía AP proporcionados por la función FG. Los comandos de actuador son utilizados por CSA para hacer volar la aeronave a la actitud especificada por los comandos de guía AP.

Los requisitos de alto nivel para la función AP son:

- REQ-AP-1: La función AP deberá generar los comandos del actuador.

Justificación: La función principal del AP es generar los comandos del actuador para la función FCS.

- REQ-AP-2: La función AP deberá establecer el estado del AP para indicar su estado.

Justificación: La función AP es responsable de proporcionar su estado actual a la función FCS.

- REQ-AP-3: La Función AP deberá generar la Solicitud de Orientación AP.

Justificación: La solicitud de guía AP requiere que la función FG sincronice su actitud de referencia para producir un comando de guía AP aceptable.

APÉNDICE C—EJEMPLO DE SISTEMA DE GUÍA DE VUELO

Este apéndice contiene una especificación de alto nivel para un sistema de guía de vuelo (FGS) simplificado. El propósito de este ejemplo es ilustrar cómo se podría asignar una especificación a subsistemas separados, como se analiza en la sección 2.10. Amplía la función de guía de vuelo (FG) del sistema de control de vuelo (FCS) especificado en el apéndice B en una especificación de subsistema separada que se podría entregar a un subcontratista. Como tal, contiene solo el subconjunto de la información de la especificación FCS relevante para el subsistema FGS. Esta especificación se utilizaría como punto de partida para que la completaran el contratista del FCS y el subcontratista del FGS antes de desarrollar el FGS. Se necesitaría una descomposición funcional adicional y la definición de los requisitos detallados de comportamiento y rendimiento para completar esta especificación.

C.1. DESCRIPCIÓN GENERAL DEL SISTEMA .

El sistema que se especifica es una parte de un FGS. El FGS compara la actitud medida de la aeronave con una actitud de referencia y genera comandos de guía del director de vuelo (FD) que se muestran como señales visibles, es decir, el FD, en las pantallas de vuelo primarias (PFD) izquierda y derecha. El piloto o copiloto puede volar manualmente la aeronave para seguir al FD y mantener la actitud de referencia. El piloto o copiloto también puede borrar el FD de los PFD, volver a encender el FD y sincronizar la actitud de referencia con la actitud actual de la aeronave.

El FGS también genera comandos de guía del piloto automático (AP) que utiliza un sistema AP para mover las superficies de control de la aeronave y seguir la guía del AP. En cualquier momento, el piloto o el copiloto pueden iniciar la dirección del volante de control, lo que indica al AP que se desacople de los actuadores de la superficie de control (CSA), lo que permite al piloto o al copiloto volar manualmente la aeronave a una nueva actitud que luego es seguida por el AP.

C.1.1 CONTEXTO DEL SISTEMA .

El contexto operativo del FGS se muestra en la figura C-1.

