

UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE GUAYAQUIL

CARRERA DE INGENIERÍA ELECTRÓNICA

**TRABAJO DE TITULACIÓN PREVIO A LA
OBTENCIÓN DEL TÍTULO DE INGENIERO ELECTRÓNICO**

PROYECTO TÉCNICO:

**“DISEÑO DE UN CONTROLADOR PID DIFUSO PARA
UN ROBOT AUTO BALANCEADO CON TECNOLOGÍA
PITSCO TETRIX PRIME Y MYRIO”**

AUTORES:

JONATHAN MIGUEL SARMIENTO CEDEÑO

JOHN STIVEN CEDEÑO CRUZ

TUTOR:

Ing. Byron Xavier Lima Cedillo MSc.

GUAYAQUIL – ECUADOR

2020

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA

Nosotros, Jonathan Miguel Sarmiento Cedeño y John Stiven Cedeño Cruz autorizamos a la **Universidad Politécnica Salesiana** la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, se declara que los conceptos y análisis desarrollados y conclusiones del presente trabajo son de exclusiva responsabilidad del autor.



Jonathan Miguel Sarmiento Cedeño
Cédula: 0920636446



John Stiven Cedeño Cruz
Cédula: 0940974173

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR

Nosotros, **Jonathan Miguel Sarmiento Cedeño**, con documento de identificación N° 0920636446 y **John Stiven Cedeño Cruz**, con documento de identificación N° 0940974173, manifestamos nuestra voluntad y ceder a la **UNIVERSIDAD POLITÉCNICA SALESIANA** la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de grado titulado: **“DISEÑO DE UN CONTROLADOR PID DIFUSO PARA UN ROBOT AUTO BALANCEADO CON TECNOLOGÍA PITSCO TETRIX PRIME Y MYRIO”** mismo que ha sido desarrollado para optar por el título de **INGENIERO ELECTRÓNICO**, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos antes cedidos.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscrito este documento en el momento que se realiza la entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.



Jonathan Miguel Sarmiento Cedeño
Cédula: 0920636446

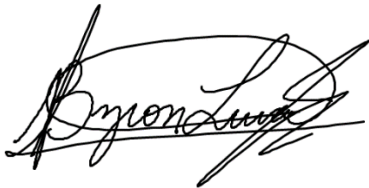


John Stiven Cedeño Cruz
Cédula: 0940974173

CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN

Yo declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación “**DISEÑO DE UN CONTROLADOR PID DIFUSO PARA UN ROBOT AUTO BALANCEADO CON TECNOLOGÍA PITSCO TETRIX PRIME Y MYRIO**” con resolución de aprobación de Consejo de Carrera N.º RESOLUCION realizado por los estudiantes **Jonathan Miguel Sarmiento Cedeño**, con documento de identificación N0920636446 y **John Steven Cedeño Cruz**, con documento de identificación N° 0940974173, obteniendo un producto que cumple con los objetivos del diseño de aprobación, informe final y demás requisitos estipulados por la Universidad Politécnica Salesiana, para ser considerados como trabajo final de titulación.

Guayaquil, octubre del 2020



Ing. Byron Xavier Lima Cedillo MSc.

Docente

c.c. 0921971768

DEDICATORIA

Quiero dedicar esta tesis a mi esposa Ivonne que con su amor, paciencia, esfuerzo y dedicación supo cómo guiar para poder escoger la carrera Universitaria de Ingeniería Electrónica y así poder llegar a cumplir hoy un sueño más en mi vida.

A mis padres Guillermo y Daysi por el apoyo a lo largo de toda mi carrera como estudiante y por inculcar en mí el ejemplo del esfuerzo, superación y valentía.

A mis hermanos y familia que me brindaron todo su apoyo cuando más lo necesitaba, un consejo o palabra de aliento que hicieron de mí una mejor persona.

A handwritten signature in black ink that reads "Jonathan S". The signature is stylized with a large, looping flourish around the letter 'S'.

Jonathan Miguel Sarmiento Cedeño

DEDICATORIA

Durante mi recorrido por la universidad me di cuenta que hay muchas cosas para lo que soy bueno, habilidades, destrezas que jamás pensé que podría realizar, pero lo más importante que pude descubrir que por más que busque todas las repuestas solo, siempre obtuve mejores resultados trabajando en equipo, compañeros, amigos y familiares que aparecieron en el momento justo, cuando creía que toda la presión, estrés y trabajo me harían abandonar este camino, aparecía la ayuda de quienes menos me los esperaba, entonces entendí que no estaba solo en este camino y comprendí que la ayuda no llega cuando la pida, llega en el momento que la necesite.

Por eso quiero dedicar mi tesis principalmente a mis padres John y Janeth, ellos me hicieron dar cuenta lo importante que es esforzarse para lograr tus metas, a toda mi familia y amigos, a mi pareja y a su familia, gracias a todos ustedes, soy el profesional que ahora soy.



John Steven Cedeño Cruz

AGRADECIMIENTO

Agradezco a Dios por permitirme alcanzar este objetivo de alcanzar el título de Ingeniería Electrónica.

A mis padres por su dedicación y esfuerzo con su trabajo que de una u otra forma me apoyaron económicamente.

A mi esposa que estuvo en cada momento tanto de felicidad como los momentos complicados en mi proceso como estudiante y así llegar hacer un profesional.

A mis compañeros de clases que estuvieron luchando en cada materia y así poder superarnos y lograr este objetivo anhelado.

Agradezco a todos los docentes como el Ing. Lima, Ing. Neira, Ing. Barcia, Ing. Mónica Miranda y el Ing. García que nos brindó sus conocimientos, experiencias y prácticas que gracias a eso nos formó como verdaderos profesionales.

A handwritten signature in black ink, reading "Jonathan S." with a stylized flourish at the end.

Jonathan Miguel Sarmiento Cedeño

AGRADECIMIENTO

En el término de esta etapa de mi vida quiero agradecer primero a Dios, a la Universidad Politécnica Salesiana por haberme aceptado ser parte de ella, así como también a los diferentes docentes que brindaron sus conocimientos día a día,

Agradezco también a mi tutor de tesis el Ing. Byron Lima por su experiencia brindada, gracias a eso pudimos terminar este proyecto.

Agradezco también a la empresa MEDILABOR S.A. que, a pesar del horario me dieron oportunidad de trabajar y poder asistir a mis clases, al Ing. Javier Quintana y a la Lcda. Sara Valle que depositaron su confianza en mí para poder terminar mis estudios.



John Steven Cedeño Cruz

RESUMEN

AÑO	ALUMNOS	DIRECTOR DE PROYECTO TÉCNICO	TEMA DE PROYECTO TÉCNICO
2020	Jonathan Miguel Sarmiento Cedeño John Steven Cedeño Cruz	ING. Byron Xavier Lima Cedillo , MSc.	Diseño de un controlador PID Difuso para un Robot Auto Balanceado con tecnología Pitsco Tetrax Prime y myRio

Actualmente la tecnología en el campo de la ingeniería electrónica, automatización y control tiene una evolución que nos ha exigido cambiar y adaptarnos para tener mejores resultados en los procesos de control.

Este presente consiste en un módulo robótico utilizando la tecnología de Pitsco Tetrax prime para el ensamble de un robot auto balanceado el cual tenga como núcleo principal un sistema embebido ni myRio permitiendo programar mediante labview un sistema de control pid difuso con la ayuda de las herramientas de creación de sistemas borrosos permitiendo desarrollar destrezas a los estudiantes de la carrera de electrónica y automatización.

El objetivo principal es que el estudiante tenga una guía de prácticas que le beneficie para el entendimiento de sistemas de control y así plasmar el conocimiento obtenido en materias como teoría de control, robótica, microcontroladores, informática industrial y electivas.

Palabras claves: Balance, Auto Balance, Pid, difuso, myRio, LabView, Pitsco, Tetrax

ABSTRACT

YEAR	STUDENTS	DIRECTOR OF TECHNICAL PROJECT	TECHNICAL PROJECT THEME
2020	Jonathan Miguel Sarmiento Cedeño John Steven Cedeño Cruz	ING. Byron Xavier Lima Cedillo , MSc.	Design of a Diffuse PID controller for a Self-Balanced Robot with Pitsco Tetrax Prime and myRio technology

Currently, the technology in the field of electronic engineering, automation and control has an evolution that has made us change and adapt to have better results in control processes.

This present consists of a robotic module using Pitsco Tetrax prime technology for the assembly of a self-balancing robot whose main core is an embedded ni myrio system allowing to program a fuzzy pid control system through labview with the help of the tools of creation of fuzzy systems allowing students to develop skills in electronics and automation.

The main objective is for the student to have a practical guide that will benefit him / her for the understanding of control systems and thus capture the knowledge obtained in subjects such as control theory, robotics, microcontrollers, industrial computing and electives.

Keywords: Balance, Auto Balance, Pid, diffuse, myRio, LabView, Pitsco, Tetrax

ÍNDICE GENERAL

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA	II
CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR	III
CERTIFICADO DE DIRECCIÓN DE TRABAJO DE TITULACIÓN	IV
DEDICATORIA	V
DEDICATORIA	VI
AGRADECIMIENTO	VII
AGRADECIMIENTO	VIII
RESUMEN	IX
ABSTRACT	X
ÍNDICE GENERAL	XI
ÍNDICE DE FIGURAS	XV
ÍNDICE DE TABLAS	XXI
INTRODUCCIÓN	1
1. EL PROBLEMA	2
1.1. Antecedentes	2
1.2. Importancia y Alcances	2
1.3. Delimitación	2
1.3.1. Temporal	2
1.3.2. Espacial	2
1.3.3. Académica	3
1.4. Innovación	3
1.5. Objetivos	3
1.5.1. Objetivo general	3
1.5.2. Objetivos específicos	3
2. FUNDAMENTOS TEÓRICOS	5
2.1. Sistemas de Control PID	5
2.2. Sistemas de control Difusos	6
2.3. Defusificación	6
2.4. Fusificación	6
2.5. Inferencia	7
2.6. Base de conocimiento	7
2.7. Función de pertenencia	7
2.8. Función triangular	8
2.9. Función trapezoidal	8

2.10. Función Gamma.....	9
2.11. Función Gaussiana.....	9
2.12. Variables lingüísticas.....	10
2.13. LabView.....	10
2.14. Ni myRio 1900.....	10
2.15. Pitsco Tetrax Prime.....	11
2.16. Sensor de giroscopio.....	12
2.17. Controladora de motores.....	13
2.18. Motor de corriente directa.....	13
2.19. Codificadores Rotativo (Encoder).....	14
2.19.1. Encoder Incrementales.....	15
2.19.2. Encoder absolutos.....	16
2.20. Max sonar Xr.....	17
2.21. Péndulo invertido.....	17
2.22. Robot de dos ruedas basado en péndulo invertido.....	18
2.23. App inventor.....	19
3. MARCO METODOLÓGICO.....	20
3.1. Diseño del robot.....	20
3.2. Diagrama de control.....	20
3.3. Esquema del robot.....	21
3.4. Conexiones del prototipo.....	22
3.4.1. Conexiones de la instrumentación.....	22
3.4.2. Conexiones de la comunicación.....	27
3.4.3. Conexiones de la potencia y alimentación.....	29
3.5. Controlador pid.....	31
3.6. Control PID Difuso.....	32
4. MANUAL DE PRACTICAS.....	36
Práctica 1: Lectura de acelerómetros con software de instrumentación virtual LabView.	36
Práctica 2: Scada mediante wifi de acelerómetros con software de instrumentación virtual LabView.....	38
Práctica 3: Conversión analógica con sensor ultrasónico.....	40
Práctica 4: Scada mediante wifi de lecturas analógica con sensor ultrasónico.....	42
Práctica 5: PID robot auto balanceado.....	44
Práctica 6: PID robot auto balanceado evitando obstáculos con sensores ultrasónicos.....	46
Práctica 7: PID Difuso robot auto balance.....	48

Práctica 8: Robot auto balanceado radio controlado mediante comunicación bluetooth. 50	50
Práctica 9: Robot auto balanceado radio controlado mediante comunicación wifi. 52	52
Práctica 10: Robot auto balanceado radio controlado mediante aplicativo en aplicación App Inventor.	53
5. RESULTADOS	55
5.1. RESULTADOS DE PRÁCTICAS	56
5.1.1. Resultados en práctica 1 - Lectura de acelerómetros con software de instrumentación virtual LabView.	56
5.1.2. Resultados en práctica 2 - Scada mediante wifi de acelerómetros con software de instrumentación virtual LabView	57
5.1.3. Resultados en práctica 3 - Conversión analógica con sensor ultrasónico.....	58
5.1.4. Resultados en práctica 4 - Scada mediante wifi de lecturas analógica con sensor ultrasónico	58
5.1.5. Resultados en práctica 5 - PID robot auto balanceado.....	59
5.1.6. Resultados en práctica 6 - PID robot auto balanceado evitando obstáculos con sensores ultrasónicos.....	60
5.1.7. Resultados en práctica 7 - PID Difuso robot auto balance.....	61
5.1.8. Resultados en práctica 8 - Robot auto balanceado radio controlado mediante comunicación bluetooth.....	63
5.1.9. Resultados en práctica 9 - Robot auto balanceado radio controlado mediante comunicación wifi	66
5.1.10. Resultados en práctica 10 - Robot auto balanceado radio controlado mediante aplicativo en aplicación App Inventor	68
5.2. COMPARACION ENTRE CONTROLADOR PID CONVENCIONAL Y CONTROLADOR PID DIFUSO.....	72
CONCLUSIONES.....	74
RECOMENDACIONES	75
REFERENCIAS BIBLIOGRAFICAS	76
ANEXOS.....	80
Anexo 1 Dimensiones de la tarjeta MyRio.....	80
Anexo 2 Lista de materiales.....	82
Anexo 3 Ensamble del prototipo	85
Anexo 4. Solución de prácticas	92
Anexo 4.0 Introducción a prácticas.....	92
Anexo 4.1 Solución de práctica 1.....	92
Anexo 4.2 Solución de práctica 2.....	98
Anexo 4.3 Solución de práctica 3.....	104
Anexo 4.4 Solución de práctica 4.....	109
Anexo 4.5 Solución de práctica 5.....	114

Anexo 4.6 Solución de práctica 6.....	126
Anexo 4.7 Solución de práctica 7.....	138
Anexo 4.8 Solución de práctica 8.....	148
Anexo 4.9 Solución de práctica 9.....	159
Anexo 4.10 Solución de práctica 10	167
Anexo 5 Configuración de red inalámbrica.....	178

ÍNDICE DE FIGURAS

Figura 1: Diagrama Pid.....	5
Figura 2: Diagrama de control difuso	6
Figura 3: Funciones de pertenencia	7
Figura 4: Funciones triangular.....	8
Figura 5: Funciones trapezoidal.....	8
Figura 6: Funciones Gamma	9
Figura 7: Funciones Gaussiana	9
Figura 8: LabView	10
Figura 9: Ni MyRio 1900.....	10
Figura 10: Ni MyRio 1900 Puerto A y B	11
Figura 11: Ni MyRio 1900 Puerto C.....	11
Figura 12: Pitsco Tetrax prime.....	12
Figura 13: Sensor de giroscopio Pitsco Tetrax.....	12
Figura 14: Tarjeta controladora de motores	13
Figura 15: Motor de corriente continua	14
Figura 16: Funcionamiento de disco rotativo del Encoder	14
Figura 17: Disposición de un Encoder incremental.....	15
Figura 18: Salidas A y B de un Encoder y valor (Q) del Biestable D según su sentido de giro	16
Figura 19: Codificador absoluto binario de 4bits. (a) Construcción Interna. (b) Señales de salida	17
Figura 20: Péndulo invertido.....	18
Figura 21: Kazuo Yamafuji Robot.....	19
Figura 22: Prototipo Pitsco Tetrax como balance	20
Figura 23: Diagrama de control.....	21
Figura 24: Esquema del prototipo.....	21
Figura 25: Modulo giroscopio conectado al ni myRio.....	22
Figura 26: Modulo giroscopio conectado al ni myRio.....	22
Figura 27: Modulo ultrasónico conectado al ni myRio.....	23
Figura 28: Modulo giroscopio conectado al ni myRio.....	24
Figura 29: Encoder conectado al ni myRio.....	25
Figura 30: Señal de Encoder en osciloscopio	25
Figura 31: Configuración de Encoder en LabView	26
Figura 32: Esquema de ni myRio con bluetooth	27
Figura 33: Configuración del uart en LabView	27
Figura 34: Aplicativo creado en app inventor	28
Figura 35: Conexión de la tarjeta ni myRio con la tarjeta de potencia.....	29
Figura 36: Configuración en LabView para controladora de motores.....	29
Figura 37: Conexión de la alimentación.....	31
Figura 38: Lazo cerrado del robot auto balance	31
Figura 39: Herramienta para creación de variables, práctica 7.	32
Figura 40: Configuración de rango en variable proporcional, práctica 7.	33
Figura 41: Configuración de rango en variable integral, práctica 7.	33
Figura 42: Configuración de rango en variable derivativa, práctica 7.	34
Figura 43: Configuración de rango en variable derivativa, práctica 7.	34
Figura 44: Configuración de reglas difusas, práctica 7.	35
Figura 45: PID FUZZY en LabView	35
Figura 46: Diseño propuesto para PRÁCTICA 1	37
Figura 47: Diseño propuesto para PRÁCTICA 2	39

Figura 48. Diseño propuesto para PRÁCTICA 1	41
Figura 49. Diseño propuesto para PRÁCTICA 1	43
Figura 50. Diseño propuesto para PRÁCTICA 5	44
Figura 51. Diseño propuesto para PRÁCTICA 6	46
Figura 52. Diseño propuesto para PRÁCTICA 7	48
Figura 53. Diseño propuesto para PRÁCTICA 8	50
Figura 54. Diseño propuesto para PRÁCTICA 9	52
Figura 55. Diseño propuesto para PRÁCTICA 10	53
Figura 56. Diseño propuesto aplicativo para PRÁCTICA 10.....	54
Figura 57. Resultados Practica 1 – compilación	56
Figura 58. Resultados Practica 1 – movimiento del prototipo de forma manual para lectura de valores en ejes x y z.....	56
Figura 59. Resultados Practica 2 – compilación	57
Figura 60. Resultados Practica 2 – movimiento del prototipo de forma manual para lectura de valores en ejes x y z.....	57
Figura 61. Resultados Practica 3 – adquisición de sensores ultrasónico.....	58
Figura 62. Resultados Practica 4 – adquisición de sensores ultrasónico.....	58
Figura 63. Resultados Practica 5 PID robot auto balanceado.	59
Figura 64. Resultados Practica 5 PID robot auto balanceado.	59
Figura 65. Resultados Practica 6 – PID robot auto balanceado evitando obstáculos con sensores ultrasónicos.....	60
Figura 66. Resultados Practica 6 – movimientos del robot auto balanceado evitando obstáculos con sensores ultrasónicos.....	60
Figura 67. Resultados Practica 6 – PID Difuso robot auto balance.	61
Figura 68. Resultados Practica 7 – Simulación difuso	61
Figura 69. Resultados Practica 7 – Estabilización del robot	62
Figura 70. Resultados Practica 8 – Adquisición del valor de distancia.....	63
Figura 71. Resultados Practica 8 – Panel frontal.....	63
Figura 72. Resultados Practica 8 – movimiento de avanzar	64
Figura 73. Resultados Practica 8 – movimiento de retroceder	64
Figura 74. Resultados Practica 8 – movimiento de giro derecha	64
Figura 75. Resultados Practica 8 – movimiento de giro izquierda.....	65
Figura 76. Resultados Practica 9 – Panel frontal.....	66
Figura 77. Resultados Practica 8 – movimiento de avanzar	66
Figura 78. Resultados Practica 9 – movimiento de retroceder	67
Figura 79. Resultados Practica 9 – movimiento de giro derecha	67
Figura 80. Resultados Practica 9 – movimiento de giro izquierda.....	67
Figura 81. Resultados Practica 8 – Adquisición del valor de distancia.....	68
Figura 82. Resultados Practica 8 – Panel frontal.....	69
Figura 83. Resultados Practica 8 – movimiento de avanzar	69
Figura 84. Resultados Practica 10 – movimiento de retroceder	70
Figura 85. Resultados Practica 10 – movimiento de giro derecha	70
Figura 86. Resultados Practica 10 – movimiento de giro izquierda	71
Figura 87. Controlador pid difuso.....	72
Figura 88. Controlador pid clásico	72
Figura 89. Controlador pid difuso vs pid clásico	73
Figura 90. Anexo 1 Dimensiones del ni myRio – vista frontal y posterior.....	80
Figura 91. Anexo 1 Dimensiones del ni myRio – vista lateral	81
Figura 92. Anexo 2 – Lista de materiales parte 1	82
Figura 93. Anexo 2 – Lista de materiales parte 2	82
Figura 94. Anexo 2 – Lista de materiales parte 3	83
Figura 95. Anexo 2 – Lista de materiales parte 4	83

Figura 96. Anexo 2 – Lista de materiales parte 5	84
Figura 97. Anexo 2 – Lista de materiales parte 6	84
Figura 98. Anexo 3 – Ensamble de parte 1 paso 1	85
Figura 99. Anexo 3 – Ensamble de parte 1 paso 2	86
Figura 100. Anexo 3 – Ensamble de parte 1 paso 3	88
Figura 101. Anexo 3 – Ensamble de parte 1 paso 4	88
Figura 102. Anexo 3 – Ensamble de parte 2 paso 1	89
Figura 103. Anexo 3 – Ensamble de parte 2 paso 2	89
Figura 104. Anexo 3 – Ensamble de parte 3 paso 1	90
Figura 105. Anexo 3 – Ensamble de parte 3 paso 2	90
Figura 106. Anexo 3 – Ensamble de parte 5 paso 1	91
Figura 107. Anexo 3 – Ensamble final de prototipo.....	91
Figura 108. Ejecución de LabView como compatible con ni myRio práctica 1	92
Figura 109. Selección de proyecto myRio práctica 1	93
Figura 110. Configuración de ruta práctica 1.....	93
Figura 111. Administrador de proyectos, selección práctica 1.....	93
Figura 112. Configuración de i2c para myRio de práctica 1.....	94
Figura 113. Bloque de variables i2c, práctica 1	94
Figura 114. Configuración de entradas digitales, práctica 1.	95
Figura 115. Conexión a lazo de selección, práctica 1	95
Figura 116. Lazo de selección con bloque i2c, práctica 1	96
Figura 117. Bundle con datos para gráfica, práctica 1	96
Figura 118. Interfaz de práctica 1.....	97
Figura 119. Ejecución de LabView como compatible con ni myRio práctica 2.....	98
Figura 120. Nuevo Proyecto, práctica 2.....	98
Figura 121. Configuración de ruta, práctica 2	99
Figura 122. Administrador de proyectos, selección práctica 2.....	99
Figura 123. Configuración de i2c para myRio de práctica 2.....	100
Figura 124. Bloque de variables i2c, práctica 2	100
Figura 125. Configuración de entradas digitales, práctica 2.	101
Figura 126. Conexión a lazo de selección, práctica 2.....	101
Figura 127. Lazo de selección con bloque i2c, práctica 2	102
Figura 128.	102
Figura 129. Interfaz practica 2.....	103
Figura 130. Ejecución de LabView como compatible con ni myRio práctica 3.....	104
Figura 131. Nuevo Proyecto, práctica 3.....	104
Figura 132. Configuración de ruta, práctica 3.	105
Figura 133. Administrador de proyectos, selección práctica 3.....	105
Figura 134. Configuración de entradas analógicas, práctica 3.....	106
Figura 135. Main Loop con entradas analógicas, práctica 3.....	107
Figura 136. Bloque unión señales practica 3	107
Figura 137. Interfaz practica 3.....	108
Figura 138. Ejecución de LabView como compatible con ni myRio práctica 4.....	109
Figura 139. Nuevo Proyecto, práctica 4	109
Figura 140. Configuración de ruta, práctica 4	110
Figura 141. Administrador de proyectos, selección práctica 4.....	110
Figura 142. Configuración de entradas analógicas, práctica 4.....	111
Figura 143. Main Loop con entradas analógicas, práctica 4	111
Figura 144. Bloque unión señales practica 4.....	112
Figura 145. Interfaz practica 4.....	112
Figura 146. Ejecución de LabView como compatible con ni myRio práctica 5.....	114
Figura 147. Nuevo Proyecto, práctica 5.....	114

Figura 148. Configuración de ruta, práctica 5	115
Figura 149. Administrador de proyectos, selección práctica 5.....	115
Figura 150. Configuración de i2c para myRio de práctica 5.....	116
Figura 151. Bloque de variables i2c, práctica 5	116
Figura 152. Configuración de entradas digitales, práctica 5.....	117
Figura 153. Conexión a lazo de selección, práctica 5.....	117
Figura 154. Lazo de selección con bloque i2c, práctica 5.....	118
Figura 155. Bundle con datos para gráfica, práctica 5.....	118
Figura 156. Ingreso de entradas analógicas, práctica 5.....	119
Figura 157. Conexión de salidas analógicas práctica 5.....	119
Figura 158. Obtención de datos de desplazamiento práctica 5.....	120
Figura 159. Ingreso de dos salidas analógicas, práctica 5.....	120
Figura 160. Señal de salida atenuada, practica 5.....	121
Figura 161. Configuración de salidas digitales, practica 5.....	121
Figura 162. Ingreso de bloques de salidas analógicas, practica 5.....	122
Figura 163. Condiciones de motores, practica 5.....	122
Figura 164. Sincronización de mediante ziegler nichols por ganancias critica, practica 5.....	123
Figura 165. Gráficas de ganancia Kp, practica 5.....	123
Figura 166. Valor de repetición de Tc, practica 5.....	124
Figura 167. Interfaz practica 5.....	124
Figura 168. Ejecución de LabView como compatible con ni myRio práctica 6.....	126
Figura 169. Nuevo Proyecto, práctica 6.....	126
Figura 170. Configuración de ruta, práctica 6	127
Figura 171. Administrador de proyectos, selección práctica 6.....	127
Figura 172. Configuración de i2c para myRio de práctica 6.....	128
Figura 173. Bloque de variables i2c, práctica 6.....	128
Figura 174. Configuración de entradas digitales, práctica 6.....	129
Figura 175. Conexión a lazo de selección, práctica 6.....	129
Figura 176. Lazo de selección con bloque i2c, práctica 6.....	130
Figura 177. Bundle con datos para gráfica, práctica 6.....	130
Figura 178. Ingreso de entradas analógicas, práctica 6.....	131
Figura 179. Conexión de salidas analógicas práctica 6.....	131
Figura 180. Obtención de datos de desplazamiento práctica 6.....	132
Figura 181. Ingreso de dos salidas analógicas, práctica 6.....	132
Figura 182. Señal de salida atenuada, practica 6.....	133
Figura 183. Configuración de salidas digitales, practica 6.....	133
Figura 184. Ingreso de dos salidas analógicas, práctica 6.....	133
Figura 185. Condiciones de motores, practica 6.....	134
Figura 186. Sincronización de mediante ziegler nichols por ganancias critica, practica 6.....	134
Figura 187. Gráficas de ganancia Kp, practica 6.....	135
Figura 188. Valor de repetición de Tc, practica 6.....	135
Figura 189. Configuración de entrada analógica de sensores, practica 6.....	136
Figura 190. Aplicación de ganancia para los sensores, practica 6.....	136
Figura 191. Ejecución de aplicación con lectura de sensores, practica 6.....	136
Figura 192. Comparación para el giro evitar obstáculo, practica 6.....	137
Figura 193. Ejecución de LabView como compatible con ni myRio práctica 7.....	138
Figura 194. Nuevo Proyecto, práctica 7.....	138
Figura 195. Configuración de ruta, práctica 7	139
Figura 196. Administrador de proyectos, selección práctica 7.....	139
Figura 197. Herramienta para creación de reglas difusas práctica 7.....	139

Figura 198. Herramienta para creación de variables, práctica 7	140
Figura 199. Configuración de rango en variable proporcional, práctica 7	140
Figura 200. Configuración de rango en variable integral, práctica 7.....	141
Figura 201. Configuración de rango en variable derivativa, práctica 7.....	141
Figura 202. Configuración de rango en variable derivativa, práctica 7.....	142
Figura 203. Configuración de entradas digitales, práctica 7	142
Figura 204. Bloque de control y simulación, práctica 7.....	142
Figura 205. Computo del pid difuso en proceso paralelo, práctica 7	143
Figura 206. Ingreso de entradas analógicas, práctica 7	143
Figura 207. Conexión de salidas analógicas práctica 7.....	144
Figura 208. Configuración de filtro, práctica 7.....	144
Figura 209. Filtro al eje “Y” y asignación en variable global, práctica 7.....	145
Figura 210. Ingreso de dos salidas analógicas, práctica 7.....	145
Figura 211. Configuración de salidas digitales, practica 7	146
Figura 212. Creación de variable global y conexión con motores, practica 7.....	146
Figura 213. Pid Fuzzy en robot auto balance, practica 7.....	147
Figura 214. Ejecución de LabView como compatible con ni myRio práctica 8.....	148
Figura 215. Ejecución de LabView como compatible con ni myRio práctica 8.....	148
Figura 216. Ejecución de LabView como compatible con ni myRio	149
Figura 217. Ejecución de LabView como compatible con ni myRio	149
Figura 218. Ejecución de LabView como compatible con ni myRio práctica 8.....	150
Figura 219. Nuevo Proyecto, práctica 8.....	150
Figura 220. Configuración de ruta, práctica 8	151
Figura 221. Administrador de proyectos, selección práctica 8.....	151
Figura 222. Ingreso de entradas analógicas, práctica 8	152
Figura 223. Conexión de salidas analógicas práctica 8.....	152
Figura 224. Configuración de filtro, práctica 8.....	152
Figura 225. Filtro al eje “Y”, práctica 8.	153
Figura 226. bloque de control y simulación para el pid, práctica 8.....	153
Figura 227. Configuración de filtro, práctica 8.....	153
Figura 228. Configuración de salidas digitales, practica 8.	154
Figura 229. Ingreso de bloques de salidas analógicas, practica 8.....	154
Figura 230. Condiciones de motores, practica 8.....	154
Figura 231. Interfaz practica 8.....	155
Figura 232. Gráficas de ganancia Kp, practica 8.....	155
Figura 233. Condicional por defecto, practica 8.....	156
Figura 234. Condicional para avanzar, practica 8.	156
Figura 235. Condicional para retroceder, practica 8.	156
Figura 236. Condicional para girar derecha, practica 8.	157
Figura 237. Condicional para girar izquierda, practica 8.....	157
Figura 238. Puerto serial con herramienta visa, practica 8.	157
Figura 239. Botoneras de control, practica 8.....	158
Figura 240. Lógica para el envío de valores al robot, practica 8.	158
Figura 241. Ejecución de LabView como compatible con ni myRio práctica 9.....	159
Figura 242. Nuevo Proyecto, práctica 9	159
Figura 243. Configuración de ruta, práctica 9	160
Figura 244. Administrador de proyectos, selección práctica 9.....	160
Figura 245. Ingreso de entradas analógicas, práctica 9	161
Figura 246. Conexión de salidas analógicas práctica 9.....	161
Figura 247. Configuración de filtro, práctica 9.....	162
Figura 248. Filtro al eje “Y”, práctica 9.	162
Figura 249. bloque de control y simulación para el pid, práctica 9.....	162

Figura 250. Configuración de filtro, práctica 9.....	163
Figura 251. Configuración de salidas digitales, practica 9.	163
Figura 252. Ingreso de bloques de salidas analógicas, practica 9.....	163
Figura 253. Condiciones de motores, practica 9.....	164
Figura 254. Interfaz practica 9.....	164
Figura 255. Botoneras de control, practica 9.....	164
Figura 256. Lógica para el envío de valores al robot, practica 9.	165
Figura 257. Condicional por defecto, practica 9.....	165
Figura 258. Condicional para avanzar, practica 9.	165
Figura 259. Condicional para retroceder, practica 9.	166
Figura 260. Condicional para girar derecha, practica 9.	166
Figura 261. Condicional para girar izquierda, practica 9.....	166
Figura 262. Diseño de aplicativo en Android, práctica 10.....	167
Figura 263. Programación en bloques de conexión, práctica 10.....	167
Figura 264. Programación en bloques de conexión, práctica 10.....	168
Figura 265. Programación en bloques de comandos, práctica 10	168
Figura 266. Ejecución de LabView como compatible con ni myRio práctica 10.....	169
Figura 267. Nuevo Proyecto, práctica 10.....	169
Figura 268. Configuración de ruta, práctica 10	170
Figura 269. Administrador de proyectos, selección práctica 10.	170
Figura 270. Ingreso de entradas analógicas, práctica 10.....	171
Figura 271. Conexión de salidas analógicas práctica 10.....	171
Figura 272. Filtro al eje “Y”, práctica 10.	172
Figura 273. Configuración de filtro, práctica 10.	172
Figura 274. bloque de control y simulación para el pid, práctica 10.	172
Figura 275. Configuración de filtro, práctica 10.	173
Figura 276. Configuración de salidas digitales, practica 10.	173
Figura 277. Ingreso de bloques de salidas analógicas, practica 10.....	173
Figura 278. Condiciones de motores, practica 10.	174
Figura 279. Interfaz practica 10.....	174
Figura 280. Configurar la adquisición de datos uart, practica 10.	175
Figura 281. Condicional por defecto, practica 10.....	175
Figura 282. Condicional para avanzar, practica 10.	176
Figura 283. Condicional para retroceder, practica 10.	176
Figura 284. Condicional para girar derecha, practica 10.	176
Figura 285. Condicional para girar izquierda, practica 10.	177
Figura 286. Programa ni Max de National Instrument.....	178
Figura 287. Dispositivo ni myRio en sistemas remotos	178
Figura 288. Configuración de red inalámbrica.....	179
Figura 289. Finalizar la configuración.....	179

ÍNDICE DE TABLAS

Tabla 1	Conexiones de módulo de giroscopio.....	23
Tabla 2	Conexiones de ultrasónico 1.....	24
Tabla 3	Conexiones de ultrasónico 2.....	24
Tabla 4	Conexiones de Encoder 1.....	26
Tabla 5	Conexiones de Encoder 2.....	26
Tabla 6	Conexiones del bluetooth.....	28
Tabla 7	Conexiones de controlador de motores.....	30
Tabla 8	Método de ziegler nichols ganancia crítica.....	32
Tabla 9	Comandos del Pitsco Tetrix en practica 8.....	63
Tabla 10	Comandos del Pitsco Tetrix en practica 8.....	66
Tabla 11	Comandos del Pitsco Tetrix en practica 10.....	69
Tabla 12	Conexiones de controlador de motores.....	106
Tabla 13	Conexiones de controlador de motores.....	111
Tabla 14	Método de ziegler nichols ganancia crítica, practica 5.....	124
Tabla 15	Método de ziegler nichols ganancia crítica, practica 6.....	135

INTRODUCCIÓN

Actualmente la competencia técnica del estudiante y por lo tanto del futuro profesional, juega un papel muy importante en el logro y mejoramiento del nivel de vida de la sociedad, para ello es fundamental que se cumplan los parámetros y normas establecidas para el buen entendimiento de los sistemas de control, pid, difusos y demás sistemas convencionales de esta forma el estudiante conocer este tipo de controladores y obteniendo estas destrezas en de forma teórica y práctica.

La Universidad Politécnica Salesiana cuenta con laboratorios necesarios para el control continuo, teniendo en cuenta el constante avance tecnológico se ve la necesidad de la creación de nuevos módulos para las cátedras como de la materia de Informática Industrial donde se trata el Control PID Difuso permitiendo a los estudiantes evidenciar una aplicación de este, lo cual no permite una mejor interacción entre lo teórico y lo práctico y se puede comprobar con el plan de estudio dado cada semestre.

Conocer diferentes tipos de controladores brinda mejores soluciones al momento de que el estudiante se enfrente a algún problema en su ambiente laboral, es por ello la importancia de demostrar las distintas aplicaciones que puede llegar a tener el Control antes mencionado.

1. EL PROBLEMA

1.1. Antecedentes

Debido a que en la materia de Informática Industrial se imparte el Control PID Difuso mediante el software LabView de National Instruments, existe la necesidad de un dispositivo didáctico que pueda servir para aplicaciones de control difuso, mediante el myRio y el Pitsco Tetrax, logrando enlazar lo programado en el software LabView mediante comunicación inalámbrica con un medio físico tal como el robot autobalance y así obtener las destrezas que ayuden a un mejor entendimiento de la materia en lo que respecta a sistemas difusos.

1.2. Importancia y Alcances

El presente trabajo contribuye al desarrollo de conocimientos tal como sistemas de control, sensores y transductores.

Con el presente proyecto el estudiante puede desarrollar rutinas aplicados a sistemas de control difusos en un robot auto balance con tecnología Pitsco Tetrax.

1.3. Delimitación

1.3.1. Temporal

La implementación de este proyecto se realizó en un intervalo de un año a partir de la aprobación de este.

1.3.2. Espacial

Este proyecto se llevó a cabo en la Universidad Politécnica Salesiana Sede Guayaquil en el bloque E (Laboratorios de Control).

1.3.3. Académica

Mediante rutinas se dará el aprendizaje necesario para las materias de control, la cual se estima un aproximado de 60 horas semestrales en los laboratorios de Control, con equipamiento específico para las prácticas de control pid difuso en un robot autobalance con tecnología Pitsco Tetrrix

1.4. Innovación

Los Controles Industriales son muy importantes en cualquier empresa para el manejo de procesos, por lo tanto, aprender diferentes tipos de controles permite una mayor variedad de posibles soluciones ante cualquier inconveniente de un proceso, el control PID es sin duda uno de lo más útil, pero al unirlo a un control Difuso se puede elaborar un control más robusto y de mejor eficiencia. El PID Difuso es considerado una de las tecnologías más importantes en la sociedad y se puede observar la gran eficiencia que nos ha enseñado como son los avances a gran escala como: Laboratorios - Equipos Médicos - Ámbito Hospitalario - Zona Industrial - Sistemas Automatizados. Se puede observar los beneficios tecnológicos mediante el banco de prácticas que nos brinda el robot de equilibrio automático.

1.5. Objetivos

1.5.1. Objetivo general

- Diseñar un controlador PID Difuso mediante el uso del dispositivo Pitsco Tetrrix prime y myRio en una planta de un robot auto balanceado

1.5.2. Objetivos específicos

- Realizar una programación de Control para el Robot Auto Balanceado.
- Elaborar un algoritmo para el uso del giroscopio y el acelerómetro del Robot Auto Balanceado.
- Exponer las diferencias del Control Difuso del Robot Auto Balanceado con un Controlador PID normal elaborado.

- Elaborar un manual de 10 prácticas didácticas para mostrar las diferentes aplicaciones del controlador mediante el uso del dispositivo myRio de National Instruments.

2. FUNDAMENTOS TEÓRICOS

2.1. Sistemas de Control PID

El control pid es un algoritmo el cual consta de tres parámetro o ganancias proporcional, integral y derivativo , el valor proporcional depende del error entre el valor medido y la referencia representada por la fórmula 1, el valor integral consta de los valores de los errores pasados representados por la fórmula 2 y el valor derivativo depende de una predicción de los valores futuros representado por la fórmula 3, fue patentado en 1939 por Albert Calender y Allan Stevenson de la firma Imperial Chemicoles (Åström, 1995), la formula del algoritmo pid está representado por la fórmula 4.

$$K_p e(t) \quad (1)$$

$$\frac{1}{T_i} \int_0^t e(\tau) d\tau \quad (2)$$

$$T_d \frac{de(t)}{dt} \quad (3)$$

$$u(t) = K_p [e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt}] \quad (4)$$

Donde:

K_p es la acción proporcional

T_i es la acción intergral

T_d es la acción derivativa

La representación del pid en diagrama se muestra en la figura 1

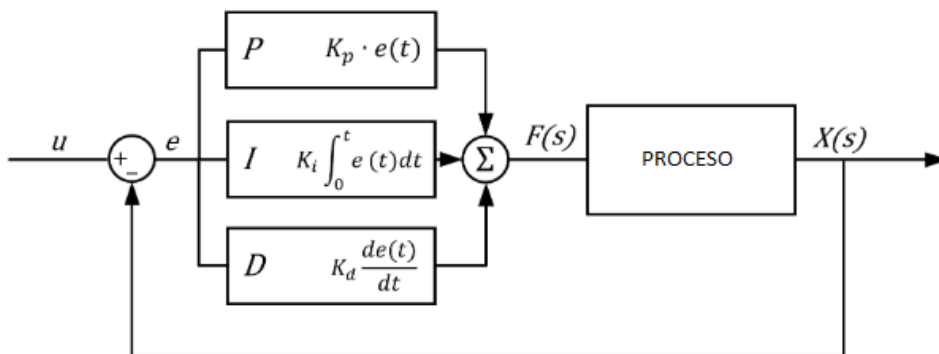


Figura 1: Diagrama Pid (Astrom, 1995)

2.2. Sistemas de control Difusos

Los sistemas difusos son una extensión de la inteligencia artificial y tiene dos tipos de estructuras una es el control directo y otra el control. (González Macías Bryan Xavier, 2018).

La capacidad de los sistemas difusos para describir la operación de un sistema complejo mediante reglas simples expresadas en lenguaje natural ha motivado su aplicación a numerosos problemas de automatización y control industrial, dando lugar a lo que ha venido en denominarse controladores difusos (Santiago Sánchez-Solano Alejandro J. Cabrera, 2015).

La estructura de un control difuso está definida por la parte de Fuzzificación, sistema de inferencia, Desfuzzificación y la base de conocimientos como se muestra en la Figura 2.

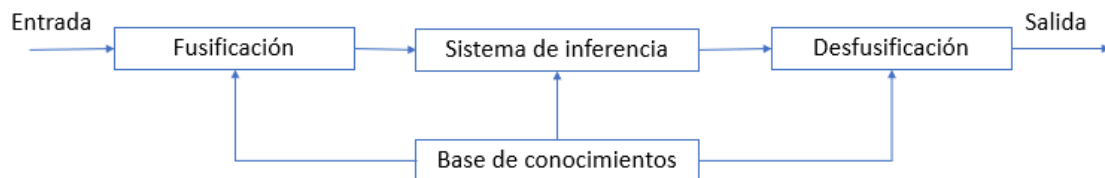


Figura 2: Diagrama de control difuso

2.3. Desfuzzificación

Es el método de procesar un valor cuantificable en Lógica clásica, tomada de un conjunto difusos y sus correspondientes grados de membresía.

Poseen una serie de reglas que transforman un valor numérico de variables en un resultado difuso, el cual depende de las reglas de membresía dadas en el sistema (Leekwijck, 1999).

2.4. Fusificación

Es el método de convertir valores reales a variables difusas que son dadas por los grados de pertenencia de cada una de las variables de entrada previamente definidos utilizados en los conjuntos difusos. (Ramos, 2008)

2.5. Inferencia

Es la relación que existe entre los conjuntos difusos de entrada y salida que son representadas por las reglas que gobiernan el sistema, se usa a la par con la base de conocimientos para generar reglas mediante casos o condiciones (Ramos, 2008).

2.6. Base de conocimiento

Contiene todo asociado a la aplicación y los objetos de control donde se definen las reglas lingüísticas que toman la decisión en el sistema (Ramos, 2008).

2.7. Función de pertenencia

La función de pertenencia nos indica el grado en que cada elemento de un universo dado pertenece a dicho conjunto. Es decir, la función de pertenencia de un conjunto A sobre un universo X será de la forma: $\mu_A: X \rightarrow [0,1]$, donde $\mu_A(x) = r$ si r es el grado en que x pertenece al conjunto A (Castillo, 2008).

Si el conjunto es nítido, su función de pertenencia (función característica) tomará los valores en $\{0,1\}$, mientras que, si es borroso, los tomará en el intervalo $[0,1]$. Si $\mu_A(x) = 0$ el elemento no pertenece al conjunto, si $\mu_A(x) = 1$ el elemento sí pertenece totalmente al conjunto (Castillo, 2008).

Las funciones de pertenencia son una forma de representar gráficamente un conjunto borroso sobre un universo (Castillo, 2008) como se muestra en la Figura 3.

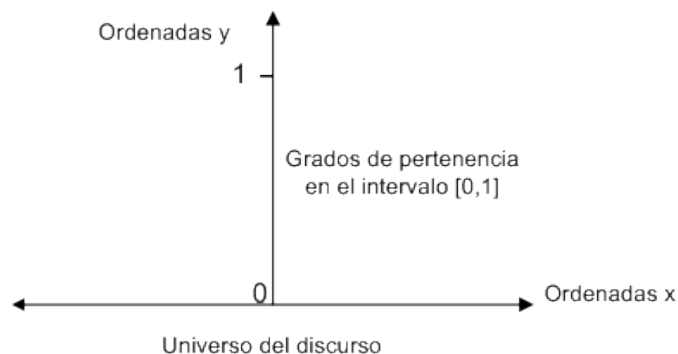


Figura 3: Funciones de pertenencia (Castillo, 2008)

2.8. Función triangular

Definida mediante el límite inferior a , el superior b y el valor modal m , tal que $a < m < b$. La función no tiene porqué ser simétrica como se muestra en la figura 4.

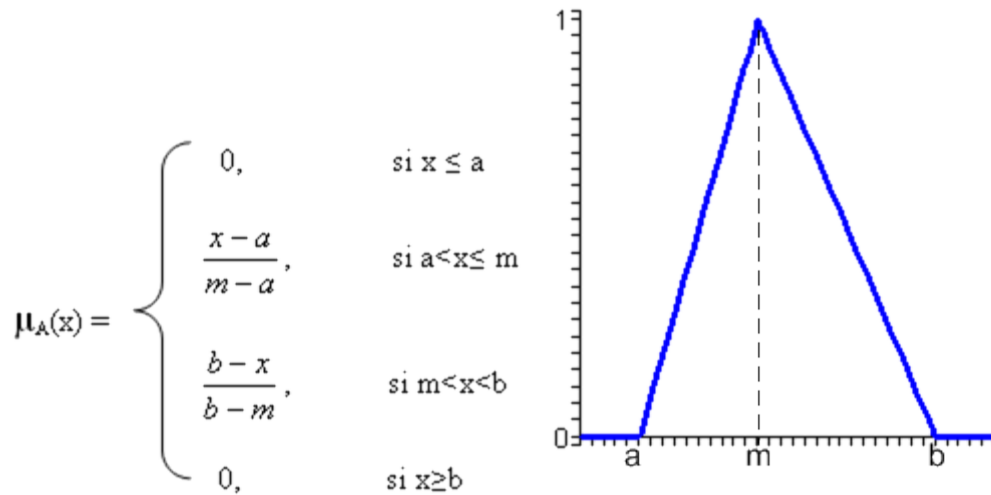


Figura 4: Funciones triangular (Castillo, 2008)

2.9. Función trapezoidal

Definida por sus límites inferior a , superior d , y los límites de soporte inferior b y superior c , tal que $a < b < c < d$.

En este caso, si los valores de b y c son iguales, se obtiene una función triangular como se muestra en la figura 5.

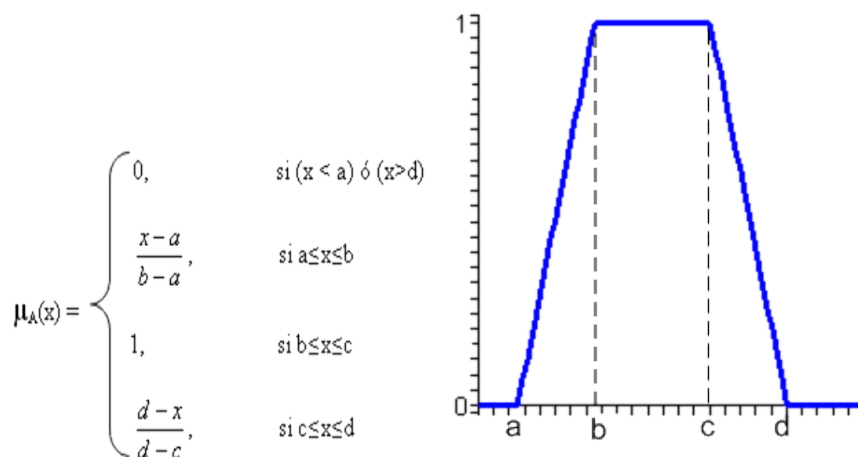


Figura 5: Funciones trapezoidal (Castillo, 2008)

2.10. Función Gamma

Definida por su límite inferior a y el valor $k > 0$. Esta función se caracteriza por un rápido crecimiento a partir de a ; cuanto mayor es el valor de k , el crecimiento es más rápido. Nunca toma el valor $\mu_A(x) = 1$, aunque tienen una asíntota horizontal en dicho valor como se muestra en la figura 6. (Castillo, 2008)

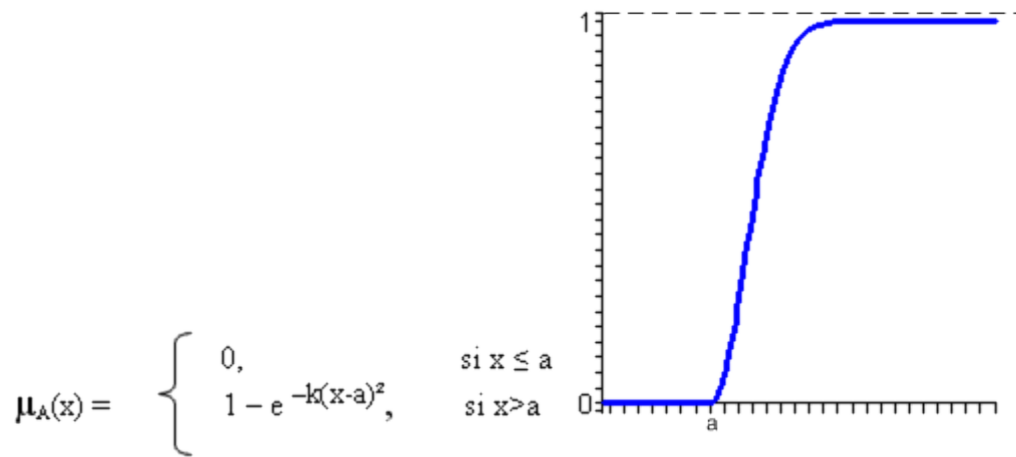


Figura 6: Funciones Gamma (Castillo, 2008)

2.11. Función Gaussiana

Definida por su valor medio m y el parámetro $k > 0$. Esta función es la típica campana de Gauss y cuanto mayor es el valor de k , más estrecha es dicha campaña que se muestra en la Figura 7 (Castillo, 2008).

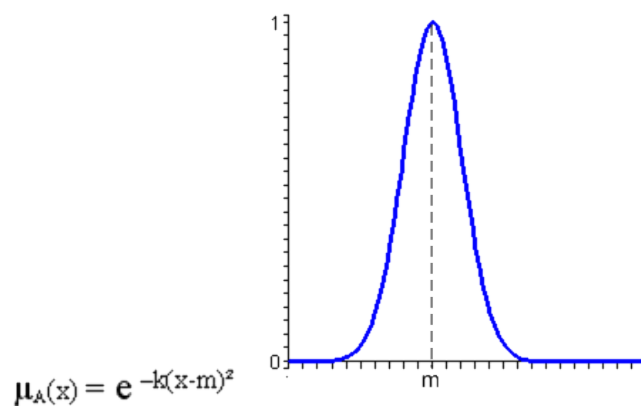


Figura 7: Funciones Gaussiana (Castillo, 2008)

2.12. Variables lingüísticas

Son aquellas que sirven para representar el conocimiento en razonamiento aproximado, es aquella cuyos valores son palabras sentencias en un lenguaje natural o artificial. De esta forma, una variable lingüística sirve para representar cualquier elemento que sea demasiado complejo, o del cual no tengamos una definición concreta; es decir, lo que no podemos describir en términos numéricos (Morcillo, 2017).

2.13. LabView

Plataforma que brinda un entorno de programación en bloque la cual nos permite la elaboración de instrumentación virtual, procesos industriales e implementar sistemas de control con sus herramientas posee dos ventanas una el panel frontal y el bloque de diagrama. (González Macías Bryan Xavier, 2018).



Figura 8: LabView (NI, 2018)

2.14. Ni myRio 1900

El hardware embebido myRio es una herramienta accesible para los estudiantes, e ideal para realizar aplicaciones y proyectos reales de ingeniería. Cuenta con un procesador ARM® Cortex™-A9 dual-Core en tiempo real, así como entradas y salidas personalizadas de FPGA Xilinx (National Instruments Corporation, 2019).



Figura 9: Ni MyRio 1900 (National Instruments Corporation, 2019)

Consta de dos puertos los cuales poseen entradas y salidas digitales, entradas y salidas análogas, pines de comunicación serial y comunicación i2c la arquitectura de los pines se muestra en la figura10 son el puerto A y B, en la Figura 11 se muestra el puerto C.

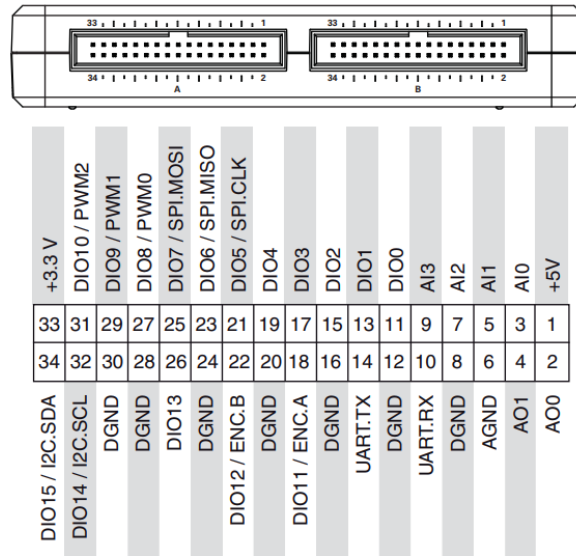


Figura 10: Ni MyRio 1900 Puerto A y B (National Instruments Corporation, 2019)

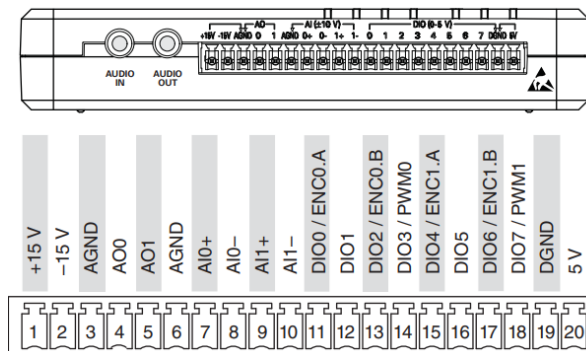


Figura 11: Ni MyRio 1900 Puerto C (National Instruments Corporation, 2019)

2.15. Pitsco Tetrax Prime

Es un dispositivo de desarrollo aplicado a la educación consta de un paquete de robótica que los estudiantes pueden usar para aplicar conceptos de ingeniería y realización de rutinas de aprendizaje para plasmar el entendimiento teórico experimental en las áreas de sistemas reales de control

y robótica que al combinar hardware myRio con software LabView, el sistema de desarrollo Pitsco TETRIX, los sensores y los actuadores en conjunto con los bloques de construcción que necesitan para crear y automatizar ensambles modelo, así como sus propios diseños véase en detalle Figura 12.



Figura 12: Pitsco Tetrix prime (prime, 2020)

2.16. Sensor de giroscopio

El sensor de giroscopio mide la velocidad angular, o la tasa de rotación, alrededor de tres ejes. Cuando está montado, proporciona actitud tridimensional (cabeceo, balanceo y guiñada), tiene una comunicación mediante protocolo SPI o I2C, incluye datos en serie (SDA), reloj en serie (SCL), dos interrupciones, alimentación y tierra (Bautista España, 2018).

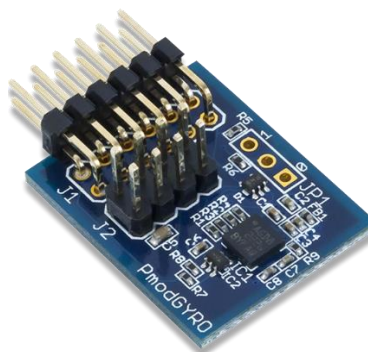


Figura 13: Sensor de giroscopio Pitsco Tetrix (prime, 2020)

2.17. Controladora de motores

Un controlador de motor es aquel que mediante señales de entrada logra el manejo de actuadores controlando su velocidad mediante el voltaje aplicado.

