# **PRUEBAS DE INTEGRACIÓN**

Tabla de contenidos

|  |  |  |
| --- | --- | --- |
| 1 | **Prueba de Integración de Locomoción y Navegación** | …1 |
| 2 | **Prueba de Integración de Interfaz de Usuario y Sistemas de Control Autónomo.** | …2 |
| 3 | **Prueba de Integración de Adaptabilidad y Eficiencia Energética.** | …3 |

1. **Prueba de Integración de comunicación en tiempo real de sensores**

|  |  |
| --- | --- |
| **PRUEBA DE INTEGRACIÓN EN TIEMPO REAL** | |
| **Requerimientos**   * La plataforma debe permitir la comunicación en tiempo real de los sensores. | |
| **Tipo de prueba** | Integración |
| **Hardware requerido** | * Raspberry Pi * Arduino Nano * Modulo CAN * Sensores |
| **Software requerido** | * Sistema operativo para Raspberry Pi (ubuntu) * IDE de Arduino * Librerías de comunicación (Serial, I2C, etc.) |
| **Objetivo** | * Verificar la comunicación en tiempo real de los sensores entre el Arduino Nano y la Raspberry Pi. |
| Descripción | |
| **Procedimiento** | * Conectar la Raspberry Pi a Arduino Nano a través de un enlace de comunicación * Conectar los Arduinos Nano con la comunicación CAN. * Conectar sensores al Arduino Nano. * Configurar el código en la Raspberry Pi para enviar una señal de inicio a Arduino Nano. * Configurar el código en Arduino Nano para solicitar datos en tiempo real de los sensores y enviarlos a Raspberry Pi. * Ejecutar el código y verificar que los datos de los sensores se muestren en la Raspberry Pi en tiempo real. |
| **Resultado esperado** | * La Raspberry Pi recibe y muestra los datos de los sensores en tiempo real. * No hay pérdida de datos ni retrasos significativos en la transmisión. |
| **Resultado obtenido** |  |
| **Comentarios** |  |

1. **Prueba de Integración : Cámara de Visión Nocturna, Lidar 2D y Micrófono**

|  |  |
| --- | --- |
| **PRUEBA DE INTEGRACIÓN** | |
| **Requerimientos**   * La red debe ser capaz de detectar objetos mediante una cámara de visión nocturna. * La red debe ser capaz de detectar objetos mediante un LiDar 2D. * La red debe generar comandos en respuesta a una entrada de voz. | |
| **Tipo de prueba** | Integración |
| **Hardware requerido** | * Raspberry Pi * Cámara de Visión Nocturna * LiDar 2D * Micrófono USB |
| **Software requerido** | * Sistema operativo para Raspberry Pi (Ubuntu) * Librerías de visión, procesamiento de imagen, LiDar y reconocimiento de voz. |
| **Objetivo** | * Verificar la correcta integración y funcionamiento de la cámara de visión nocturna, LiDar 2D, y micrófono con la Raspberry Pi. |
| Descripción | |
| **Procedimiento** | * Conectar la cámara de visión nocturna, LiDar 2D y micrófono USB a la Raspberry Pi. * Configurar el código en la Raspberry Pi para manejar cada uno de los dispositivos. * Enviar una solicitud de detección de objetos a la cámara de visión nocturna y al LiDar. * Activar el micrófono y enviar una entrada de voz. * Verificar que los datos de la cámara, LiDar y micrófono se procesen y se muestren correctamente. |
| **Resultado esperado** | * Las imágenes capturadas por la cámara de visión nocturna se muestran correctamente. * Los datos del LiDar se procesan y se muestran correctamente. * Los comandos generados en respuesta a la entrada de voz se muestran correctamente. |
| **Resultado obtenido** |  |
| **Comentarios** |  |

1. **Prueba de Integración de Sensores de Contacto, Sensor Inercial y GPS**

|  |  |
| --- | --- |
| **PRUEBA DE INTEGRACIÓN** | |
| **Requerimientos**   * La red debe ser capaz de medir el esfuerzo aplicado en cada pata de la plataforma cuadrúpeda. * La red debe ser capaz de medir la orientación de la plataforma utilizando un sensor inercial. * La red debe ser capaz de estimar su posición utilizando un GPS NEO6M. | |
| **Tipo de prueba** | Integración |
| **Hardware requerido** | * Raspberry Pi * Arduino Nano Master * Arduino Nano 1 * Sensor de Esfuerzo * Módulos CAN * Sensor Inercial * GPS NEO6M |
| **Software requerido** | * Sistema operativo para Raspberry Pi (Ubuntu) * IDE de Arduino * Librerías de sensores y comunicación. |
| **Objetivo** | * Verificar la correcta integración y funcionamiento de los sensores de contacto, sensor inercial y GPS con la Raspberry Pi. |
| Descripción | |
| **Procedimiento** | * Conectar el sensor de esfuerzo, sensor inercial y GPS a los Arduino Nano. * Conectar los Arduino Nano al Arduino Nano Master a través de los módulos CAN a la Raspberry Pi. * Configurar el código en la Raspberry Pi para enviar solicitudes de medición a los sensores. * Ejecutar el código y verificar que los datos de los sensores se transmitan correctamente a la Raspberry Pi. |
| **Resultado esperado** | * Los datos del sensor de esfuerzo se muestran correctamente. * Los datos del sensor inercial se muestran correctamente. * Los datos del GPS se muestran correctamente. |
| **Resultado obtenido** |  |
| **Comentarios** |  |

1. **Prueba de Sistema Completa con Dos Esclavos y un Maestro**

|  |  |
| --- | --- |
| **PRUEBA DE INTEGRACIÓN** | |
| **Requerimientos**   * La red debe permitir la comunicación entre todos los dispositivos y sensores. * La plataforma debe permitir la comunicación entre múltiples Arduinos. | |
| **Tipo de prueba** | Integración |
| **Hardware requerido** | * Raspberry Pi * Arduino Nano Master * Arduino Nano 1 (Esclavo) * Arduino Nano 2 (Esclavo) * Otros sensores conectados a los Arduinos * Módulos CAN |
| **Software requerido** | * Sistema operativo para Raspberry Pi (Ubuntu) * IDE de Arduino * Librerías de comunicación y sensores. |
| **Objetivo** | * Verificar el funcionamiento del sistema completo con la Raspberry Pi, un Arduino Nano Master, tres módulos CAN y dos Arduino Nano esclavos. |
| Descripción | |
| **Procedimiento** | * Ejecutar el código en la Raspberry Pi para coordinar la comunicación entre todos los dispositivos. * Ejecutar el código en los Arduino Nano para manejar las solicitudes y respuestas de los sensores. * Verificar la comunicación entre los Arduino Nano esclavos y el maestro. * Ejecutar el código y verificar que todos los datos se muestren correctamente. |
| **Resultado esperado** | * Todos los sensores envían datos correctos a la Raspberry Pi. * La comunicación entre todos los dispositivos es correcta y sin errores. * Los datos de medición de voltaje y corriente se muestran correctamente. |
| **Resultado obtenido** |  |
| **Comentarios** |  |