Figura C-1. Diagrama de contexto del FGS

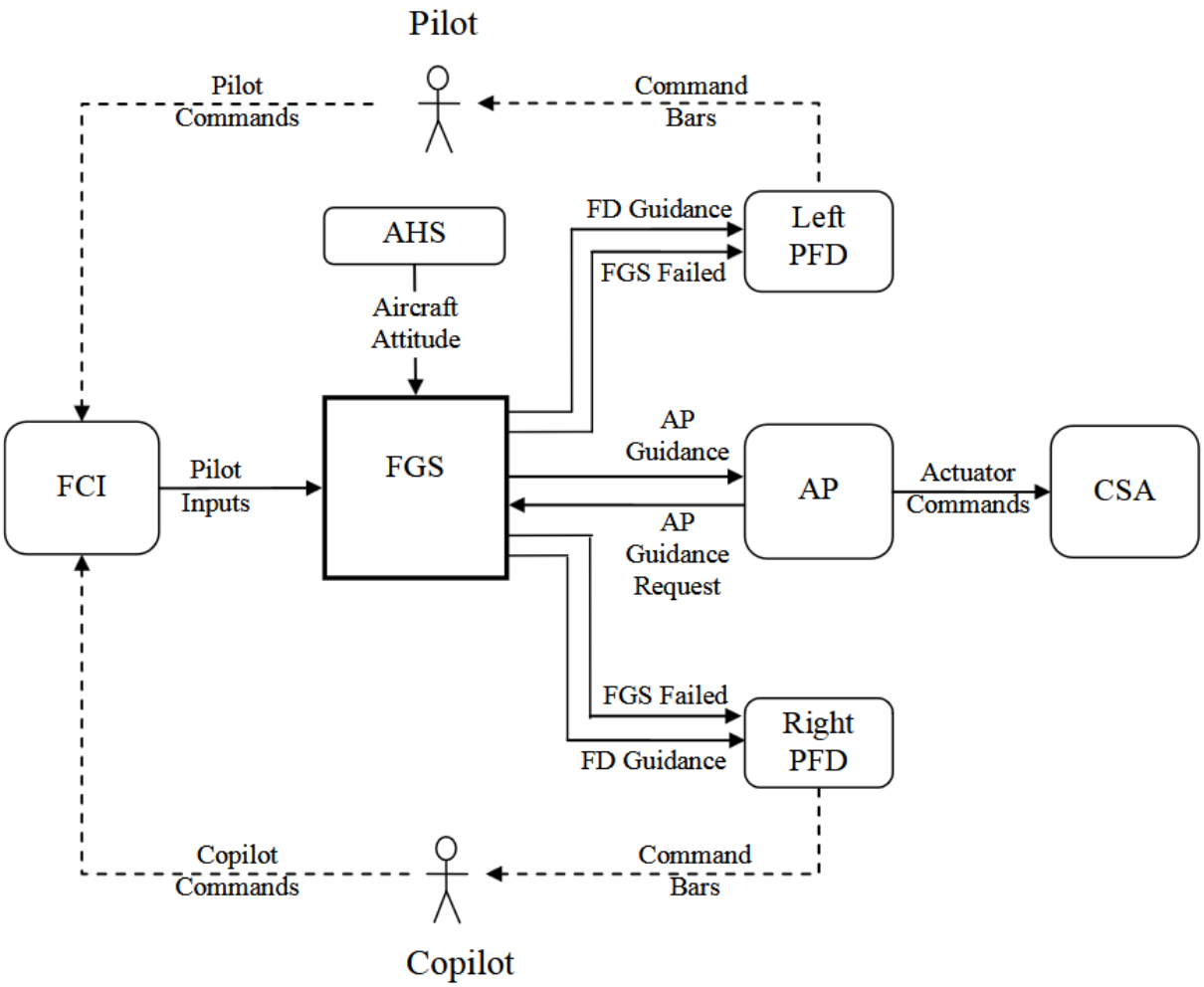


Figura C-1. Diagrama de contexto del FGS

El FGS interactúa directamente con

- la interfaz de tripulación de vuelo (FCI) que suministra las entradas proporcionadas por el piloto y el copiloto, como los comandos Display FD y Sync FD.
- el sistema de actitud y rumbo (AHS) que proporciona la actitud actual de la aeronave.
- El AP que traduce los comandos de guía de AP generados por el FGS a los comandos del actuador. El AP también envía comandos de solicitud de guía de AP al FGS para solicitar comandos de guía de AP válidos.
- los PFD derecho e izquierdo que utilizan los comandos de guía FD para posicionar las barras de comando y mostrar el indicador FGS Failed al piloto y al copiloto.

El FGS también interactúa indirectamente con:

- las aeronaves CSA que son movidas por los Comandos del Actuador generados por el AP.
- el piloto y el copiloto, quienes visualizan las barras de comando que se muestran en el FD, el estado de FGS fallido en el PFD y proporcionan las entradas del piloto a través del FCI.

C.1.2 OBJETIVOS DEL SISTEMA .

Los objetivos de alto nivel de este sistema son los siguientes:

- G1—Brindar orientación al piloto y al copiloto para permitirles volar manualmente la aeronave a una actitud deseada.
- G2: Proporcionar orientación al AP para permitirle volar automáticamente la aeronave para mantener la actitud deseada.
- G3—Garantizar que el FGS no ordene maniobras inseguras de aeronaves
- G4—Garantizar que las maniobras de la aeronave que puedan causar incomodidad a los pasajeros no sean ordenadas por el FGS
- G5—Mantener el costo de fabricación del FGS lo más bajo posible
- G6—Mantener el costo de operación del FGS lo más bajo posible

C.2. CONCEPTOS DE OPERACIÓN/AL .