La controladora del ni myRio (Figura 14) es fabricada por la empresa Digilent, es un controlador de fácil uso el cual tiene un driver de potencia Drv8432 de una potencia máxima de 6 amperios con un rango de voltaje de operación de 5-12 voltios y una frecuencia de trabajo superior a 500 KHz (Builders, 2017).

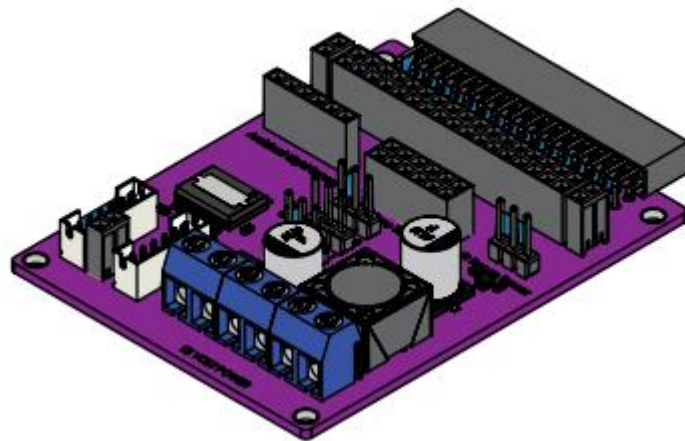


Figura 14: Tarjeta controladora de motores (Builders, 2017).

2.18. Motor de corriente directa

Los motores de corriente continua se pueden encontrar en tamaños que van desde los sub fraccionarios hasta de miles de hp, y se usan para convertir energía eléctrica en energía mecánica. Son apropiados para unidades motrices de velocidad constante, inversión de velocidad, avance poco a poco en algunos casos y unidades motrices de velocidad múltiple. (Meaton, 1991)

El motor de la figura 15 es de la familia Digilent compatible con el pisto Tetrrix por su tipo de funcionamiento mejora el factor de potencia de una planta sin que esto afecte a su trabajo, para que sea mucho mayor y que tenga un mejor tipo de trabajo en algunos sistemas de control como robótica, automatización y mecánica.



Figura 15: Motor de corriente continua (Digilent, 2018)

2.19. Codificadores Rotativo (Encoder)

En aplicaciones de automatización, los codificadores rotativos sirven como medidores de ángulo, posición, velocidad y aceleración. Los codificadores rotativos transforman un valor de rotación mecánica en una señal eléctrica que puede ser procesada por contadores, tacómetros, controladores lógicos y Pc industriales. Trabajan con un disco de vidrio o de plástico con campos alternos oscuros y claros, con una fuente de luz en un lado y un sensor sensible a la luz en el otro. A medida que el disco gira, la fuente de luz se bloquea de manera alternativa y se revela al sensor. Siempre que la fuente de luz llegue al sensor, el codificador envía un impulso eléctrico que puede ser interpretado por un controlador. El pulso finaliza cuando un campo oscuro del disco no permite el paso de la luz. La rotación del disco produce una salida de pulso de onda cuadrada. (Principles of rotary encoders, 2004)

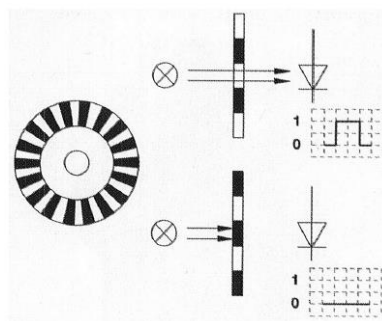


Figura 16: Funcionamiento de disco rotativo del Encoder (Principles of rotary encoders, 2004)

2.19.1. Encoder Incrementales

Los Encoder incrementales tienen en su forma más sencilla un disco transparente con una serie de marcas opacas colocadas radialmente y a la misma distancia entre sí, de un sistema de iluminación en el que la luz es colimada de forma correcta, y de un dispositivo fotorreceptor. El eje, cuya posición se quiere calcular, va acoplado al disco transparente. Para que a medida que el eje gire se irán generando pulsos en el receptor cada vez que la luz atraviese, llevando una cuenta de estos pulsos se podrá conocer la posición del eje. (Barrientos, Antonio, Peñín, Luis Felipe, & Balag, 2007)

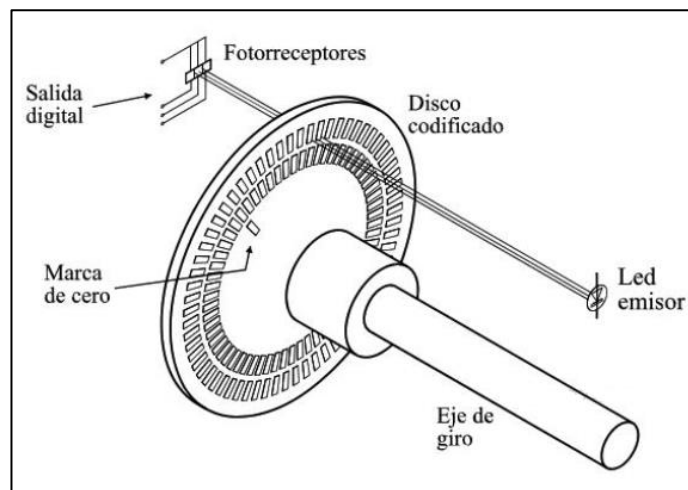


Figura 17: Disposición de un Encoder incremental (Barrientos, Antonio, Peñín, Luis Felipe, & Balag, 2007)

Los pulsos que se producen como resultado del giro, no indican por sí mismos del sentido de éste. Para poder conocer esta información, se coloca una segunda franja de marcas desplazada de la primera, de manera que el tren de pulsos que con ella se genere tenga un desfase de 90° con respecto al creado por la primera franja. Es así como mediante el uso de un biestable de tipo D, en el que la salida mantiene el valor de la entrada leída durante el flanco positivo del reloj, se puede obtener una señal adicional (Q) que indicara cuál es el sentido de giro, y que actúe sobre el contador correspondiente indicando que incremente o decremente la cuenta que se esté realizando. (Barrientos, Antonio, Peñín, Luis Felipe, & Balag, 2007).

Debido a que el contador es incremental, se deberá indicar cuál es el cero, para lo cual se incorpora una marca de referencia adicional (Z) única en todo el trazado del disco, que indicara que se ha dado una vuelta completa y que, por tanto, se debe reinicializar el contador. (Barrientos, Antonio, Peñín, Luis Felipe, & Balag, 2007)

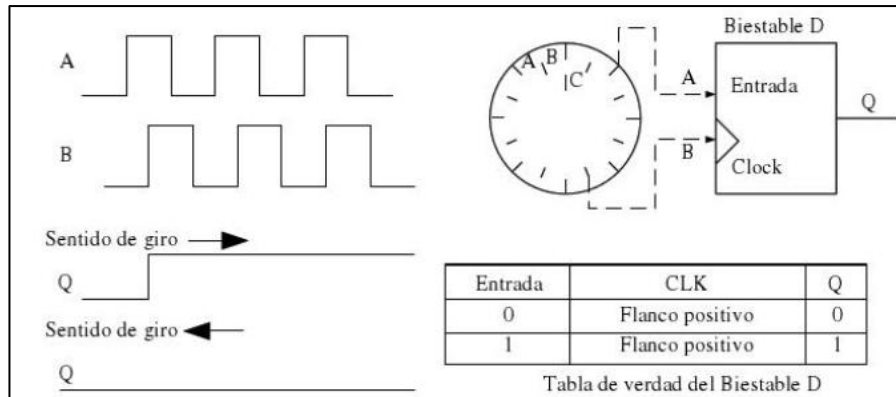


Figura 18: Salidas A y B de un Encoder y valor (Q) del Biestable D según su sentido de giro (Barrientos, Antonio, Peñín, Luis Felipe, & Balag, 2007)

2.19.2. Encoder absolutos

Un Encoder absoluto genera a la salida una palabra binaria en la que está codificada la posición absoluta que se desea medir. Soluciona las desventajas del codificador incremental, pero su precio es muy superior.

Entrega en su salida una señal digital codificada correspondiente a la posición de un disco móvil con respecto a una referencia interna. Para ello, el disco móvil se divide en sectores (2^n , donde n es el número de bits del codificador) y en cada sector se codifica el valor de su posición con zonas claras (a las que se asigna el valor binario 0) y oscuras (a las que se asigna el valor binario 1) situadas de acuerdo con el código utilizado (Gray, binario natural...). A diferencia de los codificadores incrementales, hay varias pistas con zonas diferenciadas y están agrupadas de tal forma que el sistema de lectura obtiene directamente en cada posición del elemento móvil el valor codificado de su posición. Cada pista simboliza un bit de salida, siendo la pista interna la correspondiente al bit de mayor peso. La salida se lee en paralelo para generar una representación binaria de la posición angular. (Granda Miguel, Mercedes, & Mediavilla Bolado, 2015).

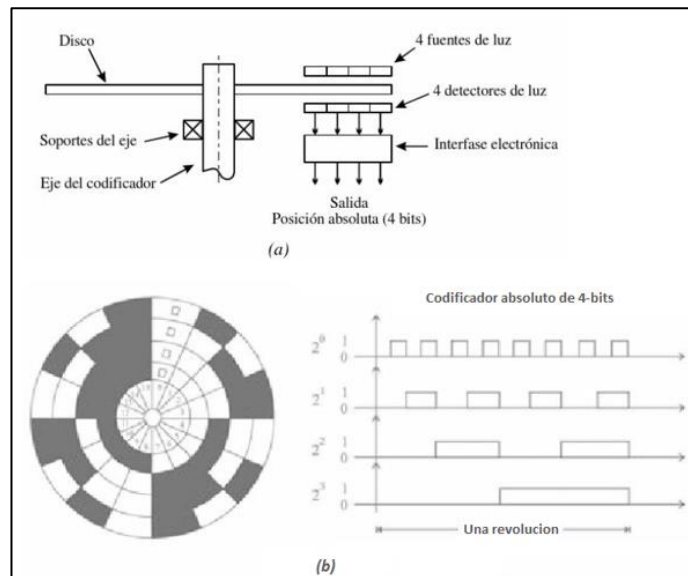


Figura 19: Codificador absoluto binario de 4bits. (a) Construcción Interna. (b) Señales de salida (Granda Miguel, Mercedes, & Mediavilla Bolado, 2015)

2.20. Max sonar Xr

Los telémetros ultrasónicos HRXL-Max Sonar-WR con clasificación IP67 ofrecen una resolución de 1 mm, funcionamiento de 2,7 a 5,5 V CC, un patrón de haz estrecho, salida de alta potencia, rechazo de ruido, calibración automática y compensación de temperatura. Las opciones de salida para esta línea de sensores ultrasónicos son anchas de pulso, voltaje analógico y una opción de serie RS232 o TTL asíncrona (MaxBotic, 2017).

2.21. Péndulo invertido

Un péndulo invertido es un péndulo que tiene su masa por encima de su punto de pivote. Es a menudo implementado con el punto de pivote montado en un carro que puede moverse horizontalmente y puede llamarse carro y poste. Mientras que un péndulo normal es estable cuando se cuelga hacia abajo, un péndulo invertido es inherentemente inestable y debe estar activamente equilibrado para permanecer en posición vertical, ya sea aplicando un par en el punto de pivote o moviendo el punto de pivote horizontalmente como parte de un sistema de retroalimentación (Abreu, 2009).

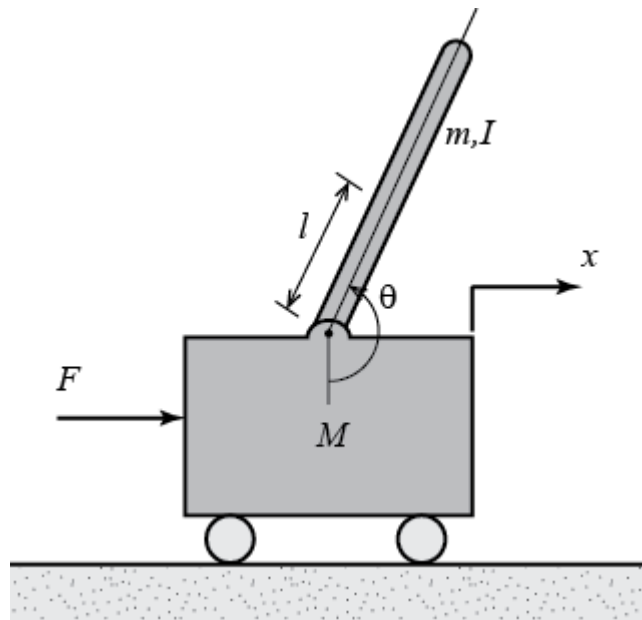


Figura 20: Péndulo invertido (CTMS, 2018)

Donde:

(M) masa del carro 0,5 kg

(m) masa del péndulo 0,2 kg

(b) coeficiente de fricción para el carro 0.1 N / m / seg

(l) longitud al centro de masa del péndulo 0,3 m

(I) momento de inercia de masa del péndulo 0.006 kg.m ²

(F) fuerza aplicada al carro

(x) coordenada de posición del carro

(theta) ángulo de péndulo desde vertical (hacia abajo)

2.22. Robot de dos ruedas basado en péndulo invertido

La idea de un robot que pueda auto equilibrarse ha ganado impulso durante la última década en varios laboratorios de robots, con fines académicos o de investigación en todo el mundo (Abreu, 2009)..

Kazuo Yamafuji, profesor emérito de la Universidad de Electro-Comunicaciones en Tokio, construyó el primer robot de péndulo invertido de dos ruedas en 1986, según un artículo en el Japón Times (Times, 2009)

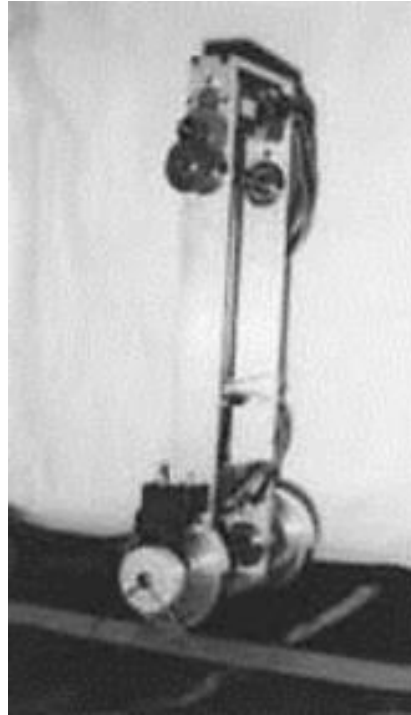


Figura 21: Kazuo Yamafuji Robot (Times, 2009)

2.23. App inventor

Es una plataforma de entorno de desarrollo web para la creación de programas ejecutables en sistemas basados en Android creada por Google Labs en el año 2010 y con colaboración del Instituto de tecnología de Massachusetts en diciembre del 2011, a través de su centro de aprendizaje para dispositivos móviles, utilizando un editor llamado Open Block Java el cual permite el desarrollo orientado a componentes más conocidos utilizado la arquitectura de programación en bloques (Margarita Torres, 2016).

3. MARCO METODOLÓGICO

3.1. Diseño del robot

El prototipo robótico balancín es un sistema el cual posee dos ruedas en la parte inferior ayudando a la estabilización de este, a la par un controlador ni myRio para el procesamiento de datos como el Angulo de inclinación del robot para controlar la estabilización o equilibrio.

En la figura 22 se muestra el robot balance y sus diferentes componentes, el ensamble de Pitsco Tetrix como balancés se detalla en el anexo 3.

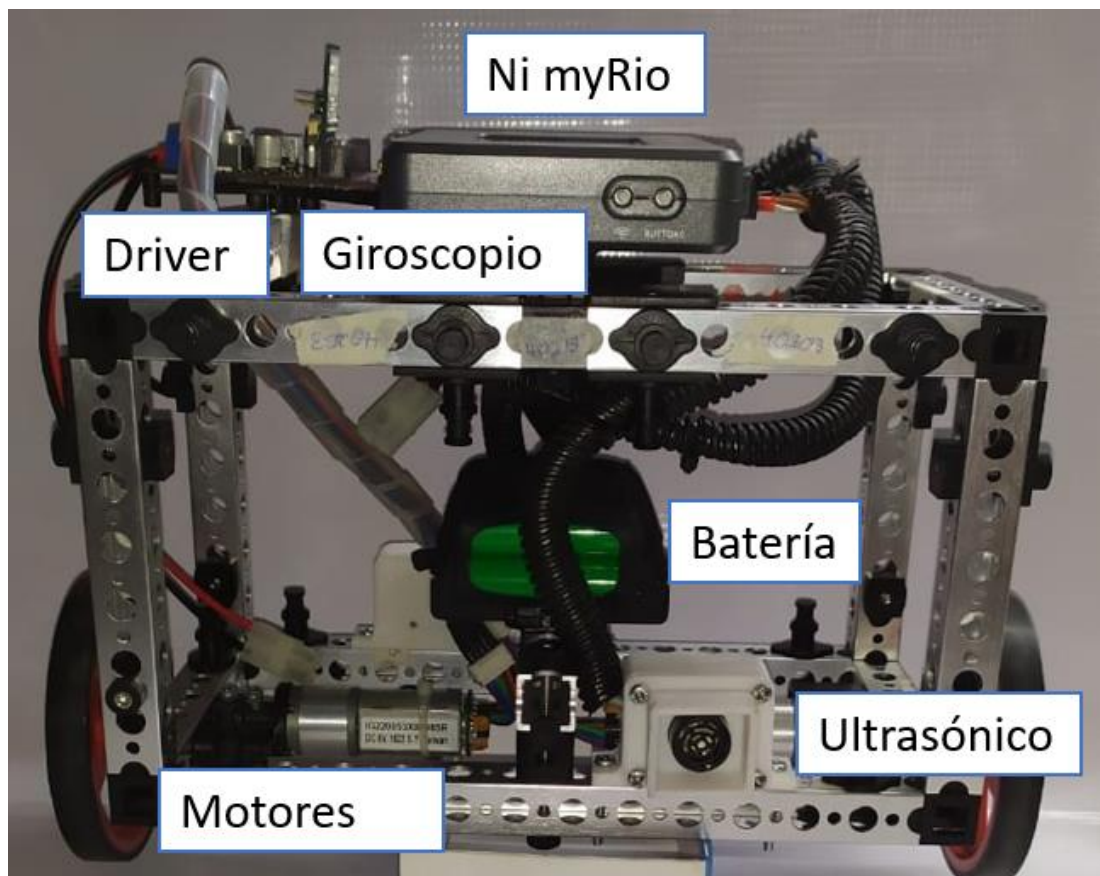


Figura 22. Prototipo Pitsco Tetrix como balance

3.2. Diagrama de control

El proyecto consta con un algoritmo de control pid difuso donde la entrada del sistema o referencia es el punto 0 de estabilización en el eje x conectado a un

controlador pid y la salida de este a un conjunto de reglas difusas para lograr la estabilización como se muestra en la figura 23.

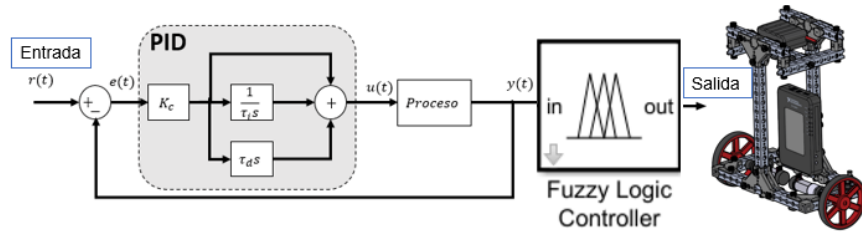


Figura 23. Diagrama de control

3.3. Esquema del robot

El robot balancín constan de un controlador ni myRio conectado con una placa de potencia para el control de los motores, los mismos que poseen un Encoder de dos canales cada uno teniendo como retroalimentación de la lectura al controlador, mediante el módulo giroscopio conectado al embebido se logra la estabilización mediante un control pid difuso.

El prototipo tiene dos entradas para la lectura del sensor de distancia y para la lectura de la comunicación serial de un bluetooth hc 05.

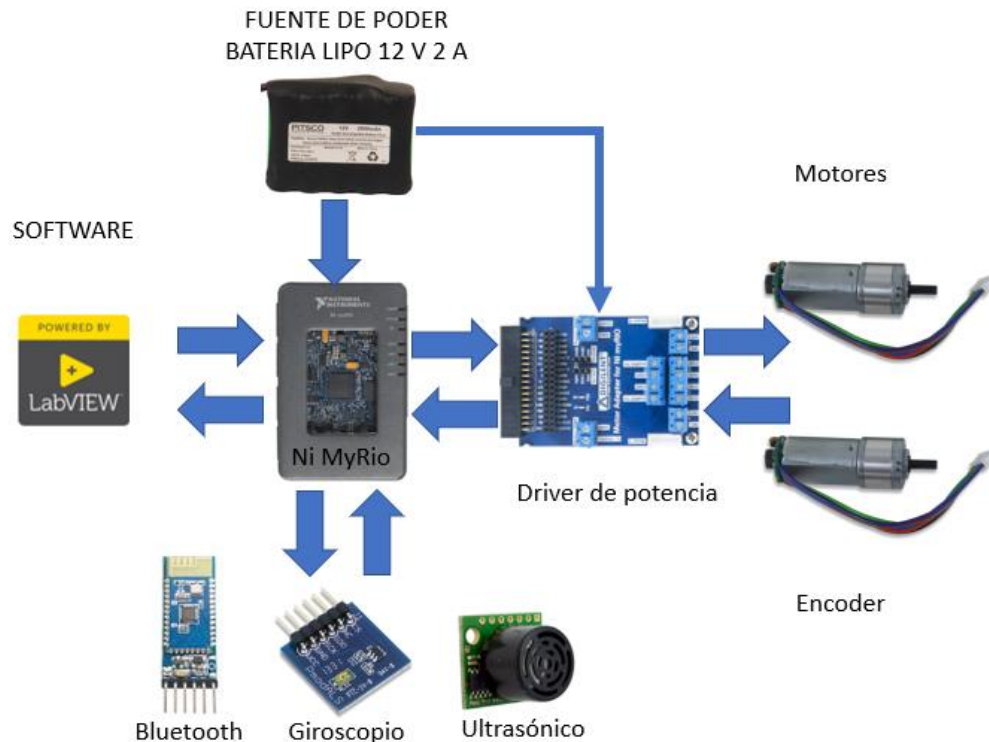


Figura 24. Esquema del prototipo

3.4. Conexiones del prototipo

3.4.1. Conexiones de la instrumentación

3.4.1.1. Sensor Giroscopio

Para la instrumentación del prototipo encargada de la adquisición de datos en tiempo real como el valor de inclinación del prototipo se utiliza un módulo giroscopio con el integrado Microelectronics L3G4200D conectado al controlador ni myRio mediante el protocolo i2c donde la dirección del dispositivo es 0X69(Dato en hexadecimal y la dirección para que el dispositivo envíe los valores es 0XA8, el tamaño del dato de respuesta es de 6 bits que sirven para obtener los datos de velocidad angular como se ve en la figura 24.

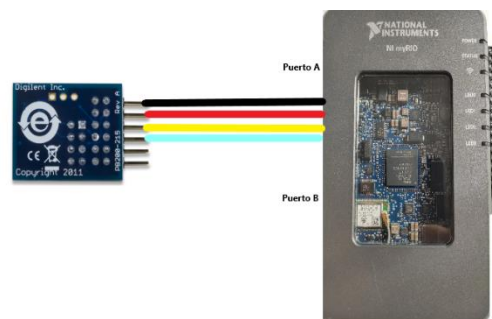


Figura 25. Modulo giroscopio conectado al ni myRio

Para leer los datos de inclinación el módulo envía un dato digital cada vez que cambie o se desplaza el robot en el espacio, la conversión del dato del i2c se toma como arreglo y se separa cada 2 bits de espacio donde cada dos bits se convierten a un valor Word de 16 bits teniendo como resultado el desplazamiento en x, y ,z como se muestra en la figura 26.

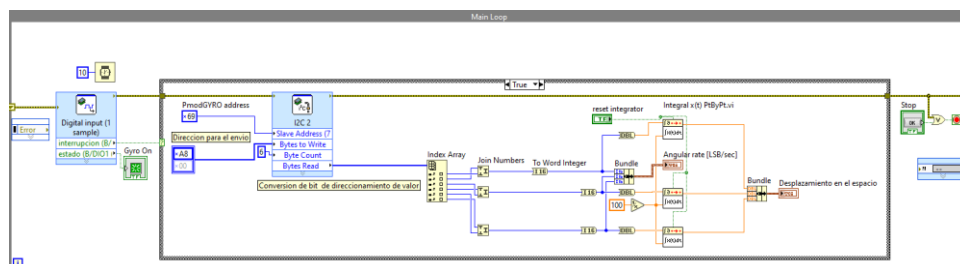


Figura 26. Modulo giroscopio conectado al ni myRio

Las conexiones del módulo de giroscopio están configuradas mediante borneras que siguen las configuraciones de la tabla 1 para conectar al panel de control mediante conectores de 7 pines.

Modulo Giroscopio	Ni MyRio
SDA	Pin 34 Puerto B
SCL	Pin 32 Puerto B
INT 1	Pin 30 Puerto B
INT 2	Pin 11 Puerto B
GND	Pin 13 Puerto B
VCC	Pin 33 Puerto B

Tabla 1 Conexiones de módulo de giroscopio

3.4.1.2. Sensor Ultrasónico

El sensor ultrasónico es el módulo Mb 1340 perteneciente a la familia de Max botix, posee una salida analógica , comunicación serial y una salida Pwm modulada dependiente de la lectura como se muestra en la figura 27.

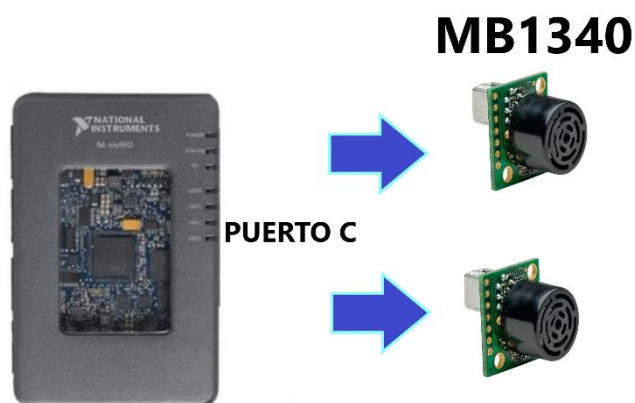


Figura 27. Modulo ultrasónico conectado al ni myRio

El prototipo consta de dos ultrasónico ubicados en la parte frontal y posterior del mismo evitando cualquier choque con algún objeto al realizar el auto equilibrio para la lectura se utiliza la configuración analógica y al valor de

lectura se le aplica una ganancia de 245 teniendo como resultante el valor en centímetros como se muestra en la figura 28

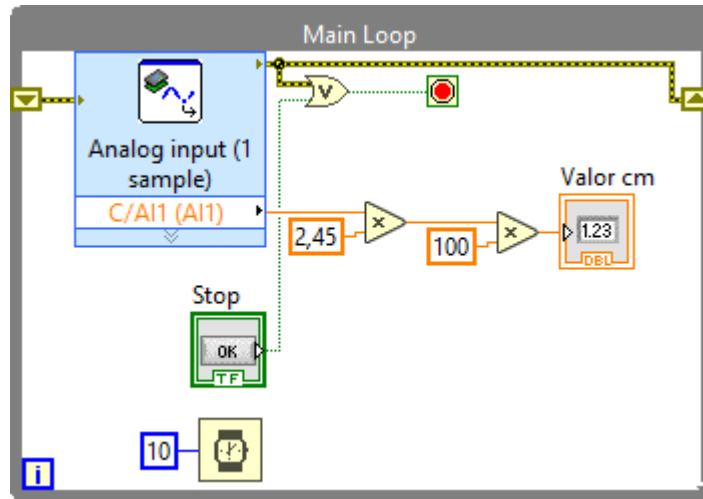


Figura 28. Modulo giroscopio conectado al ni myRio

Las conexiones del módulo ni myRio con los ultrasónicos se muestran en la tabla 2 (ultrasónico 1) y tabla 3 (ultrasónico 2) para conectar se utiliza el puerto C del ni myRio.

Modulo Ultrasónico	Ni MyRio
OUT +	Pin AI0 + Puerto C
OUT -	Pin GND Puerto C
GND	Pin VCC Puerto C
VCC	Pin GND Puerto C

Tabla 2 Conexiones de ultrasónico 1

Modulo Ultrasónico	Ni MyRio
OUT +	Pin AI1 + Puerto C
OUT -	Pin GND Puerto C
GND	Pin VCC Puerto C
VCC	Pin GND Puerto C

Tabla 3 Conexiones de ultrasónico 2

3.4.1.3. Sensor de posición -Encoder

El sensor posición que poseen los motores acoplados al eje sirven para obtener los datos mediante modulación de ancho de pulso conectado los pines a una entrada de conteo del ni myRio como se muestra en la figura 29 el esquema de conexión.

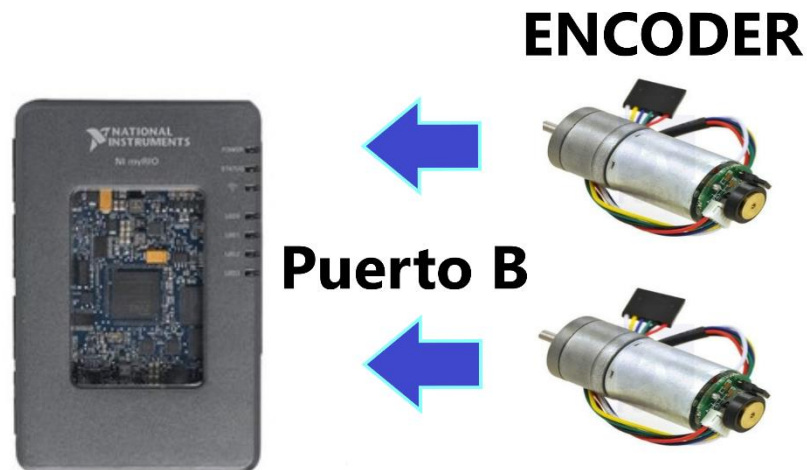


Figura 29. Encoder conectado al ni myRio

La señal del Encoder de los Encoder está desfasada la primera y la segunda se muestra en la figura 30.

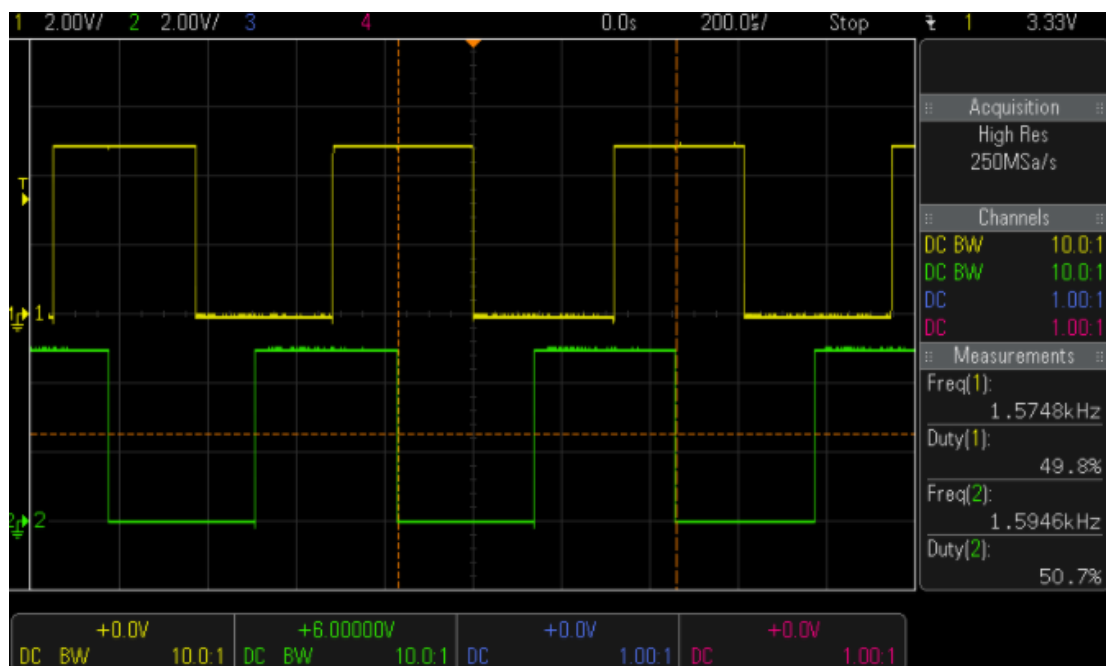


Figura 30. Señal de Encoder en osciloscopio

La configuración para la adquisición de los valores mediante el ni myRio es mediante la herramienta de lectura rápida de LabView con el complemento para el myRio, se toma en cuenta la cantidad de pulsos del Encoder y se divide el valor de lectura para 3600 y se aumenta una ganancia 360 para obtener el valor de la orientación del motor como se muestra en la figura 31.

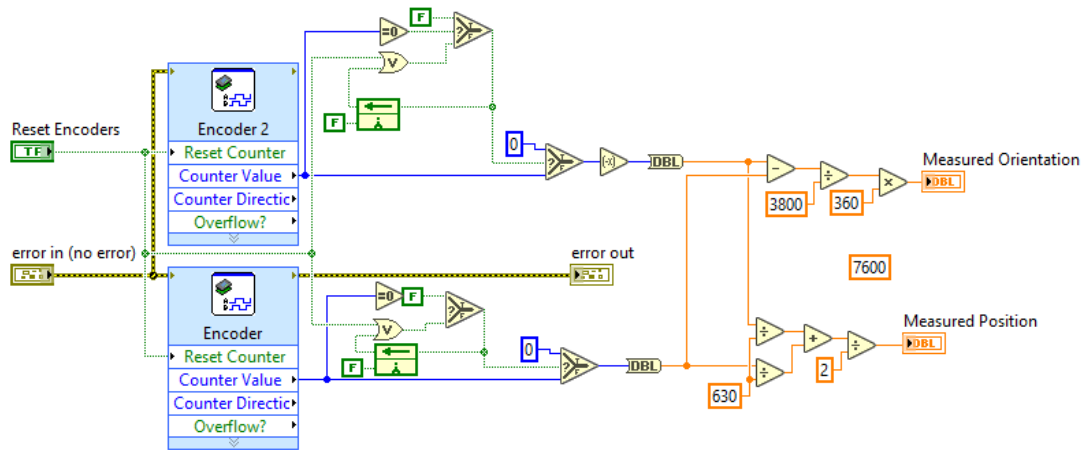


Figura 31. Configuración de Encoder en LabView

Las conexiones de los Encoder se muestran en la tabla 4 (Encoder motor derecho) y tabla 5 (Encoder motor izquierdo) para conectar se utiliza el puerto B del ni myRio.

Encoder 1	Ni MyRio
Señal A	Pin DI0+ Puerto B
Señal B	Pin DI3 Puerto B
GND	Pin VCC Puerto B
VCC	Pin GND Puerto B

Tabla 4 Conexiones de Encoder 1

Encoder 2	Ni MyRio
Señal A	Pin DI5+ Puerto B
Señal B	Pin DI7 Puerto B
GND	Pin VCC Puerto B
VCC	Pin GND Puerto B

Tabla 5 Conexiones de Encoder 2

3.4.2. Conexiones de la comunicación

3.4.2.1. Bluetooth hc-05

El bluetooth hc 05 es un dispositivo con comunicación serial, el cual tiene como propósito el de recibir datos enviados desde una aplicación creada en app inventor para el control del robot conectado al ni myRio(véase figura 32).

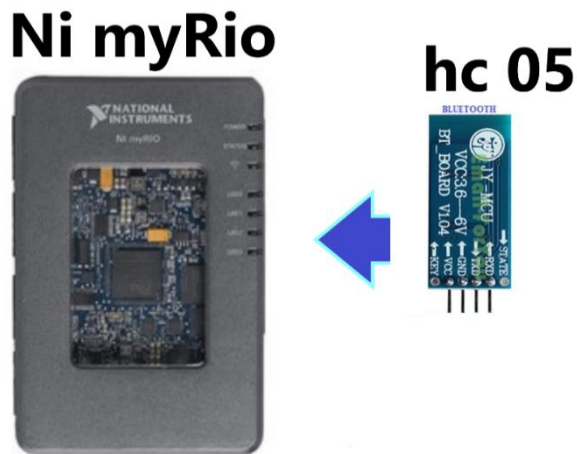


Figura 32. Esquema de ni myRio con bluetooth

En LabView la configuración para el uso del bluetooth es mediante la función de uart perteneciente a la librería de ni myRio, los parámetros son conectados en el puerto b, velocidad de transmisión de 9600, tamaño de los bits de recepción en 8 bits como se muestra en la figura 33.

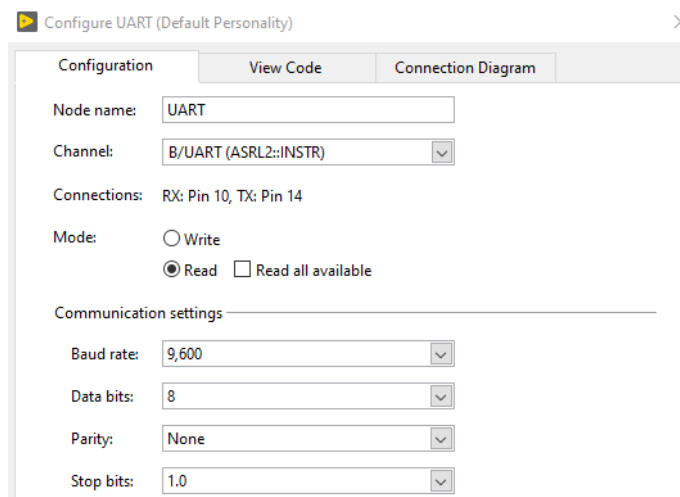


Figura 33. Configuración del uart en LabView

Las conexiones del bluetooth se muestran en la tabla 6 para conectar se utiliza el puerto A del ni myRio.

Encoder 1	Ni MyRio
TX	Pin RX+ Puerto A
RX	Pin TX Puerto A
GND	Pin VCC Puerto A
VCC	Pin GND Puerto A

Tabla 6 Conexiones del bluetooth

3.4.2.2. Aplicativo en app inventor

Para el control de una práctica se utiliza el aplicativo web App inventor el cual permite la creación de aplicaciones compatibles con dispositivos Android, mediante el cual se utiliza la interfaz que se muestra en la figura 34, donde al momento de presionar algún botón enviara un comando mediante el bluetooth dado el caso si presionan adelante envía la letra F, presionando reversa envía la letra B, presionando giro derecho envía la letra R , presionando giro izquierda envía la letra L.



Figura 34. Aplicativo creado en app inventor

3.4.3. Conexiones de la potencia y alimentación

3.4.3.1. Tarjeta controladora de motores

En la parte de potencia para el control de los actuadores de tracción del robot se seleccionó la placa de control o puente h con un integrado A4973 permitiendo conectar y controlar un motor paso a paso, dos motores de CC o dos servos de forma independiente a través del conector MXP como se muestra en la figura 35.

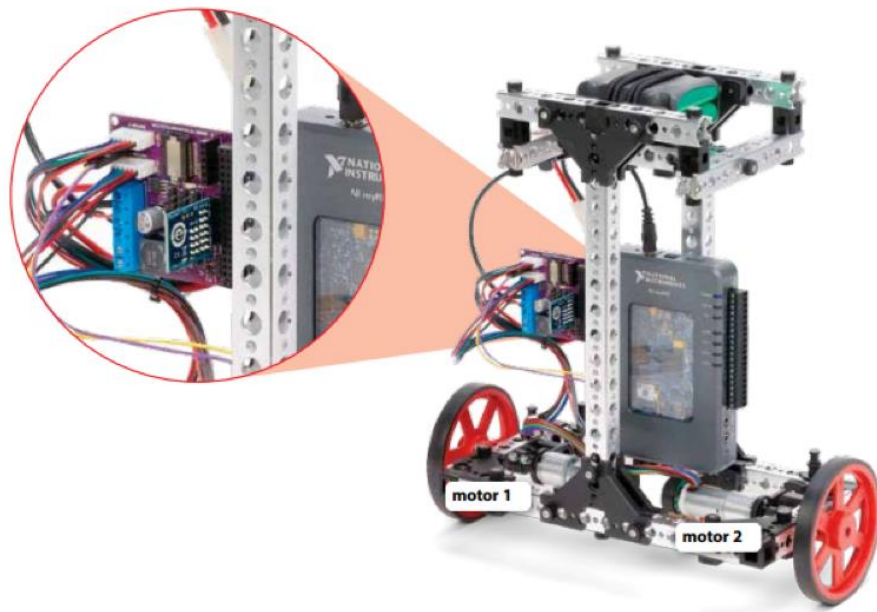


Figura 35. Conexión de la tarjeta ni myRio con la tarjeta de potencia

Para la configuración en LabView es necesario enviar un valor análogo de salida y un dato digital para el control de cada motor, donde el analógico controla la velocidad o fuerza motriz y el dato digital el sentido horario o antihorario del motor como se muestra en la figura 36.

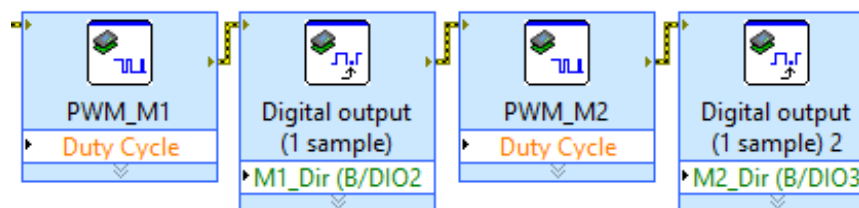


Figura 36. Configuración en LabView para controladora de motores

Las conexiones del controlador de los motores se muestran en la tabla 7 para conectar se utiliza el puerto B del ni myRio.

Controladora de motores	Ni MyRio
Señal A	Pin DI0+ Puerto B
Señal B	Pin DI3 Puerto B
GND	Pin VCC Puerto B
VCC	Pin GND Puerto B
DIR 1	Pin DI2 Puerto B
PWM 1	Pin A01 Puerto B
DIR 2	Pin DI3 Puerto B
PWM2	Pin A02 Puerto B

Tabla 7 Conexiones de controlador de motores

3.4.3.2. Fuente de poder

Para la alimentación del prototipo se utiliza una batería de níquel cadmio de 12 voltios a 2 amperio para la parte de potencia y controladora como se muestra en la figura 37.

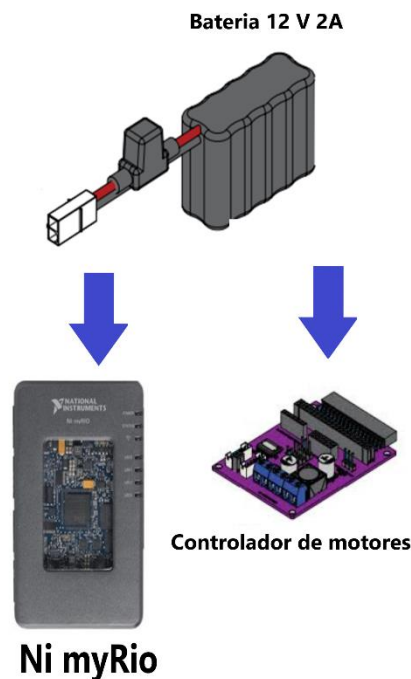


Figura 37. Conexión de la alimentación

3.5. Controlador pid

3.5.1.1. Método de sintonización de ziegler nichols

El método Ziegler Nichols permite la sintonización de controladores pid de forma empírica, sin necesidad de operaciones matemáticas o modelamiento del proceso, publicado en 1942 y desde entonces uso de los métodos más difundidos y utilizado (Ogata, 2015) .

En el proyecto se tiene como sistema de lazo cerrado lo que se muestra en la figura 38 donde la entrada es el punto 0 a estabilizar, la planta es el robot, la retroalimentación es el sensor y como salida los actuadores.

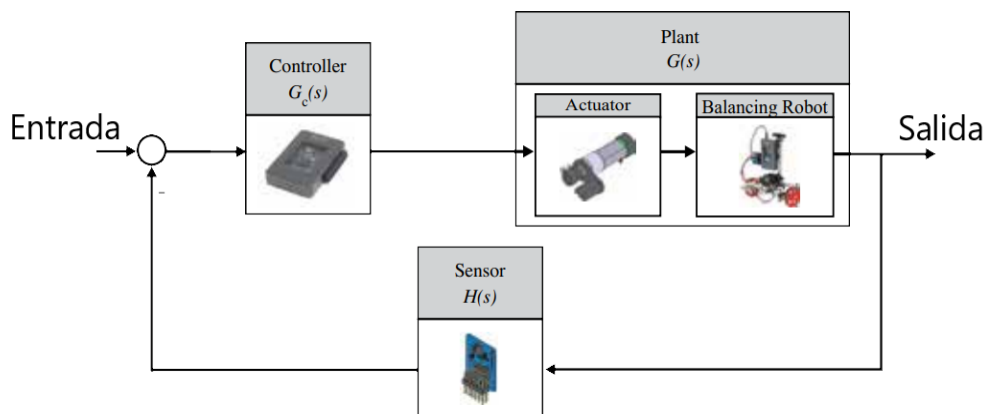


Figura 38. Lazo cerrado del robot auto balance

Este método es mediante el seteo de las variables integral y derivativa en cero y modificar el valor proporcional y aumentar poco a poco la ganancia hasta que el sistema oscile de forma mantenida ante cualquier perturbación. Esta oscilación debe ser lineal, sin saturaciones. En este momento hay que medir la ganancia proporcional, llamada ganancia crítica o K_c , y el periodo de oscilación T_c en segundos (Ogata, 2015).

Una vez medidos estos dos valores, se pueden calcular los parámetros del controlador PID con acción solo proporcional (P), proporcional e integral (PI),

proporcional y derivativa (PD) o proporcional integral y derivativa (PID) (Ogata, 2015):

	KP	Ti	Td
P	$0.50 \cdot k_c$		
PI	$0.45 \cdot k_c$	$0.83 \cdot T_c$	
PID	$0.59 \cdot k_c$	$0.50 \cdot T_c$	$0.125 \cdot T_c$

Tabla 8 Método de ziegler nichols ganancia critica

3.6. Control PID Difuso

3.6.1.1. Reglas del conjunto difuso

Para la instrumentación del prototipo encargada de la adquisición de datos en tiempo real como el valor de inclinación del prototipo se utiliza la herramienta fuzzy designer como se muestra en la figura.

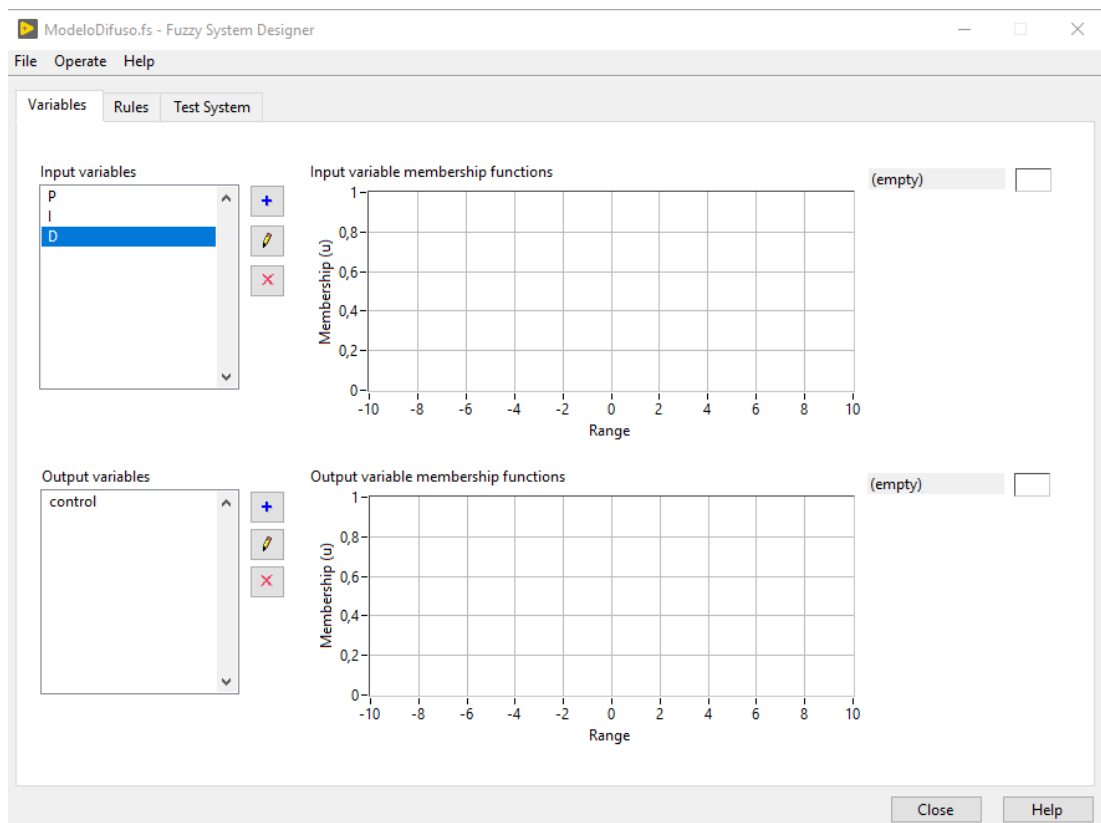


Figura 39. Herramienta para creación de variables, práctica 7.

En la variable proporcional (p) se procede a agregar las 3 reglas de correspondencia indicando cuando es cero, negativo y positivo

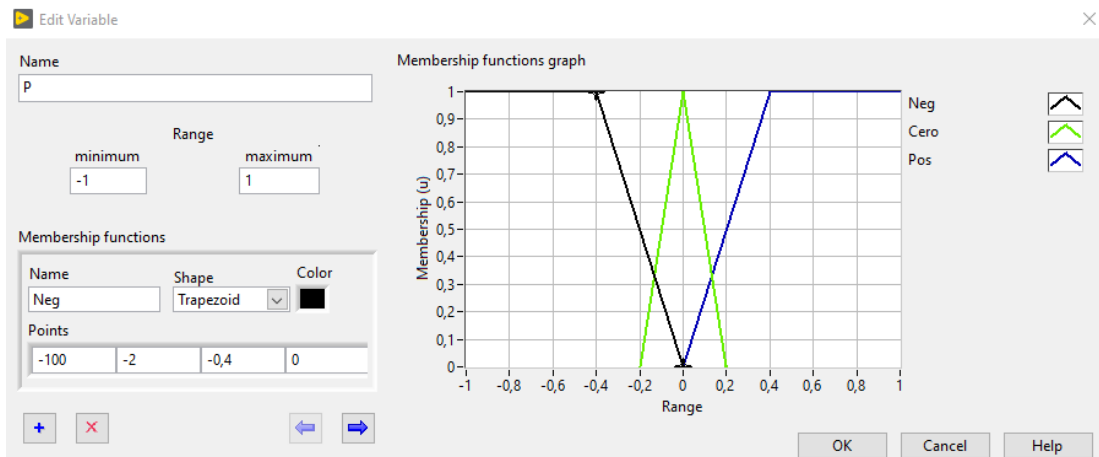


Figura 40. Configuración de rango en variable proporcional, práctica 7.

En la variable integral (i) se procede a agregar las 3 reglas de correspondencia indicando cuando es cero, negativo y positivo

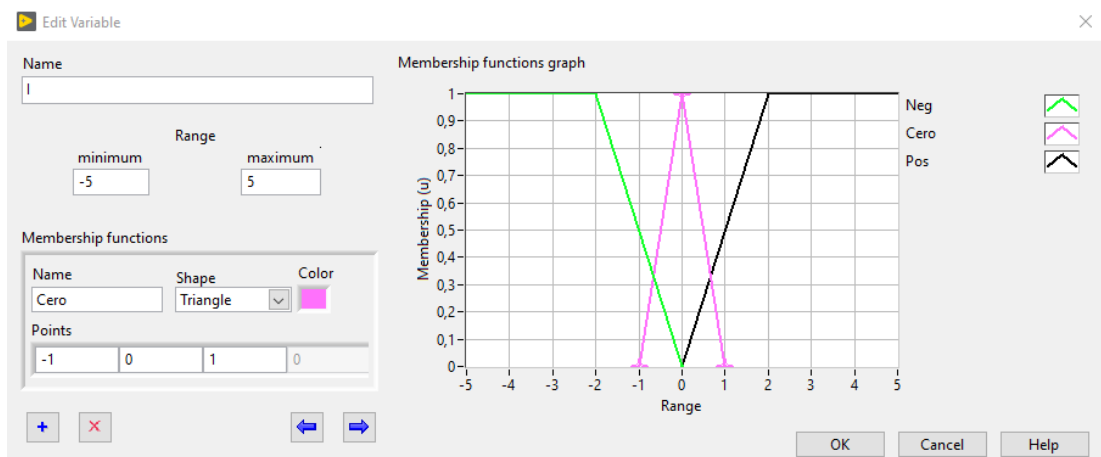


Figura 41. Configuración de rango en variable integral, práctica 7.

En la variable derivativa (d) se procede a agregar las 3 reglas de correspondencia indicando cuando es cero, negativo y positivo

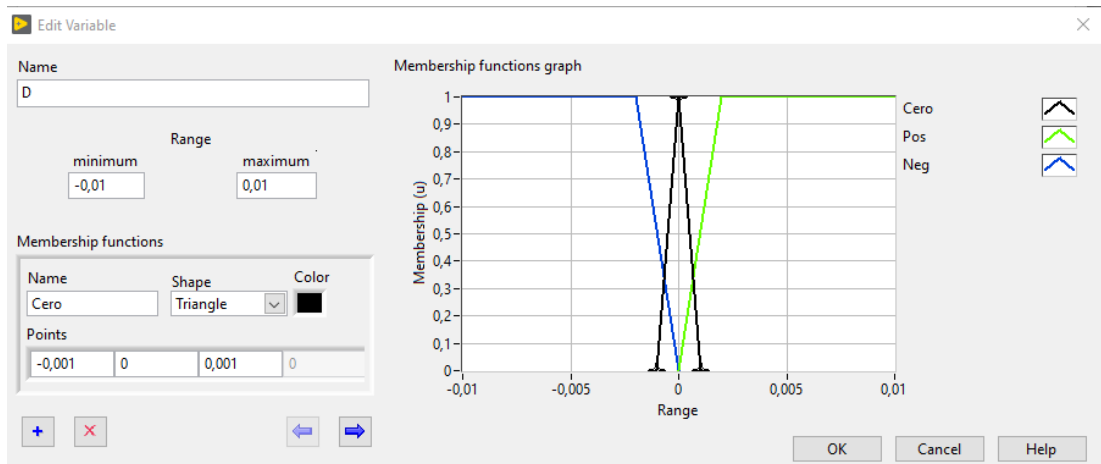


Figura 42. Configuración de rango en variable derivativa, práctica 7.

En la variable para la salida se configuran 5 se procede a agregar las 7 reglas de correspondencia indicando cuando es cero, balance poco (+) , balance poco (-), balance mucho (+), balance mucho (-). Balance largo (+). Balance largo (-) como se muestra en la figura para el control de los motores y sus respectivos cambios de sentidos el rango de operación es -1 a 1 para los actuadores.

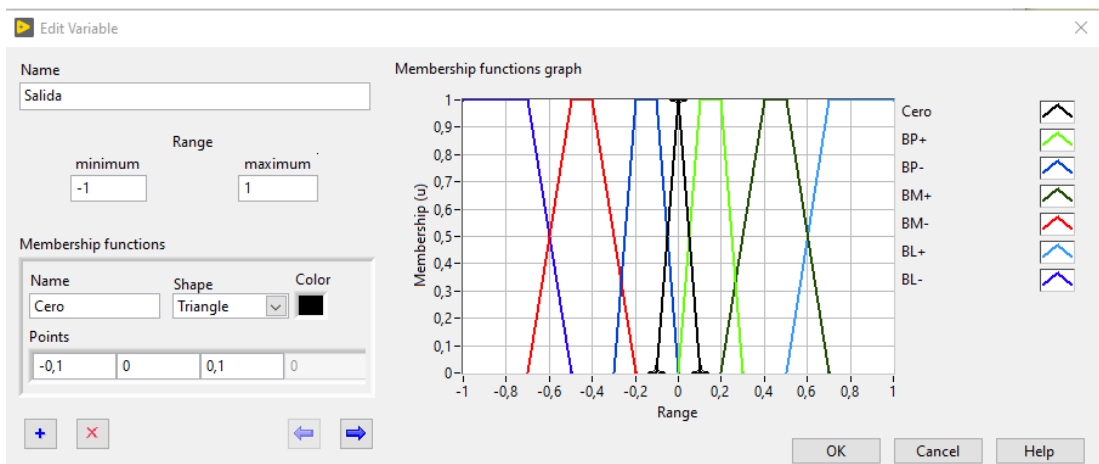


Figura 43. Configuración de rango en variable derivativa, práctica 7.