Aquí se proporcionan los conceptos operativos del FGS. Estos pueden consistir en casos de uso que describen cómo el FGS interactúa con la tripulación de vuelo y otros sistemas, como el AP, o cualquier otra información que el contratista y el subcontratista acuerden.

C.3. ENTIDADES EXTERNAS/AL .

Las siguientes secciones describen las entidades externas con las que el FGS interactúa directamente: el FCI, el AHS, el AP y el PFD izquierdo y derecho.

C.3.1 INTERFAZ CON LA TRIPULACIÓN DE VUELO .

El FCI proporciona las entradas del Piloto y del Copiloto que afectan el comportamiento del FGS. Las variables monitoreadas se muestran en la tabla C-1.

Tabla C-1. Variables monitoreadas por el FGS para FCI

Nombre	Tipo	Rango	Unidades	Interpretación física
Entradas del piloto				Comandos proporcionados por el piloto o copiloto
Mostrar Departamento de Bombero	Booleano	Falso, Verdadero		Comando para mostrar o borrar el FD
	Estado	• Inválido, Válido		
Sincronizar FD	Booleano	Falso, Verdadero		Comando para establecer la actitud de referencia a la actitud actual de la aeronave
	Estado	• Inválido, Válido		

● denota valor inicial

No se hacen suposiciones medioambientales.

C.3.2 SISTEMA DE RUMBO DE ACTITUD .

El AHS proporciona la actitud actual de la aeronave al FGS. Las variables monitoreadas se muestran en la tabla C-2.

Tabla C-2. Variables monitoreadas por el FGS para AHS

Nombre	Tipo	Rango	Unidades	Interpretación física
Actitud de la aeronave				Actitud actual de la aeronave
Balanceo de aeronave	Real	[-180.0..179.9]	Grados	Ángulo de balanceo actual de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
	Estado	• Inválido, Válido		
Paso de la aeronave	Real	[-180.0..179.0]	Grados	Ángulo de inclinación actual de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
	Estado	• Inválido, Válido		

● denota valor inicial

Se hacen los siguientes supuestos ambientales:

- EA-AHS-1: El balanceo de la aeronave oscilará entre -180,0° y +179,9°, inclusive.

Justificación: El AHS proporcionará un balanceo real de la aeronave, que puede adoptar cualquier valor en todo el rango de movimiento de la aeronave.

- EA-AHS-2: El balanceo de la aeronave se detectará con una precisión de ±0,1°.

Justificación: Esta precisión es necesaria para garantizar que la actitud de la aeronave mostrada se mueva suavemente en el PFD y para garantizar que el FGS calcule los comandos de guía AP y guía FD con la precisión necesaria.

- EA-AHS-3: El ángulo de inclinación de la aeronave oscilará entre -180,0° y +179,9°, inclusive.

Justificación: El AHS proporcionará el paso real de la aeronave, que puede adoptar cualquier valor en todo el rango de movimiento de la aeronave.

- EA-AHS-4: La inclinación de la aeronave se detectará con una precisión de ±0,1°.

Justificación: Esta precisión es necesaria para garantizar que la actitud de la aeronave mostrada se mueva suavemente en el PFD y para garantizar que el FGS calcule los comandos de guía AP y guía FD con la precisión necesaria.

C.3.3 PILOTO AUTOMÁTICO .

El AP traduce los comandos de guía generados por el FGS en comandos de actuador que moverán las superficies de control de la aeronave para lograr los cambios ordenados sobre los ejes lateral y vertical. Las variables monitoreadas y controladas se muestran en las tablas C-3 y C-4, respectivamente.

Tabla C-3. Variables monitoreadas por FGS para AP

Nombre	Tipo	Rango	Unidades	Interpretación física
Orientación AP Pedido	Booleano	Falso, Verdadero		Indicación del AP solicitando al FGS resincronizar el Actitud de referencia y proporcionar orientación válida al AP
	Estado	• Inválido, Válido		

• denota valor inicial

Tabla C-4. Variables controladas de FGS para AP

Nombre	Tipo	Rango	Unidades	Interpretación física
Orientación AP				Comandos de orientación para AP
Rollo Guía	Real	[-45.0..45.0]	Grados	Ángulo de giro deseado de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
Paso Guía	Real	[-45.0..45.0]	Grados	Ángulo de inclinación deseado de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
Guía Válido	Booleano	Falso, Verdadero		Indica si la guía de balanceo y la guía de cabeceo son válidas para su uso por parte del AP: · Falso → No válido, no utilizar · Verdadero → Válido

Se hacen los siguientes supuestos ambientales:

- EA-AP-1: La guía de balanceo oscilará entre -45,0° y +45,0°, inclusive.

Justificación: El fabricante de la aeronave especifica esto como el alcance máximo para la guía de balanceo.

- EA-AP-2: La guía de balanceo se establecerá en décimas de grado.

Justificación: Esta resolución es necesaria para lograr un movimiento suave de la aeronave durante el control del volante.

- EA-PFD-3: La guía de inclinación oscilará entre -45,0° y +45,0°, inclusive.

Justificación: El fabricante de la aeronave especifica que este es el alcance máximo para la guía de cabeceo.

- EA-AP-4: La guía de paso se establecerá en décimas de grado.

Justificación: Esta resolución es necesaria para lograr un movimiento suave de la aeronave durante el control del volante.

C.3.4 PANTALLA DE VUELO PRIMARIA .

Los PFD izquierdo y derecho muestran el estado de falla de FD y FGS. Las variables controladas se muestran en la tabla C-5.

Tabla C-5. Variables controladas por FGS para PFD

Nombre	Tipo	Rango	Unidades	Interpretación física
Orientación del Departamento de Finanzas				Comandos de guía para FD
Rollo Guía	Real	[-45.0..45.0]	Grados	Ángulo de giro deseado de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
Paso Guía	Real	[-45.0..45.0]	Grados	Ángulo de inclinación deseado de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
FD encendido	Booleano	Falso, Verdadero		Indicación si se debe visualizar FD: · Falso → No mostrar FD · Verdadero → Mostrar FD
FGS falló	Booleano	Falso, Verdadero		Indicación si la FGS falla: · Falso → FGS está funcionando · Verdadero → FGS ha fallado

Se hacen los siguientes supuestos ambientales:

- EA-PFD-1: La guía de balanceo oscilará entre -45,0° y +45,0°, inclusive.

Justificación: El fabricante de la aeronave especifica esto como el alcance máximo para la guía de balanceo.

- EA-PFD-2: La guía de balanceo se establecerá en décimas de grado.

Justificación: Esta resolución es necesaria para lograr un movimiento suave del FD durante la dirección del volante de control.

- EA-PFD-3: La guía de inclinación oscilará entre -45,0° y +45,0°, inclusive.

Justificación: El fabricante de la aeronave especifica que este es el alcance máximo para la guía de cabeceo.

- EA-PFD-4: La guía de paso se establecerá en décimas de grado.

Justificación: Esta resolución es necesaria para lograr un movimiento suave del FD durante la dirección del volante de control.

C.4. FUNCIONES DEL SISTEMA DE GUÍA DE VUELO .

El FGS compara la actitud de la aeronave medida con una actitud de referencia y genera comandos de guía FD que se muestran como señales visibles en los comandos de guía FD y AP que utiliza la función AP.

Los requisitos de alto nivel para la función FGS son los siguientes:

- REQ-FGS-1: El FGS deberá generar los comandos de guía FD.

Justificación: Una función principal del FGS es generar los comandos de guía FD para el FCS.

- REQ-FGS-2: El FGS deberá establecer la Actitud de Referencia.

Justificación: El FGS establece la actitud de referencia cuando se le solicita sincronizar la actitud de referencia con la actitud actual (ya sea por el piloto, el copiloto o el AP).

- REQ-FGS-3: El FGS deberá generar los comandos de guía AP.

Justificación: Una función principal del FGS es generar los comandos de guía AP para el FCS.

- REQ-FGS-4: La función FGS deberá establecer el estado de FCS fallido.

Fundamento: El FCS se divide entre el FGS y el AP. El estado del AP se indica de forma independiente al piloto y al copiloto en el PFD. Como resultado, el estado del FCS (FCS fallido) lo establece el FGS.