Las reglas que dominan al conjunto de reglas difusas dependen de las variables lingüísticas de entrada (valores de proporcional, integral y derivativo) teniendo como salida la variable que domina al control de los actuadores del robot como se muestra en la figura.

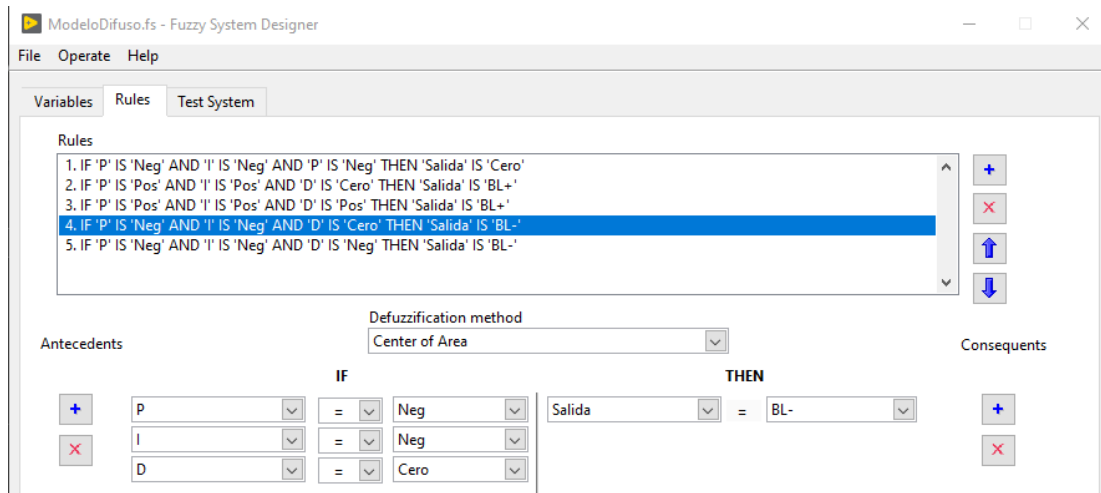


Figura 44. Configuración de reglas difusas, práctica 7.

3.6.1.2. Implementación de controlador pid difuso

Mediante el software LabView se logra la interacción entre el controlador pid y teniendo sus ganancias como entradas al bloque fuzzy como se muestra en la figura.

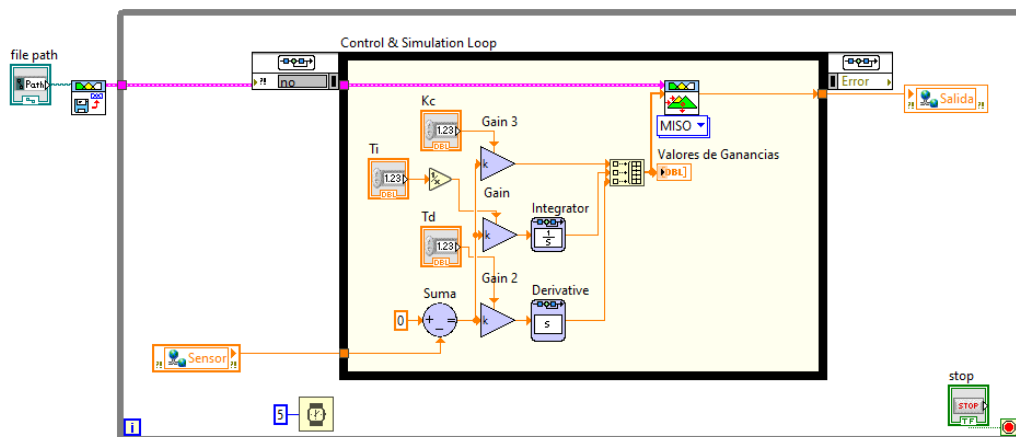
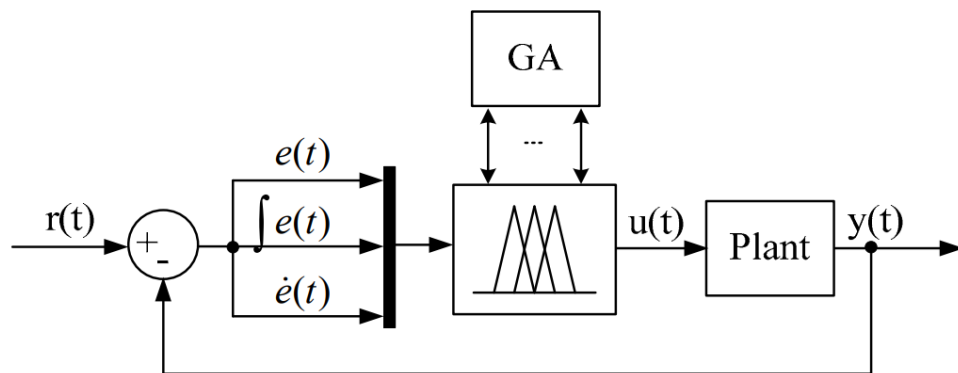




Figura 45. PID FUZZY en LabView

4. MANUAL DE PRACTICAS

Práctica 1: Lectura de acelerómetros con software de instrumentación virtual LabView.

	GUIA DE PRÁCTICA		
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil		
TÍTULO DE LA PRÁCTICA	Lectura de acelerómetros con software de instrumentación virtual LabView.		

1. Objetivos

- Reconocer las principales funciones LabView
- Programar en la herramienta de LabView para ni myRio
- Realizar la conexión entre la ni myRio y el módulo acelerómetro
- Configurar modulo acelerómetro para lectura de eje x, y ,z

2. Planteamiento del problema.

- Elaborar una herramienta virtual con LabView para la lectura de un sensor acelerómetro mediante el panel frontal de LabView. El diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 46.

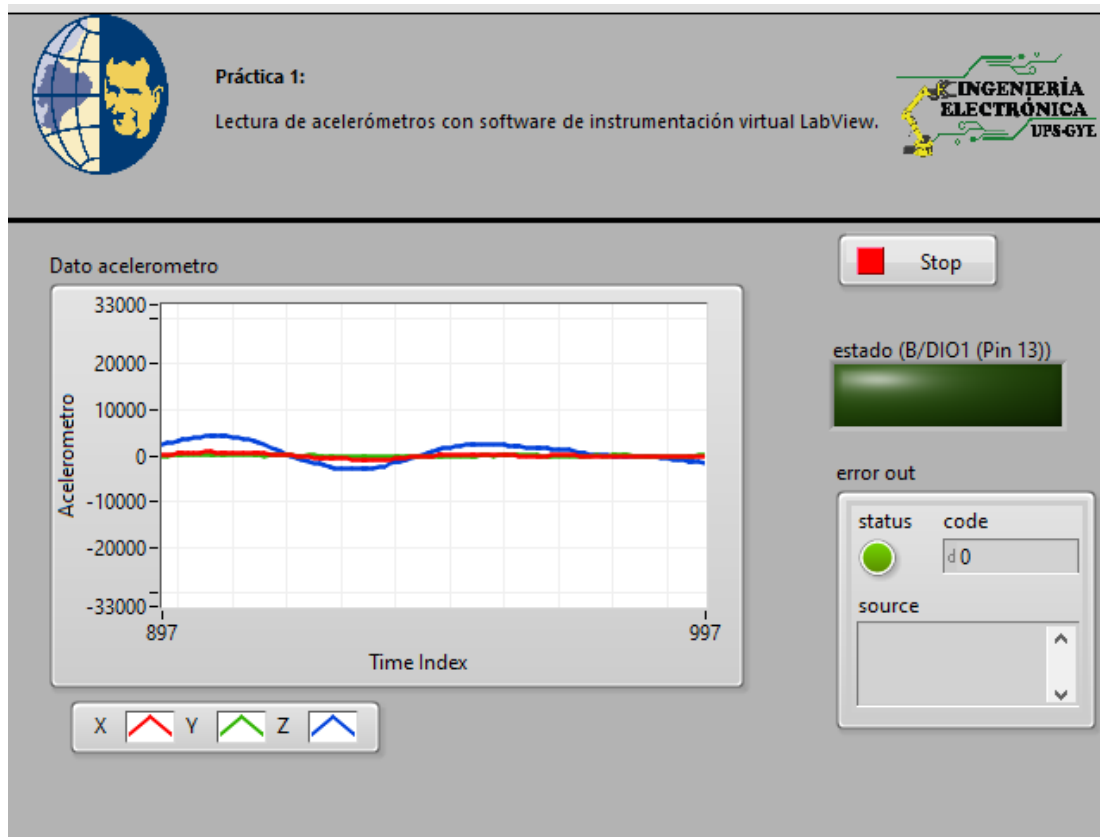




Figura 46. Diseño propuesto para PRÁCTICA 1

3. Instrucciones.

- Desarrollar los pasos en el anexo 3
- Desarrollar los pasos en el anexo 4.
- Seguir los pasos descritos en el anexo 4.1 sobre la resolución de la práctica 1.

Práctica 2: Scada mediante wifi de acelerómetros con software de instrumentación virtual LabView.

	GUIA DE PRÁCTICA	
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil	
TÍTULO DE LA PRÁCTICA	Scada mediante wifi de acelerómetros con software de instrumentación virtual LabView...	

1. Objetivos

- Configurar una red inalámbrica para la carga del programa
- Realizar la conexión entre la ni myRio y el computador por wifi
- Realizar la lectura del acelerómetro de manera remota.

2. Planteamiento del problema.

- Desarrollar una aplicación en el software LabView configurando una red wifi en el dispositivo ni myRio para la visualización de los valores de un sensor acelerómetro conectado al dispositivo , el diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 47.





Figura 47. Diseño propuesto para PRÁCTICA 2

3. Instrucciones.

- Desarrollar los pasos en el anexo 3 ensamble de prototipo
- Desarrollar los pasos en el anexo 4 pasos previos a practicas
- Desarrollar los pasos en el anexo 5 configuración de red inalámbrica
- Seguir los pasos descritos en el anexo 4.2 sobre la resolución de la práctica 2.

Práctica 3: Conversión analógica con sensor ultrasónico.

	GUIA DE PRÁCTICA	
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil	
TÍTULO DE LA PRÁCTICA	Conversión analógica con sensor ultrasónico.	

1. Objetivos

- Conectar mediante red inalámbrica le ni myRio
- Realizar un vi para la visualización de los valores del acelerómetro.
- Configurar el protocolo i2c para conectar el módulo.

2. Planteamiento del problema.

- Implementar un aplicativo en LabView capaz de realizar la conversión analógica de dos sensores ultrasónicos de un valor de voltaje a centímetros mediante un ni myRio con el Pitsco Tetrix, el diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 48.

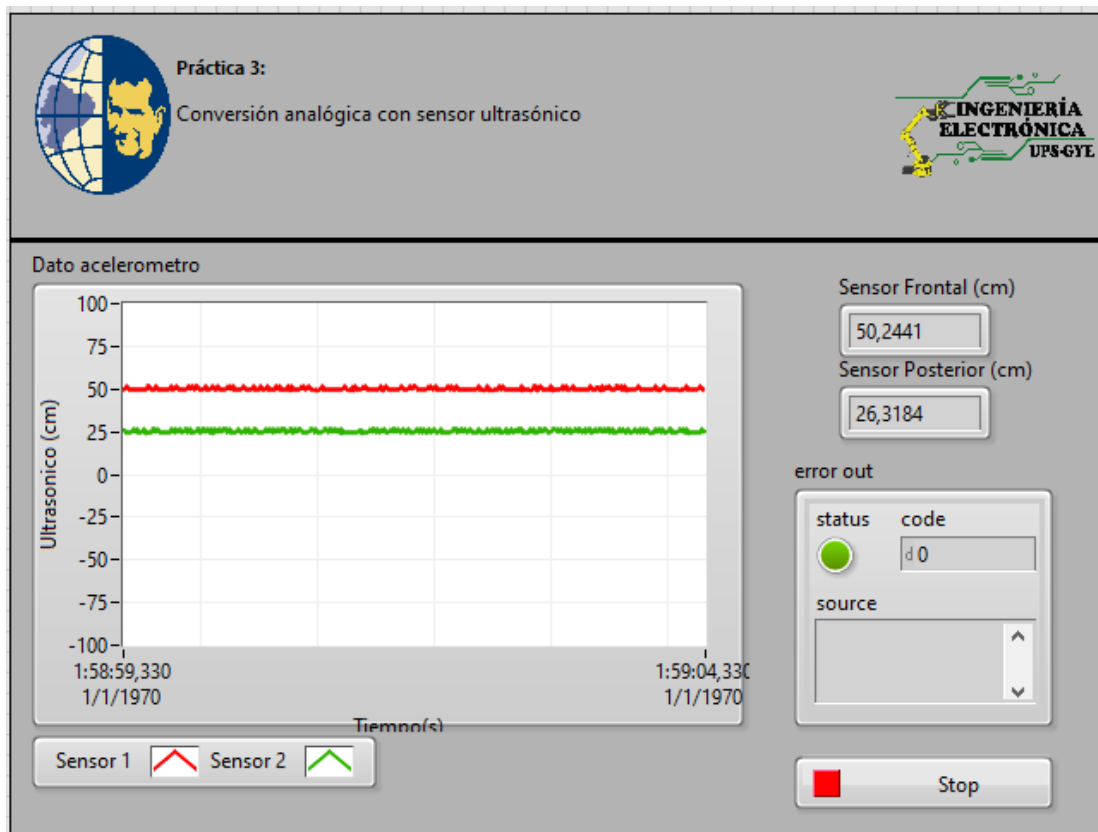




Figura 48. Diseño propuesto para PRÁCTICA 1

3. Instrucciones.

- Desarrollar los pasos en el anexo 3 ensamble de prototipo
- Desarrollar los pasos en el anexo 4 pasos previos a practicas
- Seguir los pasos descritos en el anexo 4.3 sobre la resolución de la práctica 3

Práctica 4: Scada mediante wifi de lecturas analógica con sensor ultrasónico.

	GUIA DE PRÁCTICA	
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil	
TÍTULO DE LA PRÁCTICA	Scada mediante wifi de lecturas analógica con sensor ultrasónico	

1. Objetivos

- Configurar una red inalámbrica en él ni myRio .
- Programar él ni myRio de manera inalámbrica.
- Desarrollar un vi para la lectura de dos sensores ultrasónicos.
- Implementar una aplicación usando la herramienta para él ni myRio.

2. Planteamiento del problema.

- Desarrollar un instrumento virtual para el prototipo permitiendo la conexión mediante wifi para el monitoreo de dos sensores ultrasónicos de un valor de voltaje a centímetros mediante un ni myRio con el Pitsco Tetrax, el diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 49.

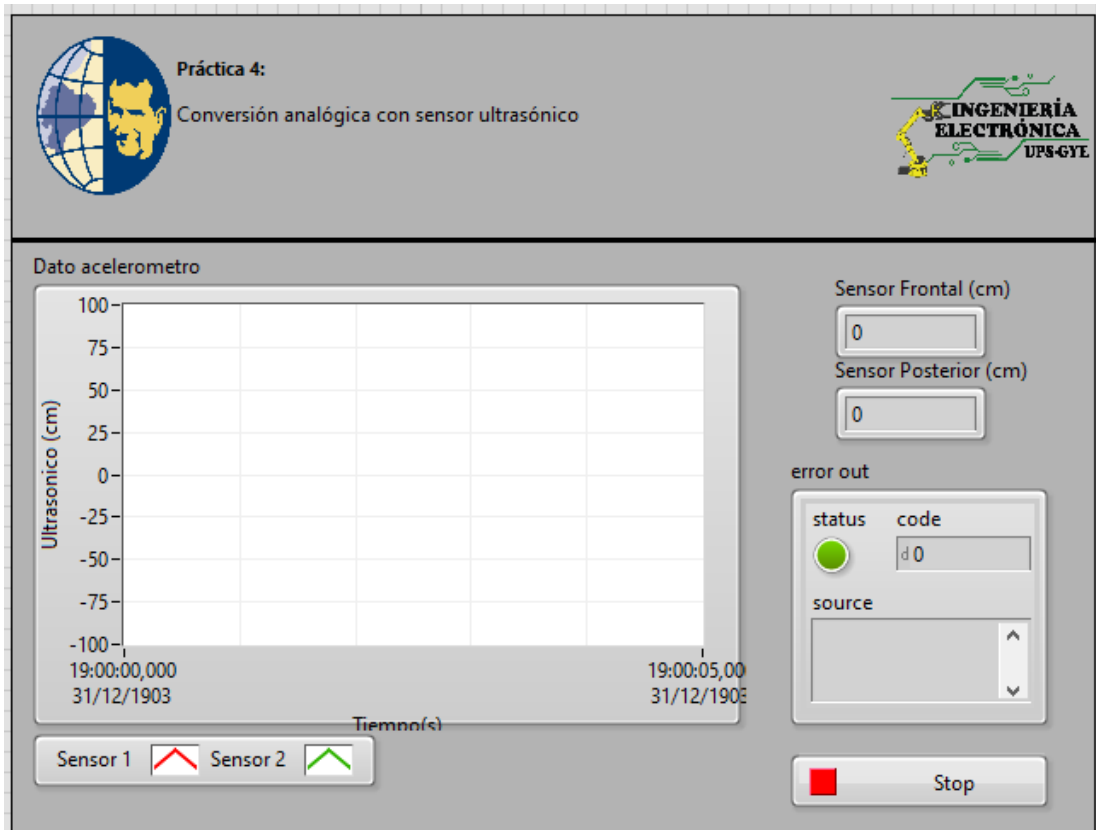

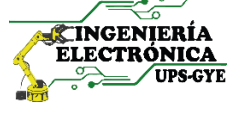


Figura 49. Diseño propuesto para PRÁCTICA 1

3. Instrucciones.

- Desarrollar los pasos en el anexo 3 ensamble de prototipo
- Desarrollar los pasos en el anexo 4 pasos previos a practicas
- Desarrollar los pasos en el anexo 5 configuración de red inalámbrica
- Seguir los pasos descritos en el anexo 4.4 sobre la resolución de la práctica 4

Práctica 5: PID robot auto balanceado.

	<h3>GUIA DE PRÁCTICA</h3>	
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil	
TÍTULO DE LA PRÁCTICA	PID robot auto balanceado.	

1. Objetivos

- Desarrollar una aplicación para el control del robot mediante pid.
- Sintonizar pid utilizando ziegler nichols.
- Programar un control en lazo cerrado en una ni myRio.
- Realizar scada de los valores de entrada y salida del controlador pid.

2. Planteamiento del problema.

- Implementar un controlador pid para la estabilización de un robot auto balance mediante el uso de la herramienta de control en LabView con él ni myRio, la retroalimentación es el sensor de acelerómetro del ni myRio , la plantilla del diseño propuesto se muestra en la figura 50.

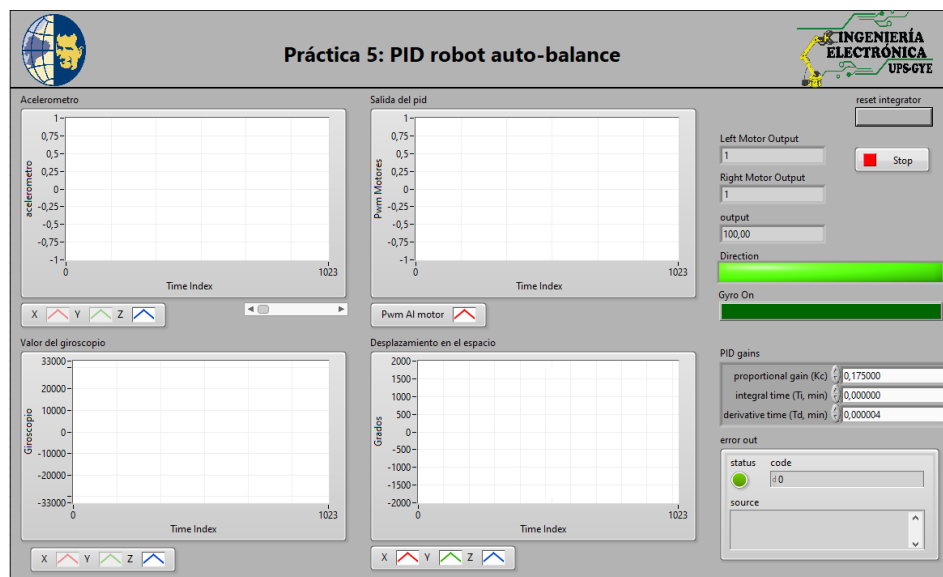




Figura 50. Diseño propuesto para PRÁCTICA 5

3. Instrucciones.

- Desarrollar los pasos en el anexo 3 ensamble de prototipo
- Desarrollar los pasos en el anexo 4 pasos previos a practicas
- Desarrollar los pasos en el anexo 5 configuración de red inalámbrica
- Seguir los pasos descritos en el anexo 4.5 sobre la resolución de la práctica 5

Práctica 6: PID robot auto balanceado evitando obstáculos con sensores ultrasónicos...

	<h3>GUIA DE PRÁCTICA</h3>	
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil	
TÍTULO DE LA PRÁCTICA	PID robot auto balanceado evitando obstáculos con sensores ultrasónicos...	

1. Objetivos

- Desarrollar un aplicativo en LabView para un robot balance esquivo obstáculos.
- Sintonizar pid utilizando ziegler nichols.
- Realizar conversión analógica para la lectura de los sensores ultrasónicos.
- Realizar scada de los valores del proceso.

2. Planteamiento del problema.

- Implementar un aplicativo mediante el cual se logre estabilizar un robot balancín y logre esquivar obstáculos en un sistema embebido ni myRio y LabView, la plantilla para practica se muestra en la figura 51.

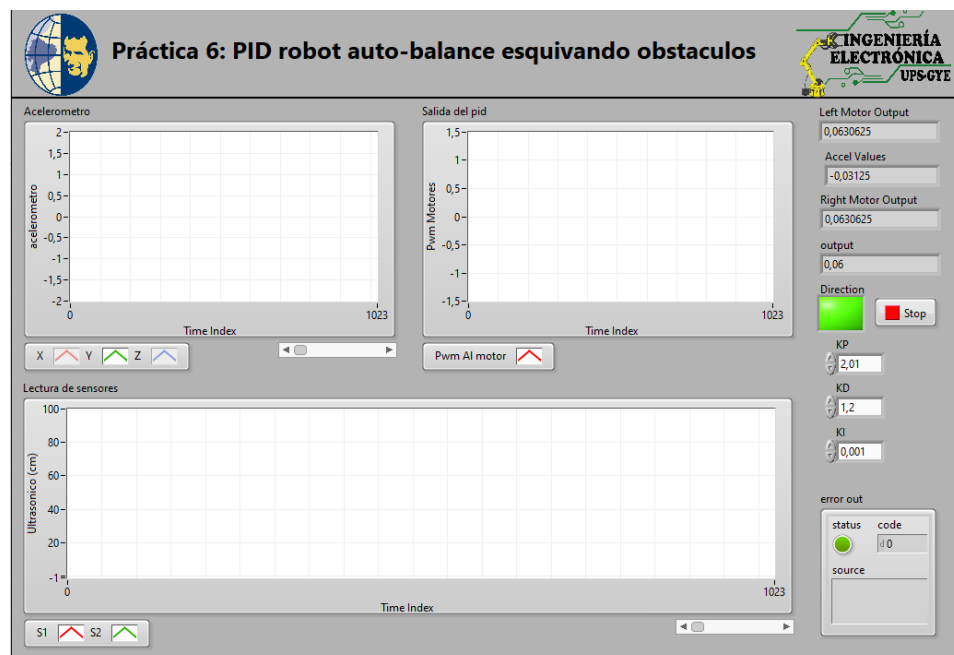




Figura 51. Diseño propuesto para PRÁCTICA 6

3. Instrucciones.

- Desarrollar los pasos en el anexo 3 ensamble de prototipo
- Desarrollar los pasos en el anexo 4 pasos previos a practicas
- Desarrollar los pasos en el anexo 5 configuración de red inalámbrica
- Seguir los pasos descritos en el anexo 4.6 sobre la resolución de la práctica 6.

Práctica 7: PID Difuso robot auto balance.

	GUIA DE PRÁCTICA	
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil	
TÍTULO DE LA PRÁCTICA	PID Difuso robot auto balance	

1. Objetivos

- Desarrollar un conjunto de reglas mediante la herramienta de LabView
- Programar un algoritmo de control pid difuso en LabView
- Implementar un conjunto de reglas difusas en LabView
- Controlar los actuadores mediante un instrumento virtual(vi)

2. Planteamiento del problema.

- Desarrollar un instrumento virtual en LabView para el funcionamiento de un robot balance mediante un controlador pid fuzzy, utilizando la herramienta de **fuzzy designer** , el diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 52.

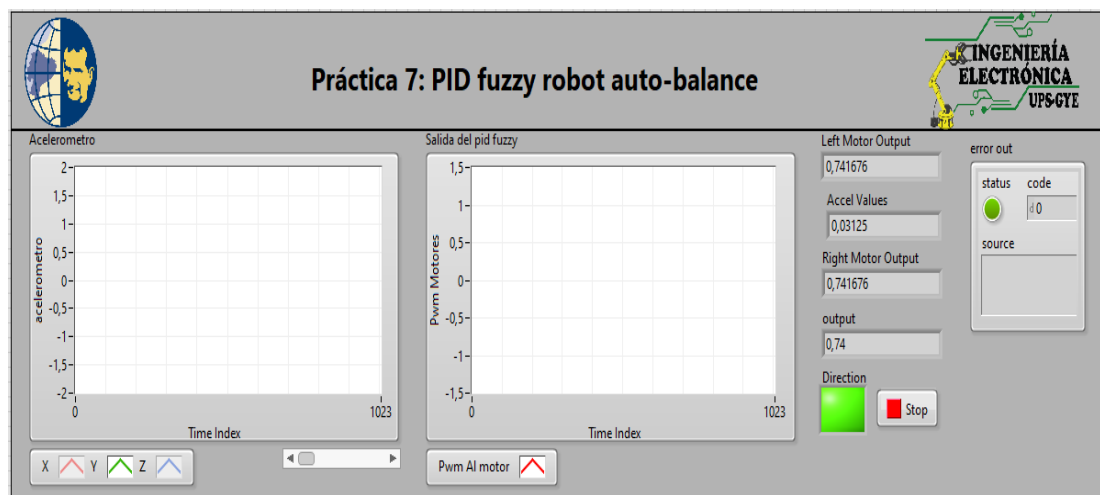




Figura 52. Diseño propuesto para PRÁCTICA 7

3. Instrucciones.

- Seguir los pasos descritos en introducción a las prácticas en anexo 3.0.
- Seguir los pasos en el anexo 1.2 sobre el ensamblado del prototipo.
- Seguir los pasos descritos en el anexo 3.7 sobre la resolución de la práctica 7.

Práctica 8: Robot auto balanceado radio controlado mediante comunicación bluetooth.

	GUIA DE PRÁCTICA	
CARRERA: Ingeniería Electrónica	ASIGNATURA: Electiva, Robótica móvil	
TÍTULO DE LA PRÁCTICA	Robot auto balanceado radio controlado mediante comunicación bluetooth.	

1. Objetivos

- Desarrollar un aplicativo para recepción de datos seriales mediante bluetooth.
- Programar en la herramienta ni myRio el bloque de uart
- Realizar la conexión entre la ni myRio y los actuadores
- Controlar el prototipo mediante él envío de valores en un buffer serial.

2. Planteamiento del problema.

- Desarrollar una aplicación en el software LabView con la herramienta para la tarjeta ni myRio para el control de un robot balance mediante el envío de datos por un puerto COM conectado por bluetooth al dispositivo, el diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 53.

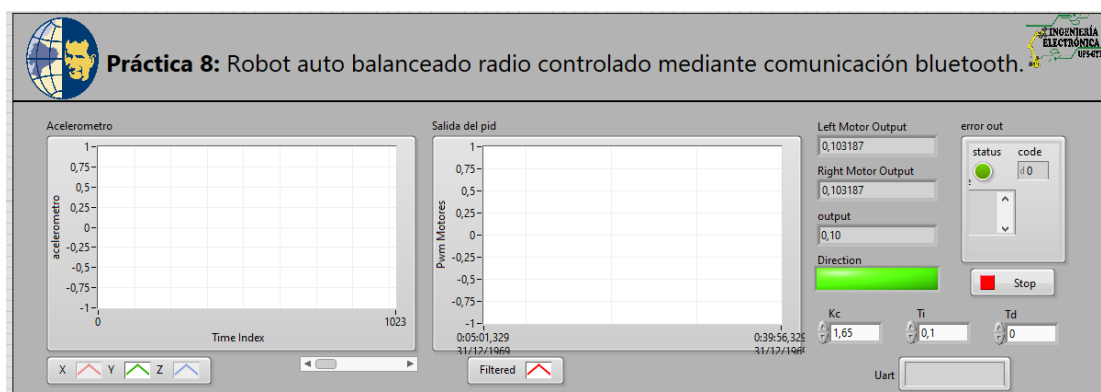




Figura 53. Diseño propuesto para PRÁCTICA 8

3. Instrucciones.

- Seguir los pasos en el anexo 3 sobre el ensamblado del prototipo.
- Seguir los pasos descritos en el anexo 3.8 sobre la resolución de la práctica 8.

Práctica 9: Robot auto balanceado radio controlado mediante comunicación wifi.

	<p>GUIA DE PRÁCTICA</p>	
<p>CARRERA: Ingeniería Electrónica</p>	<p>ASIGNATURA: Electiva, Robótica móvil</p>	
<p>TÍTULO DE LA PRÁCTICA</p>	<p>Robot auto balanceado radio controlado mediante comunicación wifi.</p>	

1. Objetivos

- Desarrollar un aplicativo para recepción de datos mediante wifi.
- Programar el control de un robot auto balance mediante pulsantes.
- Realizar la conexión entre la ni myRio y el computador mediante wifi.

2. Planteamiento del problema.

- Desarrollar una aplicación en el software LabView con la herramienta para la tarjeta ni myRio para el control de los motores Dc y servo motores del prototipo mediante el panel frontal de LabView. El diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 54.

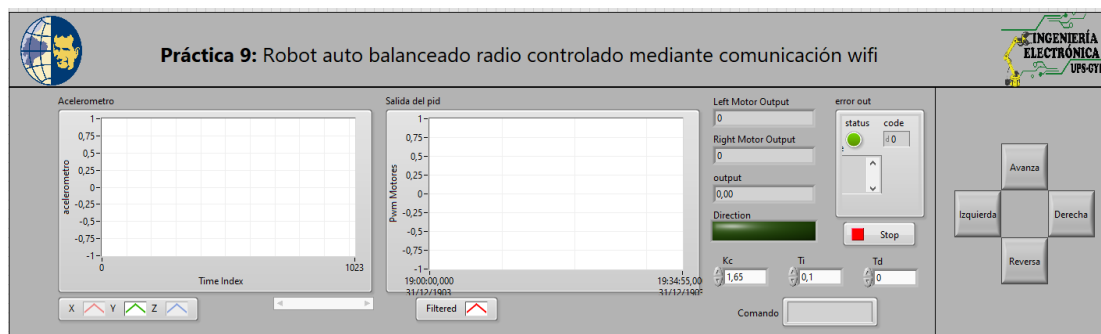

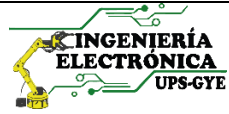


Figura 54. Diseño propuesto para PRÁCTICA 9

3. Instrucciones.

- Seguir los pasos descritos en introducción a las prácticas en anexo 4.0.
- Seguir los pasos en el anexo 3. sobre el ensamblado del prototipo.
- Seguir los pasos descritos en el anexo 4.9 sobre la resolución de la práctica 9.

Práctica 10: Robot auto balanceado radio controlado mediante aplicativo en aplicación App Inventor.

	<p>GUIA DE PRÁCTICA</p>	
<p>CARRERA: Ingeniería Electrónica</p>	<p>ASIGNATURA: Electiva, Robótica móvil</p>	
<p>TÍTULO DE LA PRÁCTICA</p>	<p>Robot auto balanceado radio controlado mediante aplicativo en aplicación App Inventor.</p>	

1. Objetivos

- Desarrollar un aplicativo en app inventor
- Adquirir valores enviados desde un hc-05
- Desarrollar un control para el robot autobalance mediante bluetooth
- Controlar los actuadores mediante un instrumento virtual(vi)

2. Planteamiento del problema.

- Desarrollar una aplicación en app inventor el cual envíe comandos al robot balance mediante comunicación serial uart para el control de este.
- Desarrollar un aplicativo en el software LabView con la herramienta para la tarjeta ni myRio para la recepción de los comandos y funcionamiento del robot, el diseño propuesto del instrumento virtual en LabView con la herramienta de ni myRio se muestra en la figura 55.

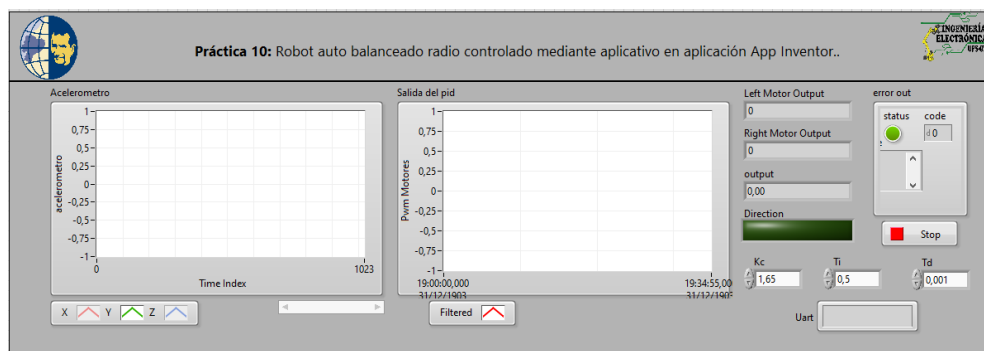


Figura 55. Diseño propuesto para PRÁCTICA 10

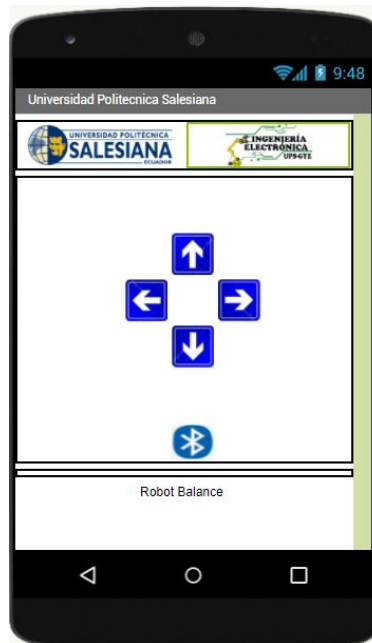


Figura 56. Diseño propuesto aplicativo para PRÁCTICA 10

3. Instrucciones.

- Seguir los pasos descritos en introducción a las prácticas en anexo 4.0.
- Seguir los pasos en el anexo 3. sobre el ensamblado del prototipo.
- Seguir los pasos descritos en el anexo 4.10 sobre la resolución de la práctica 10.

5. RESULTADOS

- Se elaboro un módulo robótico con tecnología Pitsco Tetrax para prácticas de laboratorio con un controlador lógico "ni myRio 1900 ",destinado a ser utilizados en las diferentes prácticas planteadas en el documento en las materias enfocadas a robótica móvil impartidas en la Universidad Politécnica Salesiana sede Guayaquil.
- Se planteo un modelo el cual facilitará el montaje de los elementos tales como sensores, actuadores, , batería y controlador lógico de manera eficiente.
- Se logra la compilación mediante el software labview y su herramienta ni myRio.
- Mediante las pruebas hechas se tuvo un control optimo mediante un host web configurado en la tarjeta controladora y la red inalámbrica que suministra el equipo.
- Se hizo la entrega del equipo, en este caso del robot explorador y un case protector para el robot, tal como cargador y accesorios del prototipo, un módulo completamente funcional para ser utilizado en las clases que se impartan en la Universidad Politécnica Salesiana sede Guayaquil.

5.1. RESULTADOS DE PRÁCTICAS

5.1.1. Resultados en práctica 1 - Lectura de acelerómetros con software de instrumentación virtual LabView.

El resultado de la compilación de la primera practica se muestra en la figura 57.

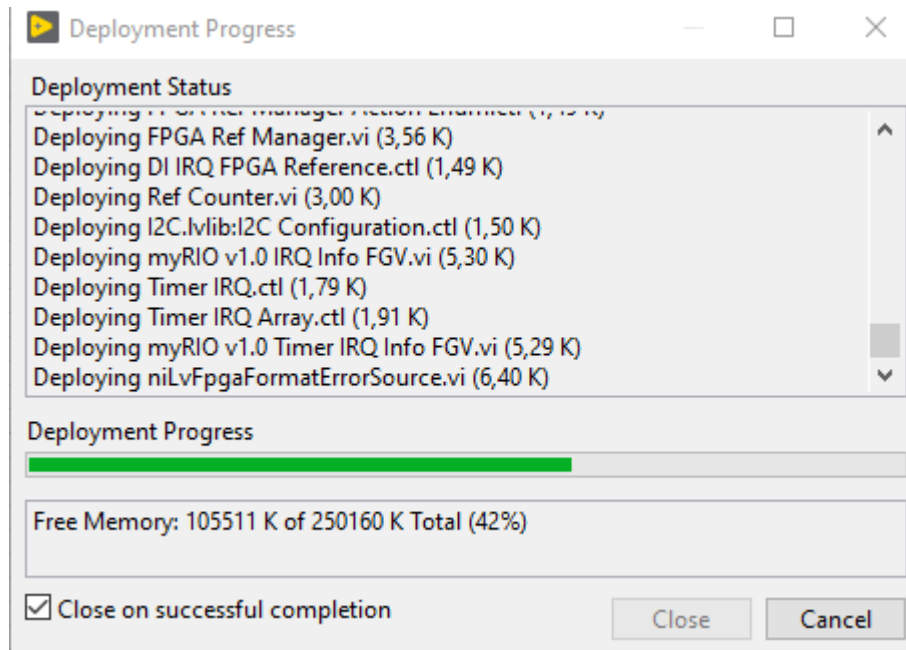


Figura 57. Resultados Practica 1 – compilación

Al compilar, el prototipo realiza la adquisición de los valores en la posición en x, y, z, conectado al computador mediante un cable USB.

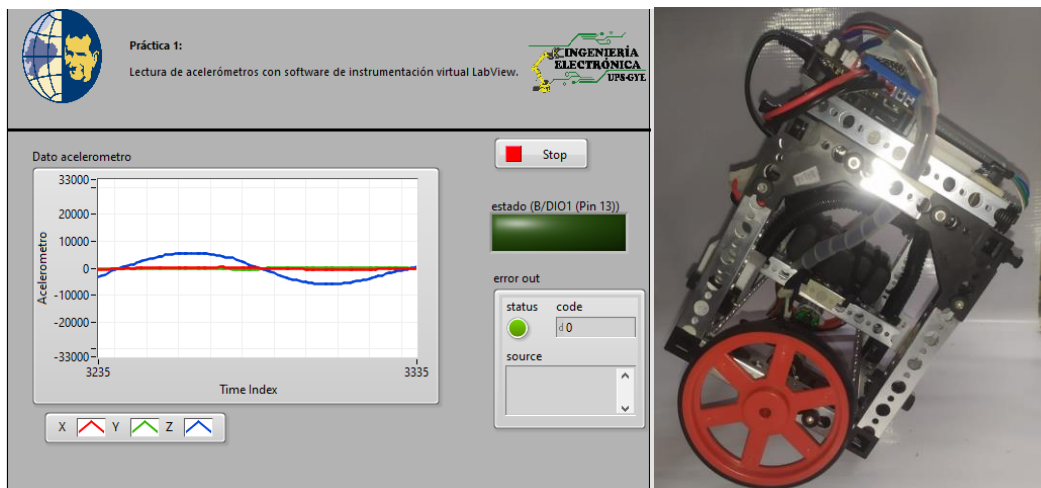


Figura 58. Resultados Practica 1 – movimiento del prototipo de forma manual para lectura de valores en ejes x y z

5.1.2. Resultados en práctica 2 - Scada mediante wifi de acelerómetros con software de instrumentación virtual LabView

El resultado de la compilación de la primera practica se muestra en la figura 57.

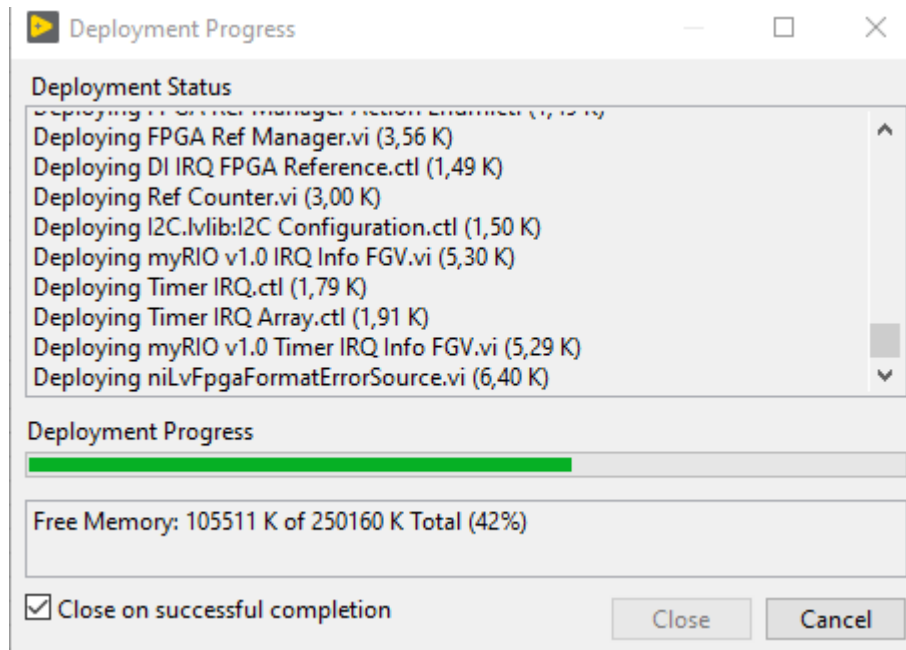


Figura 59. Resultados Practica 2 – compilación

Al compilar, el prototipo realiza la adquisición de los valores en la posición en x, y, z, conectado al computador mediante conexión inalámbrica wifi

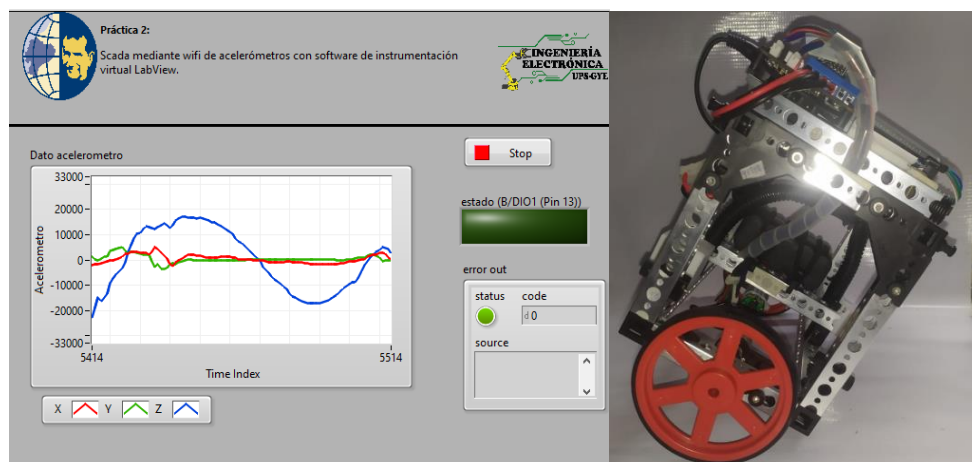


Figura 60. Resultados Practica 2 – movimiento del prototipo de forma manual para lectura de valores en ejes x y z

5.1.3. Resultados en práctica 3 - Conversión analógica con sensor ultrasónico.

Al compilar y cargar el código a la myRio se tiene como resultado la visualización de la distancia del sensor ultrasónico en un indicador gráfico , el equipo está conectado de manera USB al computador.

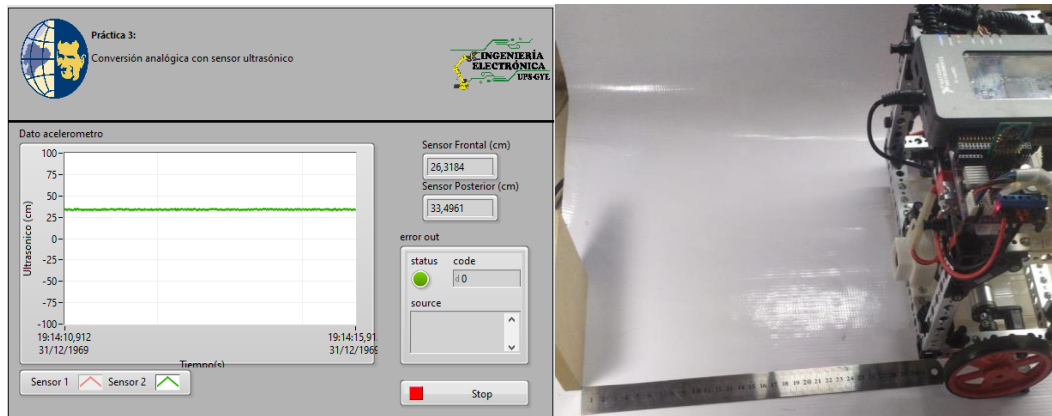


Figura 61. Resultados Practica 3 – adquisición de sensores ultrasónico

5.1.4. Resultados en práctica 4 - Scada mediante wifi de lecturas analógica con sensor ultrasónico

Al compilar y cargar el código a la myRio se tiene como resultado la visualización de la distancia del sensor ultrasónico en un indicador gráfico , el equipo está conectado de manera inalámbrica al computador.

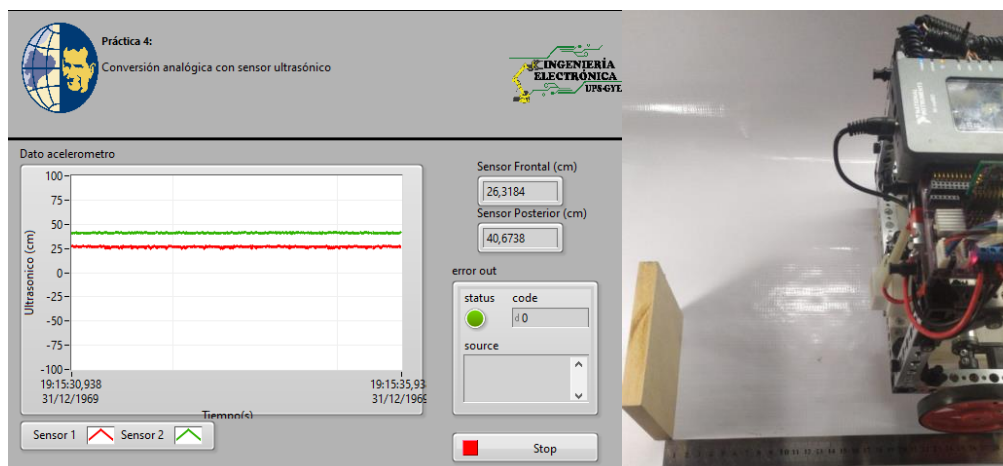


Figura 62. Resultados Practica 4 – adquisición de sensores ultrasónico

5.1.5. Resultados en práctica 5 - PID robot auto balanceado.

Al compilar y cargar el código a la myRio se obtiene como resultado la estabilización del robot auto balance teniendo como interfaz gráfica el instrumento virtual que se muestra en la imagen 63.

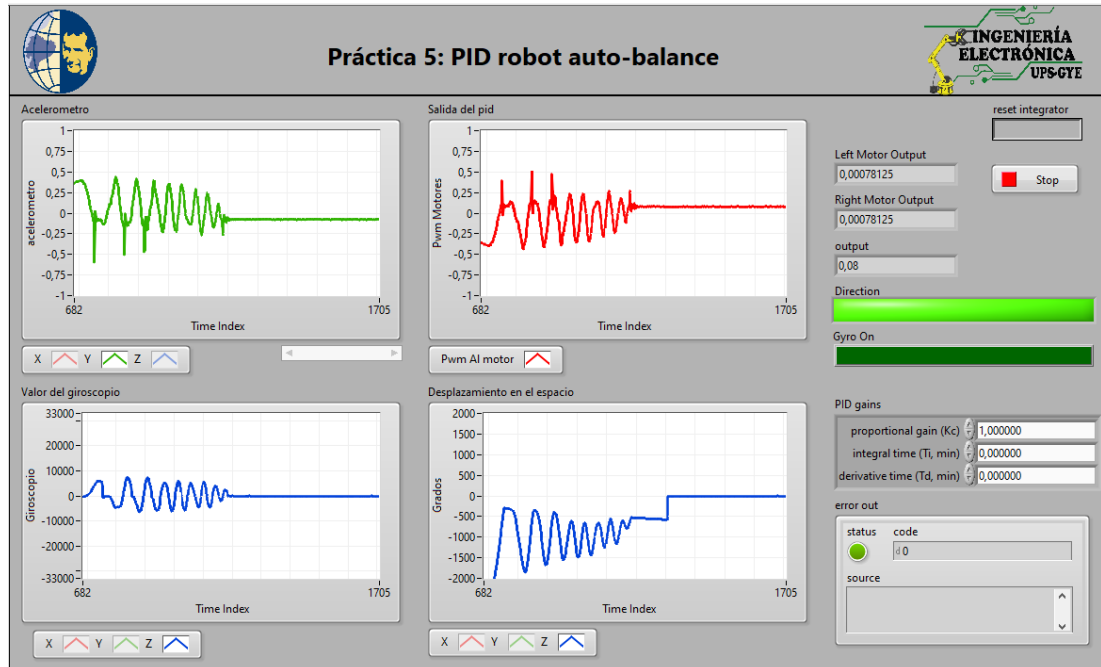


Figura 63. Resultados Practica 5 PID robot auto balanceado.

En la figura 64 se muestra el funcionamiento, el punto de referencia es el e eje y donde busca un punto cero para lograr la estabilidad del prototipo mediante el controlador pid.

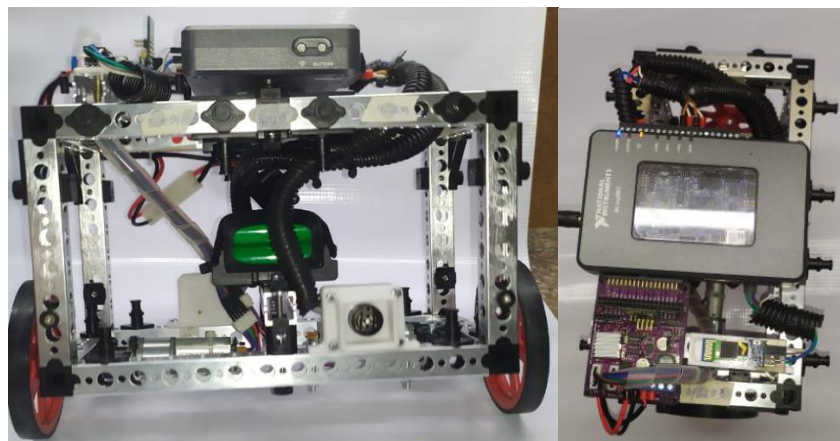


Figura 64. Resultados Practica 5 PID robot auto balanceado.

5.1.6. Resultados en práctica 6 - PID robot auto balanceado evitando obstáculos con sensores ultrasónicos

En la ejecución de la sexta practica se desarrolló un instrumento en labview el cual sirve para el control del Pitsco Tetrax mediante la interacción con los pulsantes, barras y botones para enviar comando de manera inalámbrica del computador al ni myRio en el panel frontal como se muestra en la figura 65.

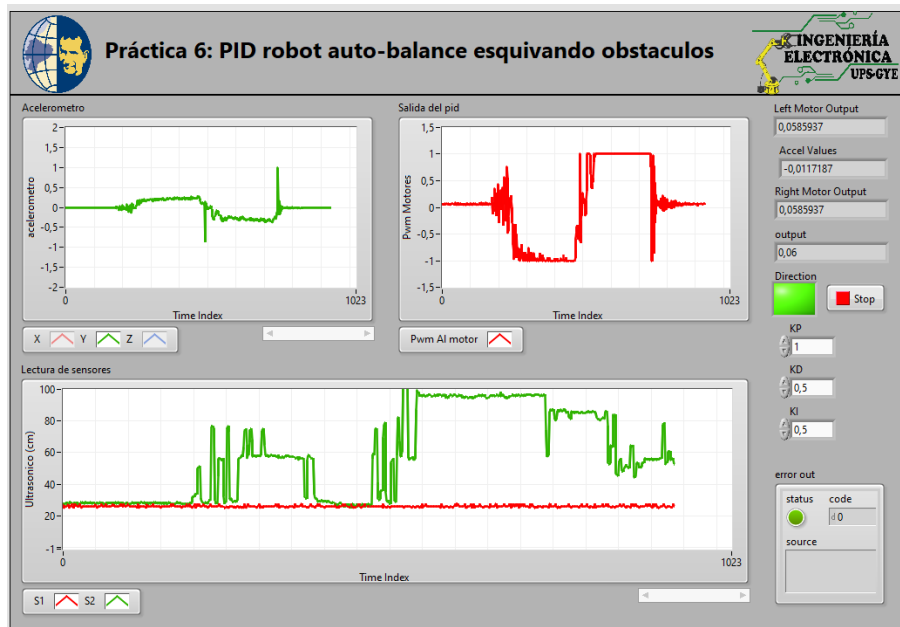


Figura 65. Resultados Practica 6 – PID robot auto balanceado evitando obstáculos con sensores ultrasónicos

Al ejecutar el controlador el prototipo comienza a avanzar y al encontrar un camino con una pared u obstáculo gira hasta encontrar un camino libre como se muestra en la figura 66.

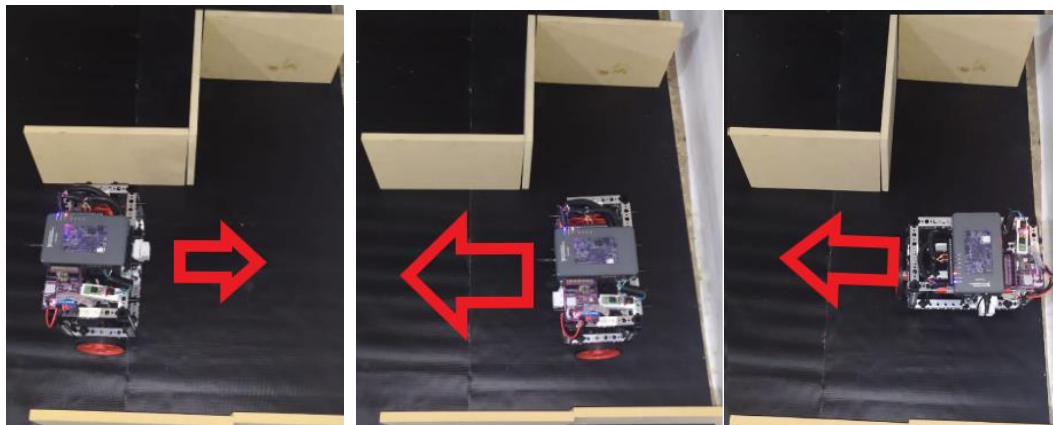


Figura 66. Resultados Practica 6 – movimientos del robot auto balanceado evitando obstáculos con sensores ultrasónicos

5.1.7. Resultados en práctica 7 - PID Difuso robot auto balance.

En la ejecución de la séptima practica se desarrolló un instrumento virtual mediante el uso de labview con la herramienta ni myRio para la implementación del controlador difuso como se muestra en la figura 66 donde un vi principal se encarga del control de los actuadores y el otro vi secundario del cómputo del controlador.