APÉNDICE D—EJEMPLO DE PILOTO AUTOMÁTICO

Este apéndice contiene una especificación de alto nivel para un piloto automático (AP) simplificado. El propósito de este ejemplo es ilustrar cómo se podría asignar una especificación a subsistemas separados, como se analiza en la sección 2.10. Amplía la función AP del sistema de control de vuelo (FCS) especificada en el apéndice B.4.2 en una especificación de subsistema separada que se podría entregar a un subcontratista. Como tal, contiene solo el subconjunto de la información de la especificación FCS relevante para el subsistema AP. Esta especificación se utilizaría como punto de partida para que la completaran el contratista FCS y el subcontratista AP antes del desarrollo del sistema AP. Se necesitaría una descomposición funcional adicional y la definición de los requisitos detallados para completar esta especificación.

D.1. DESCRIPCIÓN GENERAL DEL SISTEMA .

El sistema que se especifica es un AP simple. El AP acepta comandos de guía del AP del sistema de guía de vuelo (FGS) y genera comandos de actuador para mover las superficies de control de la aeronave para rastrear la actitud de referencia mantenida por el FGS. El piloto o copiloto puede activar el AP en cualquier momento en que la aeronave esté en una actitud segura. Antes de activarlo, el AP solicitará al FGS que resincronice la guía del AP con la actitud actual de la aeronave. Mientras el AP está activado, el piloto o copiloto puede iniciar la dirección del volante de control, que desacoplará el AP de los actuadores de la superficie de control (CSA), lo que permitirá al piloto o copiloto volar manualmente la aeronave a una nueva actitud. Una vez completado el control del volante, el AP se volverá a acoplar al CSA y rastreará la nueva actitud de referencia. El piloto o copiloto puede solicitar al AP que se desactive en cualquier momento.

El estado actual del AP se muestra en la pantalla de vuelo principal (PFD) izquierda y derecha.

D.1.1 CONTEXTO DEL SISTEMA .

El contexto operativo del AP se muestra en la figura D-1.