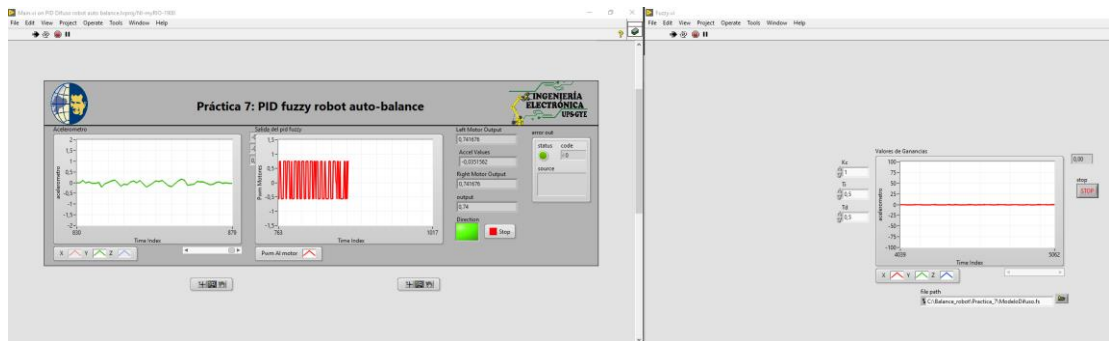


Figura 67. Resultados Practica 6 – PID Difuso robot auto balance.

Una prueba que se realiza es mediante la herramienta de diseñador difuso la cual nos permite simular el valor que tendrá las salidas variando el valor de entrada como se muestra en la figura 68.

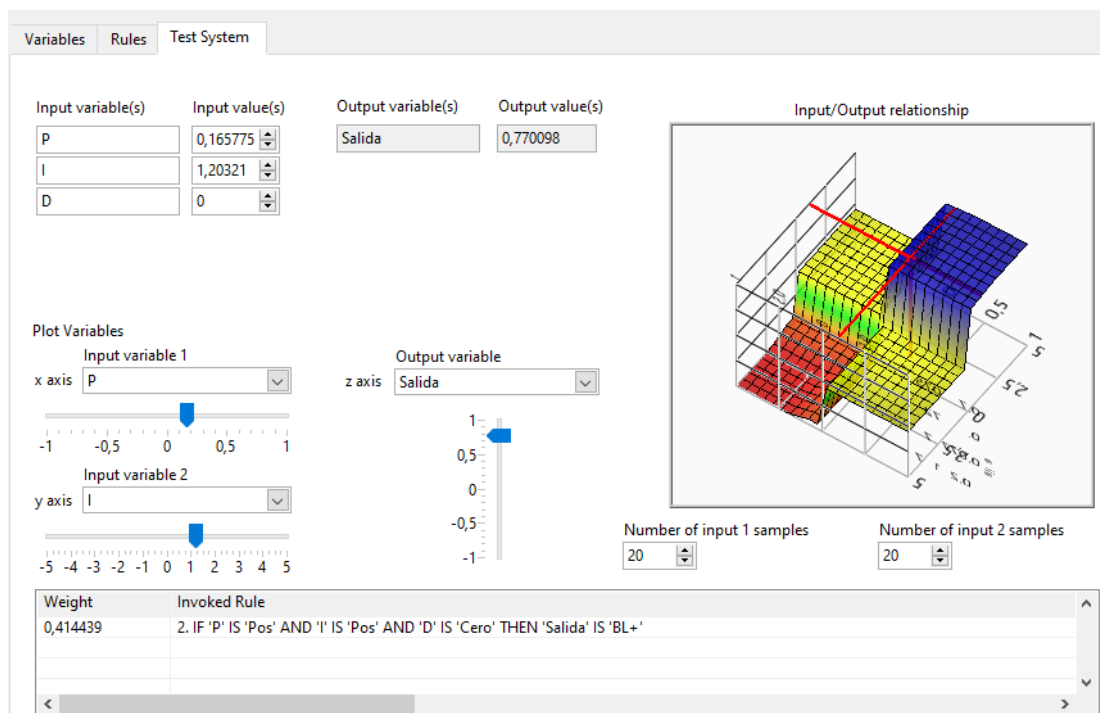


Figura 68. Resultados Practica 7 – Simulación difuso

En la figura 69 se muestra la estabilización del prototipo mediante el pid difuso.

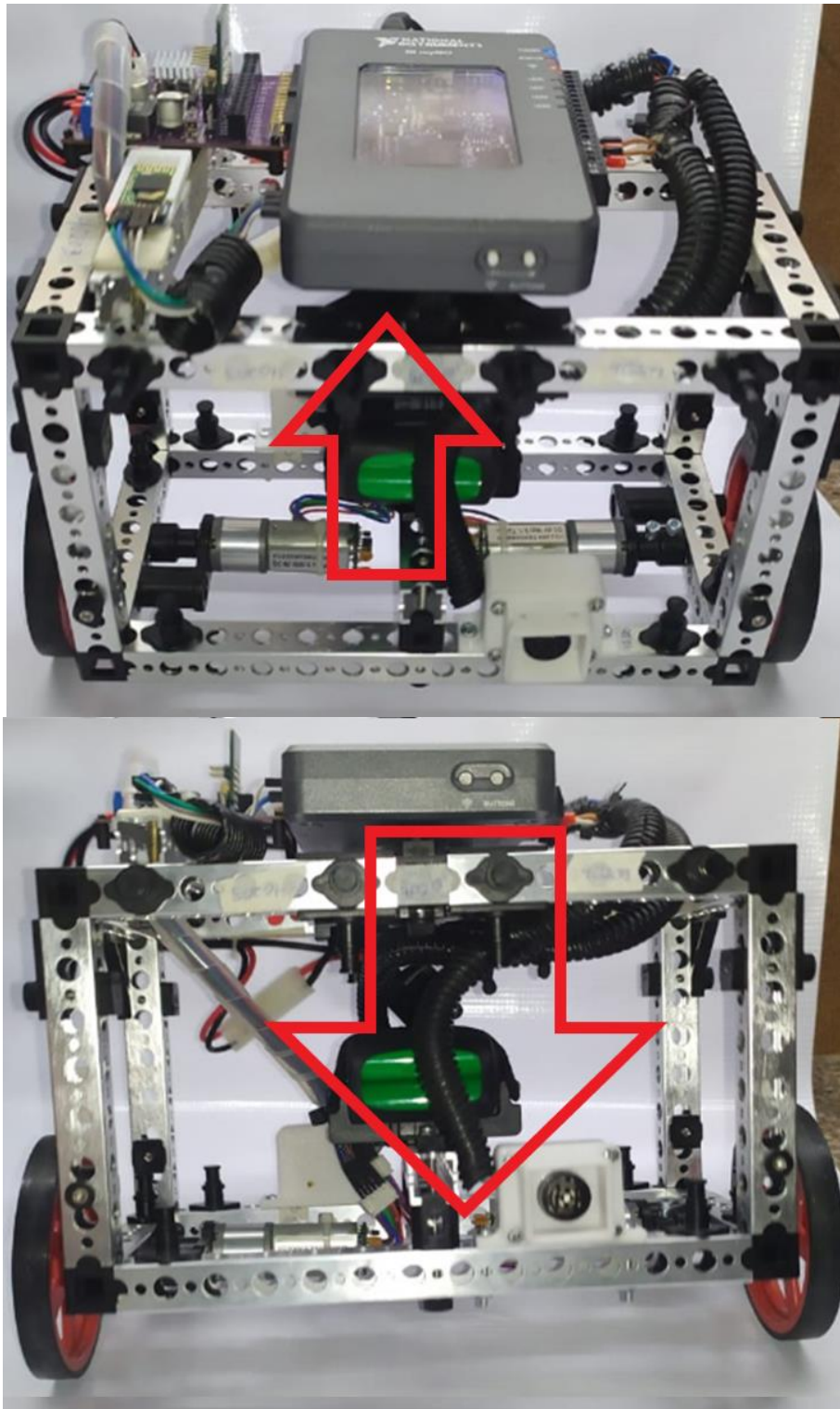


Figura 69. Resultados Practica 7 – Estabilización del robot

5.1.8. Resultados en práctica 8 - Robot auto balanceado radio controlado mediante comunicación bluetooth

En la octava practica consiste en el funcionamiento del robot auto balanceado mediante un vi secundario el cual envía valores por el puerto de comunicación bluetooth del computador el diseño es el que se muestra en la figura.

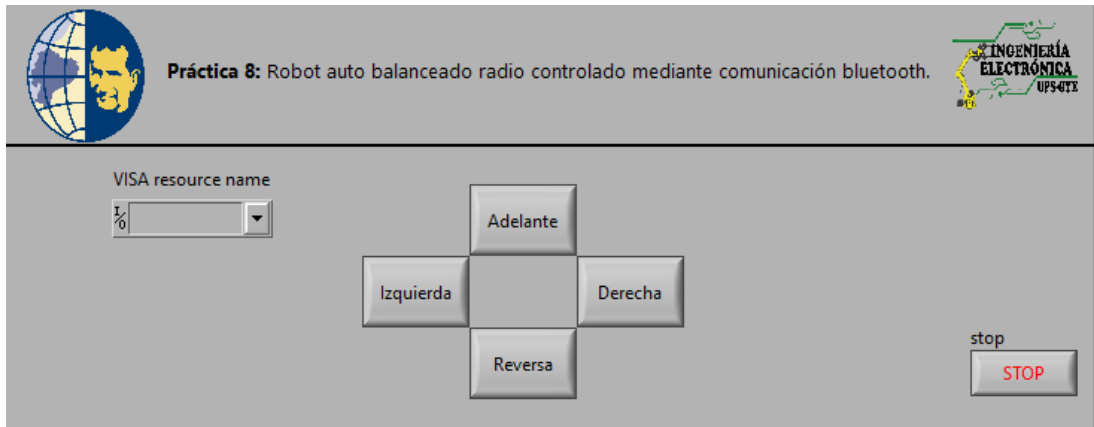


Figura 70. Resultados Practica 8 – Adquisición del valor de distancia

El vi principal es el que se muestra en la figura 71 el cual obtiene el comando mediante comunicación serial y realiza las operaciones de acuerdo con la tabla 9.

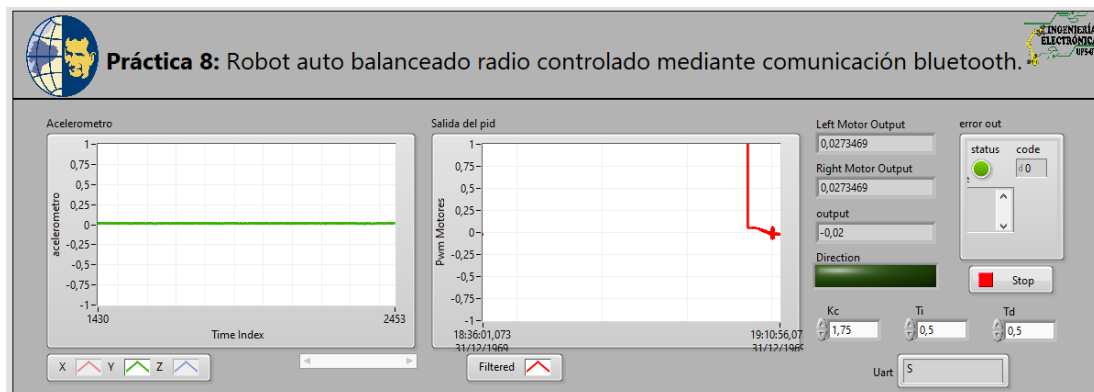


Figura 71. Resultados Practica 8 – Panel frontal

Comando de Aplicación	Acción del ni myRio
F	Avanzar
B	Retrocede
L	Giro izquierdo
R	Giro derecho
S	Detener

Tabla 9 Comandos del Pitsco Tetrax en practica 8

En la figura 72 se detalla el accionamiento del robot al enviar el comando de avanzar o adelante.

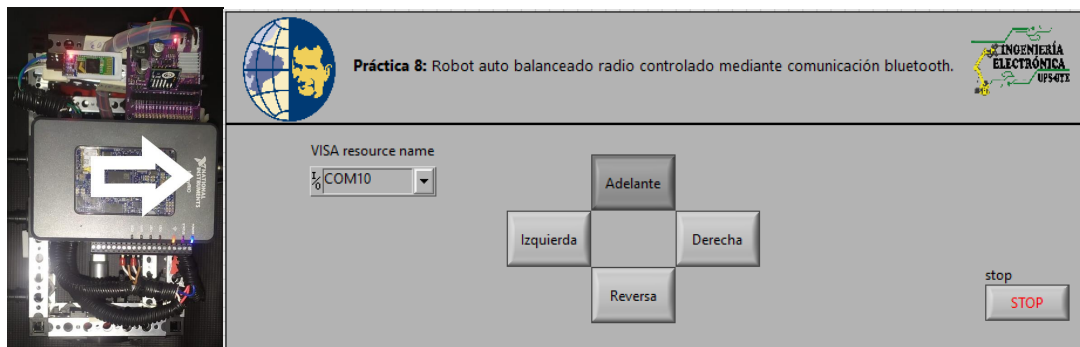


Figura 72. Resultados Practica 8 – movimiento de avanzar

En la figura 73 se detalla el accionamiento del robot al enviar el comando de retroceder o reversa.

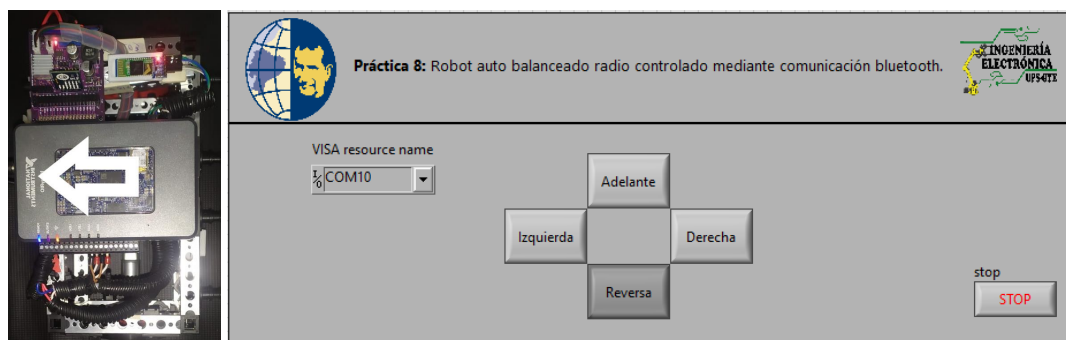


Figura 73. Resultados Practica 8 – movimiento de retroceder

En la figura 74 se detalla el accionamiento del robot al enviar el comando de giro a la derecha.

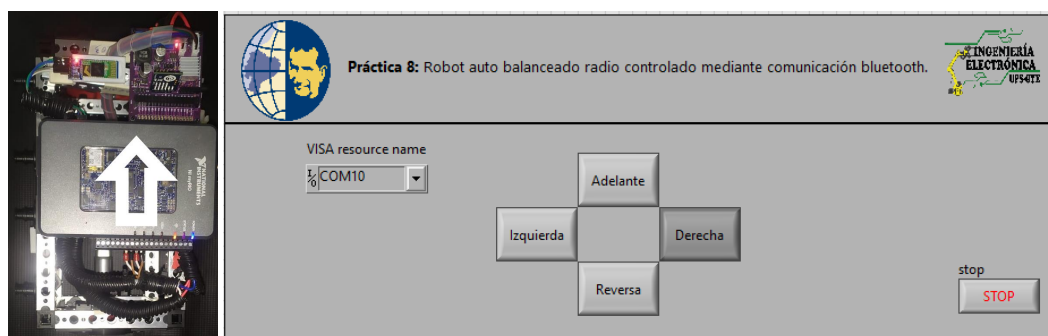


Figura 74. Resultados Practica 8 – movimiento de giro derecha

En la figura 75 se detalla el accionamiento del robot al enviar el comando de giro a la izquierda.

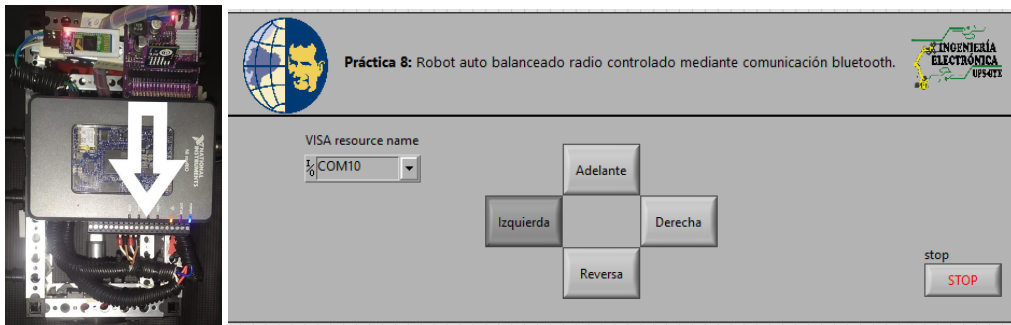


Figura 75. Resultados Practica 8 – movimiento de giro izquierda

5.1.9. Resultados en práctica 9 - Robot auto balanceado radio controlado mediante comunicación wifi

En la compilación de la octava carga el robot balance tiene una conexión wifi mediante el accionamiento de pulsadores realizara los comandos de avanzar, retroceder, giro derecha y giro izquierda como se muestra en la figura.

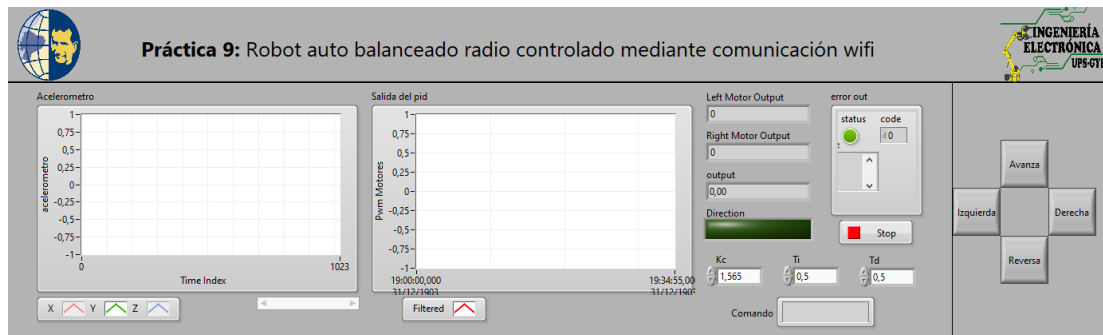


Figura 76. Resultados Practica 9 – Panel frontal

Botón	Acción del ni myRio
Avanza	Avanzar
Reversa	Retrocede
Izquierda	Giro izquierdo
Derecha	Giro derecho
Sin accionar	Balance

Tabla 10 Comandos del Pitsco Tetrax en practica 8

En la figura 77 se detalla el accionamiento del robot al enviar el comando de avanzar o adelante.

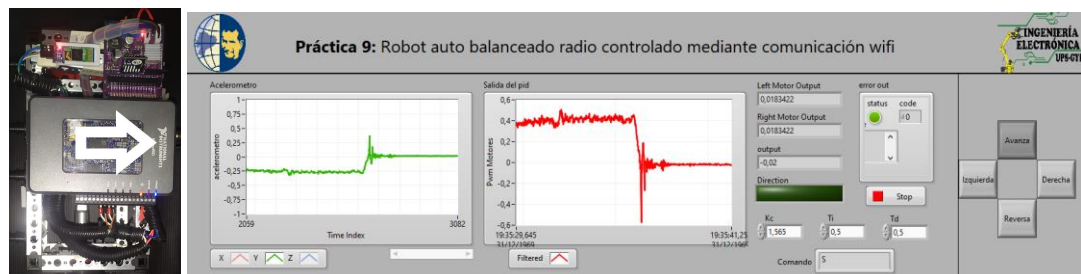


Figura 77. Resultados Practica 8 – movimiento de avanzar

En la figura 78 se detalla el accionamiento del robot al enviar el comando de retroceder o reversa.

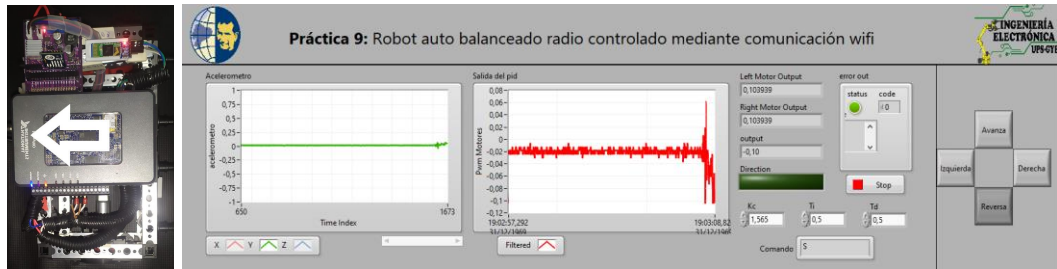


Figura 78. Resultados Practica 9 – movimiento de retroceder

En la figura 79 se detalla el accionamiento del robot al enviar el comando de giro a la derecha.

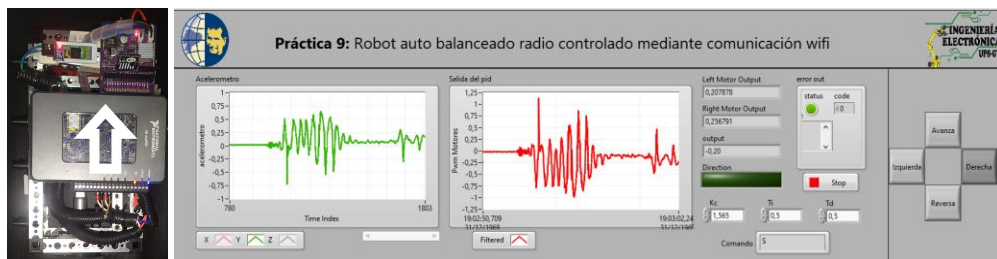


Figura 79. Resultados Practica 9 – movimiento de giro derecha

En la figura 80 se detalla el accionamiento del robot al enviar el comando de giro a la izquierda.

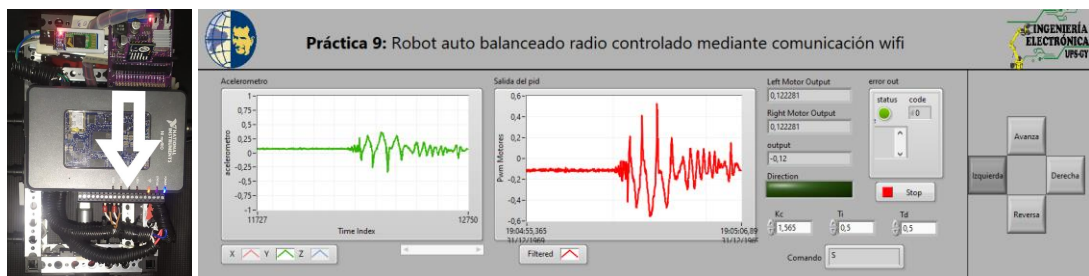


Figura 80. Resultados Practica 9 – movimiento de giro izquierda

5.1.10. Resultados en práctica 10 - Robot auto balanceado radio controlado mediante aplicativo en aplicación App Inventor

En la décima practica consiste en el control mediante un aplicativo creado app inventor que se muestra en la figura 81, el cual envía los comandos para tracción del robot.

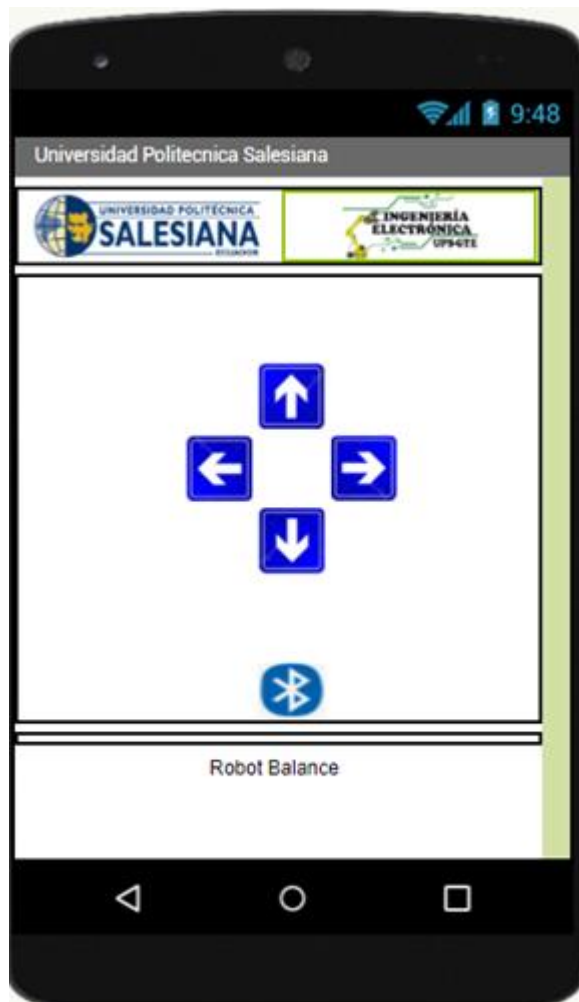


Figura 81. Resultados Practica 8 – Adquisición del valor de distancia

El vi principal es el que se muestra en la figura 82 el cual obtiene el comando mediante comunicación serial y realiza las operaciones de acuerdo con la tabla 10.

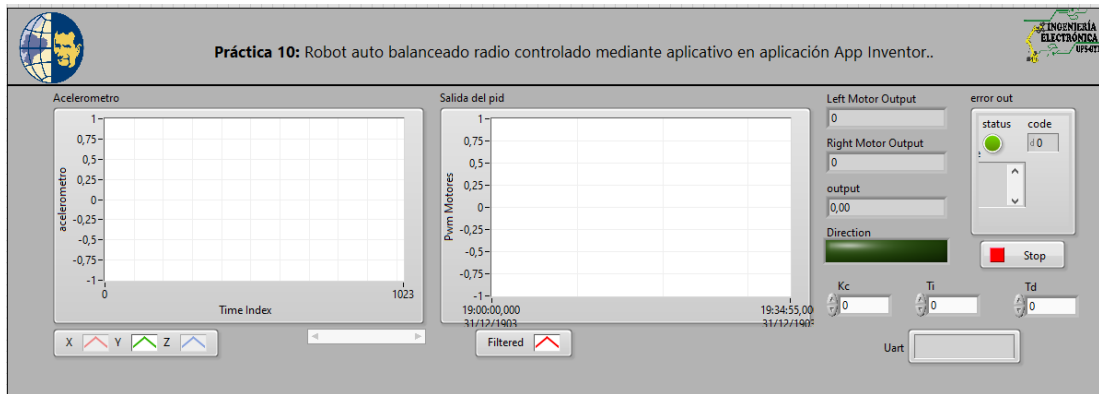


Figura 82. Resultados Practica 8 – Panel frontal

Comando de Aplicación	Acción del ni myRio
F	Avanzar
B	Retrocede
L	Giro izquierdo
R	Giro derecho
S	Detener

Tabla 11 Comandos del Pitsco Tetrix en practica 10

En la figura 83 se detalla el accionamiento del robot al enviar el comando de avanzar o adelante.

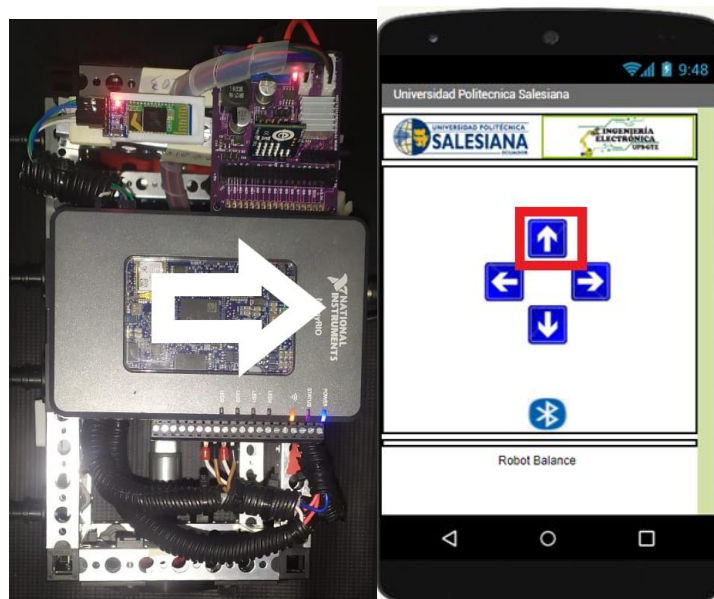


Figura 83. Resultados Practica 8 – movimiento de avanzar

En la figura 84 se detalla el accionamiento del robot al enviar el comando de retroceder o reversa.

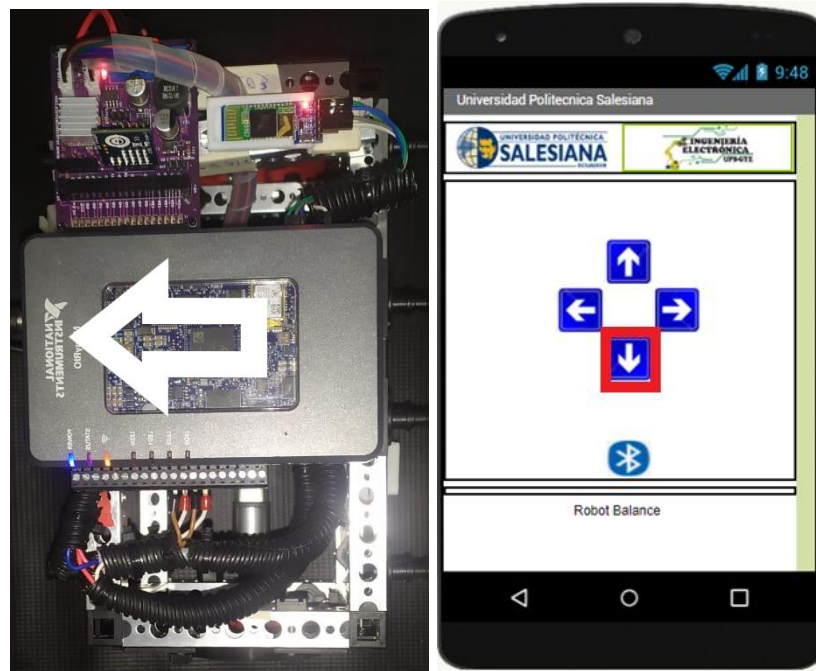


Figura 84. Resultados Practica 10 – movimiento de retroceder

En la figura 85 se detalla el accionamiento del robot al enviar el comando de giro a la derecha.

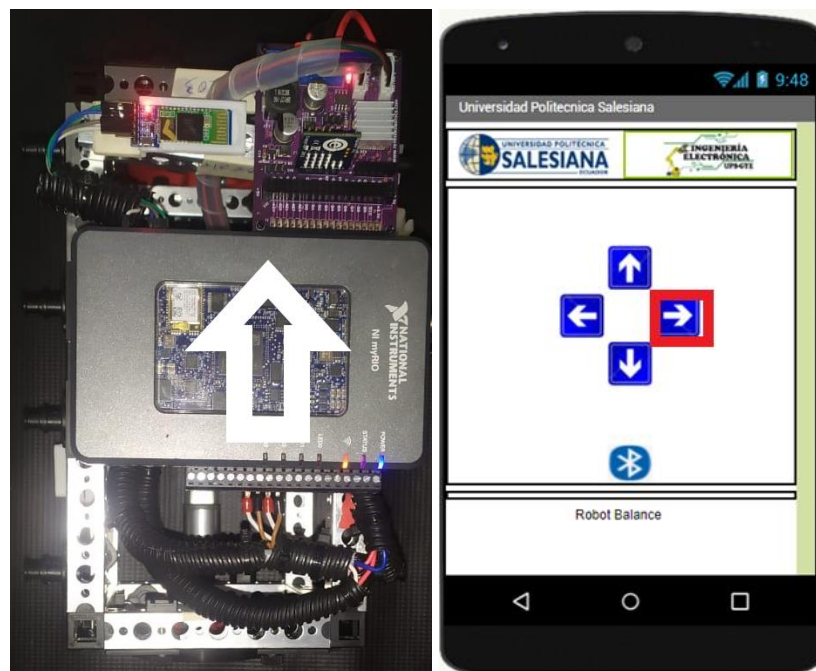


Figura 85. Resultados Practica 10 – movimiento de giro derecha

En la figura 86 se detalla el accionamiento del robot al enviar el comando de giro a la izquierda.

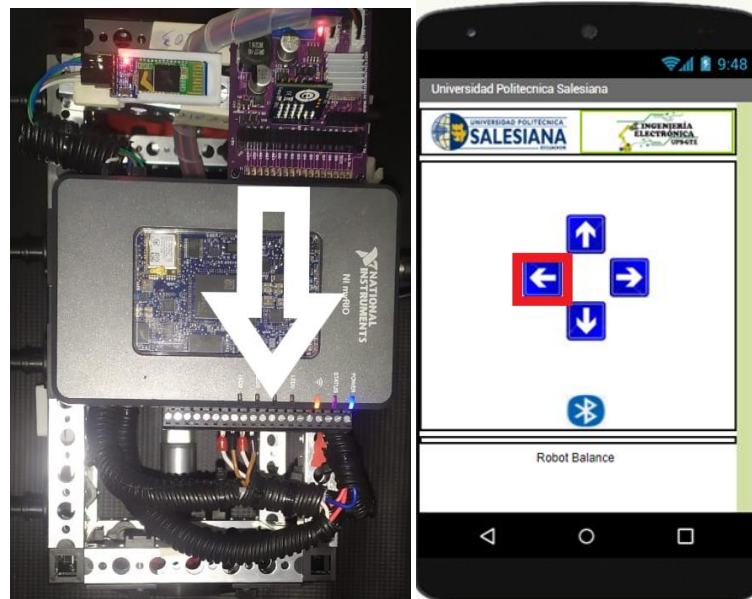


Figura 86. Resultados Practica 10 – movimiento de giro izquierda

5.2. COMPARACIÓN ENTRE CONTROLADOR PID CONVENCIONAL Y CONTROLADOR PID DIFUSO

En los sistemas de control difusos se pueden definir algunas formas para el controlador , donde a mayor cantidad de reglas se obtiene mejores resultados sin embargo esto afecta la estabilización o sintonización al momento de usar el controlador pid difuso , tiene ventajas sobre el controlador dependiendo el tipo de sistema o planta de control a implementarse .

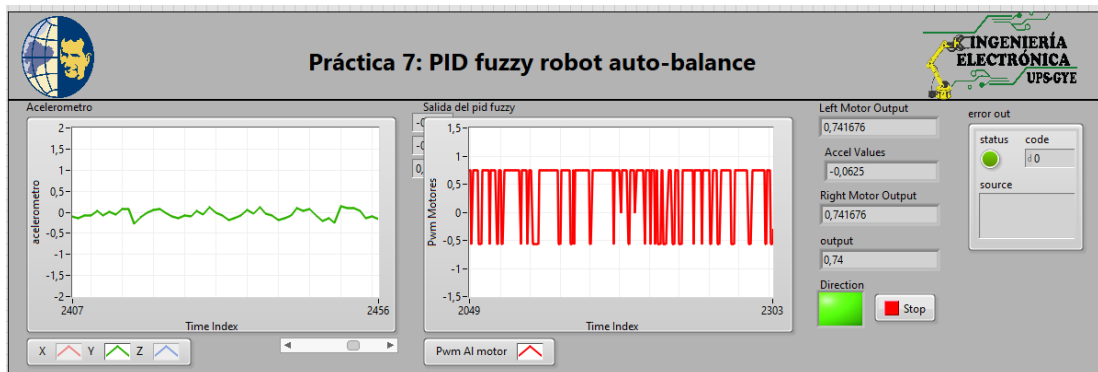


Figura 87. Controlador pid difuso

En un controlador pid es más factible usarle en sistemas que no necesiten un sobre pico , para no obtener un error en estado estacionario que logra un error acumulativo en la planta.

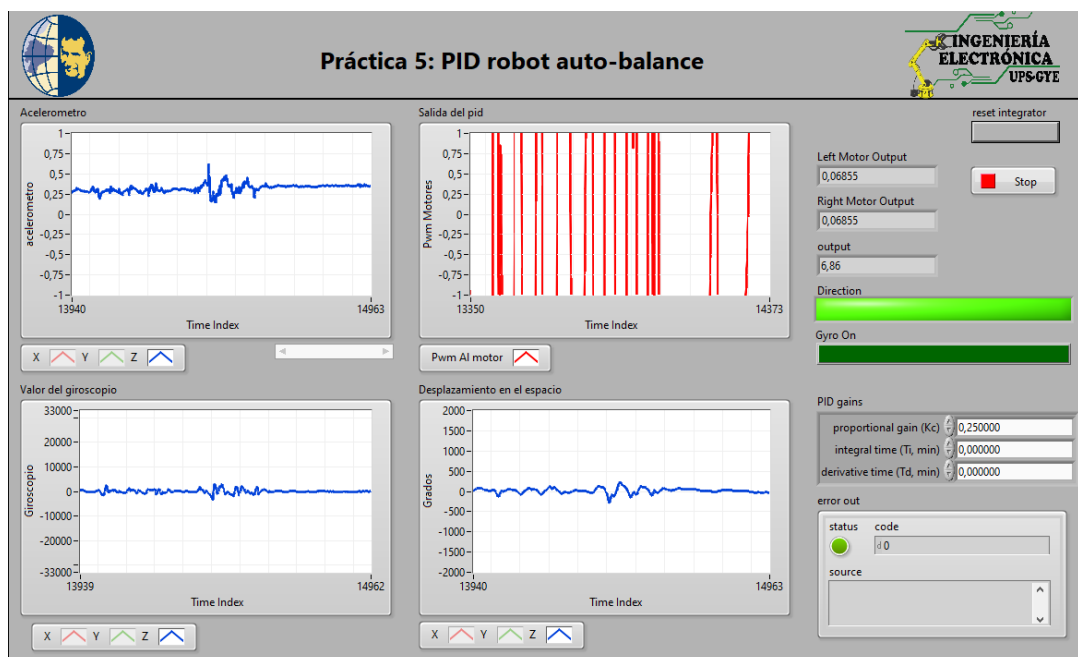


Figura 88. Controlador pid clásico

Con lo anterior se puede definir que el controlador pid es mejor para los sistemas de control aplicados en robot autobalance, mediante el control del error de estado estacionario logra mayor eficacia al momento de la implementación.

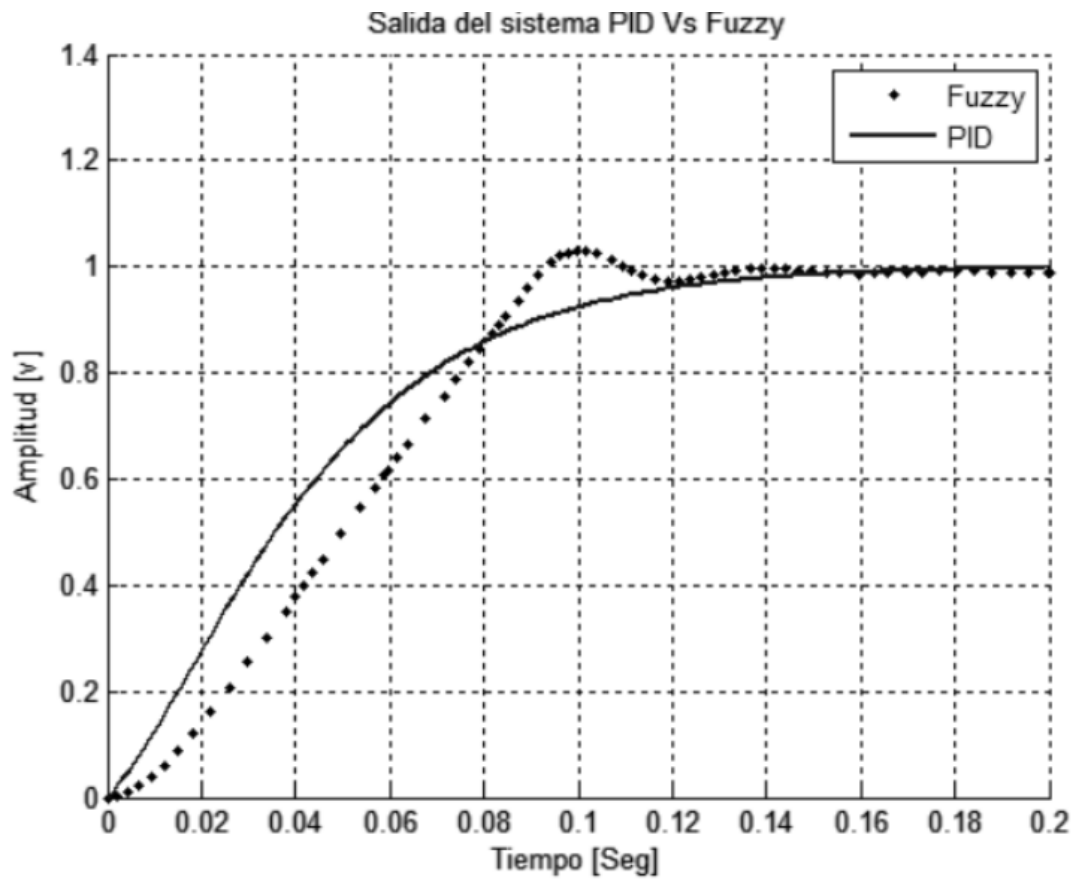


Figura 89. Controlador pid difuso vs pid clásico

CONCLUSIONES

- Se desarrolló un instrumento virtual en labview el cual permite el control de un robot auto balance mediante un controlador PID difuso con la herramienta de diseño y control de labview.
- Se diseñó un algoritmo para la adquisición de los valores del giroscopio y acelerómetro con la herramienta del ni myRio.
- Se realizó un manual de 10 prácticas para el robot Pitsco Tetrax controlado con una tarjeta ni myRio en el software labview para complementar la cátedra de diversas asignaturas de la carrera.
- Se implementó un controlador pid clásico en un robot balance y se comparó su operación con un pid difuso lo cual podrá aumentar las alternativas de diseño para los futuros Ingenieros.
- En los sistemas de control difusos se pueden definir algunas formas para el controlador donde a mayor cantidad de reglas se obtiene mejores resultados.
- Un controlador pid es más factible usarle en sistemas que no necesiten un sobre pico.

RECOMENDACIONES

- Seguir los pasos descritos en los anexos para el ensamble, verificar la carga de la batería antes de cualquier operación.
- En la creación de las reglas difusas es necesario que se creen una mayor cantidad de reglas para mejorar el control del prototipo.
- El prototipo implementado tiene un controlador robusto el cual nos permite agregar sistemas de control de mayor complejidad como una actualización futura.
- Se recomienda que al momento de realizar prueba el robot este en una posición estable y segura antes de iniciar alguna operación.
- Realizar nuevos controladores en el sistema embebido ni myRio como controladores predictivos, neuronales, neuro difusos y realizar comparación con los controladores del presente proyecto.

REFERENCIAS BIBLIOGRAFICAS

- Abreu, V. V. (2009). Balance Bot. *Universidad de Madeira*, 85.
- Adafruit. (2018). *Adafruit*. (Adafruit) Obtenido de <https://www.adafruit.com/product/3328>
- Analog Dialoge. (2012). *Analog Dialoge*. Obtenido de <https://www.analog.com/en/analog-dialogue.html>
- Astrom, K. J. (1995). *PID Controllers: Theory, Design and Tuning*. ISA; Edición: Second Edition.
- Åström, K. J. (1995). *PID Controllers: Theory, Design, and Tuning. Second Edition. Research Triangle Park*. Tore: Instrument Society of America.
- Avelectronic. (2018). *Avelectronic*. (Avelectronic) Obtenido de <https://avelectronics.cc/producto/convertidor-de-nivel-logico-i2c-bidireccional/>
- Barrientos, Antonio, Peñín, Luis Felipe, & Balag. (2007). *Fundamentos de robótica (2a. ed.)*. Madrid: McGraw-Hill España.
- Bautista España, G. M. (2018). *Diseño e implementación del algoritmo de control para un robot balance, usando fuzzy logic en la plataforma de national instruments*. Obtenido de <http://dspace.ups.edu.ec/handle/123456789/16009>
- Boris Joel Carriel Montoya, O. I. (marzo de 2015). *Diseno de un banco de pruebas neumatico*. Obtenido de <https://dspace.ups.edu.ec:https://dspace.ups.edu.ec/bitstream/123456789/10265/1/UPS-GT001294.pdf>
- Builders, G. (2017). *Ni.com*. Obtenido de https://download.ni.com/evaluation/academic/Builders_Guide_Pitsco_TETRIX_PRIME_myRIO.pdf
- Castillo, M. Á. (2008). *dma.fi.upm.es*. Obtenido de Tutorial de Introducción de Lógica Borrosa: http://www.dma.fi.upm.es/recursos/aplicaciones/logica_borrosa/web/tutorial_fuzzy/contenido3.html
- CENAM. (2018). *Centro Nacional de Metrología de México (CENAM)*. Obtenido de http://rcm.gov.co/images/2016/docs/Julie_Galindo_2016-07-15.pdf
- CTMS. (2018). *Péndulo invertido: modelado del sistema*. Obtenido de <http://ctms.engin.umich.edu/CTMS/index.php?example=InvertedPendulum§ion=SystemModeling>
- Digilent. (2018). *DC Motor/Gearbox (1:53 Gear Ratio): Custom 6V Motor Designed for Digilent Robot Kits*. Obtenido de Digilent: <https://store.digilentinc.com/dc-motor-gearbox-1-53-gear-ratio-custom-6v-motor-designed-for-digilent-robot-kits/>
- Digital, M. (s.f.). *Efecto Peltier*. Obtenido de <http://www.mundodigital.net/ques-es-el-efecto-peltier/>
- Drok. (2017). *DROK*. Obtenido de <https://www.droking.com/>

- Ecured. (2015). *Termómetros digitales*. Obtenido de https://www.ecured.cu/Term%C3%B3metros_digitales
- Fluke. (s.f.). *Fluke*. (Calibradores de termómetro de microbaño) Obtenido de <https://la.flukecal.com/products/process-calibration-tools/temperature-calibrators/dry-block-calibrators-and-micro-baths/cal>
- Fritzing. (2015). *Fritzing*. Obtenido de Fritzing: <http://fritzing.org/home/>
- García Tinoco, J. P. (2019). *Diseño e implementación de un sistema domótico ininterrumpido con iluminación, sistemas de vigilancia y automatización, sistemas de vigilancia y automatización de portones de ingreso utilizando control pid y labview*. Obtenido de <https://dspace.ups.edu.ec>:
<https://dspace.ups.edu.ec/handle/123456789/17070>
- González Macías Bryan Xavier, B. E. (2018). *Dspace.ups.edu.ec*. Obtenido de Diseño e implementación del algoritmo de control para un robot balance, usando fuzzy logic en la plataforma de national instruments.: <http://dspace.ups.edu.ec/handle/123456789/16009>
- Granda Miguel, Mercedes, & Mediavilla Bolado. (2015). *Instrumentación electrónica: transductores y acondicionadores de señal*. Madrid: Editorial de la Universidad de Cantabria.
- Leekwijck, V. (1999). *Defuzzification: criteria and classification*. Kerre, E. E.: Fuzzy Sets and Systems.
- Manuel Baltieri, C. L. (Marzo de 2018). <http://dx.doi.org/10.1101/284562>. Obtenido de University of Sussex: <https://www.biorxiv.org/content/biorxiv/early/2018/03/19/284562.full.pdf>
- Margarita Torres, M. V. (2016). *Análisis, diseño e implementación de la aplicación móvil G.U.I.D.I.D.O (Guía didáctica docente para favorecer la inclusión educativa de estudiantes con discapacidad auditiva) de la carrera Ingeniería de Sistemas en la universidad Politécnica Salesiana Sed*. Obtenido de <https://dspace.ups.edu.ec/>:
<https://dspace.ups.edu.ec/bitstream/123456789/12672/1/UPS-GT001700.pdf>
- MaxBotic. (2017). *www.maxbotix.com*. Obtenido de https://www.maxbotix.com/Ultrasonic_Sensors.htm
- Meaton, R. W. (1991). *MotoresElectricos: Selección mantenimiento y reparación*. McGraw-Hill Interamericana.
- Metrosens. (s.f.). *Metrosens S.a*. Obtenido de <http://www.metrosens.com.ec/ecuador/caracterizacion.html>
- meyca, M. (s.f.). *Mediciones meyca*. (Meyca) Obtenido de http://www.medicionesmeyca.com/?page_id=79
- Morcillo, C. G. (2017). *www.esi.uclm.es/*. Obtenido de Lógica Difusa: https://www.esi.uclm.es/www/cglez/downloads/docencia/2011_Softcomputing/LogicaDifusa.pdf

MSP430. (Septiembre de 2014). *MSP430 LaunchPad Value Line Development kit*. Obtenido de <http://www.ti.com/tool/MSP-EXP430G2>

National Instruments Corporation. (2019). *National Instrument*. Obtenido de ni.com: <https://www.ni.com/es-cr/shop/select/myrio-student-embedded-device>

NI. (2018). *national instrument*. Obtenido de national instrument: <https://www.ni.com/es-cr/shop/labview.html>

Nicolas Goilav, G. L. (2018). *Arduino Aprender a desarrollar objetos inteligentes*. EEUU: Eni ediciones.

Nolla, X. (2019). (Instrumentos WIKA S.A.) Obtenido de <https://www.bloginstrumentacion.com/productos/como-funciona-un-transmisor-de-presion/>

Ogata, K. (2015). *Ingeniería de Control Moderna*. Editorial Prentice Hall. Obtenido de <https://www.picuinio.com/es/arduprog/control-ziegler-nichols.html>

prime, P. t. (2020). *TETRIX® PRIME Dual-Control Robotics Set*. Obtenido de TETRIX® PRIME Dual-Control Robotics Set: <https://www.pitsco.com/EC/Shop/TETRIX-Robotics/TETRIX-PRIME/TETRIX-PRIME-Dual-Control-Robotics-Set>

Principles of rotary encoders. (22 de 10 de 2004). *Robotics World*. Obtenido de <https://search.proquest.com/docview/218407823?accountid=32861>

Products, C. (2015). *Marsh-Bellofram*. Obtenido de <http://www.cpinc.com/bellofram-t1000.html>

profesionalreview. (2018). *profesional review*. Obtenido de <https://www.profesionalreview.com/2018/10/16/que-celula-peltier/>

Radio-electronics. (2016). *Solid State Relays*. Obtenido de physics.unlv.edu/~bill/PHYS483/ssr.pdf

Ramos, O. R. (12 de junio de 2008). *Colección de Tesis Digitales*. Obtenido de Universidad de las Americas Puebla: http://catarina.udlap.mx/u_dl_a/tales/documentos/lmt/ramirez_r_o/

Raspberrypi. (2019). *Raspberrypi*. (Raspberrypi) Obtenido de <https://www.raspberrypi.org/>

Rivas F. Dario, C. A. (2013). AUTOMATIZACIÓN DEL PROCESO DE CARACTERIZACIÓN. *UNIVERSIDAD DE LA SALLE PROGRAMA DE INGENIERÍA EN AUTOMATIZACIÓN*, 111. Obtenido de <http://repository.lasalle.edu.co/bitstream/handle/10185/22253/T45.13%20R116a.pdf?sequence=1&isAllowed=y>

sabelotodo. (2017). *Bimetalicapresion*. Obtenido de <http://www.sabelotodo.org/automovil/bimetalicapresion.html>

Santiago Sánchez-Solano Alejandro J. Cabrera, M. B. (2015). *digital.csic.es/*. Obtenido de CONTROLADORES DIFUSOS ADAPTATIVOS COMO MÓDULOS DE PROPIEDAD INTELECTUAL PARA FPGAS:

<https://digital.csic.es/bitstream/10261/86598/1/Controladores%20difusos.pdf>

- Tcmetrologia. (30 de Junio de 2014). *Tcmetrologia*. Obtenido de Tcm: <https://www.tcmetrologia.com/blog/calibracion-de-equipos-isotermos/>
- Times, T. J. (2009). Scientist claims he made Segway predecessor in '86. *The Japan Times Online*. Retrieved June 18, 2009.
- Wojciech Walendziuk, A. I. (2015). Evaluation of Pt100 Sensor Deflection Effect. *Department of Electrical Engineering, Bialystok University of Technology*, 21, 4.
- Zambrano Mendoza, E. L., & Martillo Ayala, D. H. (Febrero de 2015). Diseño de aplicaciones de sistemas embebidos basados en tecnología Raspberry y Odroid-U3. Guayaquil, Guayas, Ecuador. Obtenido de <https://dspace.ups.edu.ec/bitstream/123456789/11409/1/UPS-GT001580.pdf>

ANEXOS

Anexo 1 Dimensiones de la tarjeta MyRio

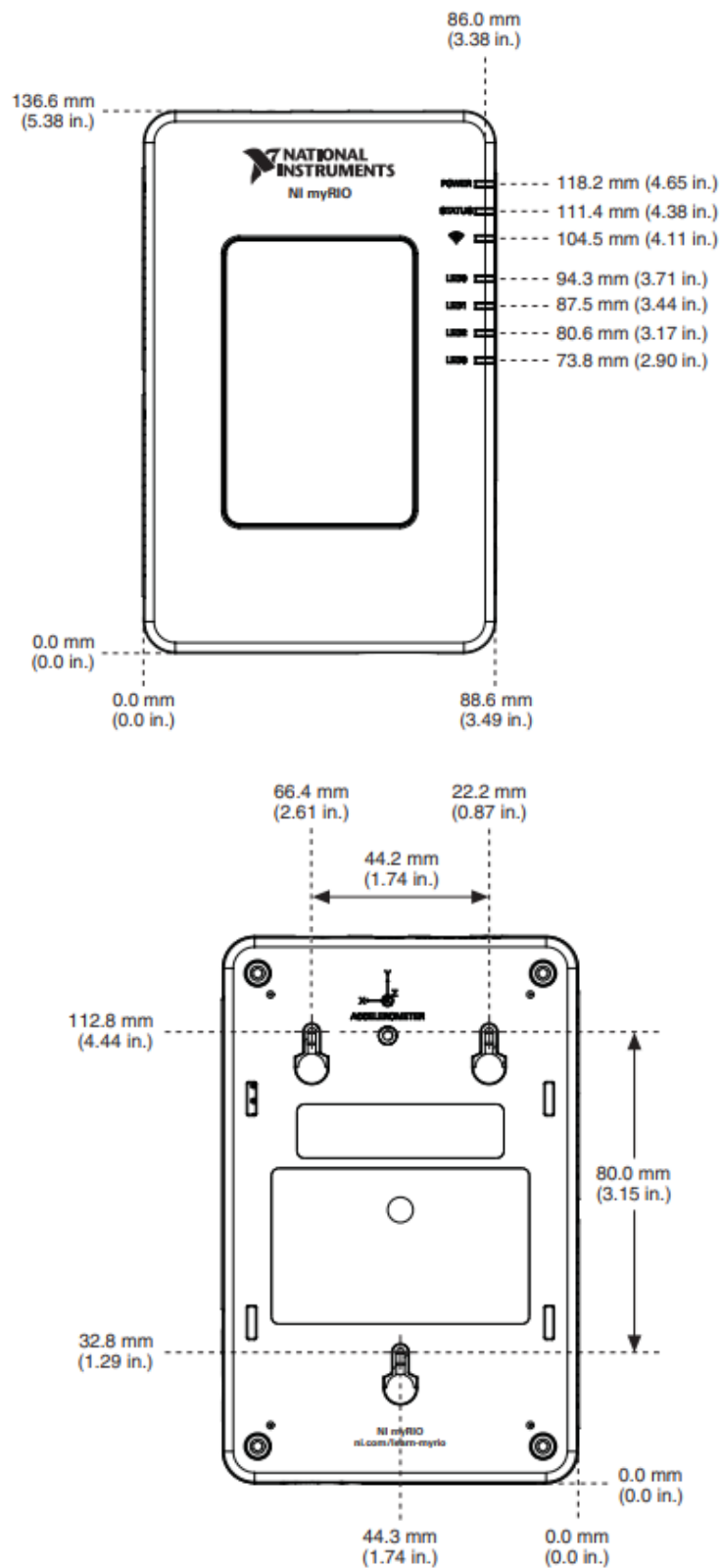


Figura 90. Anexo 1 Dimensiones del ni myRIO – vista frontal y posterior

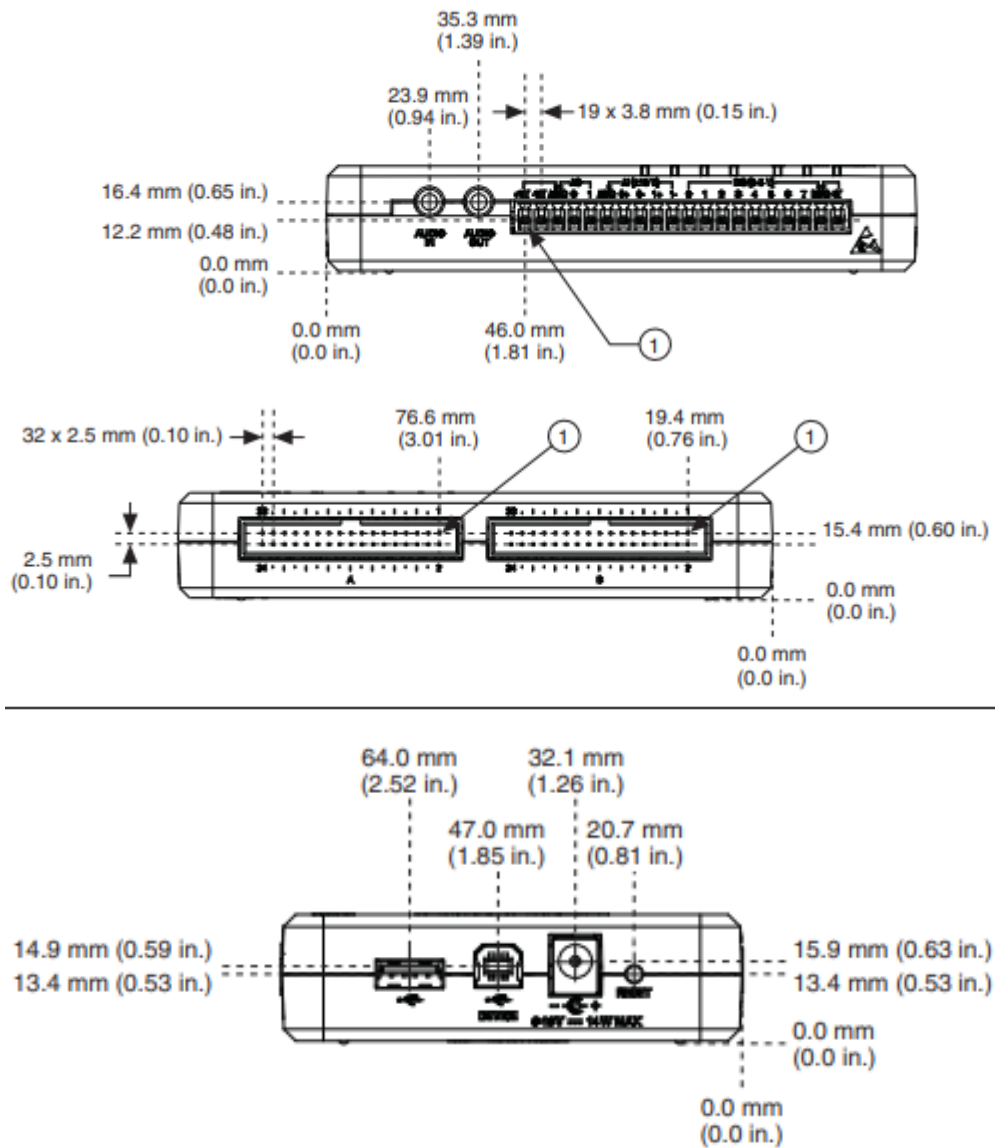


Figura 91. Anexo 1 Dimensiones del ni myRio – vista lateral

Anexo 2 Lista de materiales

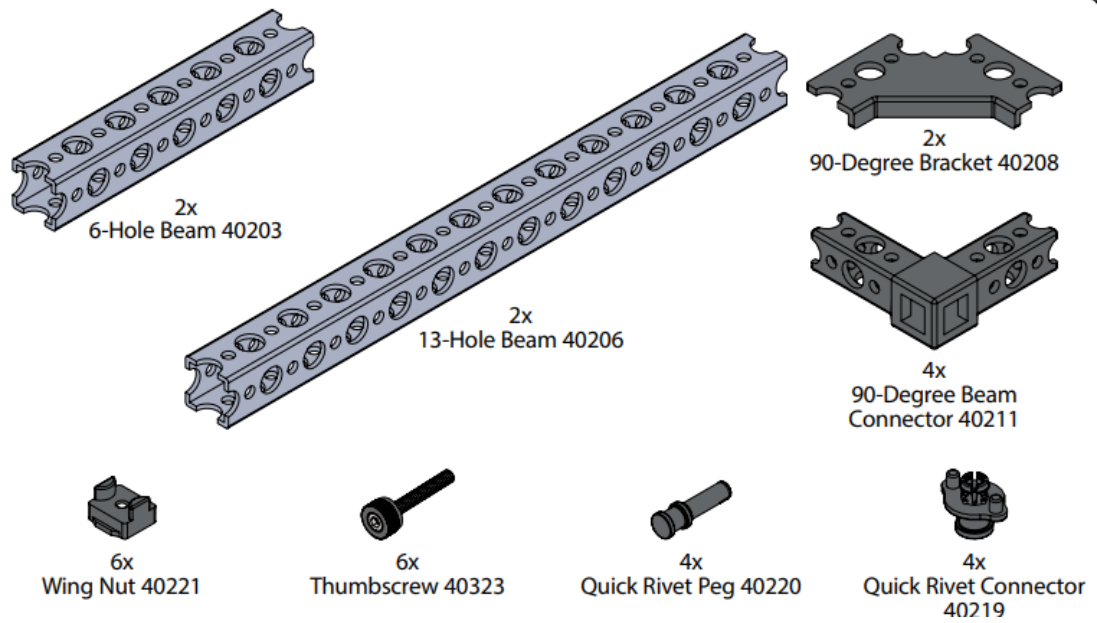


Figura 92. Anexo 2 – Lista de materiales parte 1

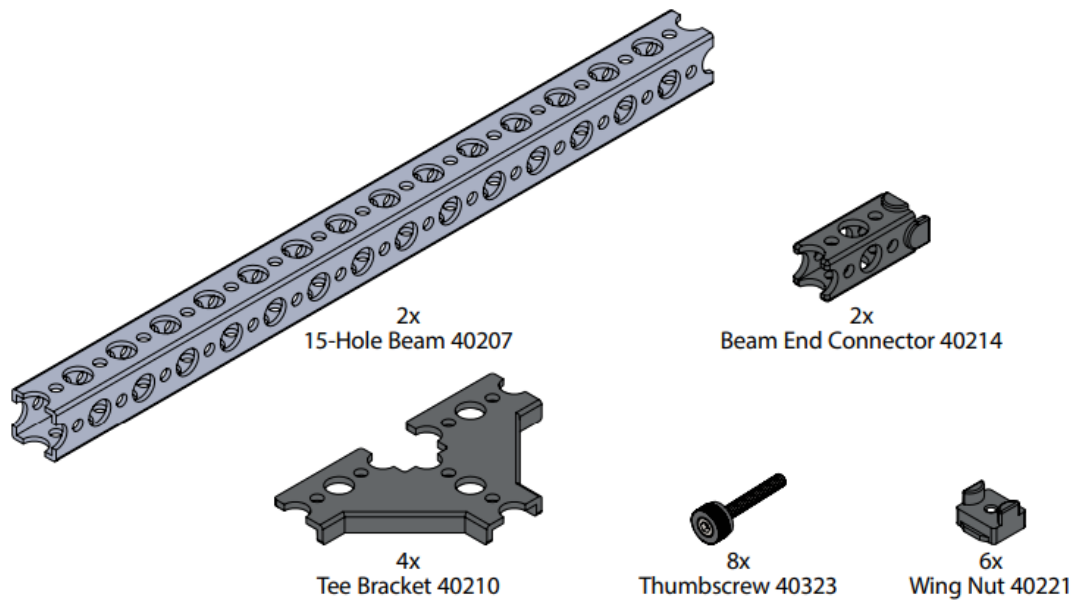


Figura 93. Anexo 2 – Lista de materiales parte 2

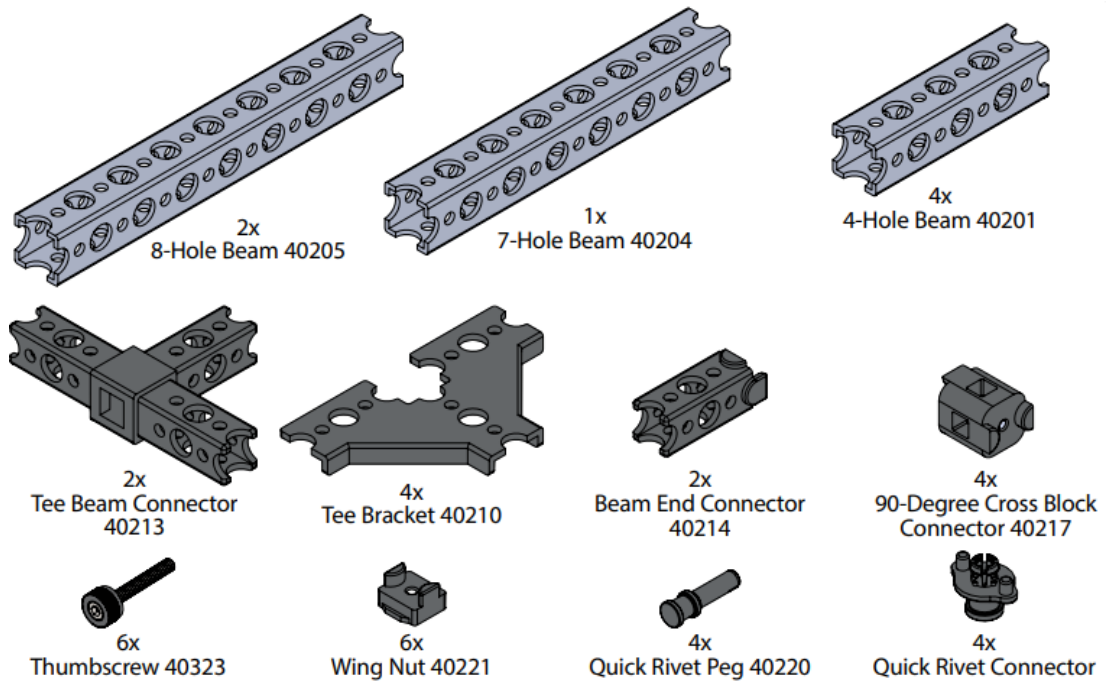


Figura 94. Anexo 2 – Lista de materiales parte 3

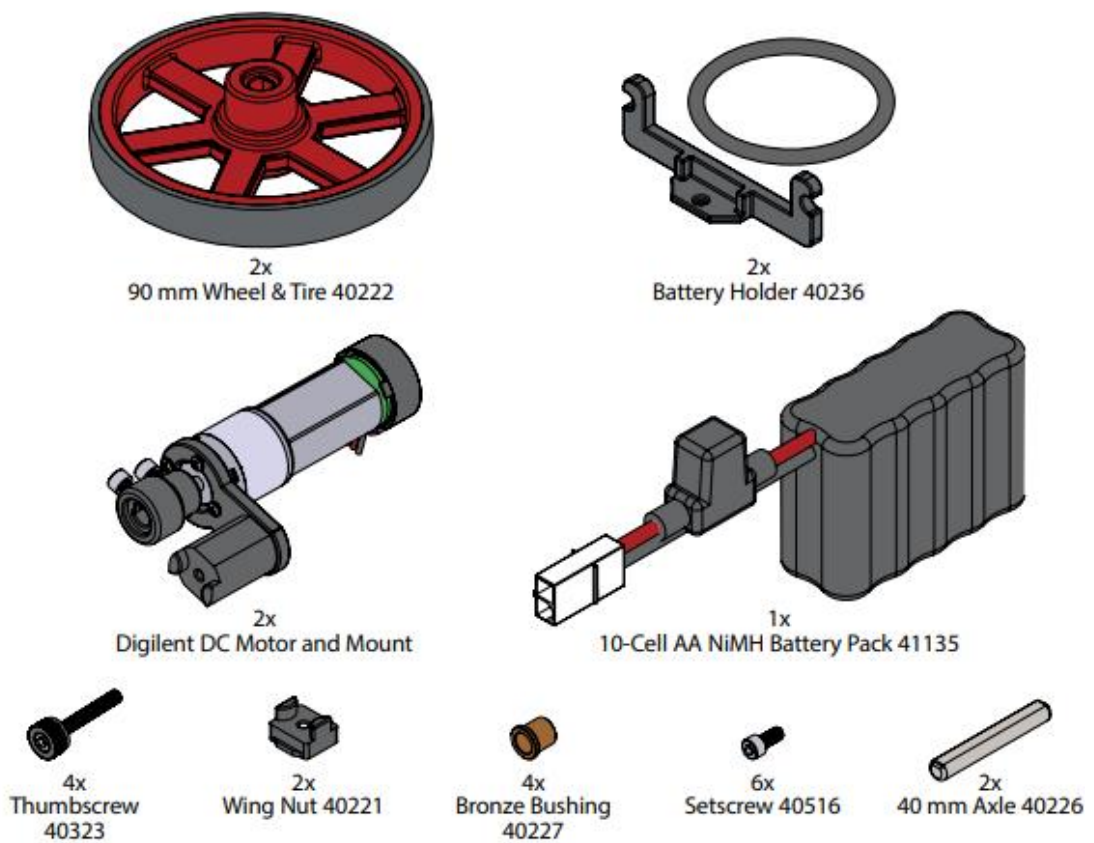


Figura 95. Anexo 2 – Lista de materiales parte 4

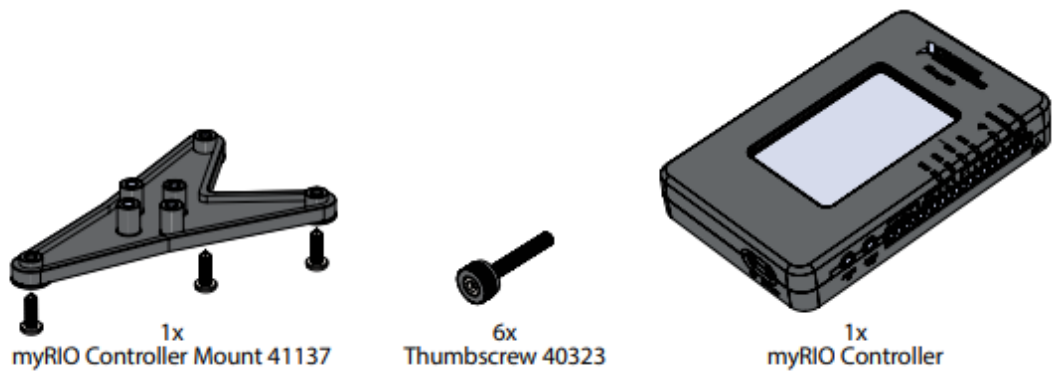


Figura 96. Anexo 2 – Lista de materiales parte 5

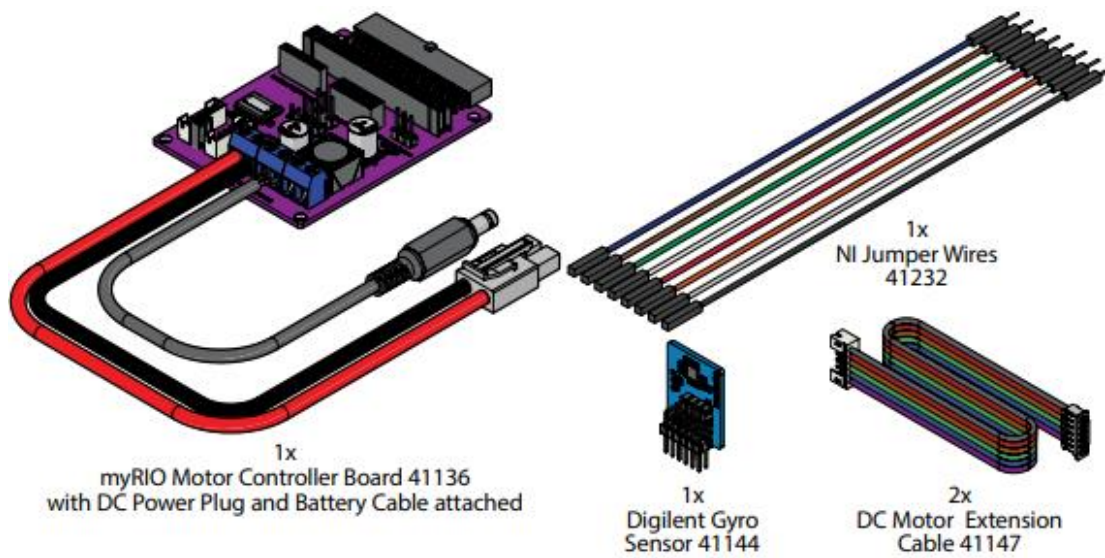


Figura 97. Anexo 2 – Lista de materiales parte 6

Anexo 3 Ensamble del prototipo

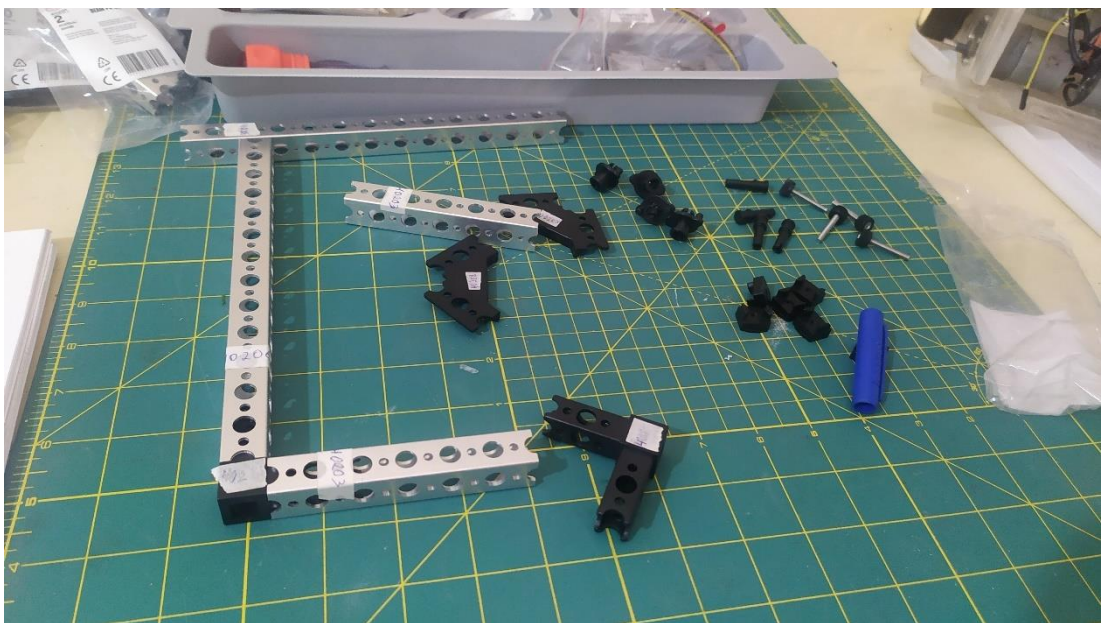


Figura 98. Anexo 3 – Ensamble de parte 1 paso 1

Step 1.1

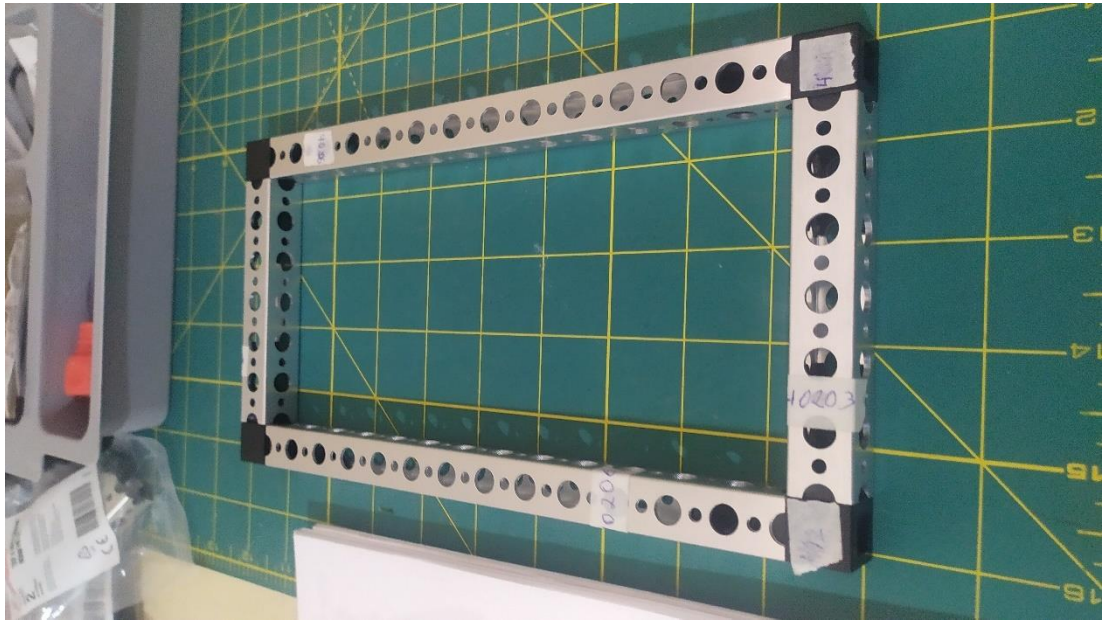
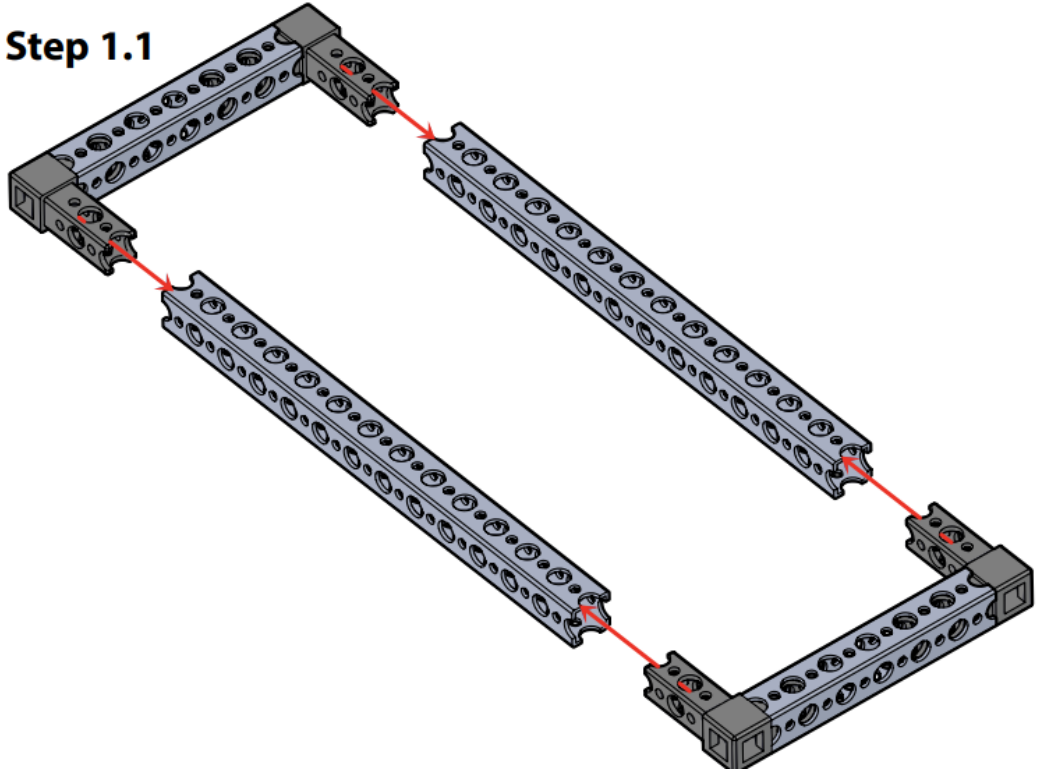
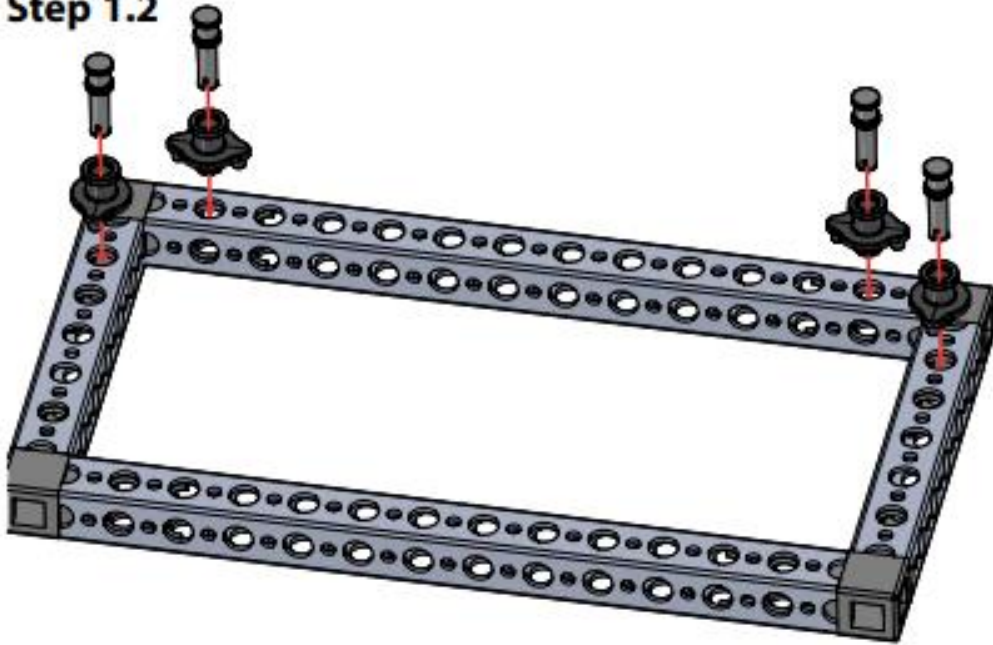


Figura 99. Anexo 3 – Ensamble de parte 1 paso 2

Step 1.2



Step 1.3

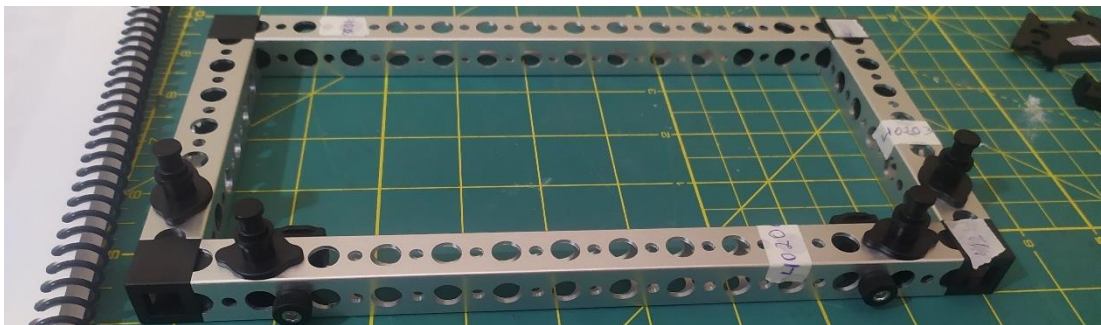
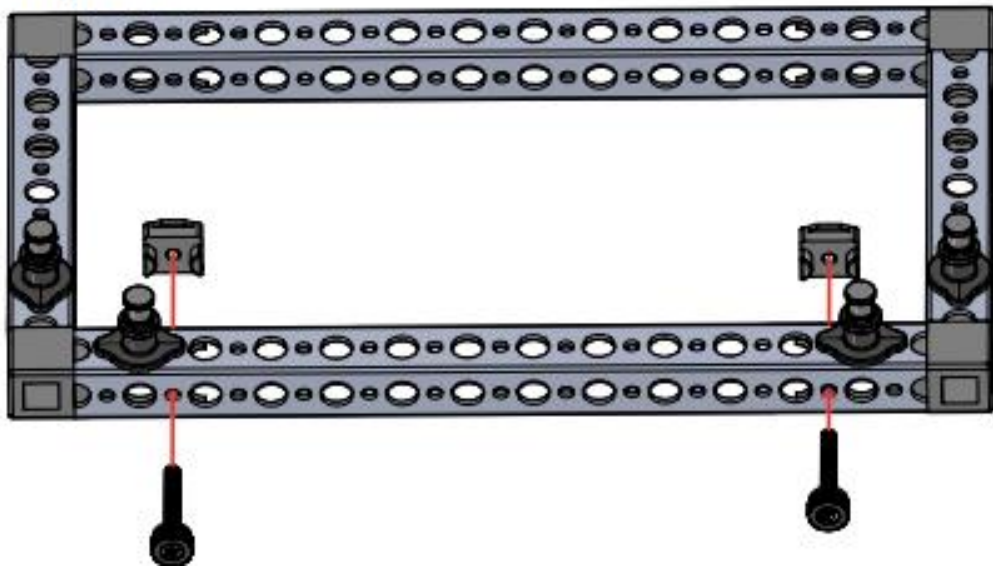


Figura 100. Anexo 3 – Ensamble de parte 1 paso 3

Step 1.4



Step 1.5

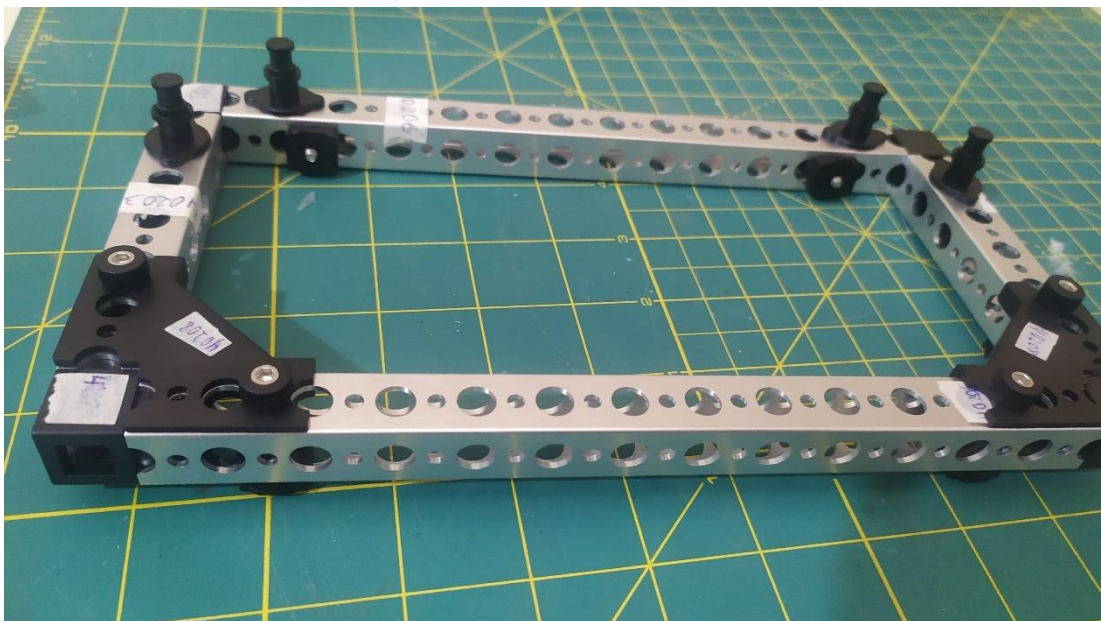


Figura 101. Anexo 3 – Ensamble de parte 1 paso 4



Figura 102. Anexo 3 – Ensamble de parte 2 paso 1

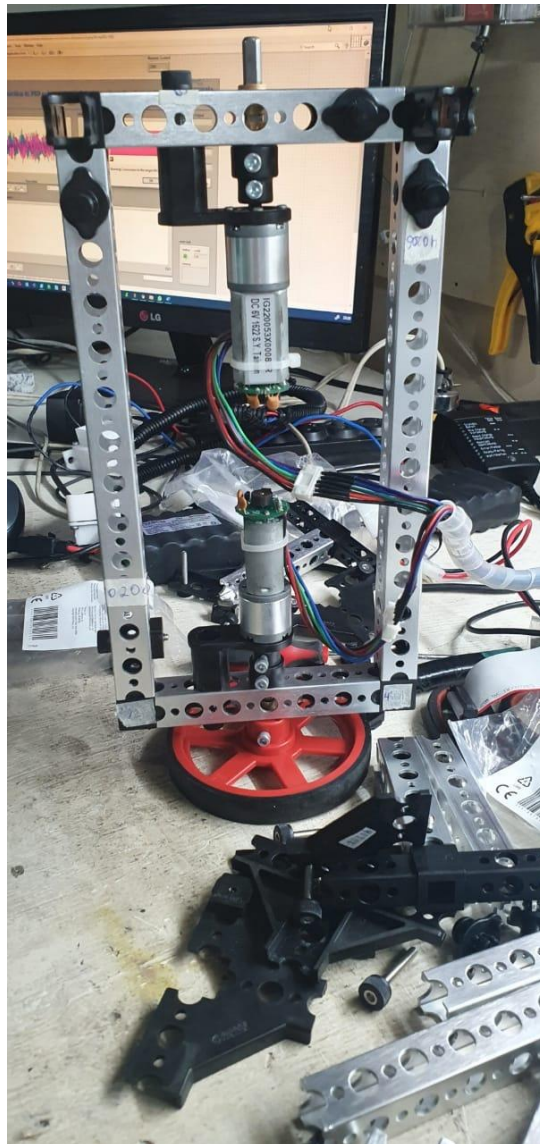


Figura 103. Anexo 3 – Ensamble de parte 2 paso 2

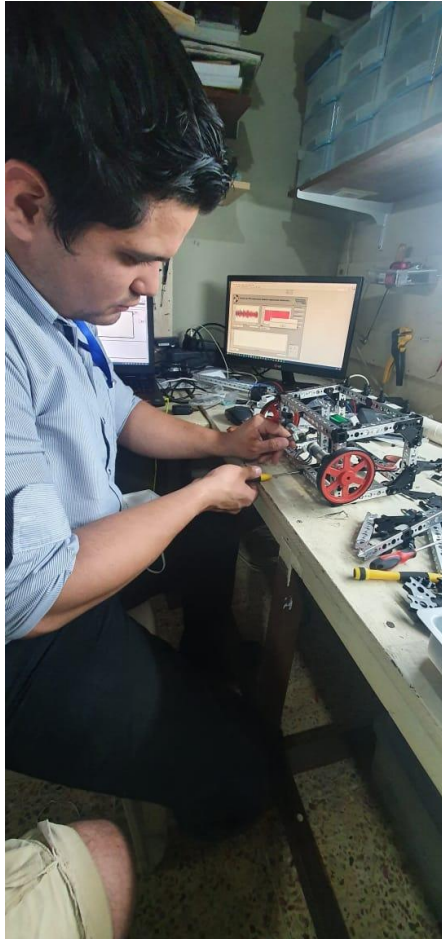


Figura 104. Anexo 3 – Ensamble de parte 3 paso 1



Figura 105. Anexo 3 – Ensamble de parte 3 paso 2



Figura 106. Anexo 3 – Ensamble de parte 5 paso 1

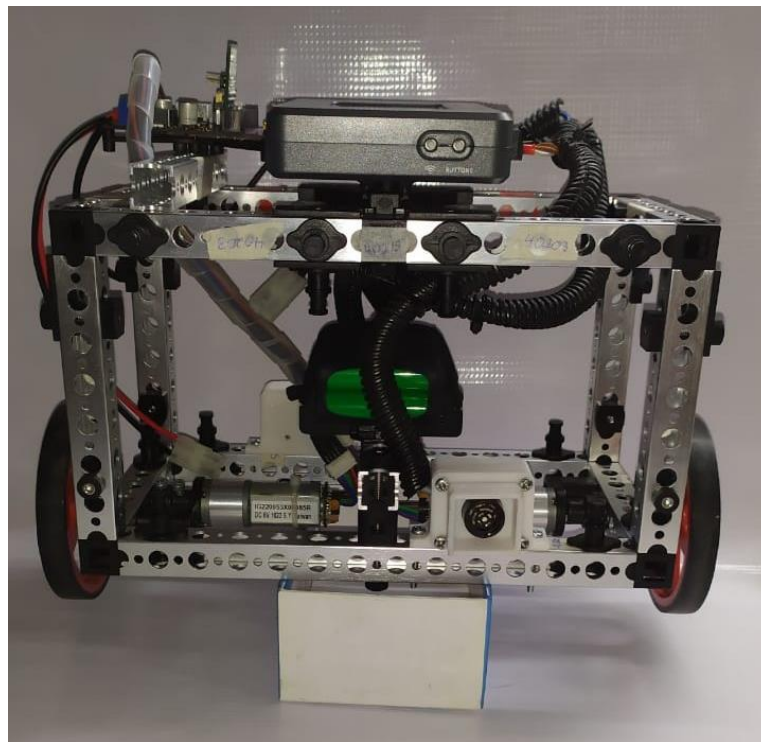


Figura 107. Anexo 3 – Ensamble final de prototipo

Anexo 4. Solución de prácticas

Anexo 4.0 Introducción a prácticas

1. Revisar el estado de la batería del prototipo.
2. Revisar el estado de las piezas del Pitsco Tetrix.
3. Revisar el estado llantas estén en óptimas condiciones.
4. Revisar que los motores estén sin ninguna obstrucción mecánica.

Anexo 4.1 Solución de práctica 1

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 108

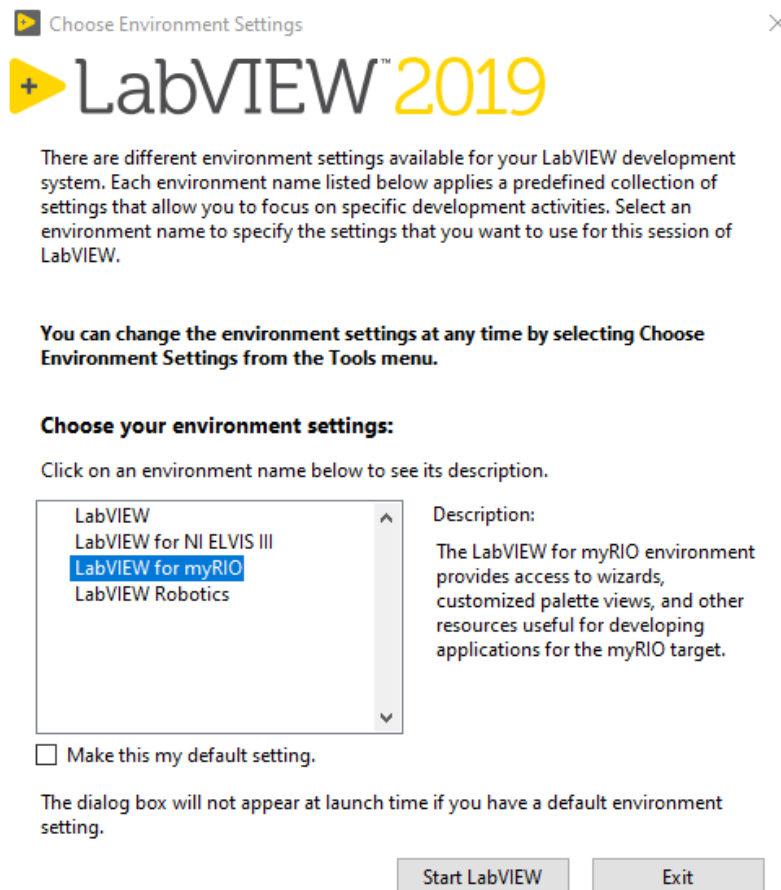


Figura 108. Ejecución de LabView como compatible con ni myRio práctica 1

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 109.

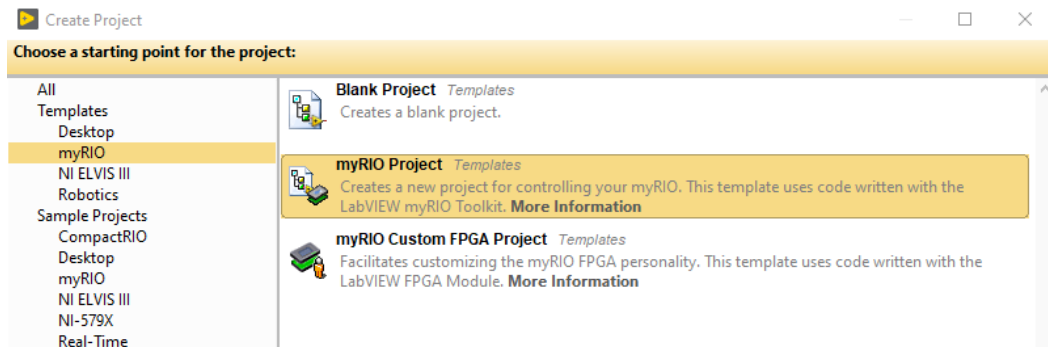


Figura 109. Selección de proyecto myRio práctica 1

Configurar la ruta donde se guarda el archivo (carpeta practica 1), nombre (Lectura de acelerómetros con software de instrumentación virtual) y el tipo de conexión como USB y dar en finalizar como se muestra en la imagen 110.

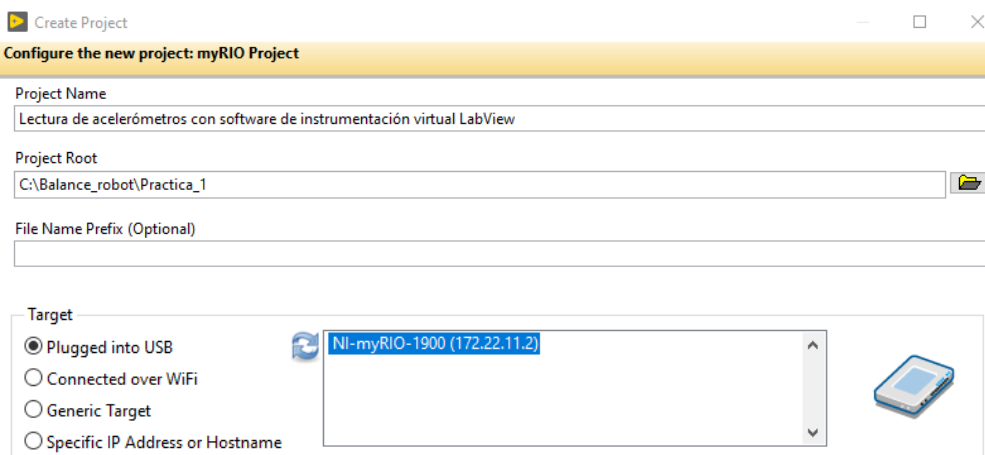


Figura 110. Configuración de ruta práctica 1

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la figura 111.

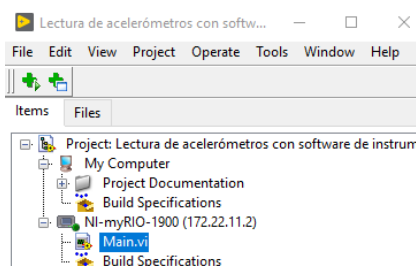


Figura 111. Administrador de proyectos, selección práctica 1

En el bloque de diagrama se agrega un bloque i2c(bloque de diagrama>>myRio >> i2c) para nimirio y se configura como escritura , en el puerto a y su velocidad de 100 kbps como se muestra en la figura 112.

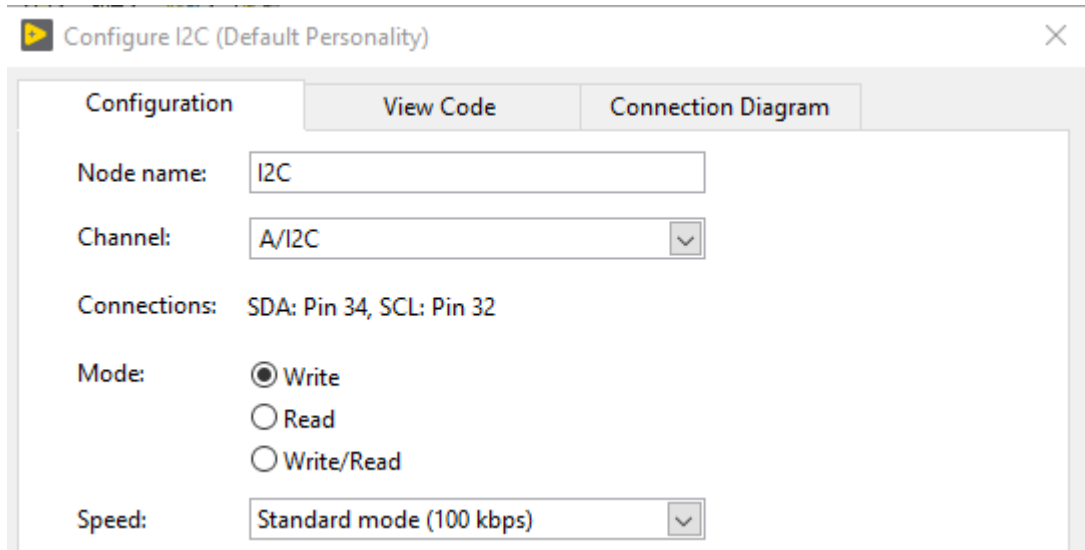


Figura 112. Configuración de i2c para myRio de práctica 1

En el bloque i2c se crea una variable que contenga la direccion del dispositivo **0x69** y un arreglo donde se configuran los parametros en formato de arreglo bidimensional para el dispositivo como se muestra en la figura 113.

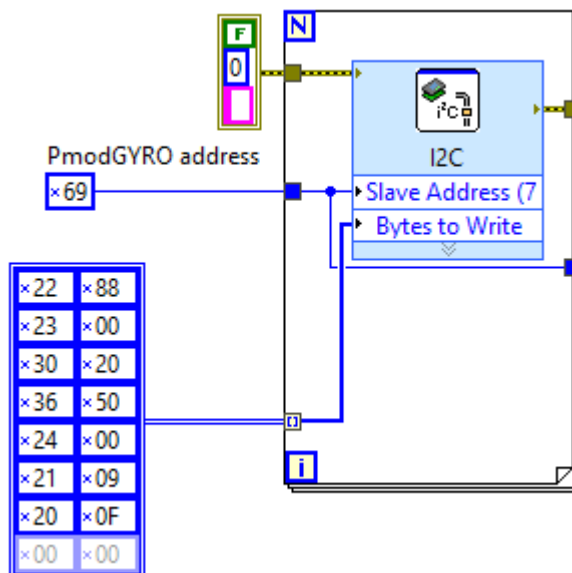


Figura 113. Bloque de variables i2c, práctica 1

Crear un lazo de repetición (while loop) en el cual se agrega dos datos de entradas digitales (bloque de diagrama >>myRio >> digital input) para el puerto A DIO-0 y el A DIO 1 los cuales tienen la función de captar la señal del módulo acelerómetro como se muestra en la imagen 114.

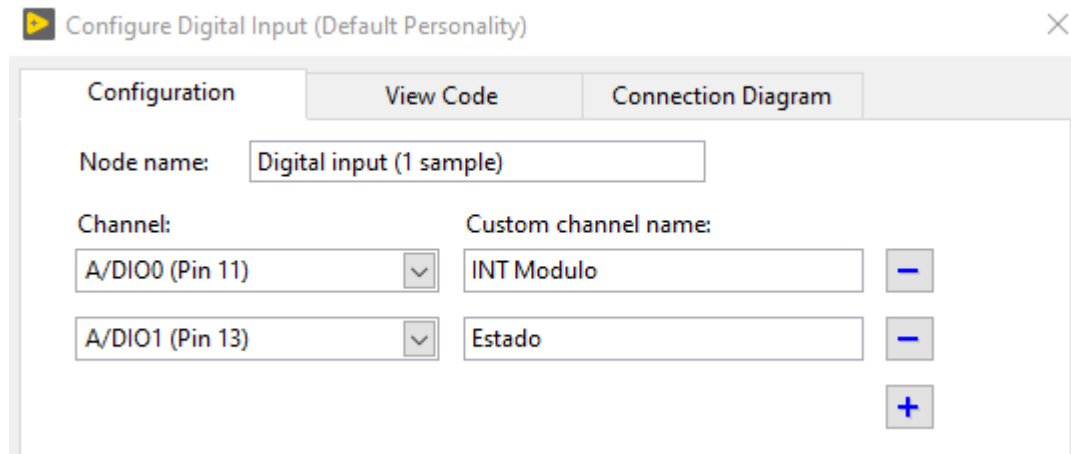


Figura 114. Configuración de entradas digitales, práctica 1.

Se procede a conectar a la salida digital de la interrupción a un lazo de selección de caso para la ejecución cuando se realice una variación en la inclinación del prototipo como se muestra en la figura 115.

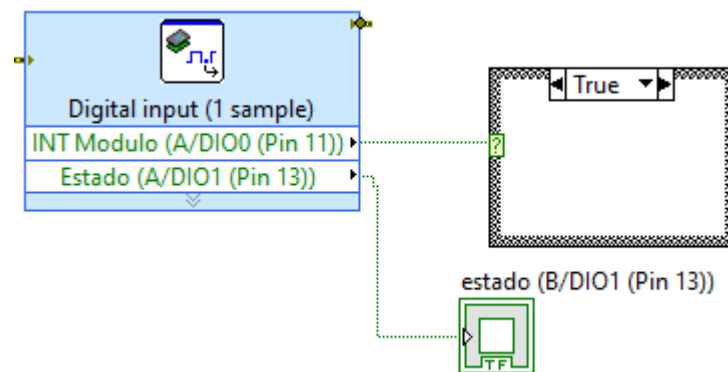


Figura 115. Conexión a lazo de selección, práctica 1

Agregar un bloque de lectura i2c donde la dirección del dispositivo es 0x69 y el dato para que envíen el dato es 0xA8, con un tamaño de 6 bits en el bloque de diagrama como se muestra en la figura 116.

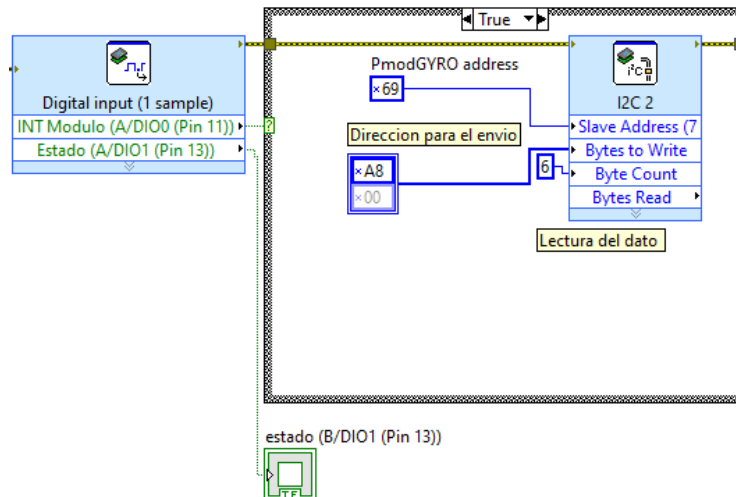


Figura 116. Lazo de selección con bloque i2c, práctica 1

Posteriormente se procede a separar el dato en 2 bits cada uno y convirtiendo a datos del tipo Word teniendo una dimensión o tamaño máximo de 2^{16} teniendo como resultado los valores en los ejes “X”, “Y”, “Z” agregando en un bundle para mostrar los datos en una sola grafica como se muestra en la figura 117.

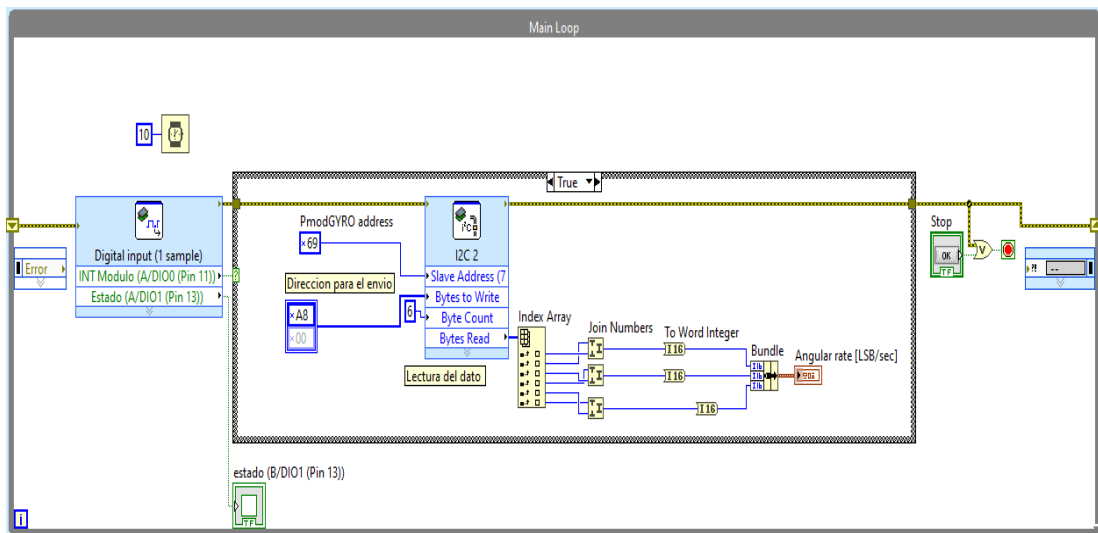


Figura 117. Bundle con datos para gráfica, práctica 1

Se procede a compilar el programa en la ni myRio en la figura se muestra el funcionamiento del prototipo para la lectura de los ejes, cada movimiento del robot es de manera manual para constatar el valor de inclinación como se muestra en la figura 118.

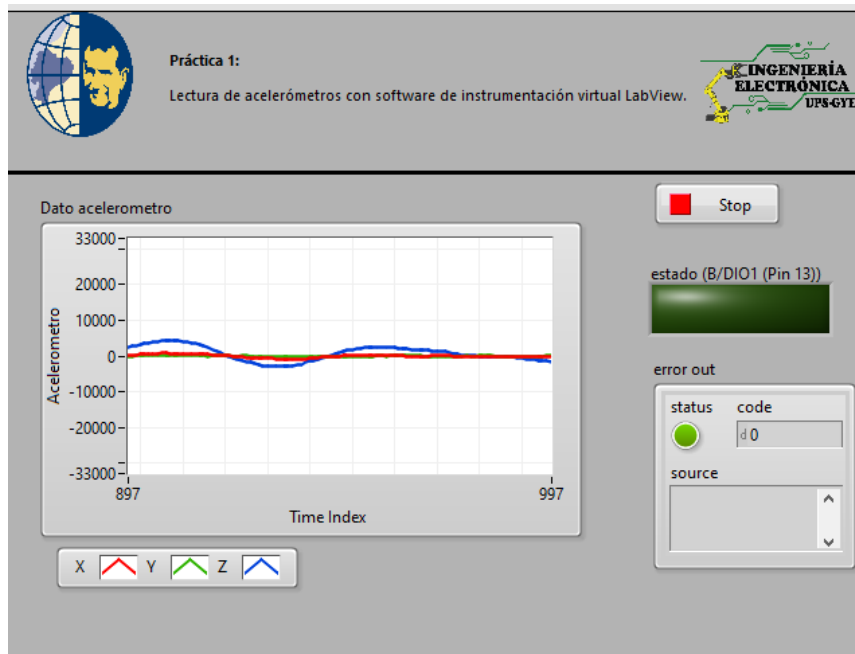


Figura 118. Interfaz de práctica 1

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Configurar correctamente el proyecto en LabView.

Anexo 4.2 Solución de práctica 2

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 119.

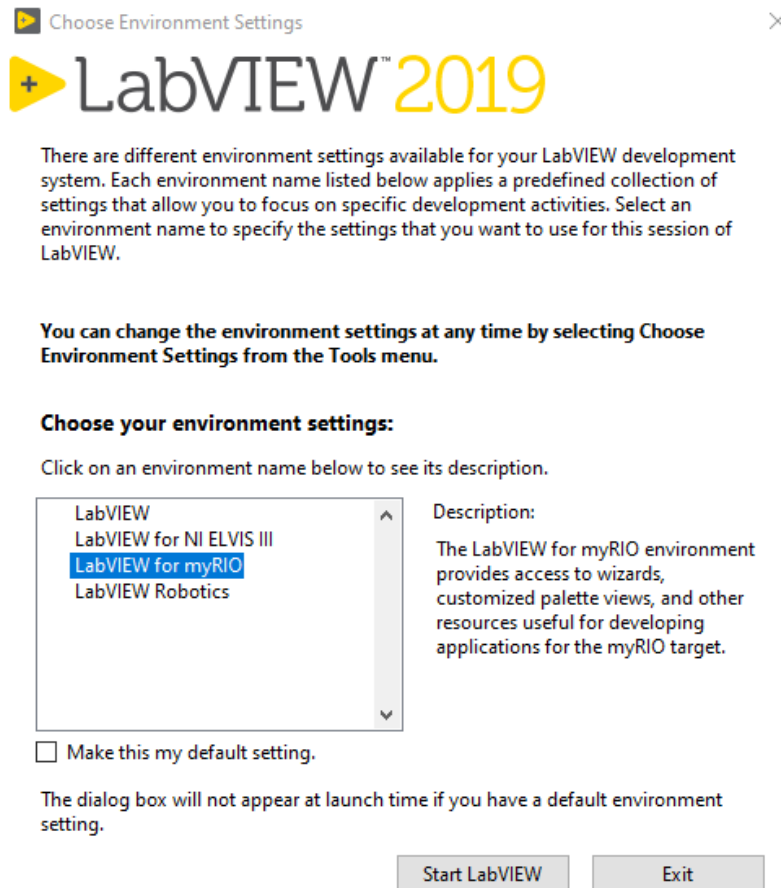


Figura 119. Ejecución de LabView como compatible con ni myRio práctica 2

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 120.