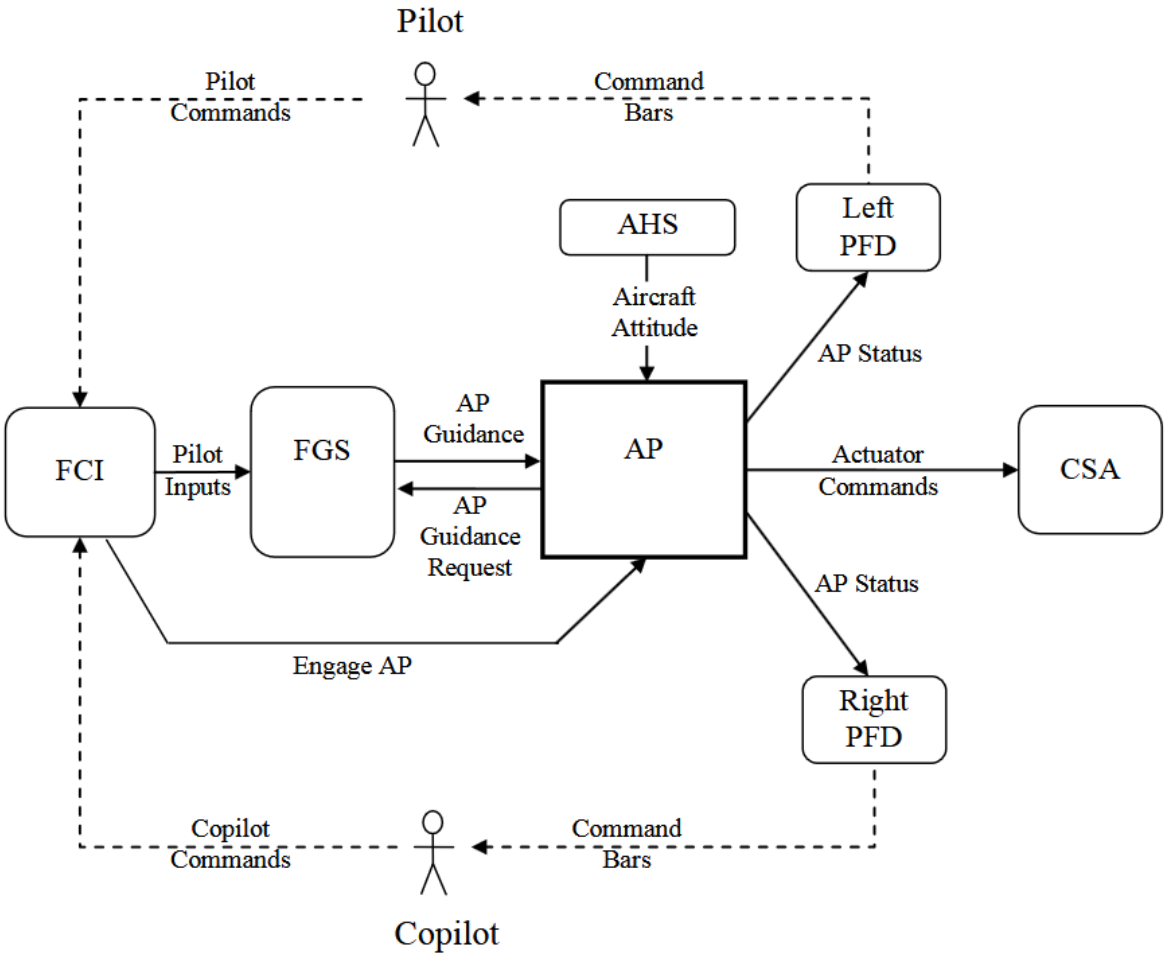


Figura D-1. Diagrama de contexto del sistema de piloto automático

La AP interactúa directamente con

- la interfaz de tripulación de vuelo (FCI), que proporciona el comando Engage AP iniciado por el piloto o el copiloto.
- el sistema de actitud y rumbo (AHS), que proporciona la actitud actual de la aeronave.
- el FGS, que proporciona los comandos de guía AP utilizados por el AP para generar los comandos del actuador. El AP también envía comandos de solicitud de guía AP al FGS para solicitar que la guía AP se sincronice con la actitud actual de la aeronave.
- los PFD derecho e izquierdo, que muestran el estado del AP.

La AP también interactúa indirectamente con

- el piloto y el copiloto, quienes inician el comando Engage AP y ven el estado del AP que se muestra en el PFD.

D.1.2 OBJETIVOS DEL SISTEMA .

Los objetivos de alto nivel de este sistema son los siguientes:

- G1: Generar los comandos del actuador para mantener la aeronave en la actitud proporcionada por el FGS
- G2—Asegurarse de que el AP no ordene maniobras inseguras de aeronaves
- G3—Asegurarse de que las maniobras de la aeronave que puedan causar incomodidad a los pasajeros no sean ordenadas por el AP
- G4—Mantener el costo de fabricación del AP lo más bajo posible
- G5—Mantenga el costo de operación del AP lo más bajo posible

D.2. CONCEPTOS DE OPERACIÓN/AL .

Aquí se proporcionan los conceptos operativos del AP. Estos podrían consistir en casos de uso que describan cómo interactúa el sistema AP con la tripulación de vuelo y otros sistemas, como el FGS, o cualquier otra información que el contratista y el subcontratista acuerden.

D.3. ENTIDADES EXTERNAS/AL .

Las siguientes secciones describen las entidades externas con las que el AP interactúa directamente: la FCI, la AHS, la FGS, el PFD izquierdo y derecho y la CSA.

D.3.1 INTERFAZ CON LA TRIPULACIÓN DE VUELO .

El FCI proporciona las entradas del Piloto y del Copiloto que afectan el comportamiento del AP. Las variables monitoreadas se muestran en la tabla D-1.

Tabla D-1. Variables monitoreadas por AP para FCI

Nombre	Tipo	Rango	Unidades	Interpretación física
Involucrar a AP	Booleano	Falso, Verdadero		Comando para activar o desactivar el AP
	Estado	● Inválido, Válido		

● denota valor inicial

No se hacen suposiciones medioambientales.

D.3.2 SISTEMA DE RUMBO DE ACTITUD .

El AHS proporciona la actitud actual de la aeronave al AP. Las variables monitoreadas se muestran en la tabla D-2.

Tabla D-2. Variables monitoreadas por el piloto automático para AHS

Nombre	Tipo	Rango	Unidades	Interpretación física
Actitud de la aeronave				Actitud actual de la aeronave
Balanceo de aeronave	Real	[-180.0..179.9]	Grados	Ángulo de balanceo actual de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
	Estado	● Inválido, Válido		
Paso de la aeronave	Real	[-180.0..179.9]	Grados	Ángulo de inclinación actual de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
	Estado	● Inválido, Válido		

● denota valor inicial

Se hacen los siguientes supuestos ambientales:

- EA-AHS-1: El balanceo de la aeronave oscilará entre -180,0° y +179,9°, inclusive.

Justificación: El AHS proporcionará un balanceo real de la aeronave, que puede adoptar cualquier valor en todo el rango de movimiento de la aeronave.

- EA-AHS-2: El balanceo de la aeronave se detectará con una precisión de ±0,1°.

Justificación: Esta precisión es necesaria para garantizar que la actitud de la aeronave mostrada se mueva suavemente en el PFD y para garantizar que el FGS calcule los comandos de guía AP y guía FD con la precisión necesaria.

- EA-AHS-3: El ángulo de inclinación de la aeronave oscilará entre -180,0° y +179,9°, inclusive.

Justificación: El AHS proporcionará el paso real de la aeronave, que puede adoptar cualquier valor en todo el rango de movimiento de la aeronave.

- EA-AHS-4: La inclinación de la aeronave se detectará con una precisión de ±0,1°.

Justificación: Esta precisión es necesaria para garantizar que la actitud de la aeronave mostrada se mueva suavemente en el PFD y para garantizar que el FGS calcule los comandos de guía AP y guía FD con la precisión necesaria.

D.3.3 SISTEMA DE GUÍA DE VUELO .

El FGS proporciona los comandos de guía que el AP utiliza para generar comandos de actuador. Las variables monitoreadas y controladas se muestran en las tablas D-3 y D-4, respectivamente.

Tabla D-3. Variables monitoreadas por el piloto automático para FGS

Nombre	Tipo	Rango	Unidades	Interpretación física
Orientación AP				Comandos de orientación para AP
Guía de balanceo	Real	[-45.0..45.0]	Grados	Ángulo de giro deseado de la aeronave: · 0° indica el nivel de las alas · -X° indica X° banco a la izquierda • +X° indica X° banco a la derecha
	Estado	● Inválido, Válido		
Paso Guía	Real	[-45.0..45.0]	Grados	Ángulo de inclinación deseado de la aeronave: · 0° indica vuelo nivelado · -X° indica X° con la nariz hacia arriba • +X° indica X° nariz hacia abajo
	Estado	● Inválido, Válido		
Guía Válido	Booleano	Falso, Verdadero		Indica si la guía de balanceo y la guía de cabeceo son válidas para su uso por parte del AP: · Falso → No válido, no utilizar · Verdadero → Válido
	Estado	● Inválido, Válido		

● denota valor inicial

Tabla D-4. Variables controladas por el piloto automático para FGS

Nombre	Tipo	Rango	Unidades	Interpretación física
Orientación AP Pedido	Booleano	Falso, Verdadero		Solicitar al FGS resincronizar la Actitud de Referencia y proporcionar orientación válida al AP.

Se hacen los siguientes supuestos ambientales:

- EA-FGS-1: La guía de balanceo oscilará entre -45,0° y +45,0°, inclusive.

Justificación: El fabricante de la aeronave especifica esto como el alcance máximo para la guía de balanceo.

- EA-FGS-2: La guía de balanceo se establecerá en décimas de grado.

Justificación: Esta resolución es necesaria para lograr un movimiento suave de la aeronave durante el control del volante.

- EA-FGS-3: La guía de inclinación oscilará entre -45,0° y +45,0°, inclusive.

Justificación: El fabricante de la aeronave especifica que este es el alcance máximo para la guía de cabeceo.

- EA-FGS-4: La guía de tono se establecerá en décimas de grado.

Justificación: Esta resolución es necesaria para lograr un movimiento suave de la aeronave durante el control del volante.

D.3.4 PANTALLA DE VUELO PRIMARIA .

El PFD izquierdo y derecho muestra el estado del AP. Las variables controladas se muestran en la tabla D-5.

Tabla D-5. Variables controladas por piloto automático para PFD

Nombre	Tipo	Rango	Unidades	Interpretación física
Estado de AP	Enumerado	Fallido, Apagado, En		Estado de la función AP: · Falló → La función AP ha fallado · Apagado → La función AP está desactivada · En → La función AP está activada
	Estado latente	0,1	segundo	
	Tolerancia	N / A		

N/A = no aplicable

No se hacen suposiciones medioambientales.