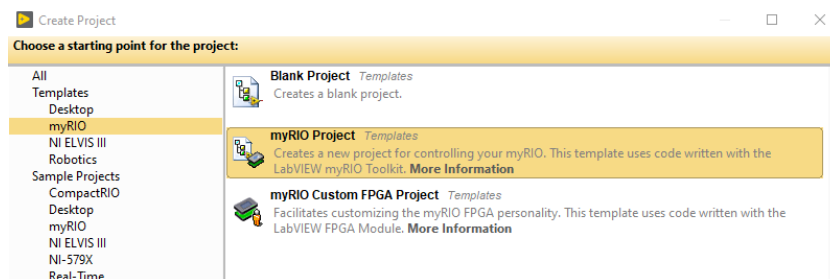


Figura 120. Nuevo Proyecto, práctica 2

Configurar la ruta donde se guarda el archivo (carpeta practica 2), nombre (Scada mediante wifi de acelerómetros con software de instrumentación virtual LabView) y el tipo de conexión como wifi y dar en finalizar como se muestra en la imagen 121.

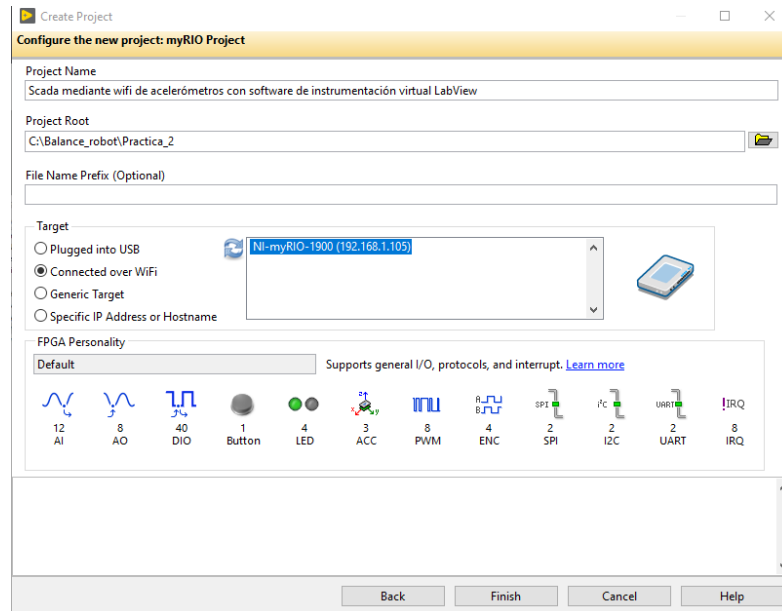


Figura 121. Configuración de ruta, práctica 2

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 122.

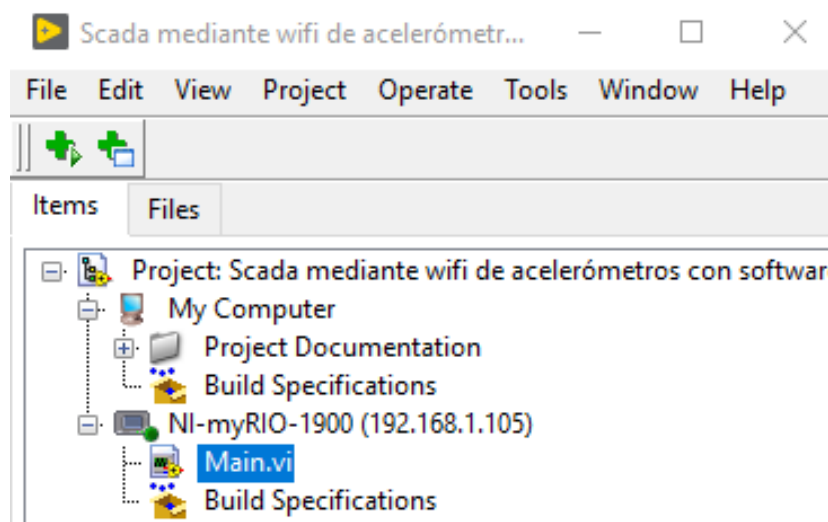


Figura 122. Administrador de proyectos, selección práctica 2

En el bloque de diagrama se agrega un bloque i2c para ni myrio y se configura como escritura , en el puerto a y su velocidad de 100 kbps como se muestra en la figura 123.

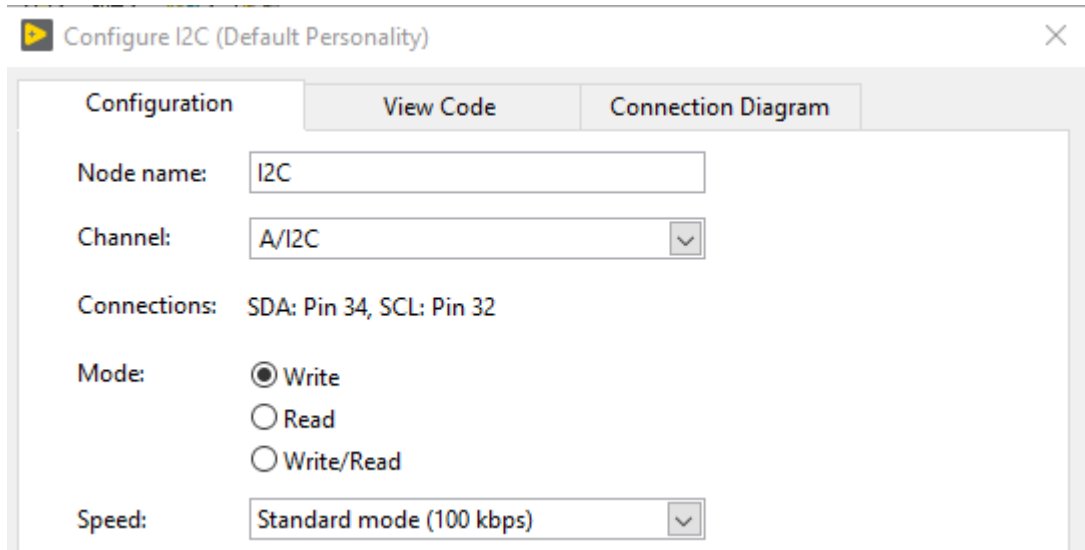


Figura 123. Configuración de i2c para myRio de práctica 2

En el bloque i2c se crea una variable que contenga la direccion del dispositivo **0x69** y un arreglo donde se configuran los parametros en formato de arreglo bidimensional para el dispositivo como se muestra en la figura 124.

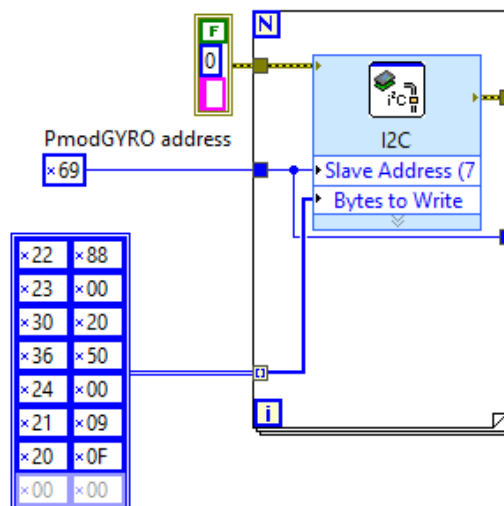


Figura 124. Bloque de variables i2c, práctica 2

Crear un lazo de repetición (while loop) en el cual se agrega dos datos de entradas digital para el puerto A DIO-0 y el A DIO 1 los cuales tienen la función de captar la señal del módulo acelerómetro como se muestra en la imagen 125.

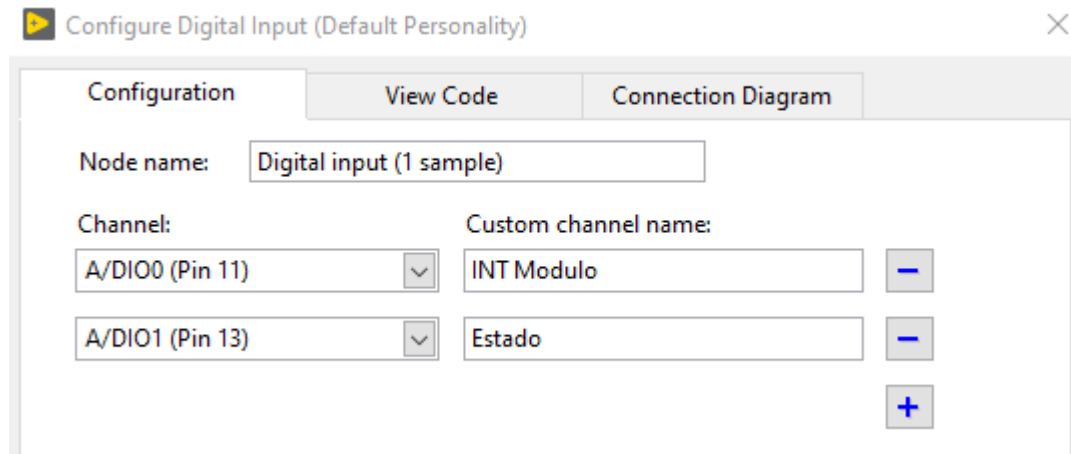


Figura 125. Configuración de entradas digitales, práctica 2.

Se procede a conectar a la salida digital de la interrupción a un lazo de selección de caso para la ejecución cuando se realice una variación en la inclinación del prototipo como se muestra en la figura 126.

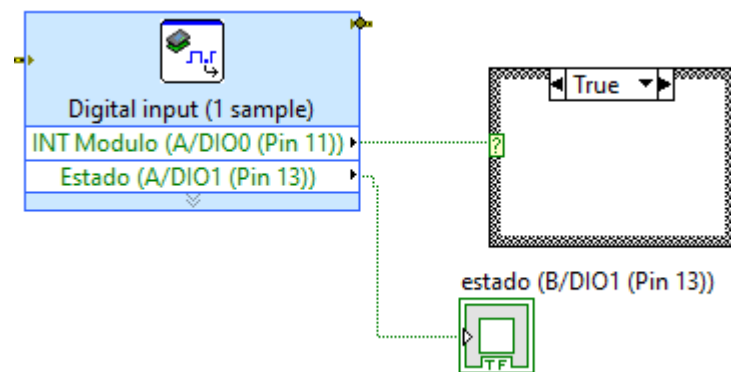


Figura 126. Conexión a lazo de selección, práctica 2

Agregar un bloque de lectura i2c donde la dirección del dispositivo es 0x69 y el dato para que envíen el dato es 0xA8, con un tamaño de 6 bits en el bloque de diagrama como se muestra en la figura 127.

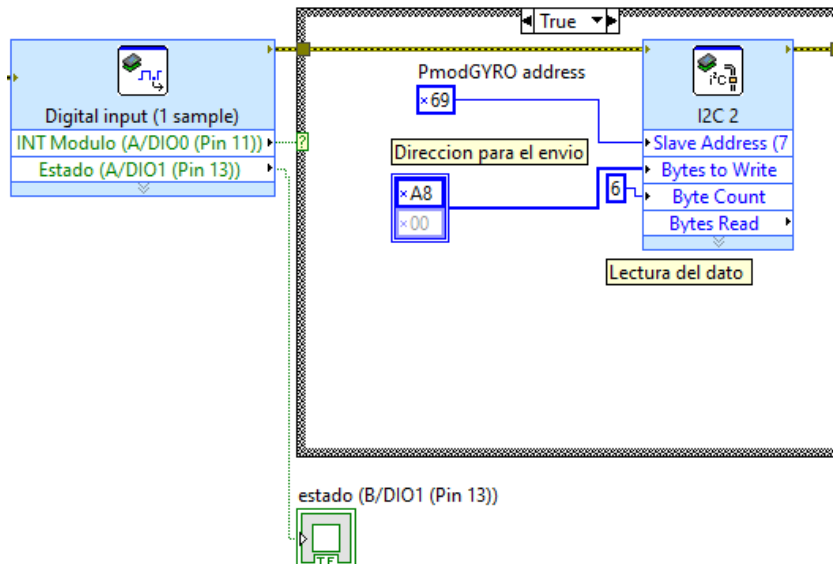
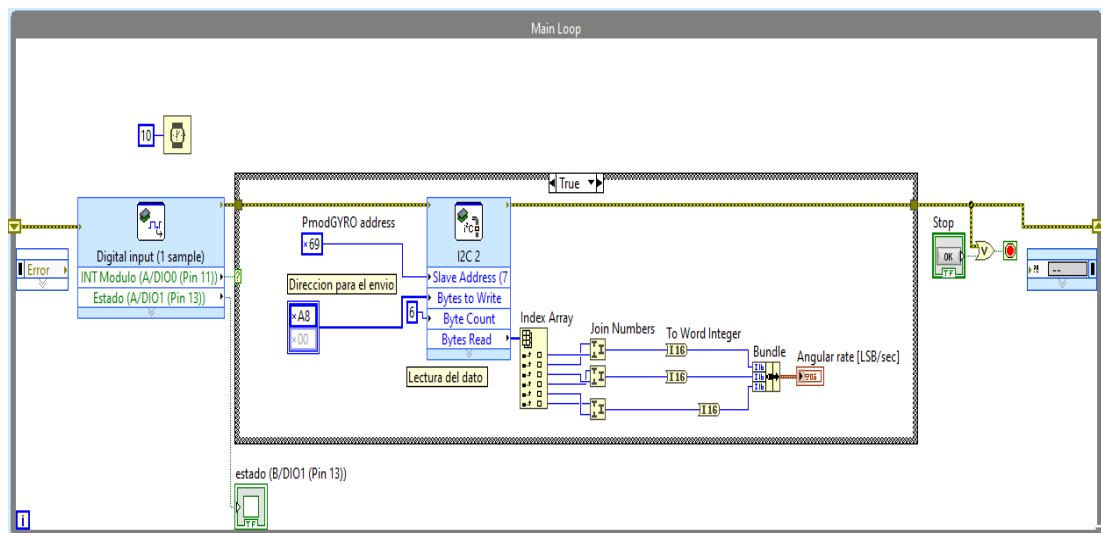


Figura 127. Lazo de selección con bloque i2c, práctica 2

Posteriormente se procede a separar el dato en 2 bits cada uno y convirtiendo a datos del tipo Word teniendo una dimensión o tamaño máximo de 2^{16} teniendo como resultado los valores en los eje “X”, “Y”, “Z” como se muestra en la figura 128.



Bundle con datos para gráfica, práctica 2

Figura 128.

Se procede a compilar el programa de manera inalámbrica en la ni myRio en la figura se muestra el funcionamiento del prototipo para la lectura de

los ejes , cada movimiento del robot es de manera manual para constatar el valor de inclinación como se muestra en la figura 129.

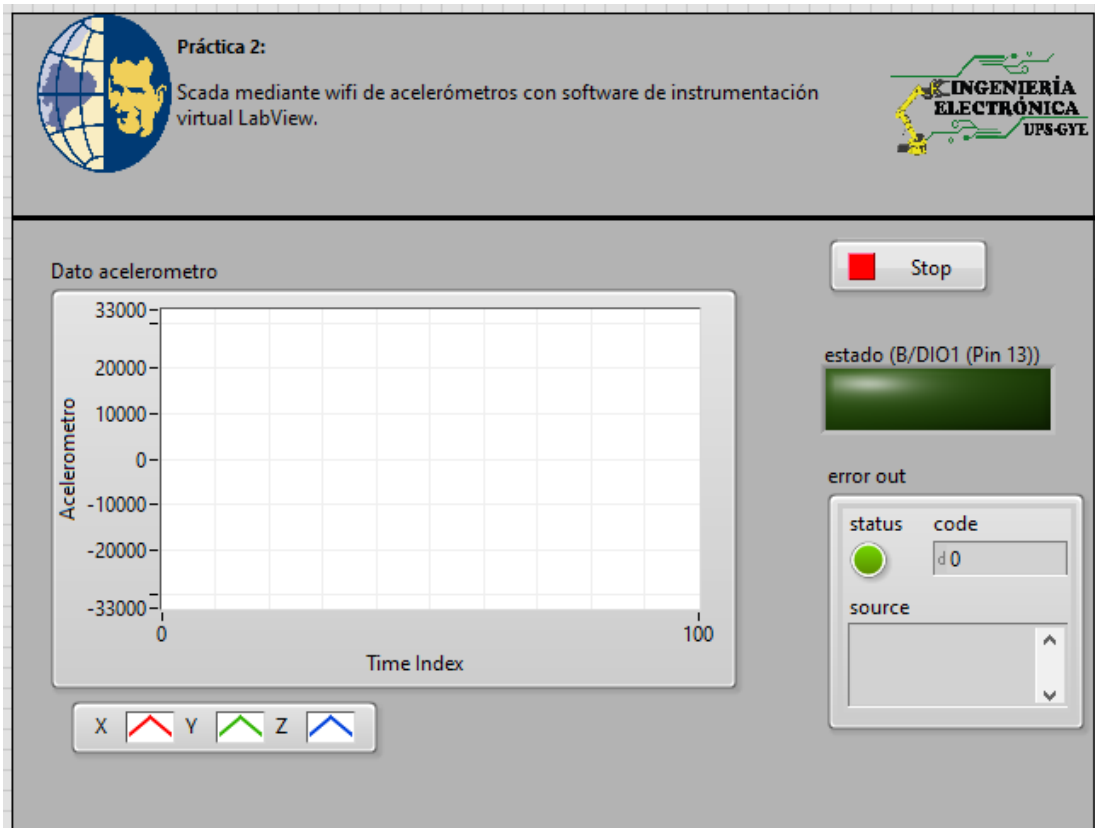


Figura 129. Interfaz practica 2.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Configurar correctamente el proyecto en LabView.
- Verificar que tenga conexión a la red inalámbrica

Anexo 4.3 Solución de práctica 3

Ejecutar LabView y seleccionar el modo de LabView compatible con el ni myRio como se muestra en la figura 130.

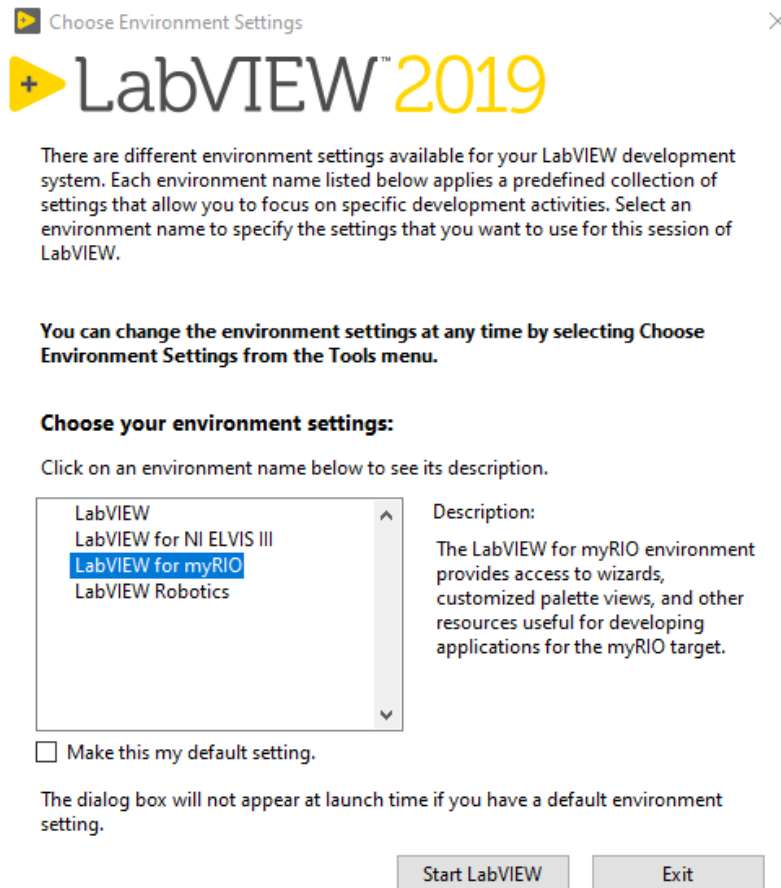


Figura 130. Ejecución de LabView como compatible con ni myRio
práctica 3

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 131.

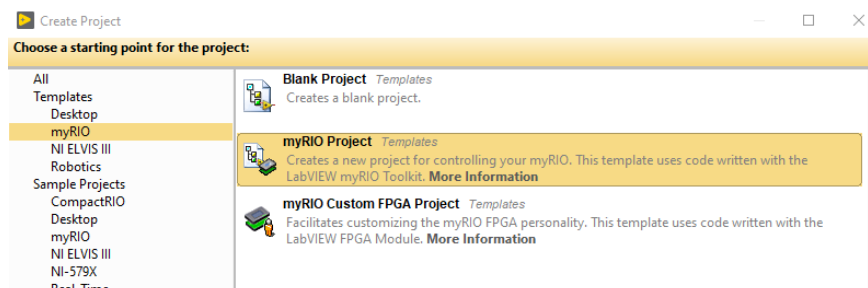


Figura 131. Nuevo Proyecto, práctica 3

Configurar la ruta donde se guarda el archivo (carpeta practica 3), nombre (Conversión analógica con sensor ultrasónico.) y el tipo de conexión como USB y dar en finalizar como se muestra en la imagen 132.

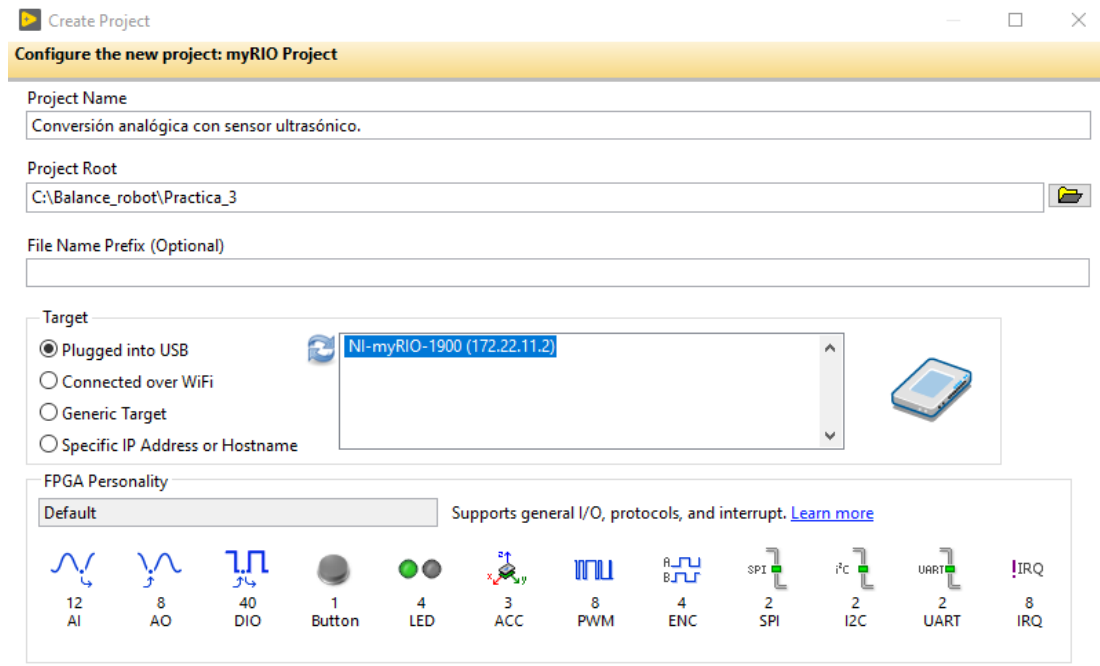


Figura 132. Configuración de ruta, práctica 3.

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 133.

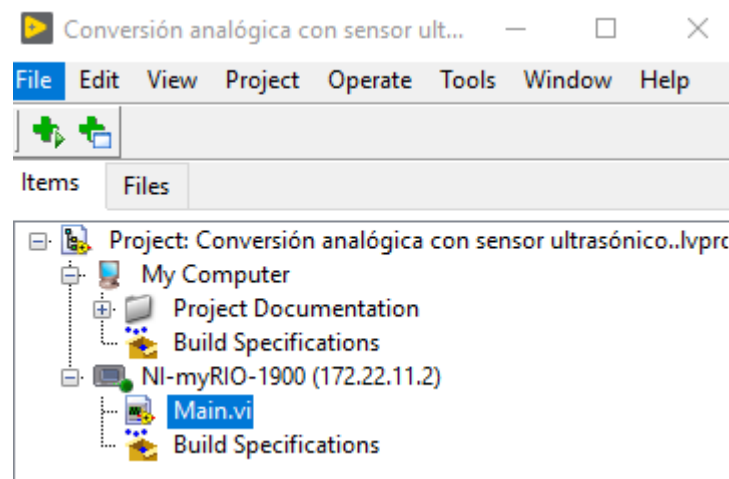


Figura 133. Administrador de proyectos, selección práctica 3.

En el bloque de diagrama agregar un bloque de lectura analógica y configurar dos entradas una en el puerto C AI0 para el sensor ultrasónico análogo delantero y otra en el puerto C AI1 para el sensor posterior como se muestra en la figura 134

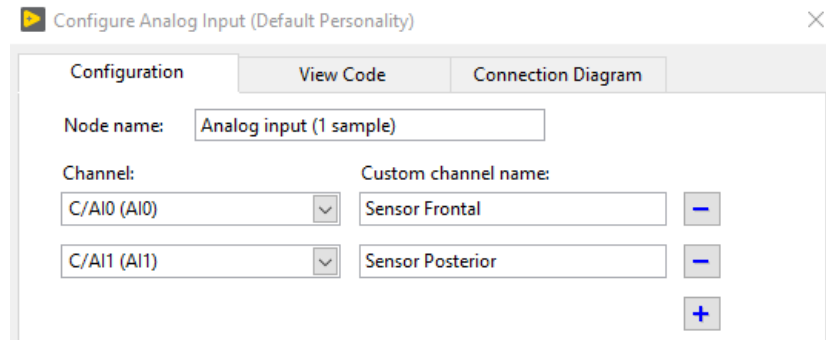


Figura 134. Configuración de entradas analógicas, práctica 3.

Las conexiones del sensor ultrasónico 1 frontal y sensor ultrasónico 2 posterior con él ni myRio se muestra en la tabla 12

Sensores	Ni MyRio
VCC Ultrasónico 1	Pin VCC Puerto C
Salida Ultrasónico 1	Pin Ai0+ Puerto C
GND Ultrasónico 1	Pin GND Puerto C
VCC Ultrasónico 2	Pin VCC Puerto C
Salida Ultrasónico 2	Pin Ai0+ Puerto C
GND Ultrasónico 2	Pin GND Puerto C

Tabla 12 Conexiones de controlador de motores

Luego en el bloque de diagrama a la salida de las lecturas se aplica una ganancia de 245 para la conversión de los valores de voltaje a un dato en centímetros como se muestra en la figura 136

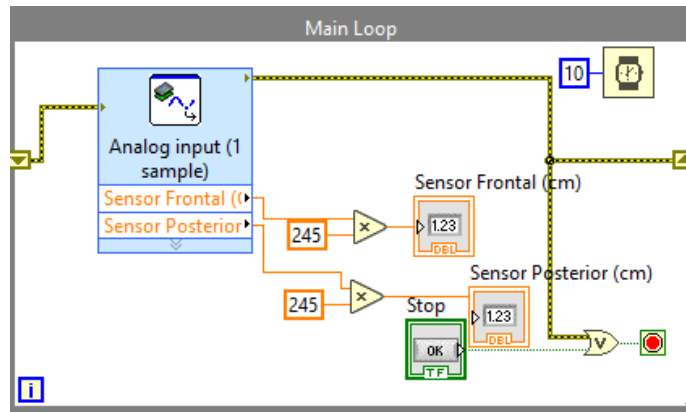


Figura 135. Main Loop con entradas analógicas, práctica 3

Posteriormente se procede a agregar un bloque para unir señales para mostrar los datos en una gráfica como se muestra en la figura 136.

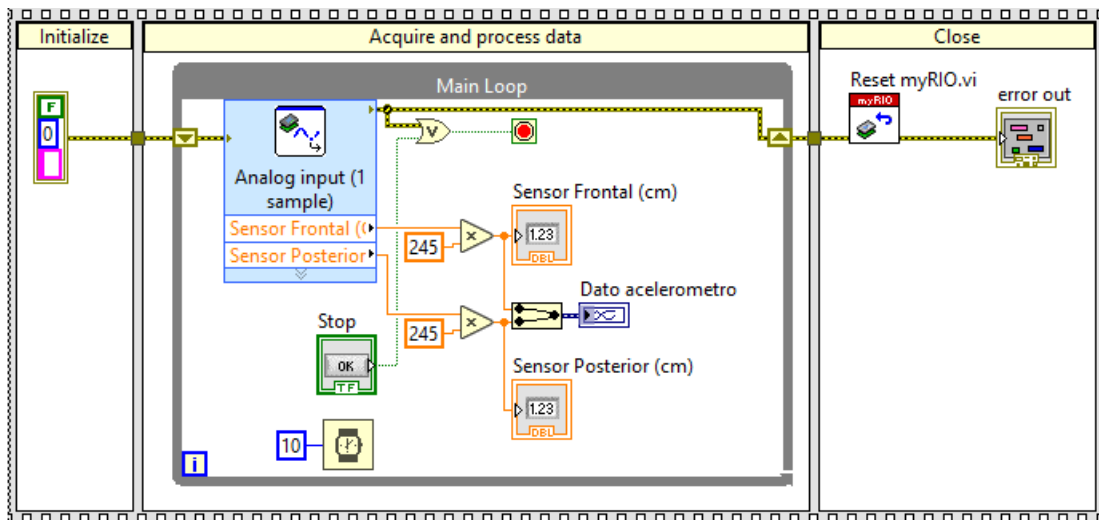


Figura 136. Bloque unión señales practica 3

Se procede a ejecutar el programa y verificar el funcionamiento como se muestra en la figura 137.

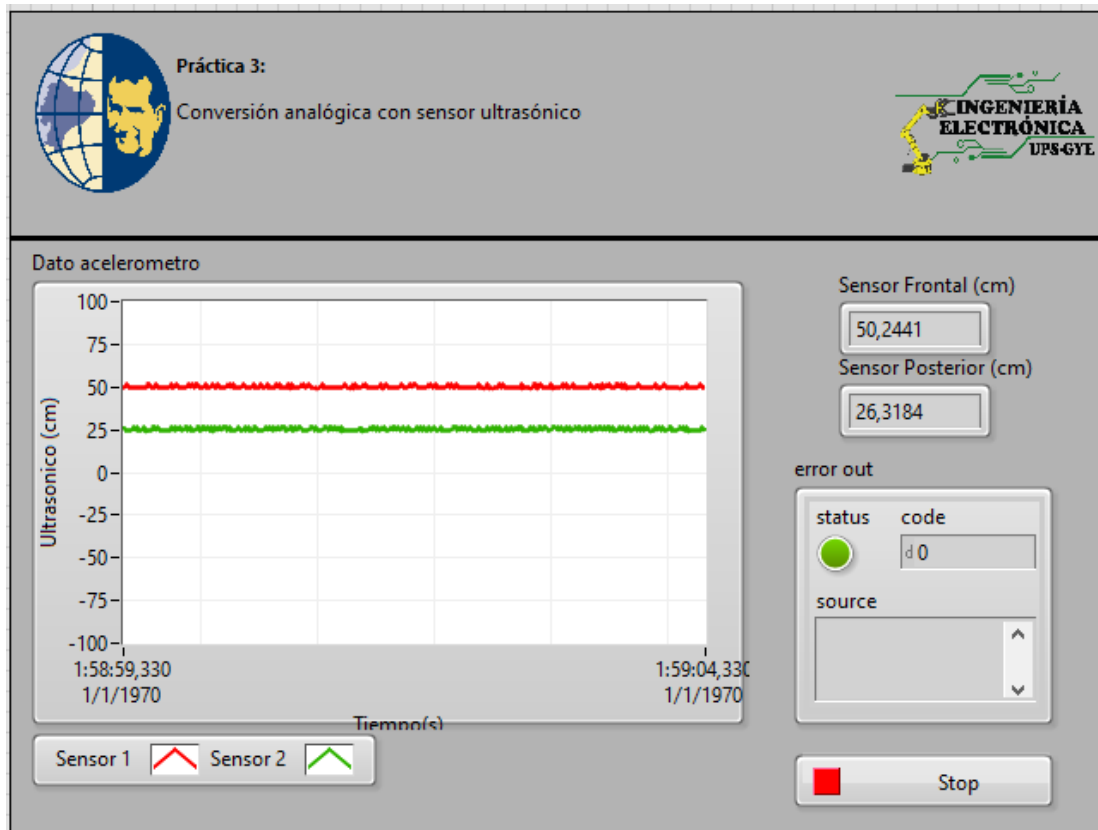


Figura 137. Interfaz practica 3.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Conectar correctamente los sensores al puerto C.
- Verificar la alimentación a los sensores

Anexo 4.4 Solución de práctica 4

Ejecutar LabView y seleccionar el modo de LabView compatible con el ni myRio como se muestra en la figura 138.

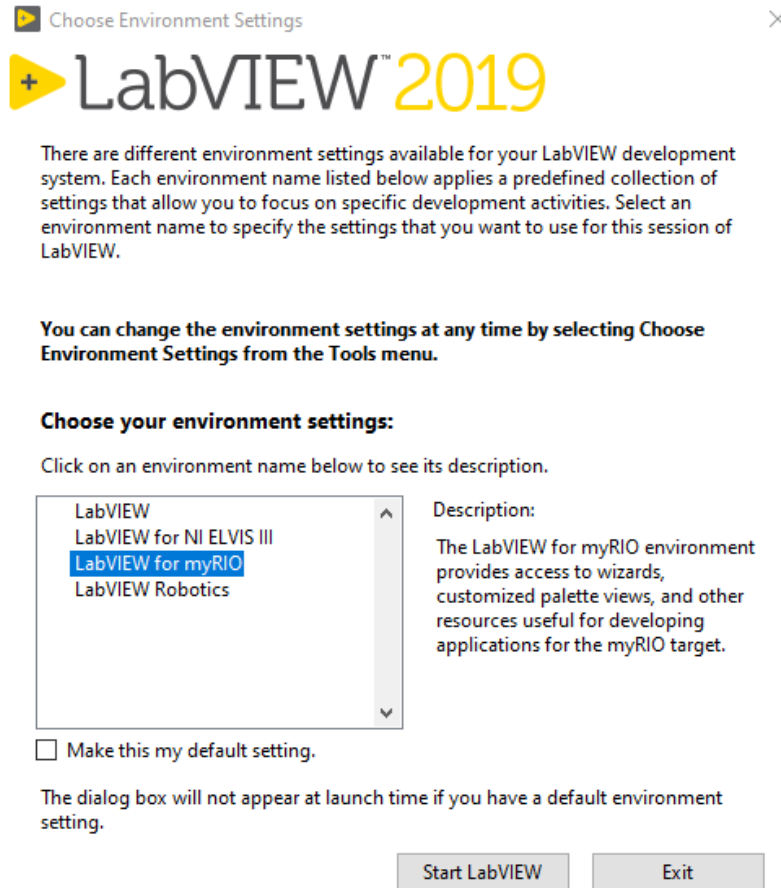


Figura 138. Ejecución de LabView como compatible con ni myRio práctica 4

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 139.

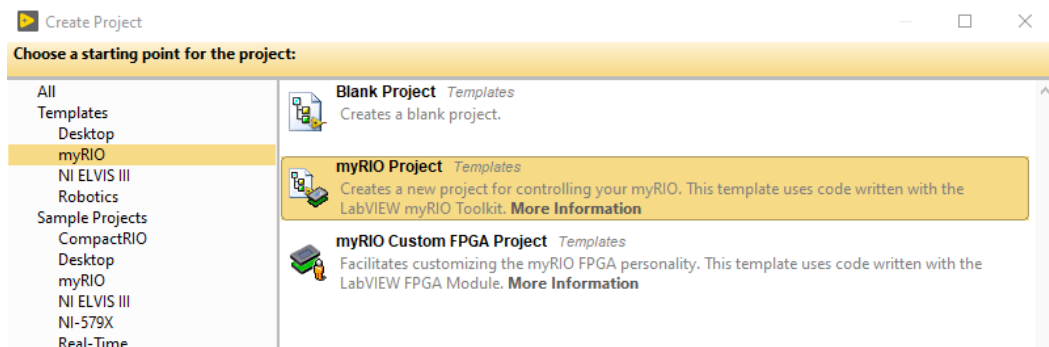


Figura 139. Nuevo Proyecto, práctica 4

Configurar la ruta donde se guarda el archivo (carpeta practica 4), nombre (Scada mediante wifi de lecturas analógica con sensor ultrasónico y el tipo de conexión como USB y dar en finalizar como se muestra en la imagen 140.

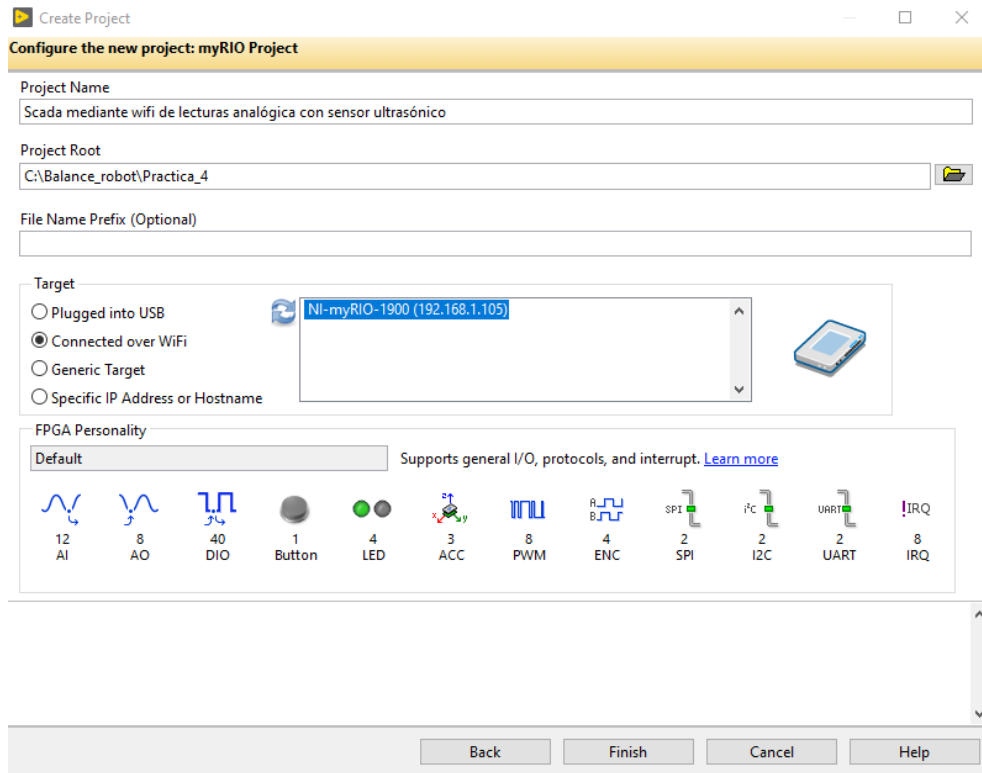


Figura 140. Configuración de ruta, práctica 4

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 141.

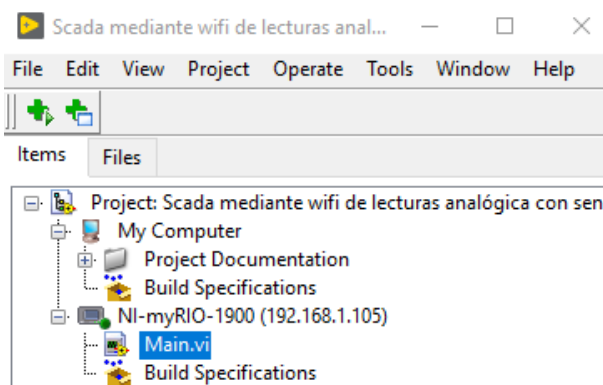


Figura 141. Administrador de proyectos, selección práctica 4.

En el bloque de diagrama agregar un bloque de lectura analógica y configurar dos entradas una en el puerto C AI0 para el sensor ultrasónico análogo delantero y otra en el puerto C AI1 para el sensor posterior como se muestra en la figura 142

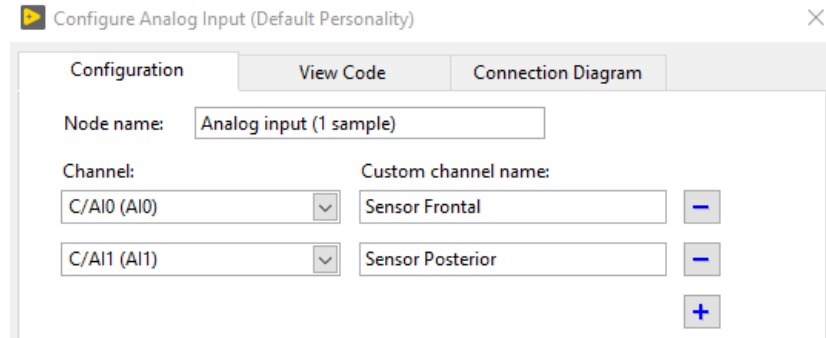


Figura 142. Configuración de entradas analógicas, práctica 4.

Las conexiones del sensor ultrasónico 1 frontal y sensor ultrasónico 2 posterior con él ni myRio se muestra en la tabla 13

Sensores	Ni MyRio
VCC Ultrasónico 1	Pin VCC Puerto C
Salida Ultrasónico 1	Pin Ai0+ Puerto C
GND Ultrasónico 1	Pin GND Puerto C
VCC Ultrasónico 2	Pin VCC Puerto C
Salida Ultrasónico 2	Pin Ai0+ Puerto C
GND Ultrasónico 2	Pin GND Puerto C

Tabla 13 Conexiones de controlador de motores

Luego en el bloque de diagrama a la salida de las lecturas se aplica una ganancia de 245 para la conversión de los valores de voltaje a un dato en centímetros como se muestra en la figura 143.

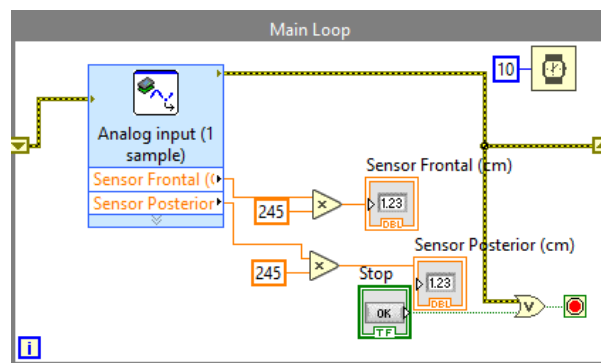


Figura 143. Main Loop con entradas analógicas, práctica 4

Posteriormente se procede a agregar un bloque para unir señales para mostrar los datos en una gráfica como se muestra en la figura 144.

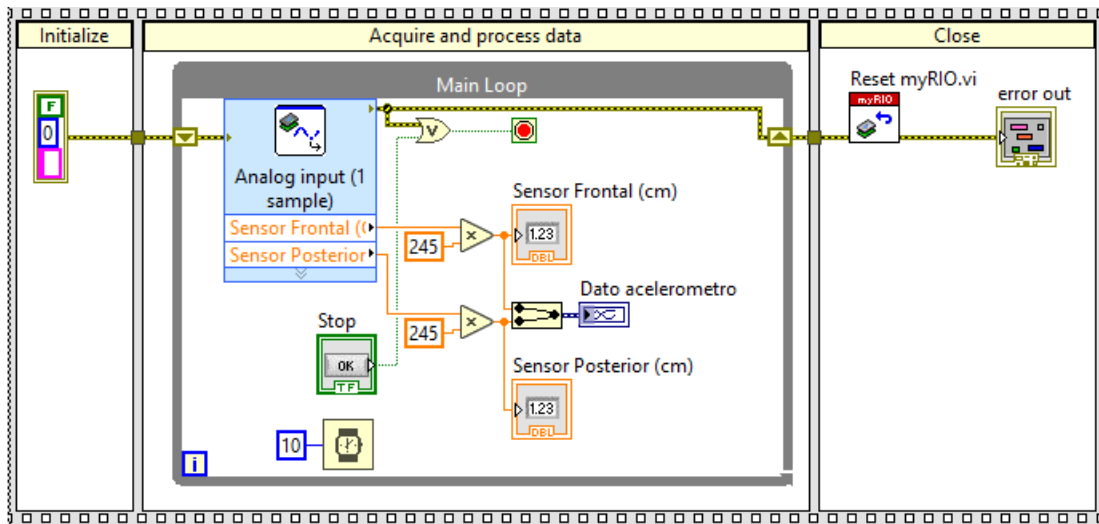


Figura 144. Bloque unión señales practica 4

Se procede a ejecutar el programa y verificar el funcionamiento como se muestra en la figura 145.

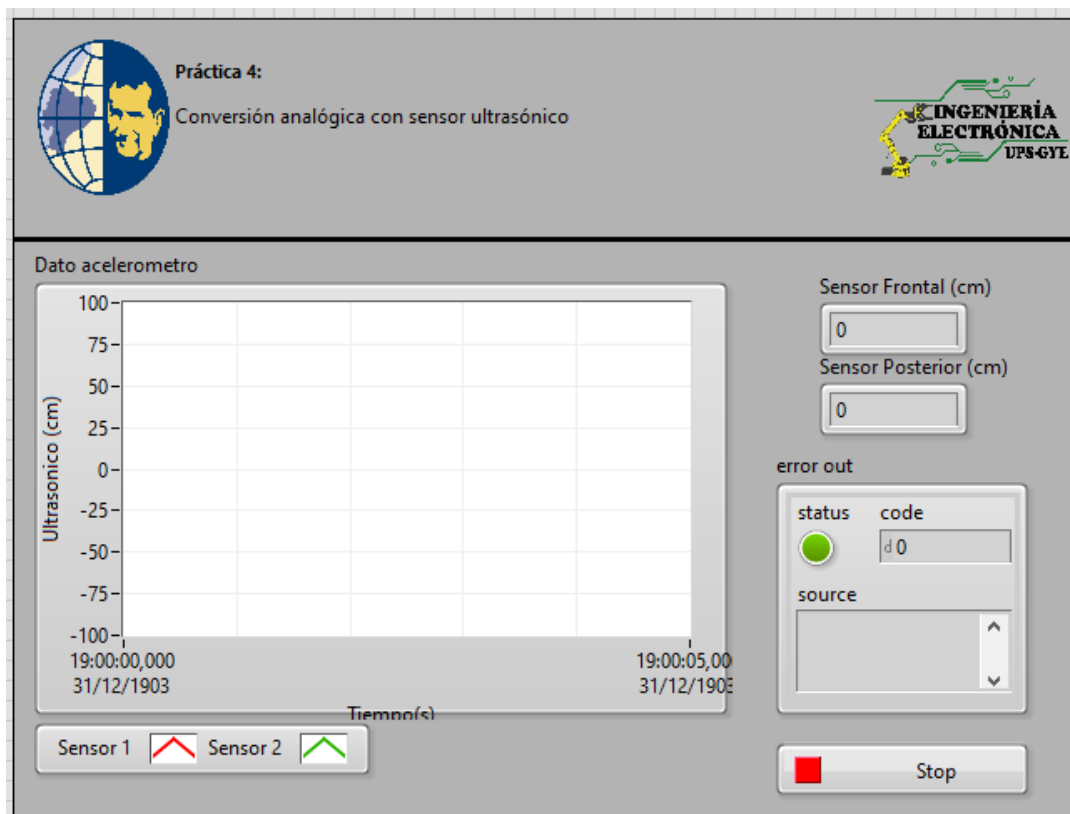


Figura 145. Interfaz practica 4.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Conectar correctamente los sensores al puerto C.
- Verificar la alimentación a los sensores

Anexo 4.5 Solución de práctica 5

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 146.

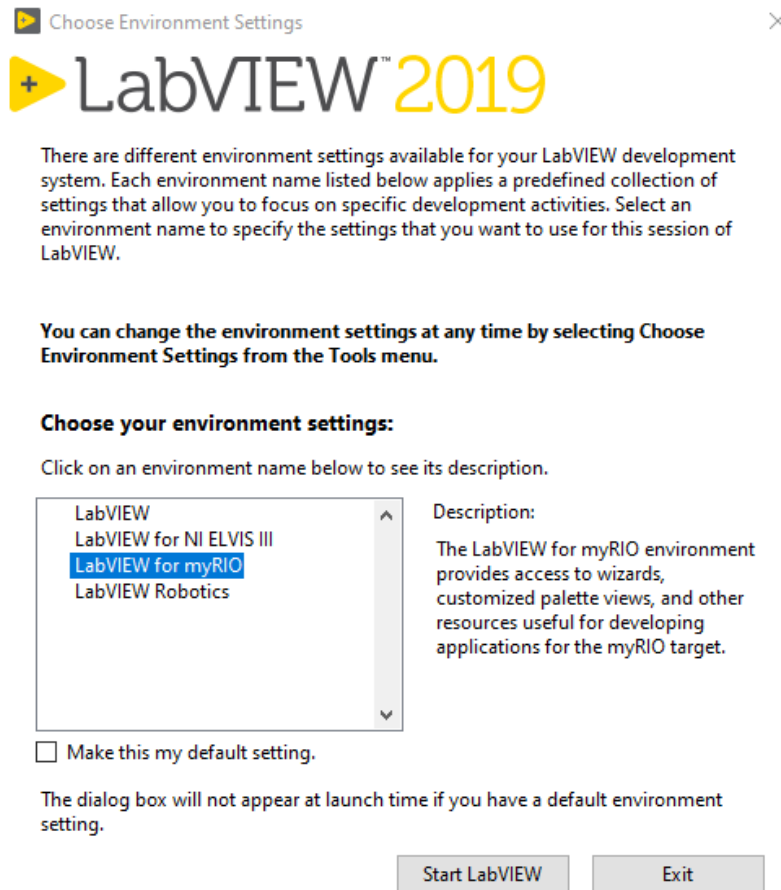


Figura 146. Ejecución de LabView como compatible con ni myRio
práctica 5

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 147.

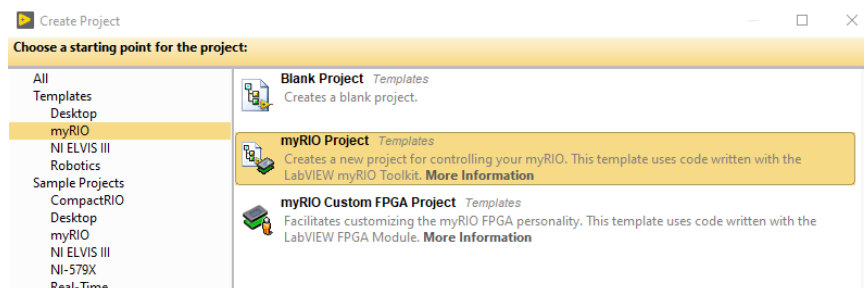


Figura 147. Nuevo Proyecto, práctica 5

Configurar la ruta donde se guarda el archivo (carpeta practica 5), nombre (PID robot auto balanceado) y el tipo de conexión como USB y dar en finalizar como se muestra en la imagen 148.

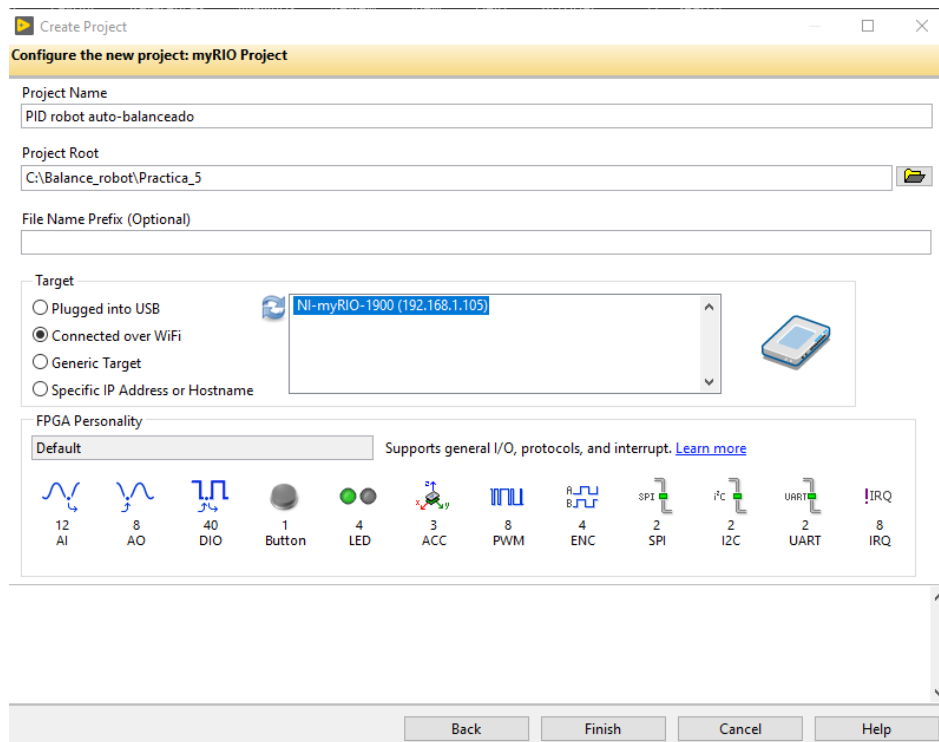


Figura 148. Configuración de ruta, práctica 5

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 149.

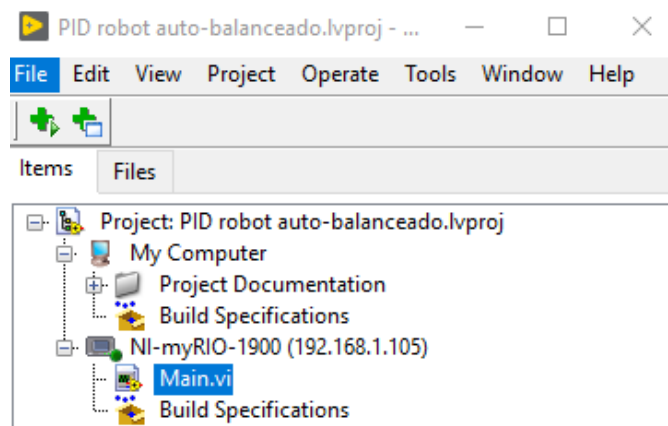


Figura 149. Administrador de proyectos, selección práctica 5.

En el bloque de diagrama se agrega un bloque i2c para ni myrio y se configura como escritura , en el puerto a y su velocidad de 100 kbps como se muestra en la figura 150.