D.3.5 ACTUADORES DE SUPERFICIE DE CONTROL .

El CSA posiciona las superficies de control de la aeronave, en función de los comandos de actuador generados por el FCS para mantener la aeronave en la actitud de referencia. Las variables controladas se muestran en la tabla D-6.

Tabla D-6. Variables controladas por piloto automático para CSA

Nombre	Tipo	Rango	Unidades	Interpretación física
Comandos del actuador				Tasas de actuadores comandados
Rollo Solenoide	Real	[-20.0..20.0]	grado superficie por segundo	Velocidad comandada del actuador de vuelco: · 0 → sin cambios en las superficies de control · -X → ala izquierda abajo • +X → ala derecha hacia abajo
Paso Solenoide	Real	[-20.0..20.0]	grado superficie por segundo	Velocidad comandada del actuador de paso: · 0 → sin cambios en las superficies de control · -X → nariz hacia arriba • +X → nariz hacia abajo
AP En	Booleano	Falso, Verdadero		Indicación de si la función AP está activada y se utilizarán los comandos del actuador

Se hacen los siguientes supuestos ambientales:

- EA-CSA-1: La velocidad del actuador de alabeo oscilará entre -20,0 y +20,0° superficie/segundo, inclusive.

Fundamento: el fabricante de la aeronave especifica este valor como el rango necesario para controlar adecuadamente la aeronave.

- EA-CSA-2: La velocidad del actuador de alabeo se establecerá con una resolución de 0,1° superficie/segundo.

Fundamento: un análisis de controlabilidad muestra que es necesaria una resolución de 0,1° superficie/segundo para mantener el control de la aeronave.

- EA-CSA-3: La velocidad del actuador de cabeceo oscilará entre -20,0 y +20,0° superficie/segundo, inclusive.

Fundamento: el fabricante de la aeronave especifica este valor como el rango necesario para controlar adecuadamente la aeronave.

- EA-CSA-4: La velocidad del actuador de cabeceo se establecerá con una resolución de 0,1° superficie/segundo.

Justificación: Un análisis de controlabilidad muestra que es necesaria una resolución de 0,1° de superficie por segundo para mantener el control de la aeronave.

D.4. FUNCIONES DEL PILOTO AUTOMÁTICO .

El AP genera comandos de actuador a partir de los comandos de guía AP proporcionados por la función FG. Los comandos de actuador son utilizados por CSA para volar la aeronave a la actitud especificada por los comandos de guía AP.

Los requisitos de alto nivel para el sistema AP son los siguientes:

- REQ-APS-1: El sistema AP deberá generar los comandos del actuador.

Justificación: La función principal del AP es generar los comandos del actuador para el FCS.

- REQ-APS-2: El AP debe configurar el estado del AP para indicar su estado.

Fundamento: El AP es responsable de proporcionar su estado actual al FCS.

- REQ-APS-3: El AP deberá generar la Solicitud de Orientación de AP.

Justificación: La solicitud de orientación AP requiere que el FGS sincronice su actitud de referencia para producir un comando de orientación AP aceptable.

Revisiones de la traducción al español

Fecha	Revisión	Modificaciones	Autor
4/01/25	01	Se crea el documento Se crean los estilos para aplicar formato Se aplica formato a la Sección 2.1 – Traducción revisada. Se aplica formato a la Sección 2.2 – Traducción revisada. Se aplica formato a la Sección 2.3 – Traducción revisada. Se aplica formato a la Sección 2.4 – Traducción revisada. Se aplica formato a la Sección 2.5 – Traducción revisada. Se aplica formato a la Sección 2.6 – Traducción revisada. Se aplica formato a la Sección 2.7 – Traducción revisada. Se aplica formato a la Sección 2.8 – Traducción revisada.	GS
19/01/25	02	Se aplica formato a la Sección 2.9 Se aplica formato a la Sección 2.10 Se aplica formato a la Sección 2.11	GS
21/01/25	03	Se aplica formato a Anexo A	GS
30/01/25	03	– Traducción revisada Anexo A	GS
31/01/25	04	Se aplica formato al Anexo B Se aplica formato al Anexo C Se aplica formato al Anexo D	GS