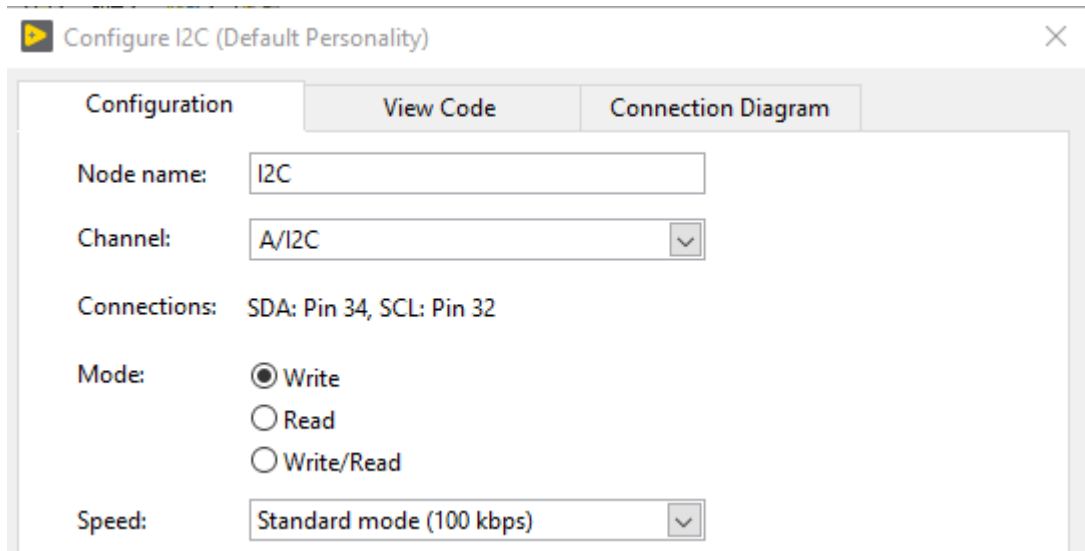


Figura 150. Configuración de i2c para myRio de práctica 5.

En el bloque i2c se crea una variable que contenga la direccion del dispositivo **0x69** y un arreglo donde se configuran los parametros en formato de arreglo bidimensional para el dispositivo como se muestra en la figura 151

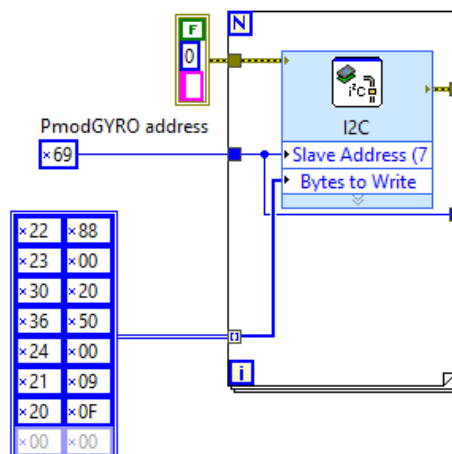


Figura 151. Bloque de variables i2c, práctica 5

Crear un lazo de repetición (while loop) en el cual se agrega dos datos de entradas digital (bloque de diagrama >>myRio >> digital input) para el puerto A DIO-0 y el A DIO 1 los cuales tienen la función de captar la señal del módulo acelerómetro como se muestra en la imagen 152.

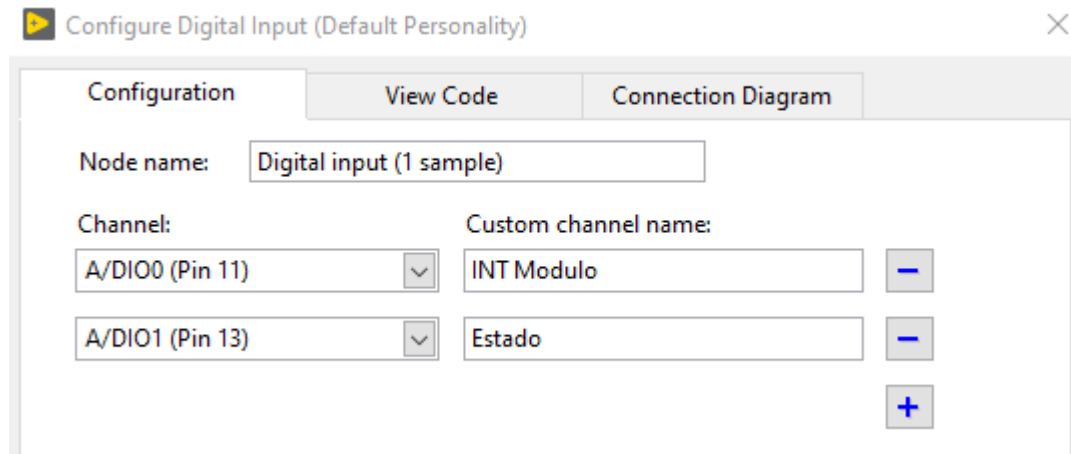


Figura 152. Configuración de entradas digitales, práctica 5.

Se procede a conectar a la salida digital de la interrupción a un lazo de selección de caso para la ejecución cuando se realice una variación en la inclinación del prototipo como se muestra en la figura 153

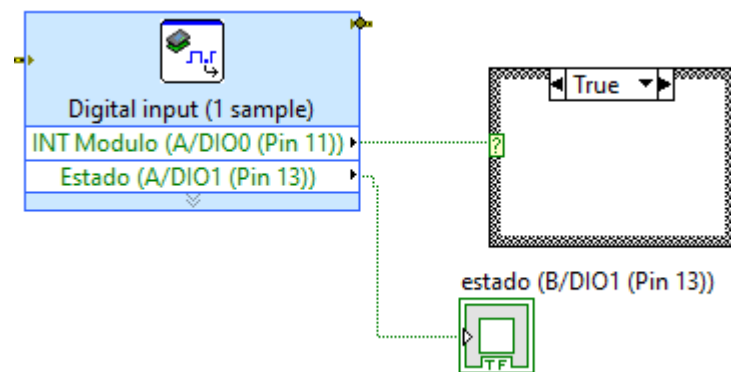


Figura 153. Conexión a lazo de selección, práctica 5.

Agregar un bloque de lectura i2c donde la dirección del dispositivo es 0x69 y el dato para que envíen el dato es 0xA8, con un tamaño de 6 bits en el bloque de diagrama como se muestra en la figura 154

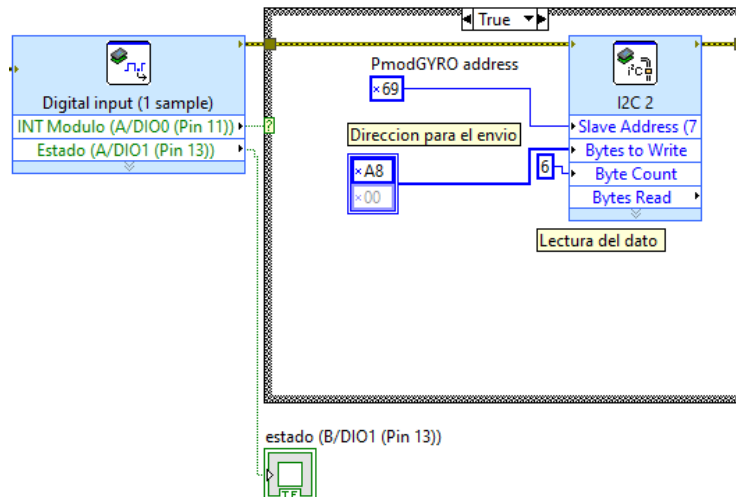


Figura 154. Lazo de selección con bloque i2c, práctica 5.

Posteriormente se procede a separar el dato en 2 bits cada uno y convirtiendo a datos del tipo Word teniendo una dimensión o tamaño máximo de 2^{16} teniendo como resultado los valores en los eje “X”, “Y”, “Z” agregando en un bundle para mostrar los datos en una sola grafica como se muestra en la figura 155

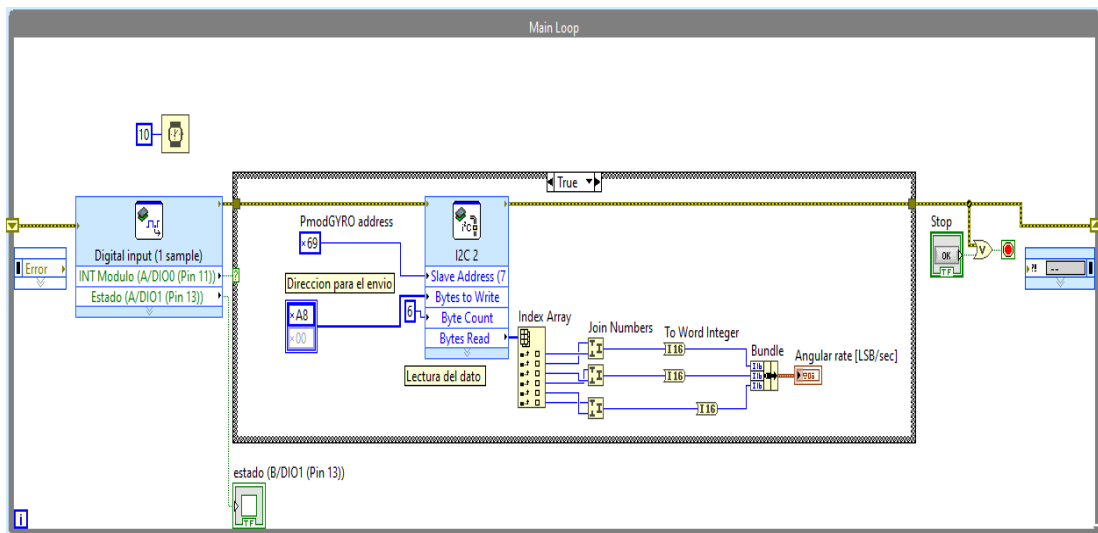


Figura 155. Bundle con datos para gráfica, práctica 5.

Agregar tres datos de lectura analógica (bloque de diagrama >>myRio >> accelerometer), las entradas del acelerómetro se activan los ejes “x”, “y”, “z” como se muestra en la imagen 156.

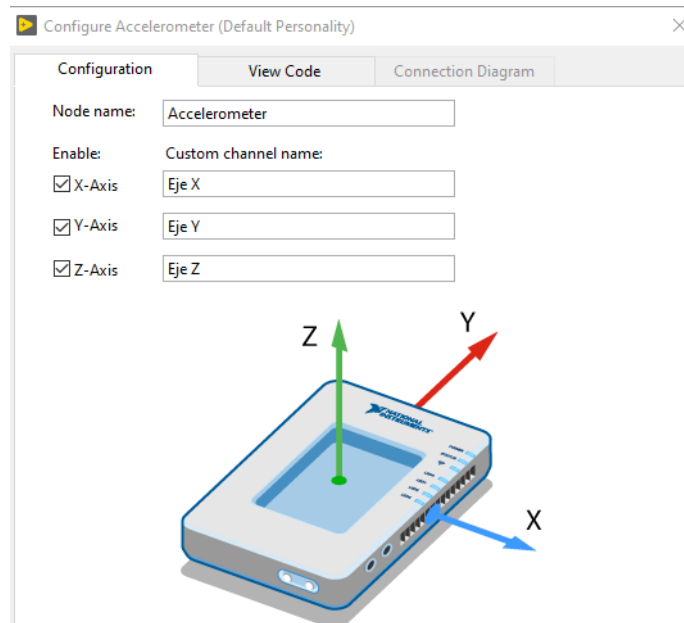


Figura 156. Ingreso de entradas analógicas, práctica 5.

Se procede a conectar las salidas a un indicador gráfico y se crea un indicador numérico para la variable del eje “z” como se muestra en la figura 157

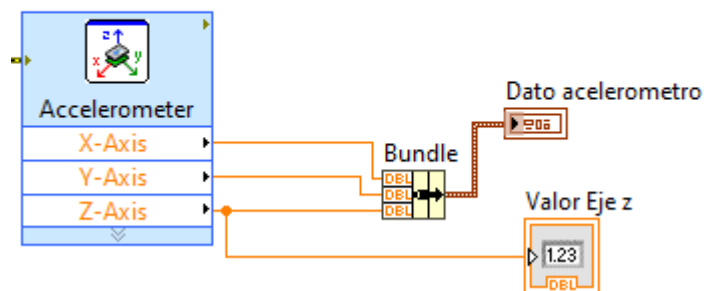


Figura 157. Conexión de salidas analógicas práctica 5.

En las salidas del giroscopio realizar una integral a cada salida para obtener el dato de desplazamiento y en el selector de caso donde cuando no se cumpla el rango de operación de 45 grados (2500) y -45 grados (-2500) el prototipo entre en inactividad mediante la escritura de dos valores análogos en cada motor como se muestra en la figura 158

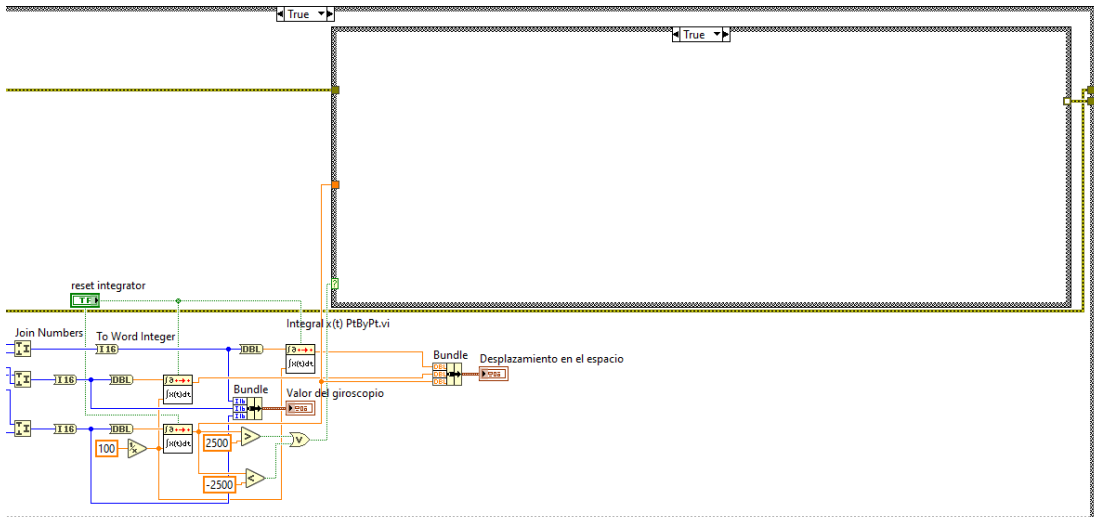


Figura 158. Obtención de datos de desplazamiento práctica 5.

En el selector de casos cuando la condicional es verdadero y se encuentra en un rango de operación no óptimo, agregar dos salidas análogas (bloque de diagrama >> myRio >> Analog Output) para el control de los motores, la primera en el puerto B A01 y la segunda en el puerto B A02 como se muestra la configuración en la figura 159

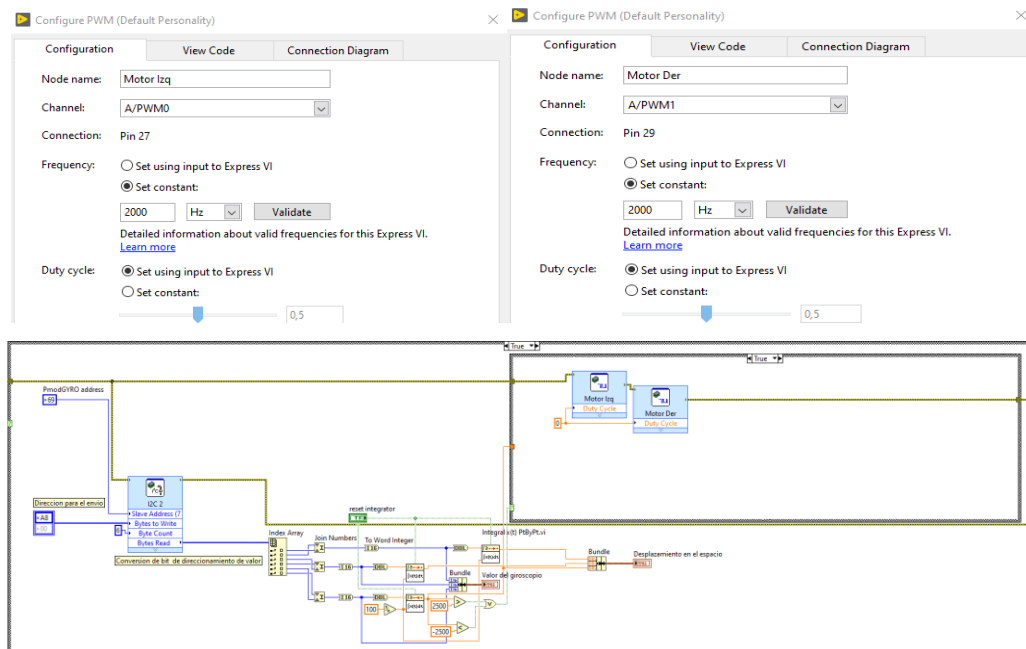


Figura 159. Ingreso de dos salidas analógicas, práctica 5.

En el selector de casos cuando la condicional es falso, agregar un bloque de control pid (bloque de diagrama >> control y simulación >> pid >> pid controlador) donde el rango de la salida es -1 a 1 y la referencia es 0, a la

par en salida agregar un bloque para atenuar la señal de salida mediante un controlador de adelanto como se muestra en la figura 160

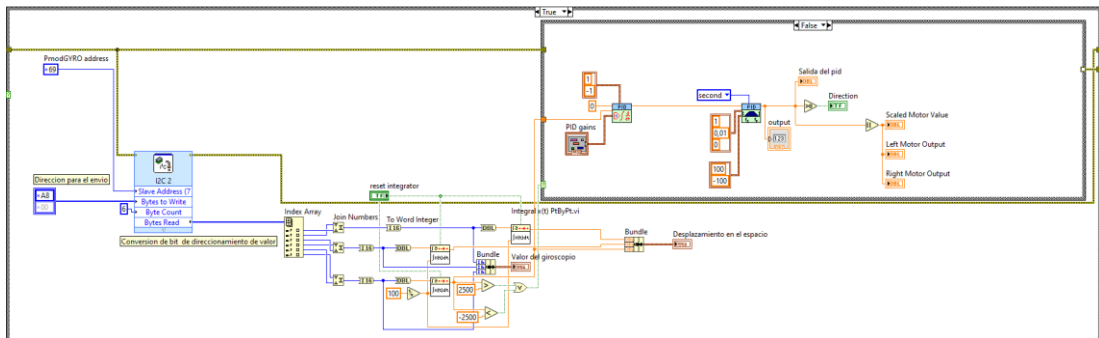


Figura 160. Señal de salida atenuada, practica 5.

Luego a la salida del controlador se agrega las salidas digitales para el control de sentido horario y antihorario en cada motor en el puerto A DIO 1 y puerto A DIO2 como se muestra en la figura 161.

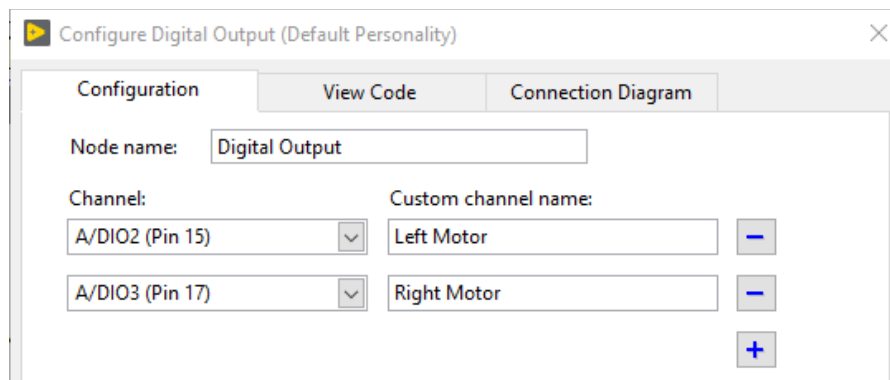


Figura 161. Configuración de salidas digitales, practica 5.

Posteriormente agregar dos bloques de salida analógicas para controlar el voltaje suministrado a los motores como se muestra en la figura 162.

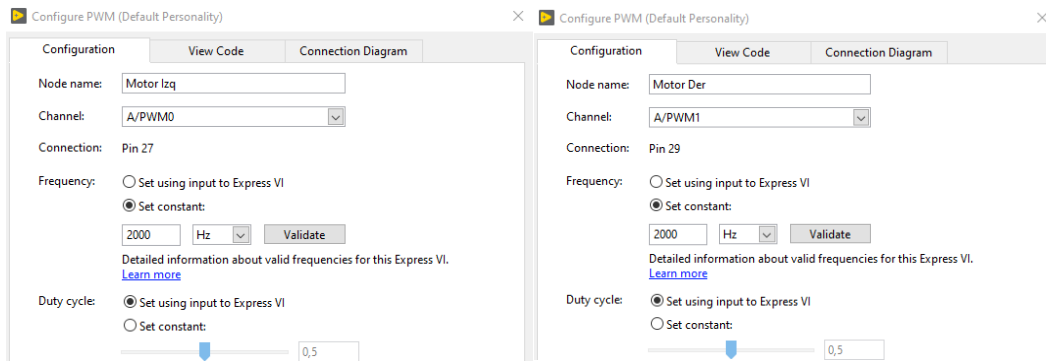


Figura 162. Ingreso de bloques de salidas analógicas, practica 5.

En la salida agregar un condicional cuando sea mayor que 0 enviar un dato a al motor 1 y en el segundo motor enviar la misma señal negada y controlar los voltajes mediante el dato de salida como valor absoluto como se muestra en la figura 163

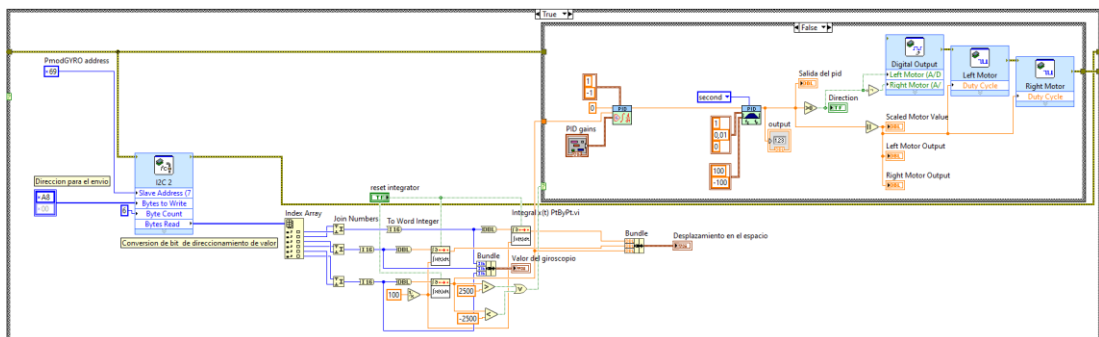


Figura 163. Condiciones de motores, practica 5.

Se inicia el método de sintonizando mediante ziegler nichols por ganancias critica donde se ubica la ganancia derivativa e integral en cero, y se cambia el valor proporcional hasta generar oscilaciones continuas donde se utiliza el valor de Kp en 1 como se muestra en la gráfica 164

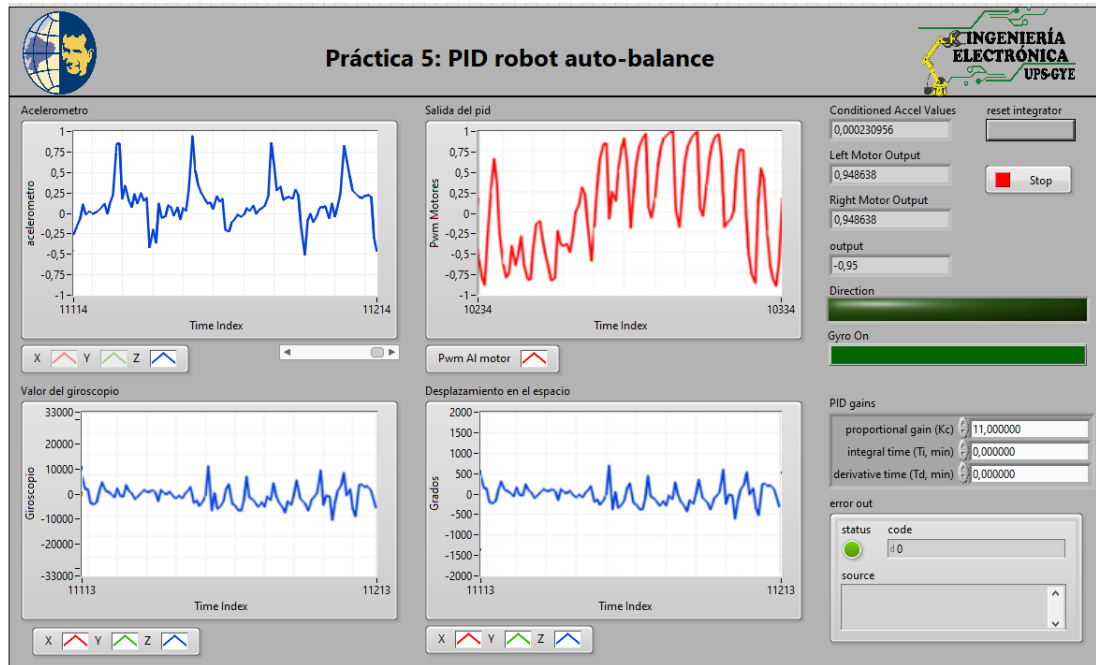


Figura 164. Sincronización de mediante ziegler nichols por ganancias critica, practica 5.

En la figura se utiliza como ganancia k_p el valor de 11 donde se aprecia las oscilaciones periódicas y se toma como k_c el valor de k_p como se aprecia en la figura 165.

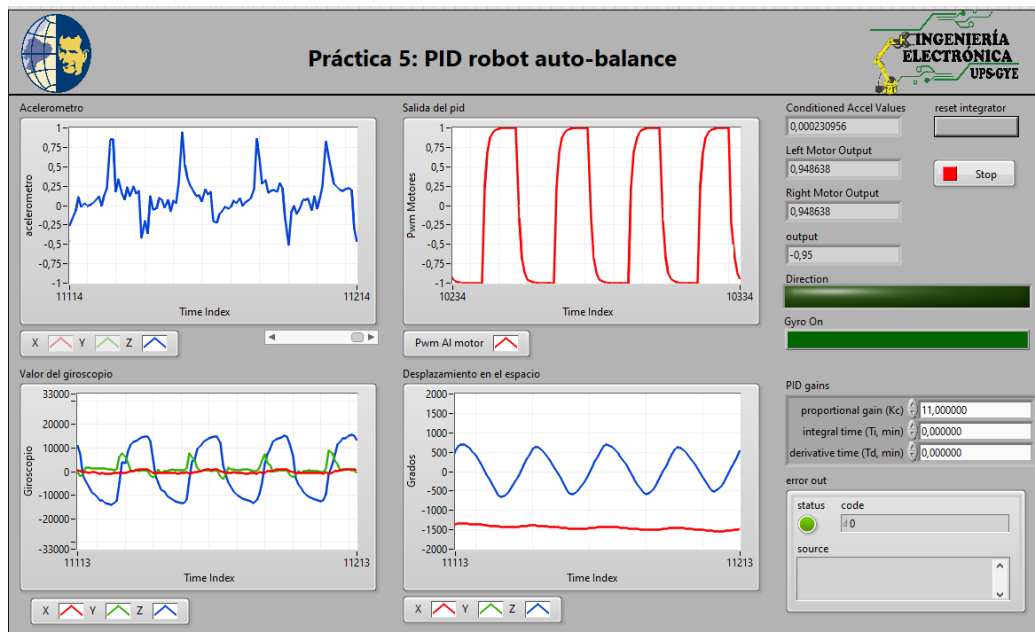


Figura 165. Gráficas de ganancia K_p , practica 5.

Se procede a tomar los valores de repetición cíclica y se encuentra que el valor de t_c es 23 como se muestra en la figura 166.

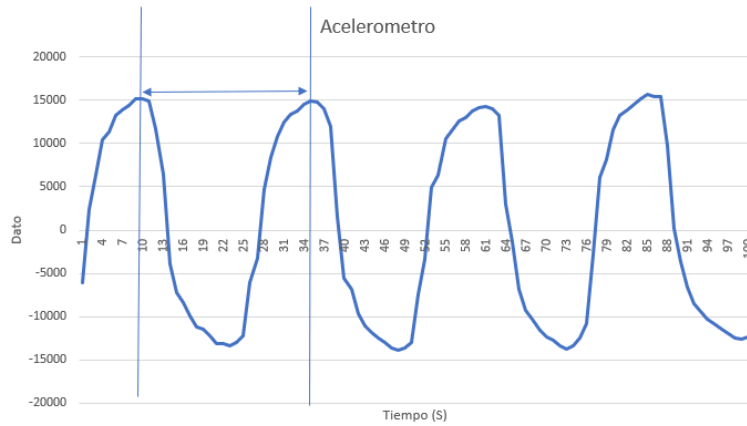


Figura 166. Valor de repetición de T_c , practica 5.

Se procede a utilizar la tabla 14 con el valor de $k_c = 11$ y $T_c = 23.5$ donde el resultado de $k_p = 6.785$, $T_i = 11.75$, $T_d = 1.4375$.

	KP	Ti	Td
P	$0.50 \cdot k_c$		
PI	$0.45 \cdot k_c$	$0.83 \cdot T_c$	
PID	$0.59 \cdot k_c$	$0.50 \cdot T_c$	$0.125 \cdot T_c$

Tabla 14 Método de ziegler nichols ganancia critica, practica 5

Luego se procede a compilar y comprobar la sintonización como se muestra en la figura 167

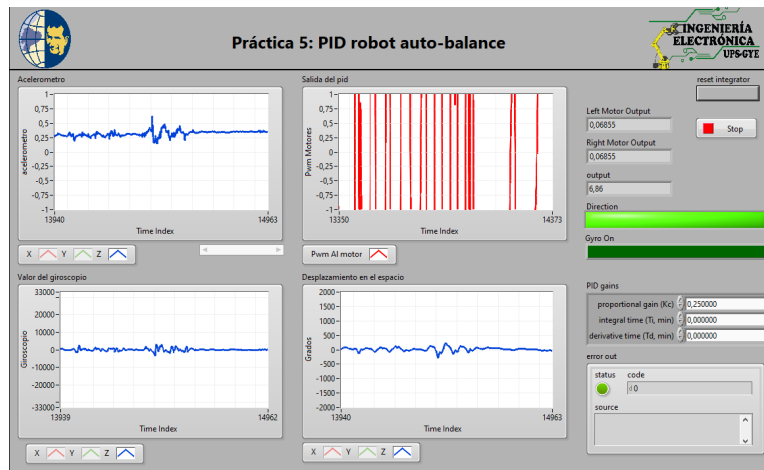


Figura 167. Interfaz practica 5

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Revisar el correcto funcionamiento del sensor al momento de la ejecución
- Comprobar los valores de ganancias e ir aumentando según el criterio para la practicas para obtener óptimos resultados.

Anexo 4.6 Solución de práctica 6

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 168

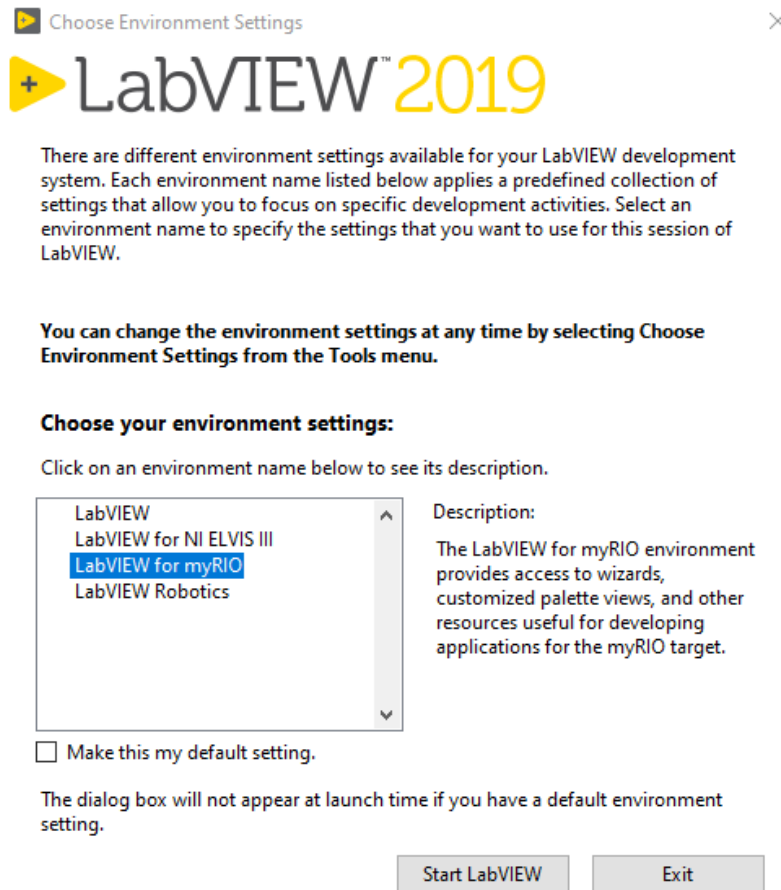


Figura 168. Ejecución de LabView como compatible con ni myRio
práctica 6

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 169.

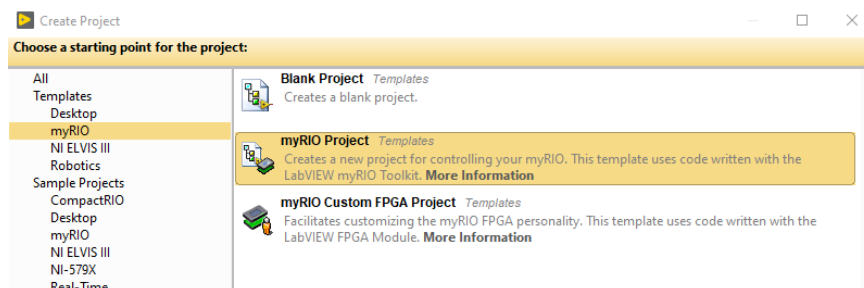


Figura 169. Nuevo Proyecto, práctica 6

Configurar la ruta donde se guarda el archivo (carpeta practica 6), nombre (PID robot auto balanceado evitando obstáculos con sensores ultrasónicos) y el tipo de conexión como USB y dar en finalizar como se muestra en la imagen 170.

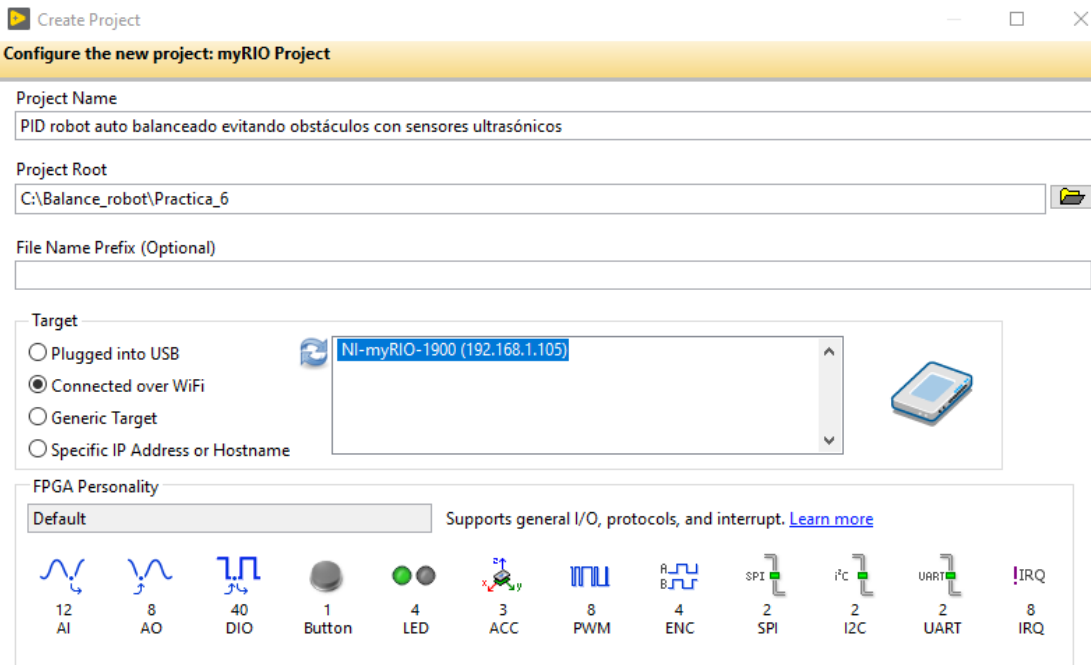


Figura 170. Configuración de ruta, práctica 6

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 171.

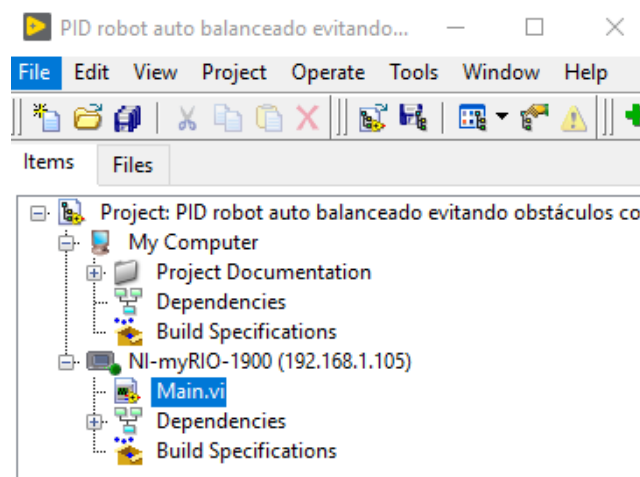


Figura 171. Administrador de proyectos, selección práctica 6.

En el bloque de diagrama se agrega un bloque i2c para ni myrio y se configura como escritura , en el puerto a y su velocidad de 100 kbps como se muestra en la figura 172.

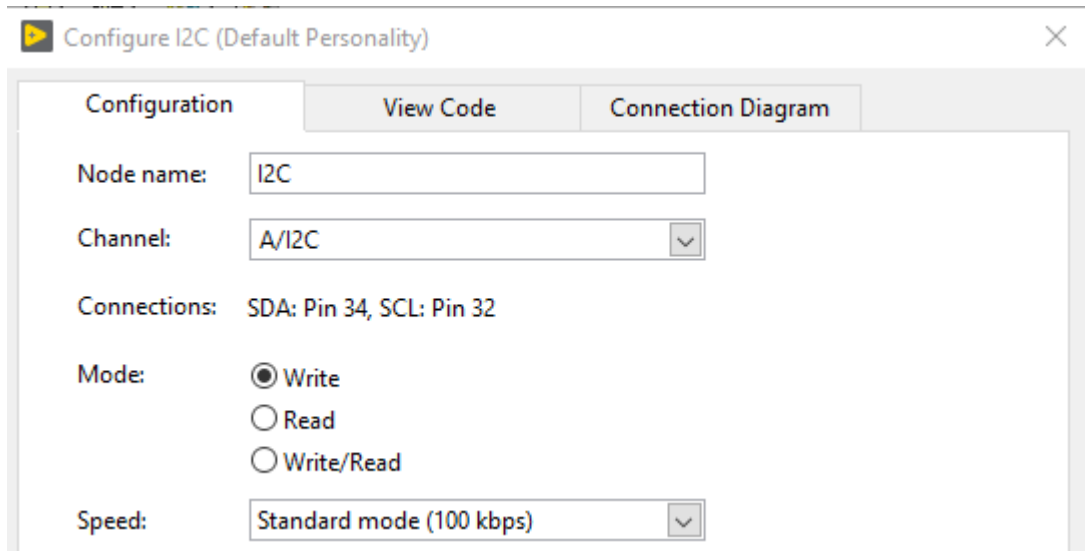


Figura 172. Configuración de i2c para myRio de práctica 6.

En el bloque i2c se crea una variable que contenga la dirección del dispositivo **0x69** y un arreglo donde se configuran los parámetros en formato de arreglo bidimensional para el dispositivo como se muestra en la figura 173.

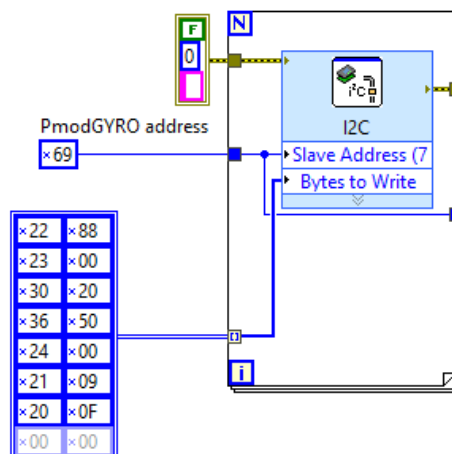


Figura 173. Bloque de variables i2c, práctica 6.

Crear un lazo de repetición (while loop) en el cual se agrega dos datos de entradas digital (bloque de diagrama >>myRio >> digital input) para el puerto A DIO-0 y el A DIO 1 los cuales tienen la función de captar la señal del módulo acelerómetro como se muestra en la imagen 174.

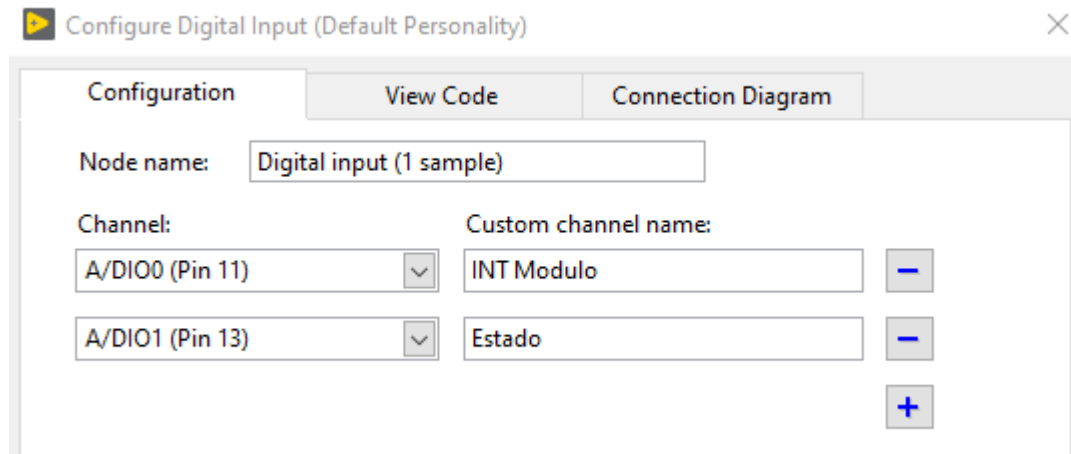


Figura 174. Configuración de entradas digitales, práctica 6.

Se procede a conectar a la salida digital de la interrupción a un lazo de selección de caso para la ejecución cuando se realice una variación en la inclinación del prototipo como se muestra en la figura 175.

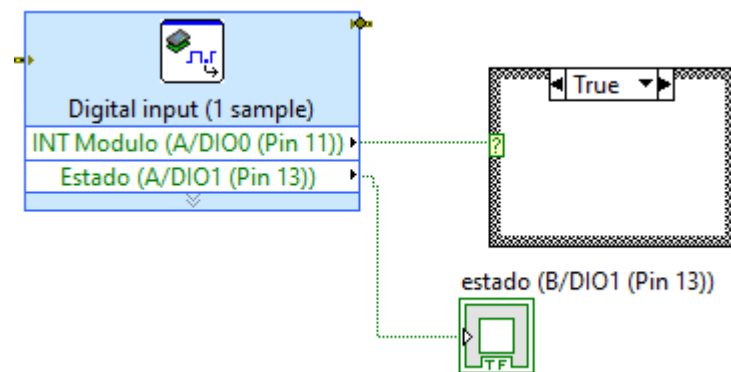


Figura 175. Conexión a lazo de selección, práctica 6.

Agregar un bloque de lectura i2c donde la dirección del dispositivo es 0x69 y el dato para que envíen el dato es 0xA8, con un tamaño de 6 bits en el bloque de diagrama como se muestra en la figura 176.

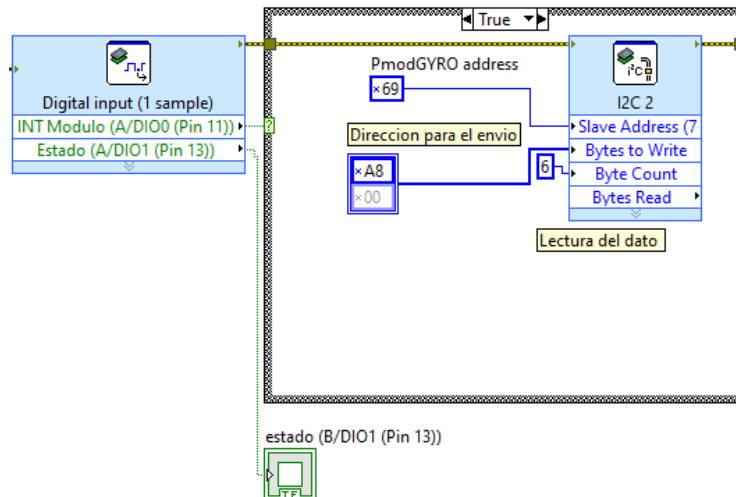


Figura 176. Lazo de selección con bloque i2c, práctica 6.

Posteriormente se procede a separar el dato en 2 bits cada uno y convirtiendo a datos del tipo Word teniendo una dimensión o tamaño máximo de 2^{16} teniendo como resultado los valores en los eje “X”, “Y”, “Z” agregando en un bundle para mostrar los datos en una sola grafica como se muestra en la figura 177

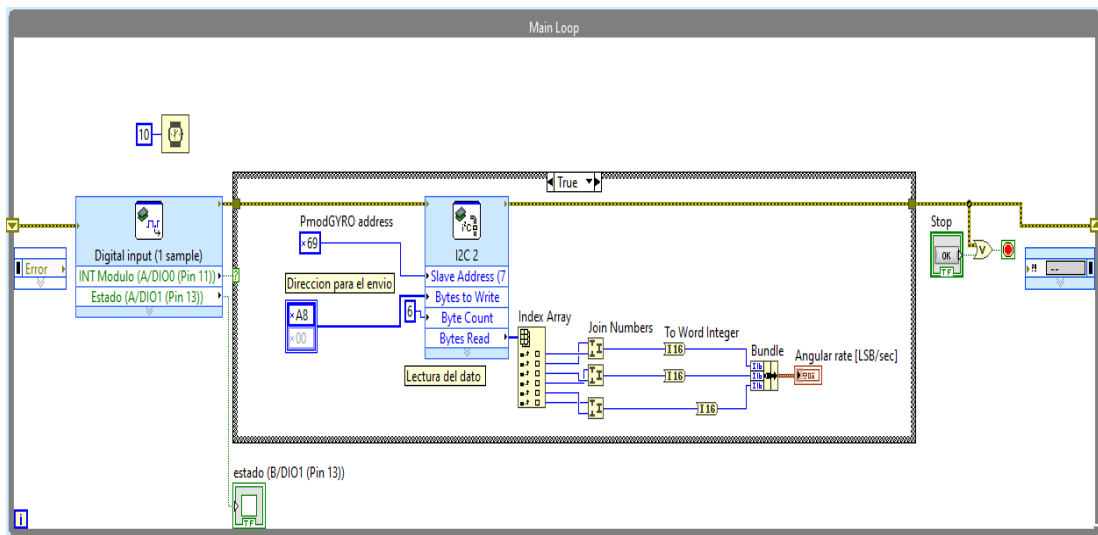


Figura 177. Bundle con datos para gráfica, práctica 6.

Agregar tres datos de lectura analógica (bloque de diagrama >>myRio >> accelerometer), las entradas del acelerómetro se activan los ejes “x”, “y”, “z” como se muestra en la imagen 178.

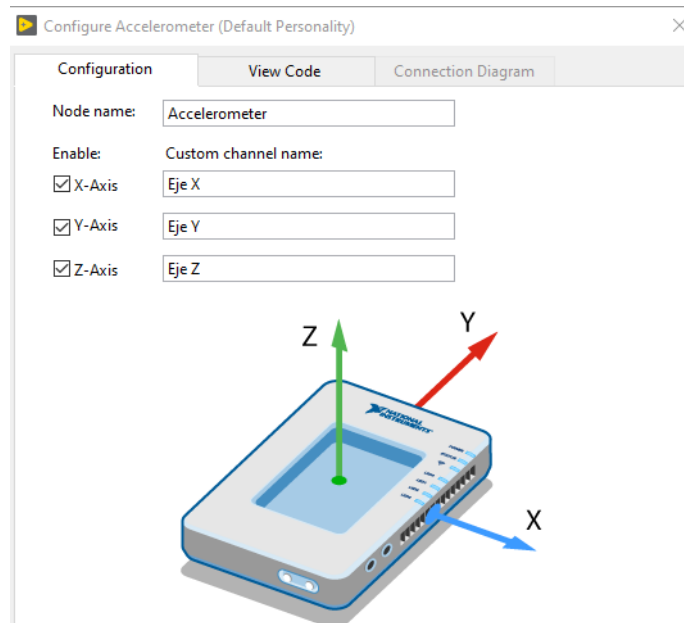


Figura 178. Ingreso de entradas analógicas, práctica 6.

Se procede a conectar las salidas a un indicador gráfico y se crea un indicador numérico para la variable del eje “z” como se muestra en la figura 179

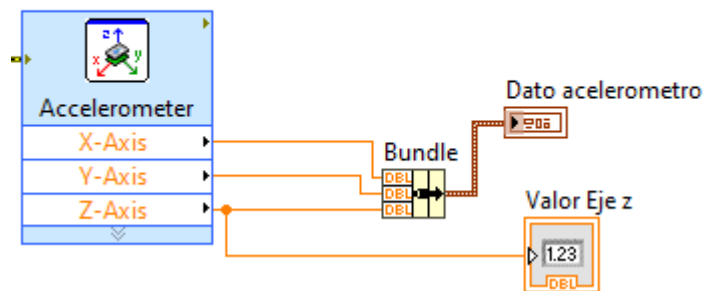


Figura 179. Conexión de salidas analógicas práctica 6.

En las salidas del giroscopio realizar una integral a cada salida para obtener el dato de desplazamiento y en el selector de caso donde cuando no se cumpla el rango de operación de 45 grados (2500) y -45 grados (-2500) el prototipo entre en inactividad mediante la escritura de dos valores análogos en cada motor como se muestra en la figura 180

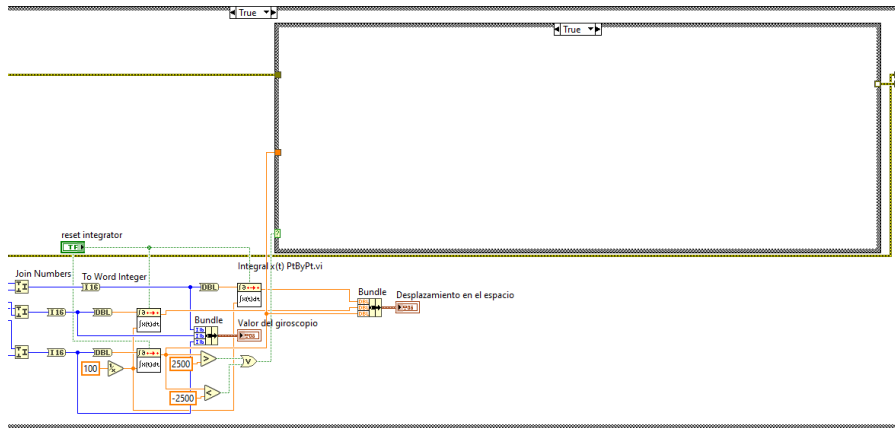


Figura 180. Obtención de datos de desplazamiento práctica 6.

En el selector de casos cuando la condicional es verdadero y se encuentra en un rango de operación no óptimo, agregar dos salidas análogas (bloque de diagrama >> myRio >> Analog Output) para el control de los motores, la primera en el puerto B A01 y la segunda en el puerto B A02 como se muestra la configuración en la figura 181

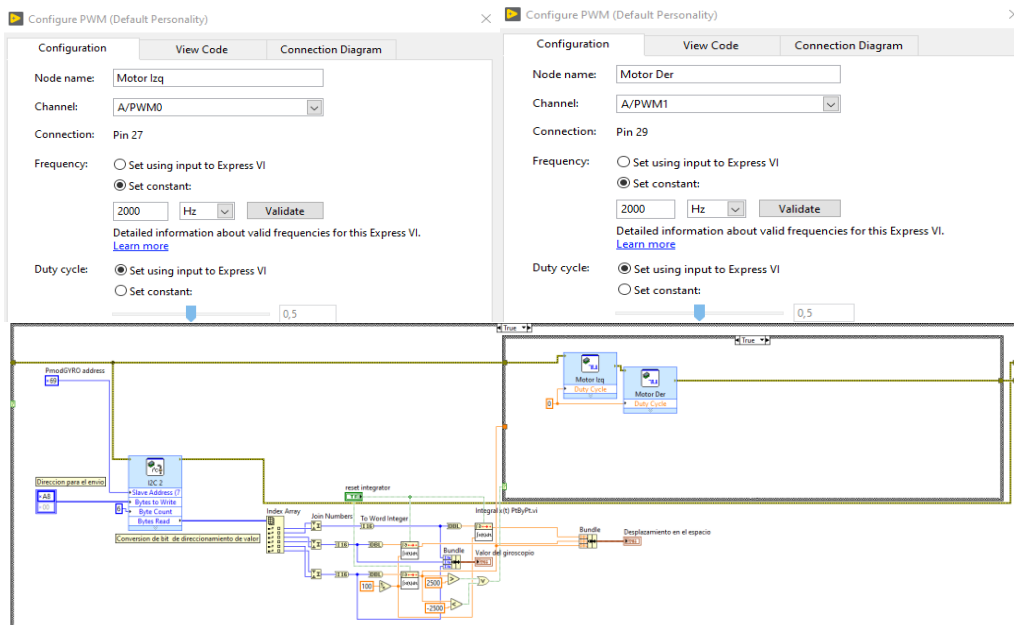


Figura 181. Ingreso de dos salidas analógicas, práctica 6.

En el selector de casos cuando la condicional es falso, agregar un bloque de control pid (bloque de diagrama >> control y simulación >> pid >> pid controlador) donde el rango de la salida es -1 a 1 y la referencia es 0, a la par en salida agregar un bloque para atenuar la señal de salida mediante un controlador de adelanto como se muestra en la figura 182

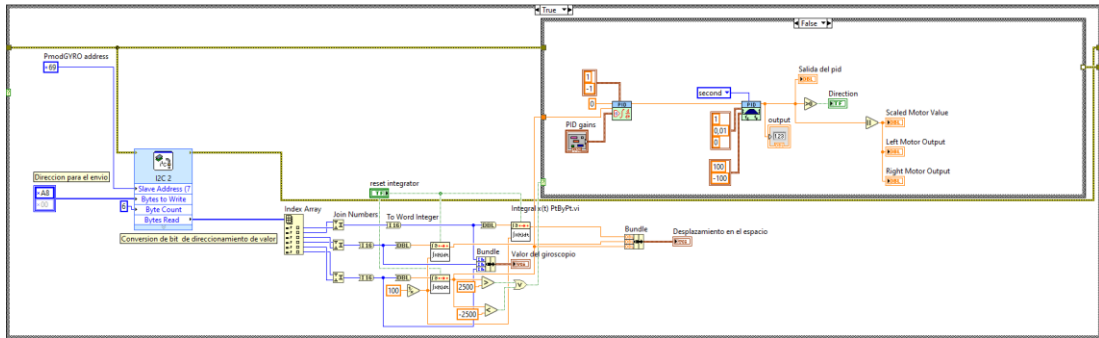


Figura 182. Señal de salida atenuada, practica 6.

Luego a la salida del controlador se agrega las salidas digitales para el control de sentido horario y antihorario en cada motor en el puerto A DIO 1 y puerto A DIO2 como se muestra en la figura 183.

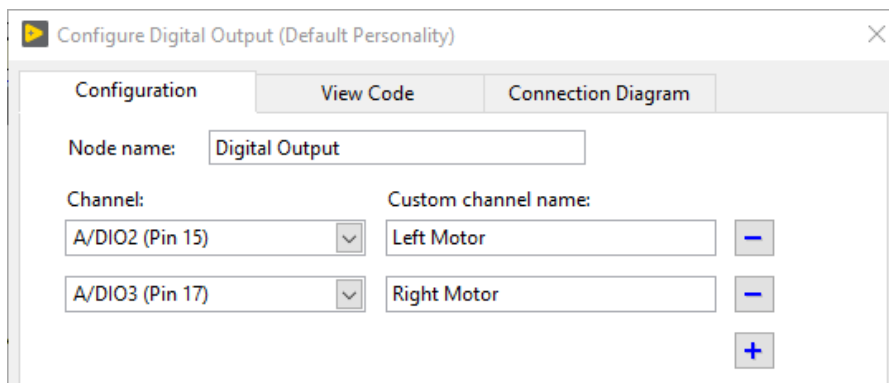


Figura 183. Configuración de salidas digitales, practica 6.

Posteriormente agregar dos bloques de salida analógicas para controlar el voltaje suministrado a los motores como se muestra en la figura 184.

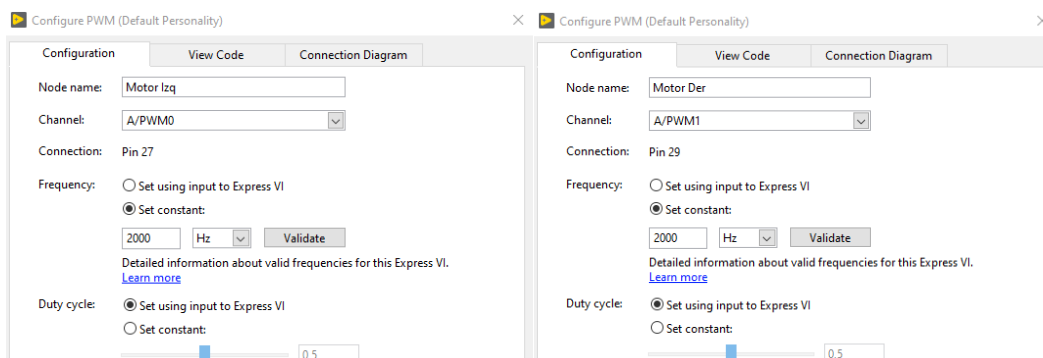


Figura 184. Ingreso de dos salidas analógicas, práctica 6.

En la salida agregar un condicional cuando sea mayor que 0 enviar un dato a al motor 1 y en el segundo motor enviar la misma señal negada y controlara los voltajes mediante el dato de salida como valor absoluto como se muestra en la figura 185

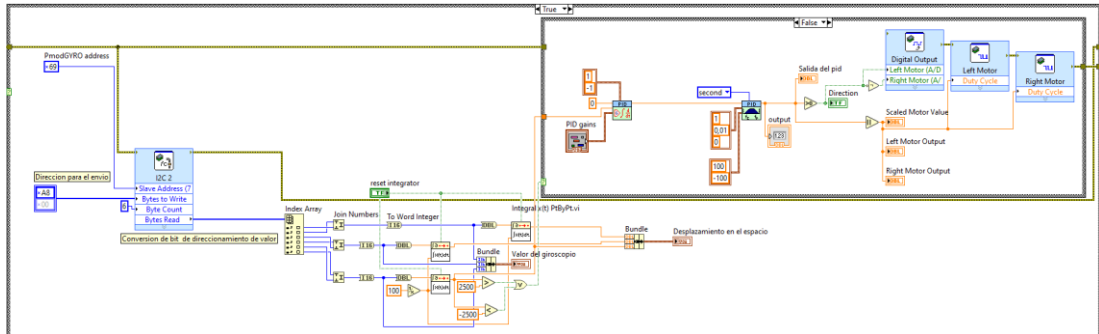


Figura 185. Condiciones de motores, practica 6.

Se inicia el método de sintonizando mediante ziegler nichols por ganancias critica donde se ubica la ganancia derivativa e integral en cero, y se cambia el valor proporcional hasta generar oscilaciones continuas donde se utiliza el valor de Kp en 1 como se muestra en la gráfica 186

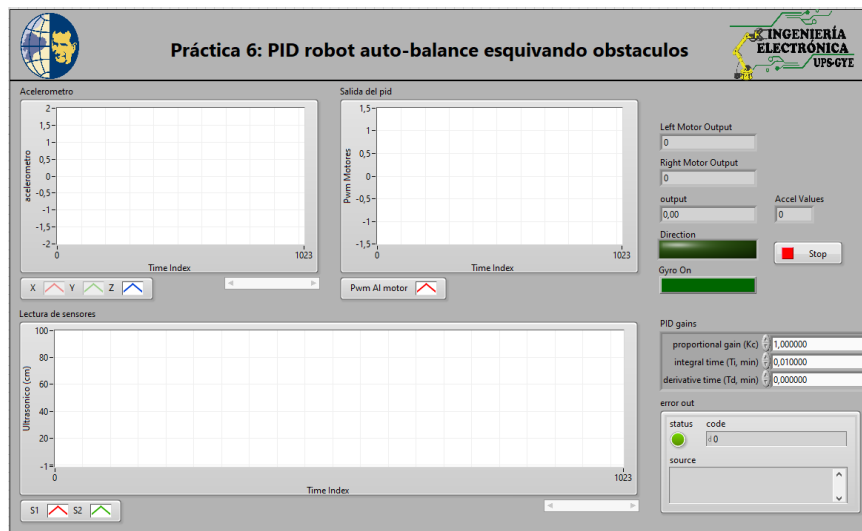


Figura 186. Sincronización de mediante ziegler nichols por ganancias critica, practica 6.

En la figura se utiliza como ganancia kp el valor de 11 donde se aprecia las oscilaciones periódicas y se toma como kc el valor de kp como se aprecia en la figura 187.

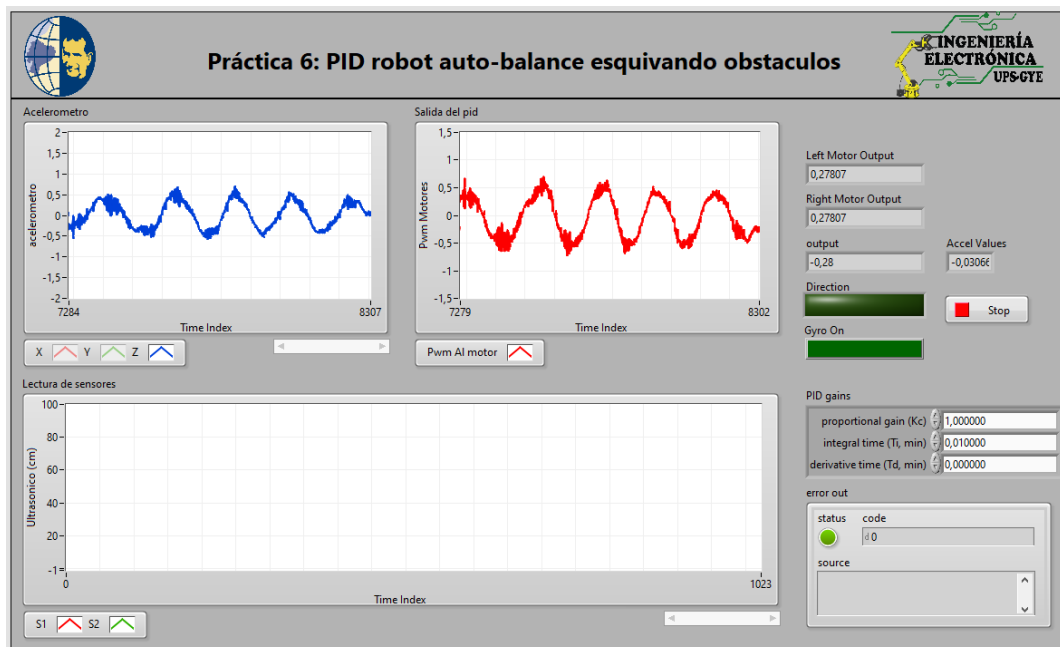


Figura 187. Gráficas de ganancia Kp, practica 6.

Se procede a tomar los valores de repetición cíclica y se encuentra que el valor de t_c es 23 como se muestra en la figura 188.

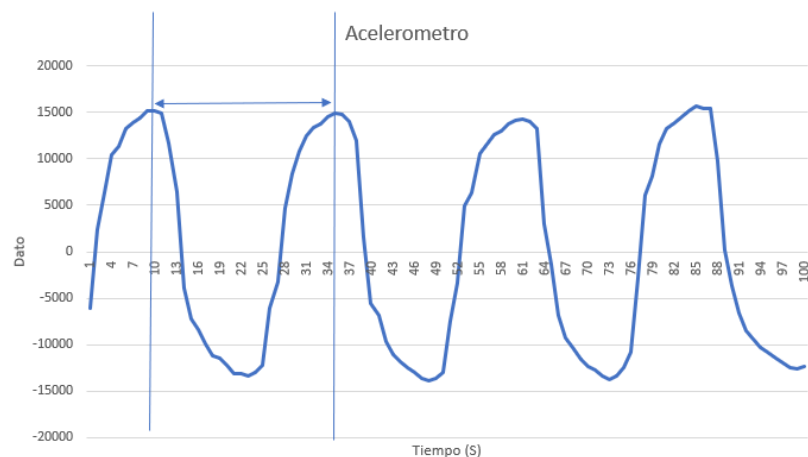


Figura 188. Valor de repetición de T_c , practica 6.

Se procede a utilizar la tabla 15 con el valor de $k_c = 11$ y $T_c = 23.5$ donde el resultado de $k_p = 6.785$, $T_i = 11.75$, $T_d = 1.4375$.

	KP	Ti	Td
P	$0.50 \cdot k_c$		
PI	$0.45 \cdot k_c$	$0.83 \cdot T_c$	
PID	$0.59 \cdot k_c$	$0.50 \cdot T_c$	$0.125 \cdot T_c$

Tabla 15 Método de ziegler nichols ganancia critica, practica 6

Se procede a agregar 2 entradas analógicas en el puerto C Ai0 para el sensor frontal y en el puerto C Ai1 para el sensor posterior

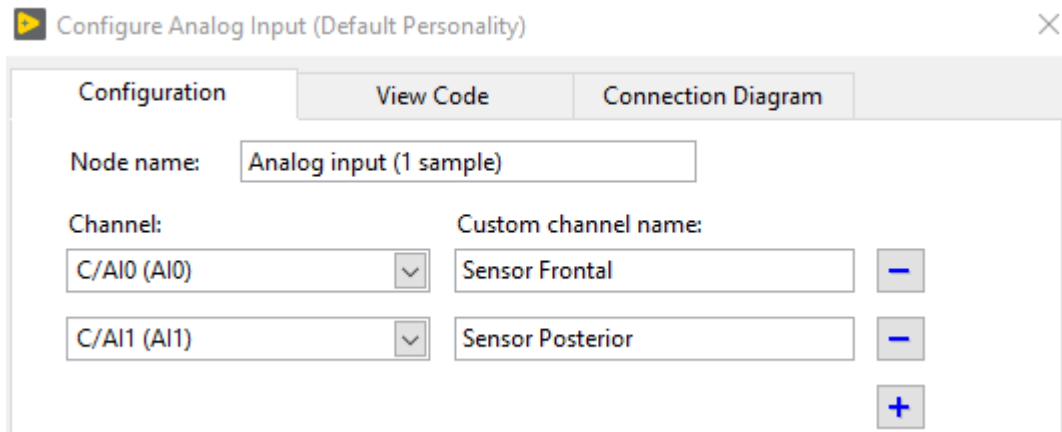


Figura 189. Configuración de entrada analógica de sensores, practica 6.

Posteriormente se agrega las ganancias a la salida de los sensores para la conversión a centímetros

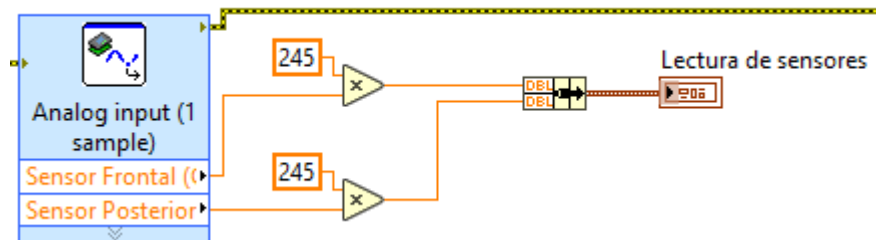


Figura 190. Aplicación de ganancia para los sensores, practica 6.



Figura 191. Ejecución de aplicación con lectura de sensores, practica 6.

Agregar una comparación , “Si el valor de sensor 1 o 2 es menor a 20”, envía un valor booleano para el giro del robot como se muestra en la figura 192.

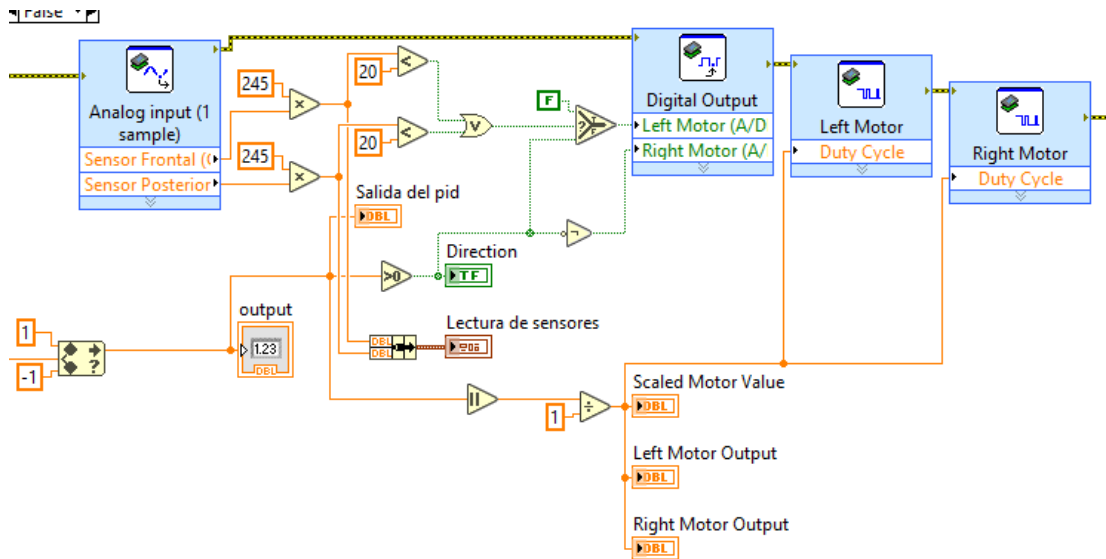


Figura 192. Comparación para el giro evitar obstáculo, practica 6.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Revisar el correcto funcionamiento del sensor al momento de la ejecución
- Comprobar los valores de ganancias e ir aumentando según el criterio para la practicas para obtener óptimos resultados.
- Verificar la correcta lectura de los sensores y condiciones del sensor.

Anexo 4.7 Solución de práctica 7

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 193

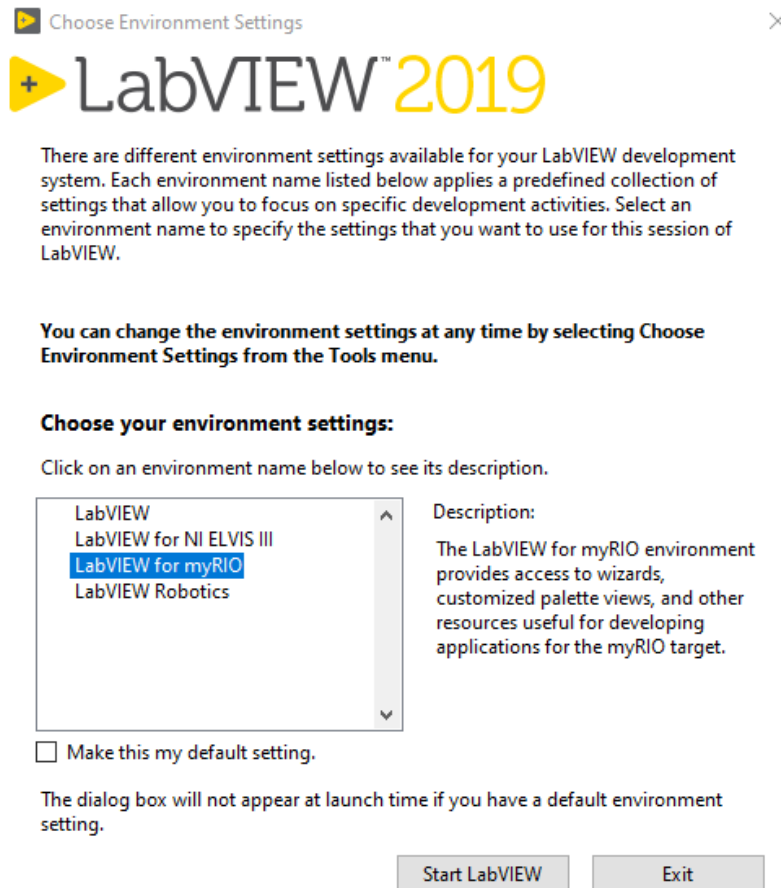


Figura 193. Ejecución de LabView como compatible con ni myRio práctica 7

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 194

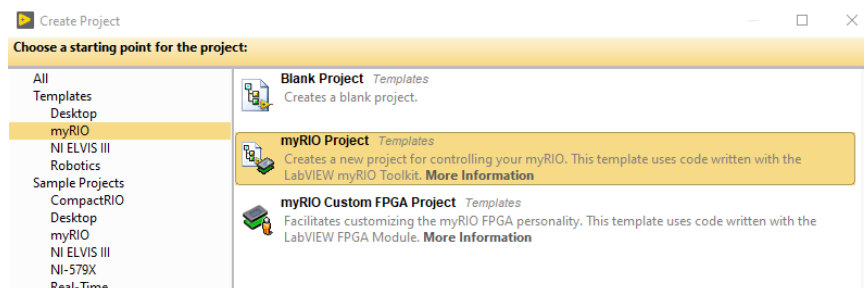


Figura 194. Nuevo Proyecto, práctica 7

Configurar la ruta donde se guarda el archivo (carpeta practica 7), nombre (PID difuso robot auto) y el tipo de conexión como wifi y dar en finalizar como se muestra en la imagen 195.

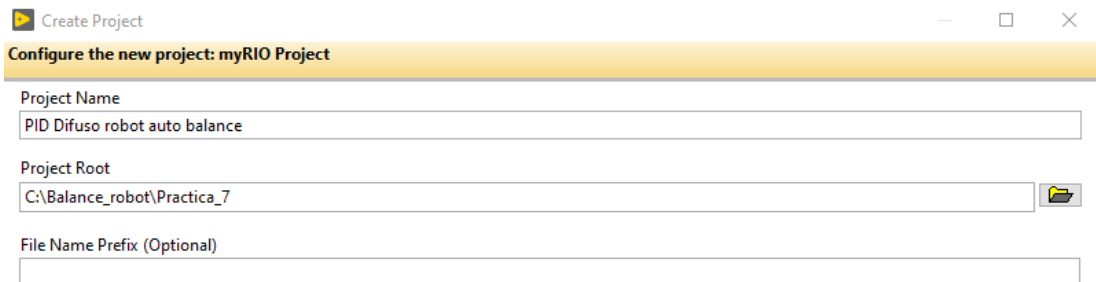


Figura 195. Configuración de ruta, práctica 7

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 196.

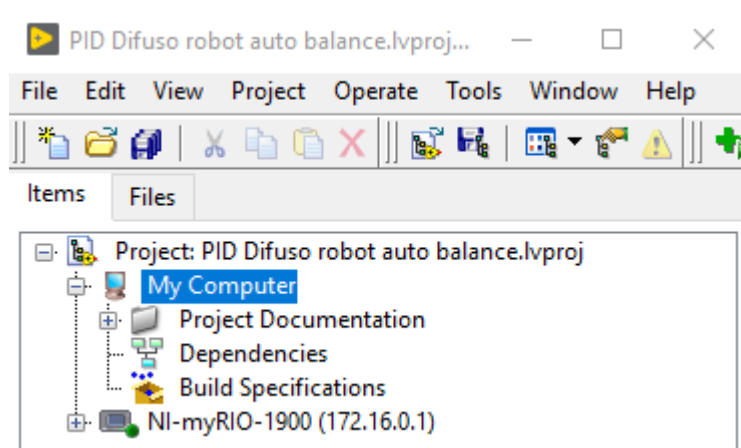


Figura 196. Administrador de proyectos, selección práctica 7.

Ejecutar la herramienta para creación de reglas difusas como se muestra en la figura 197.

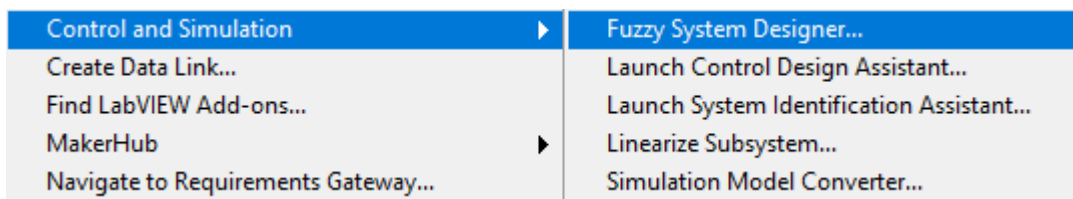


Figura 197. Herramienta para creación de reglas difusas práctica 7.

Se procede a la creación de las variables lingüísticas de entrada para los valores de proporcional, integral y derivativo y la señal de salida del control.

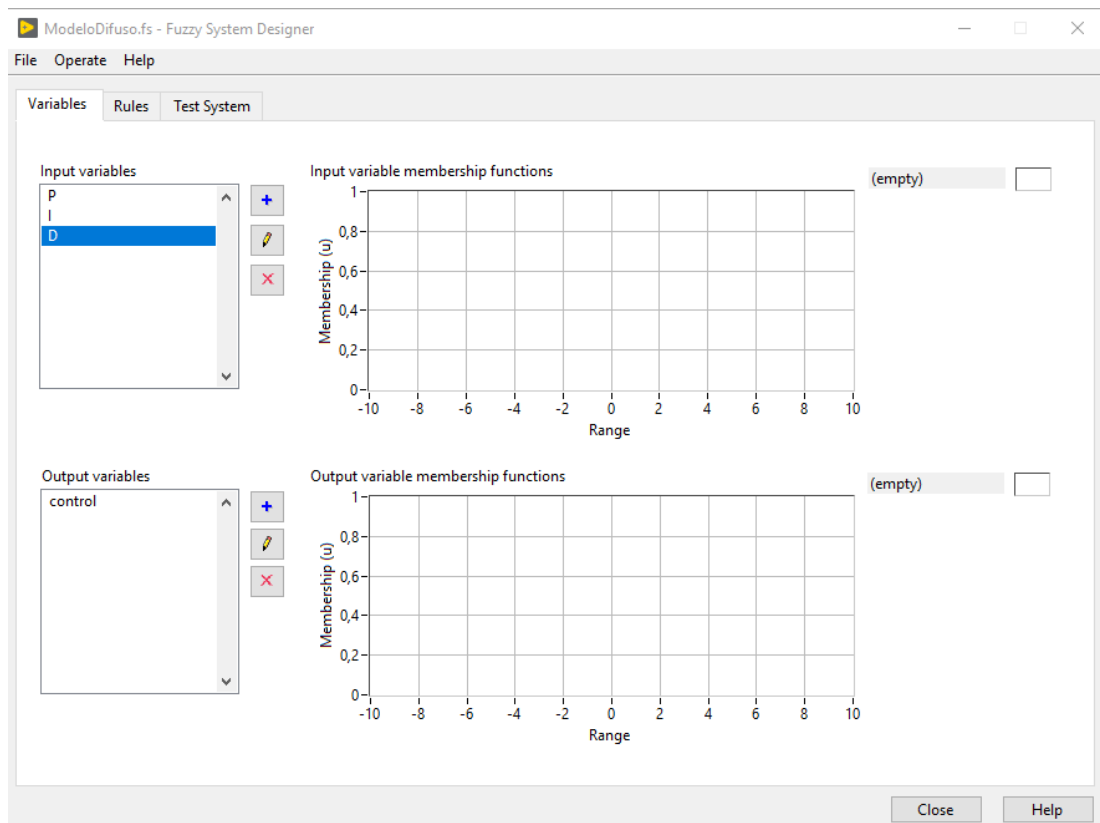


Figura 198. Herramienta para creación de variables, práctica 7.

En la variable proporcional (p) se procede a agregar las 3 reglas de correspondencia indicando cuando es cero, negativo y positivo

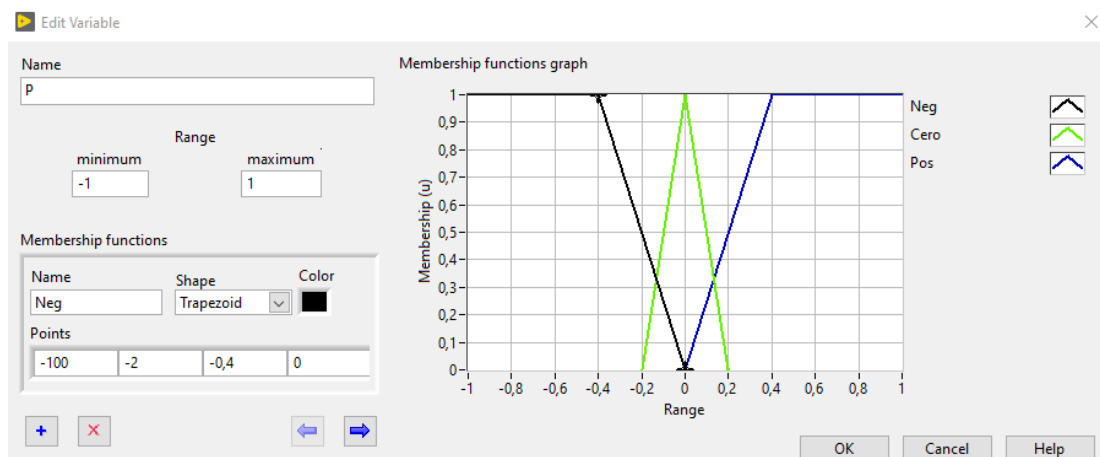


Figura 199. Configuración de rango en variable proporcional, práctica 7.

En la variable integral (i) se procede a agregar las 3 reglas de correspondencia indicando cuando es cero, negativo y positivo

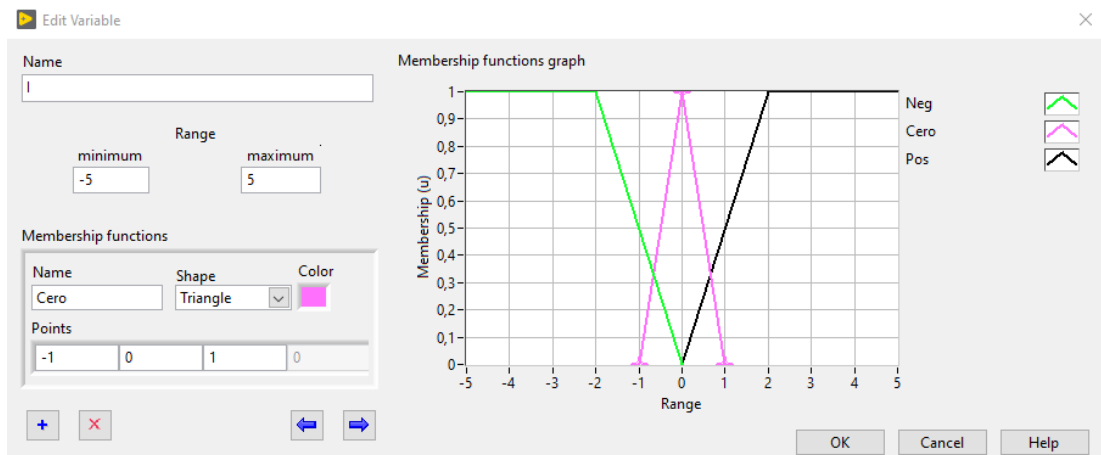


Figura 200. Configuración de rango en variable integral, práctica 7.

En la variable derivativa (d) se procede a agregar las 3 reglas de correspondencia indicando cuando es cero, negativo y positivo

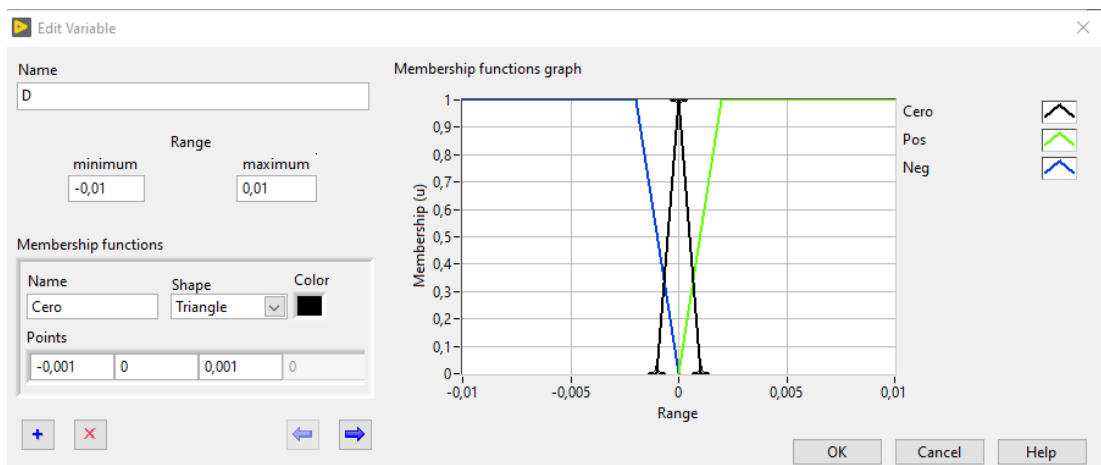


Figura 201. Configuración de rango en variable derivativa, práctica 7.

En la variable para la salida se configuran 5 se procede a agregar las 7 reglas de correspondencia indicando cuando es cero, balance poco (+) , balance poco (-), balance mucho (+), balance mucho (-). Balance largo (+). Balance largo (-) como se muestra en la figura para el control de los motores y sus respectivos cambios de sentidos el rango de operación es -1 a 1 para los actuadores.

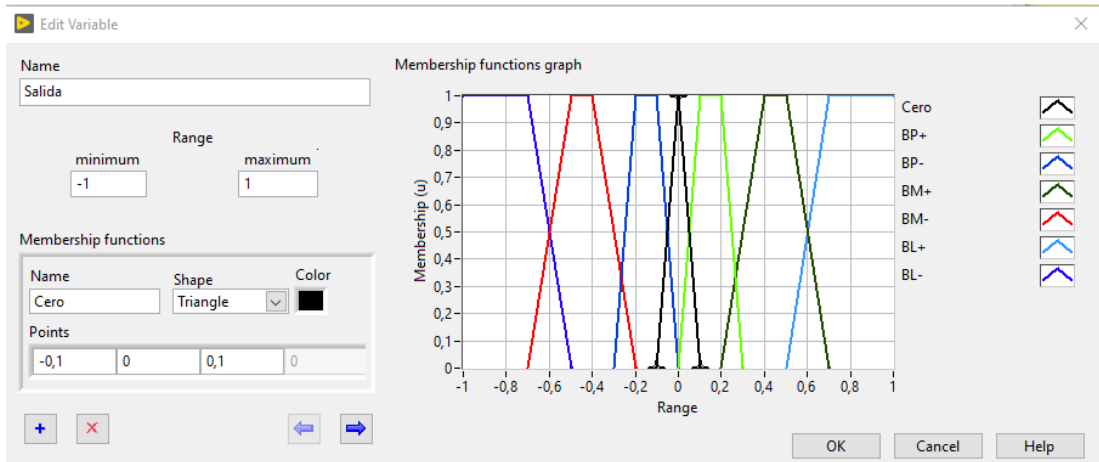


Figura 202. Configuración de rango en variable derivativa, práctica 7.

Crear el conjunto de reglas difusas con las variables lingüísticas como se muestra en la imagen.

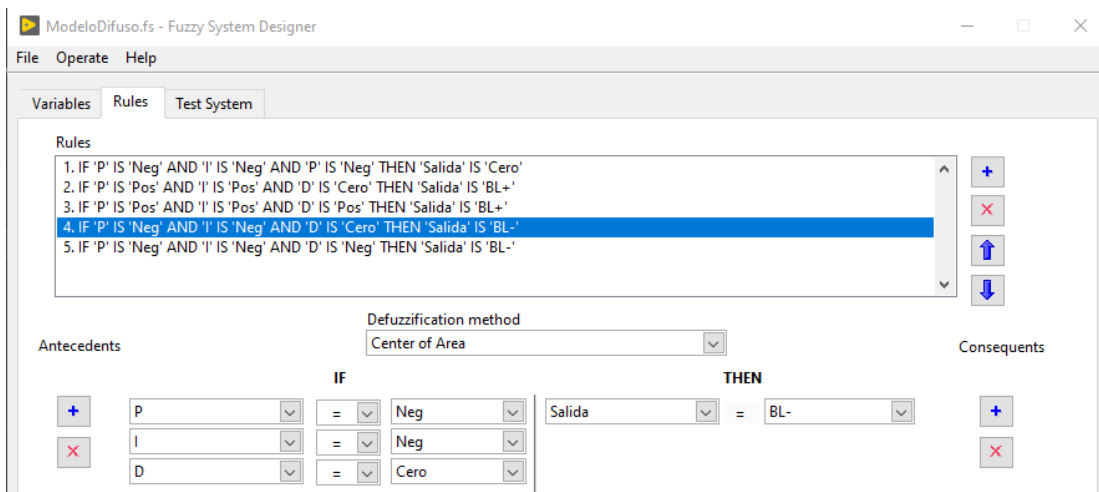


Figura 203. Configuración de entradas digitales, práctica 7.

Se procede a crear un vi donde se realiza el computo del controlador difuso en el bloque de diagrama agregar un bloque de control y simulación como se muestra en la imagen 204.

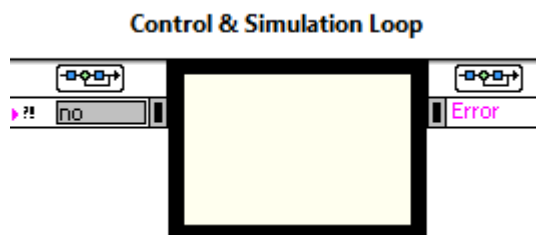


Figura 204. Bloque de control y simulación, práctica 7.

Agregar el bloque de control difuso y el lazo de control pid como se muestra en la imagen, como entrada agregar una variable global llamada sensor y una variable llamada salida para comunicarse con el proceso en él ni myRio

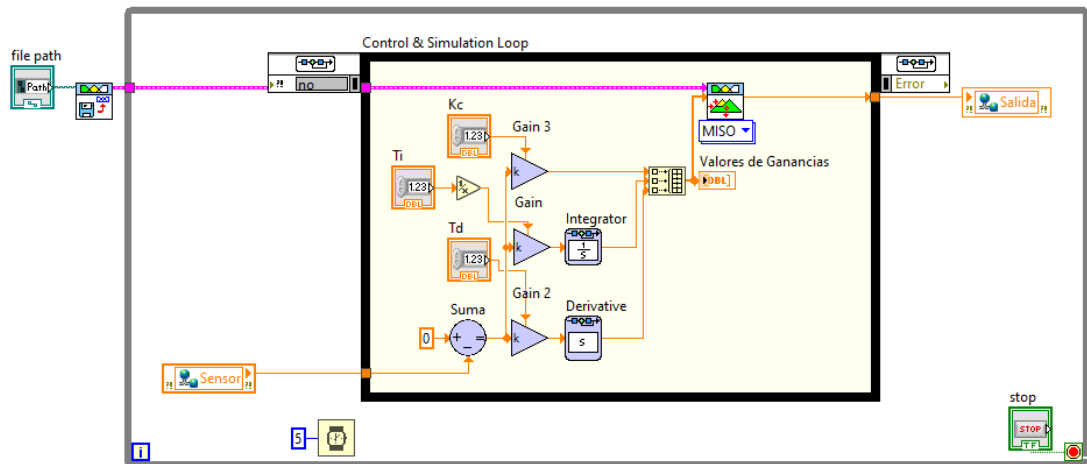


Figura 205. Computo del pid difuso en proceso paralelo, práctica 7.

Agregar tres datos de lectura analógica (bloque de diagrama >> myRio >> acelerometer), las entradas del acelerómetro se activan los ejes “x”, “y”, “z” como se muestra en la imagen.

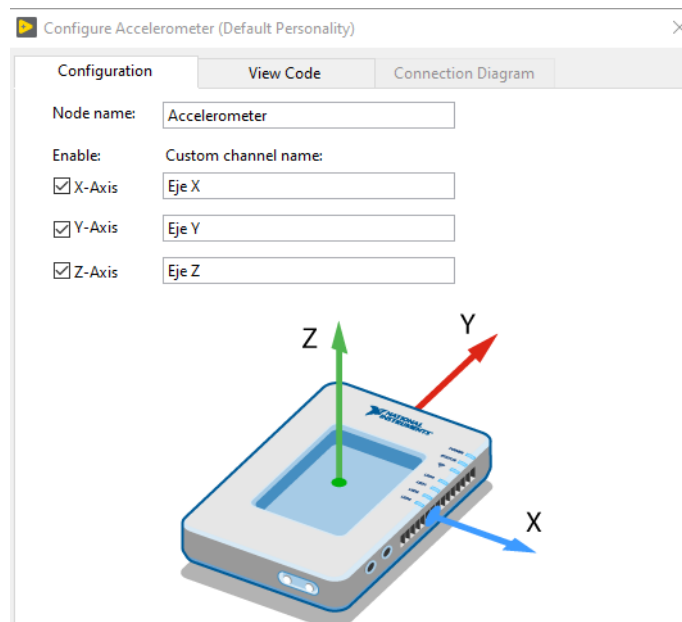


Figura 206. Ingreso de entradas analógicas, práctica 7.

Se procede a conectar las salidas a un indicador gráfico y se crea un indicador numérico para la variable del eje “y” como se muestra en la figura 207.

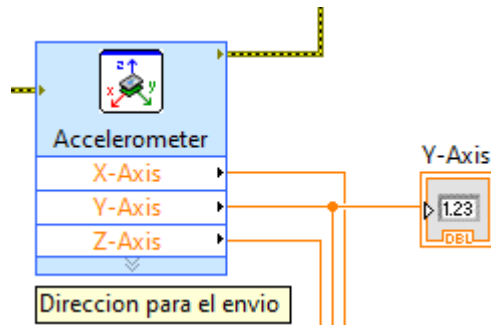


Figura 207. Conexión de salidas analógicas práctica 7.

En la salida del eje y se agrega un filtro para la eliminación de ruido mediante un pasa banda como se muestra en la figura.

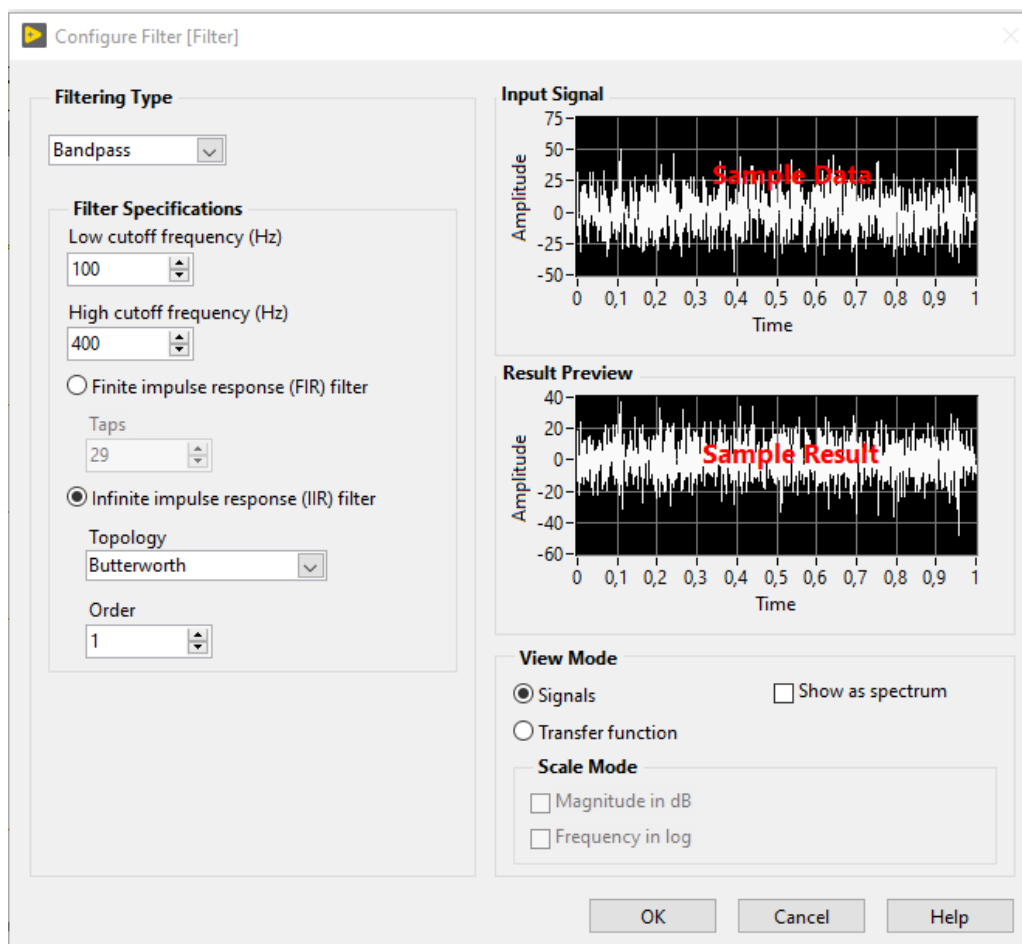


Figura 208. Configuración de filtro, práctica 7.

Se procede a ubicar los límites (1 , -1) y asignamos a la variable global para enviar los valores al vi de cómputo como se muestra en la figura 209.

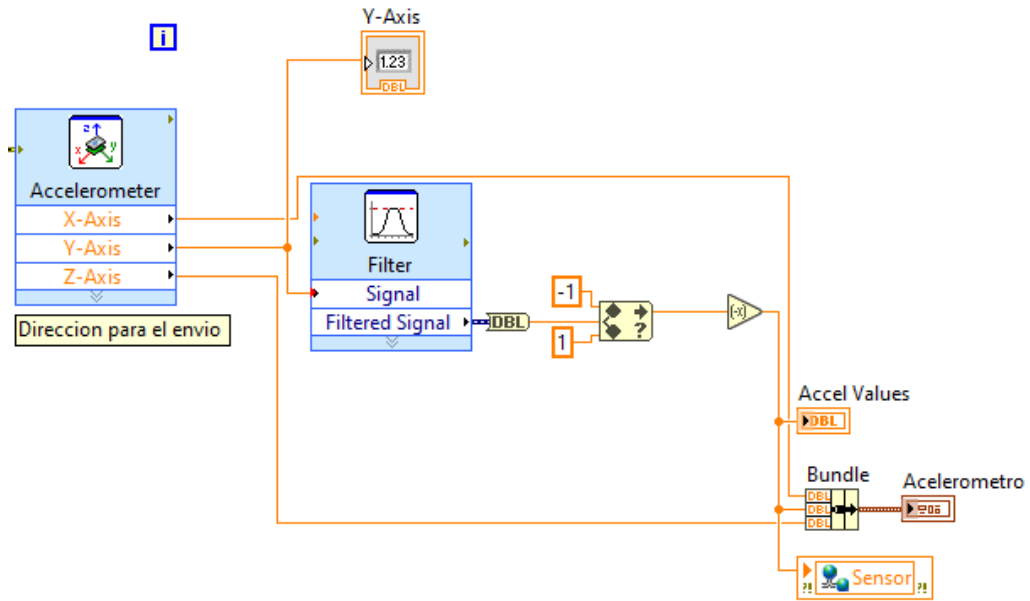


Figura 209. Filtro al eje “Y” y asignación en variable global, práctica 7.

Posteriormente se agrega un condicional donde si el valor absoluto del sensor es mayor a 0.7 , el accionar de los actuadores se detendrá mediante dos salidas analógicas (bloque de diagrama >> myRio >> Analog Output) para el control de los motores, la primera en el puerto B A01 y la segunda en el puerto B A02 como se muestra la configuración en la figura 210.

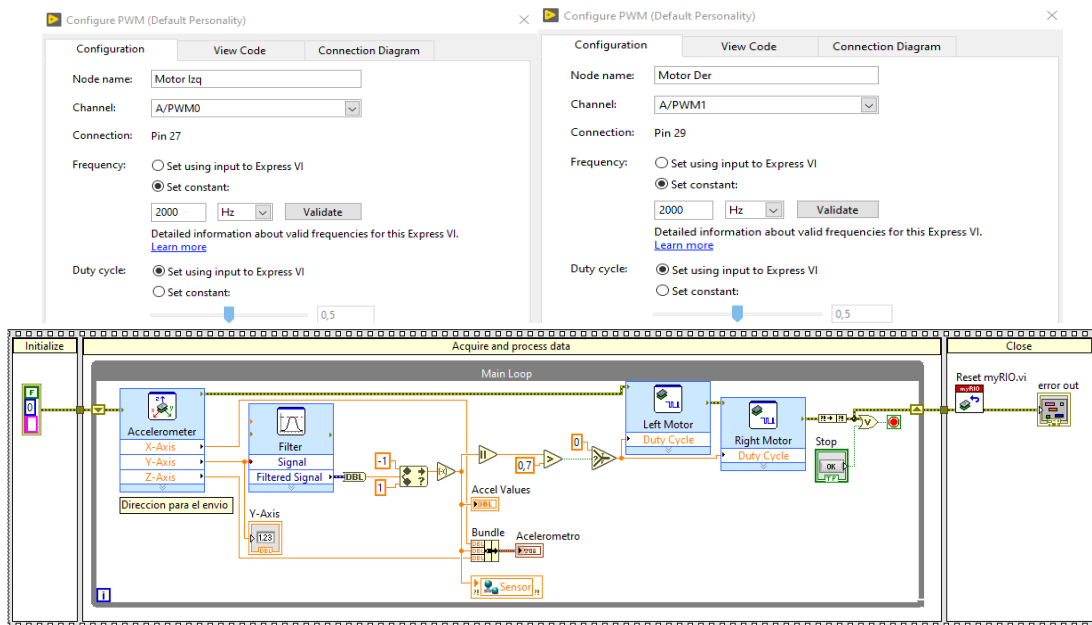


Figura 210. Ingreso de dos salidas analógicas, práctica 7.

Luego a la salida del controlador se agrega las salidas digitales para el control de sentido horario y antihorario en cada motor en el puerto A DIO 1 y puerto A DIO2 como se muestra en la figura 211.

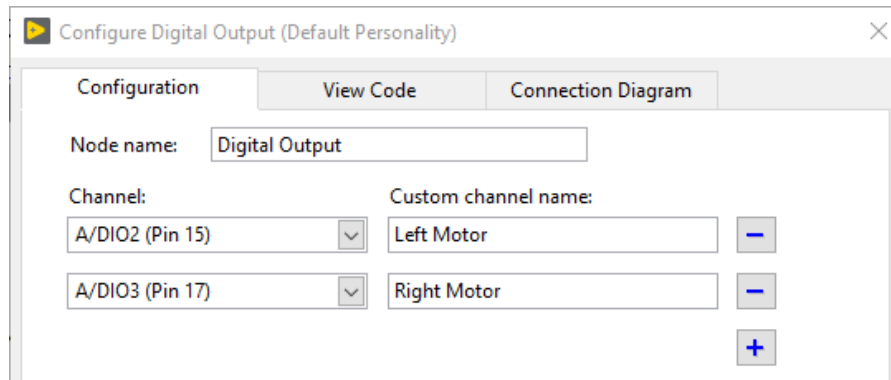


Figura 211. Configuración de salidas digitales, practica 7.

Crear una variable global llamada salida , permitiendo la interacción con el vi de computar y conectar a las entradas de los motores y configurar un condicional (cuando el valor sea mayor a 0 enviar el giro horario caso contrario antihorario) como se muestra en la figura 212

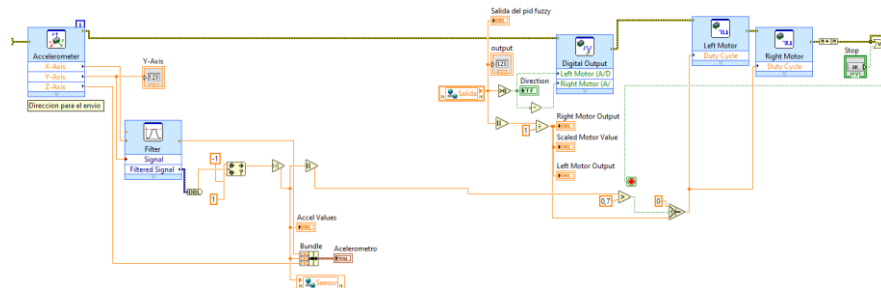


Figura 212. Creación de variable global y conexión con motores, practica 7.

Se procede a compilar el programa principal y el programa de cómputo donde se realiza el pid fuzzy como se muestra en la figura la ejecución del programa y funcionamiento del controlador.

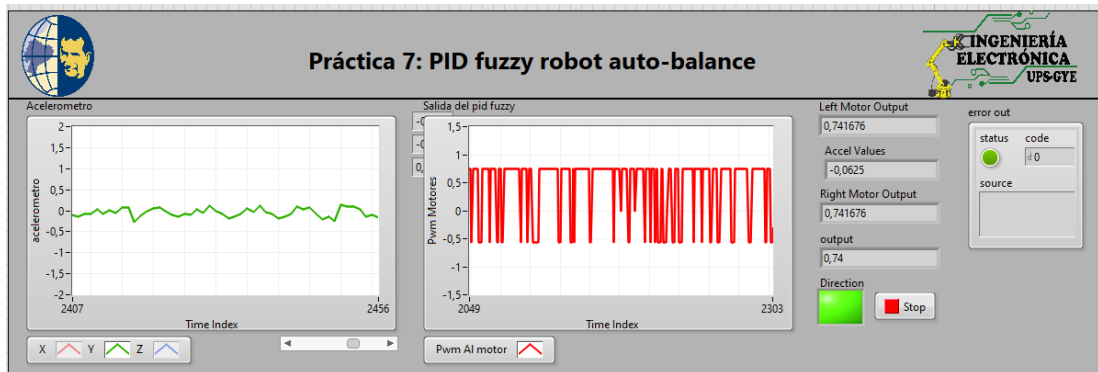


Figura 213. Pid Fuzzy en robot auto balance, practica 7.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Revisar el correcto funcionamiento del sensor al momento de la ejecución
- Comprobar la dirección del archivo de reglas difusas.
- Verificar la correcta lectura de los sensores y condiciones del sensor.

Anexo 4.8 Solución de práctica 8

Se procede a configurar el computador con el dispositivo bluetooth hc 05 como esclavo(modo por fabrica),seleccionar la opción del sistema en el menú de inicio (**Bluetooth y otras opciones de configuración de dispositivos**) como se muestra en la imagen 214

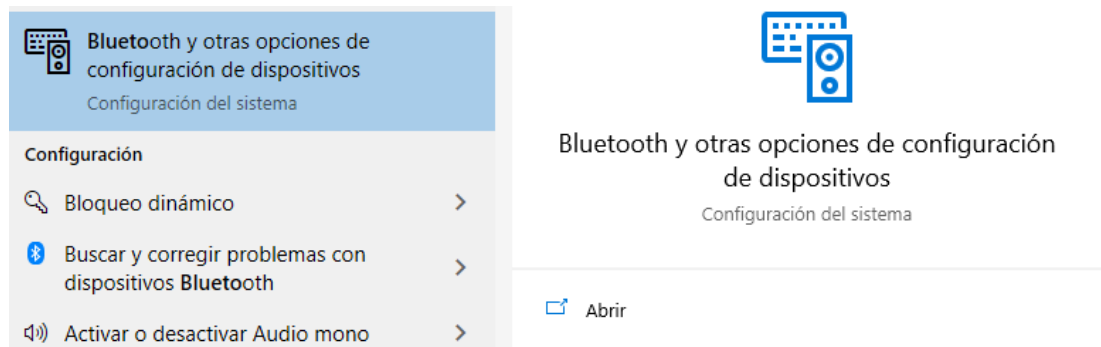


Figura 214. Ejecución de LabView como compatible con ni myRio
práctica 8

Seleccionar la opción de agregar nuevo dispositivo bluetooth como se muestra en la imagen 215

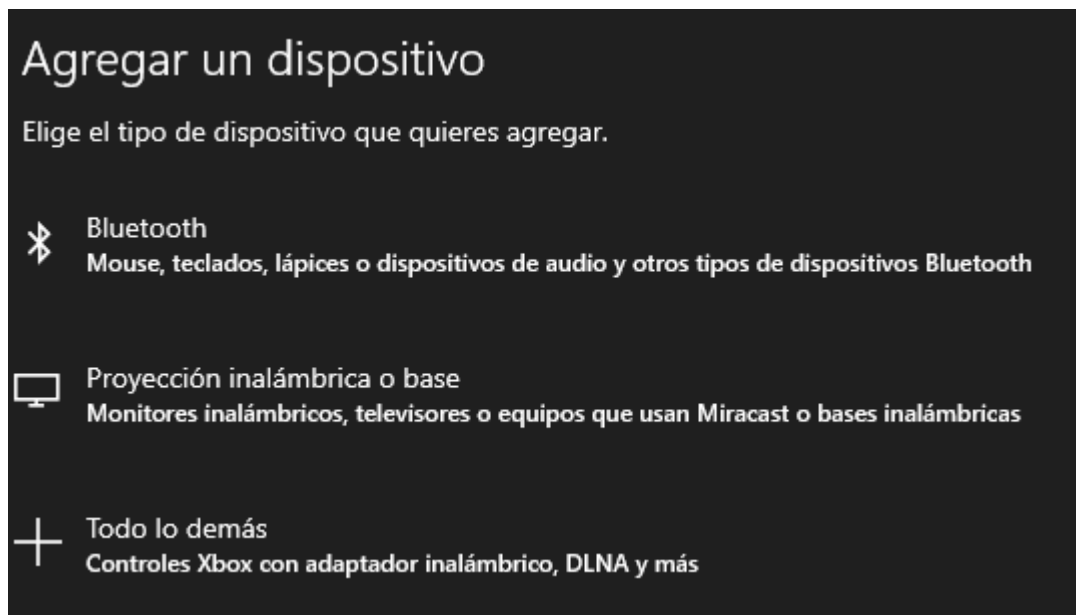


Figura 215. Ejecución de LabView como compatible con ni myRio
práctica 8

Se procede a agregar la clave de fabrica del hc05 (**1234** o **0000**) como se muestra en la figura 216

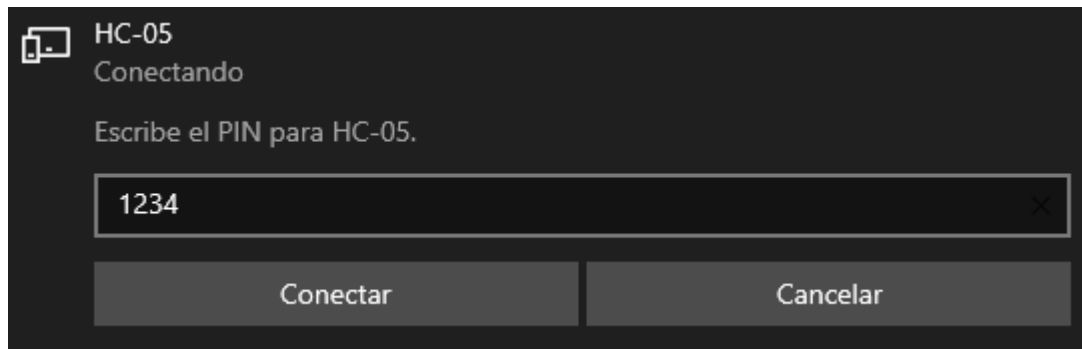


Figura 216. Ejecución de LabView como compatible con ni myRio

Posteriormente al enlazar el computador se verifica en que puerto de comunicación esta seteado mediante la configuración del bluetooth y puertos COM como se muestra en la figura 217

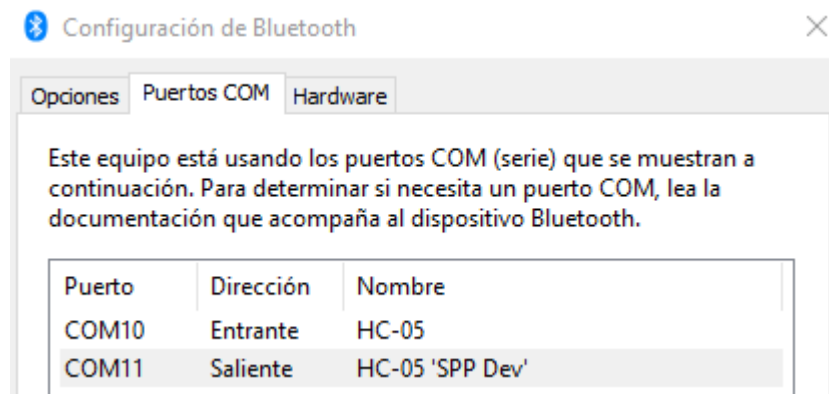


Figura 217. Ejecución de LabView como compatible con ni myRio

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 218

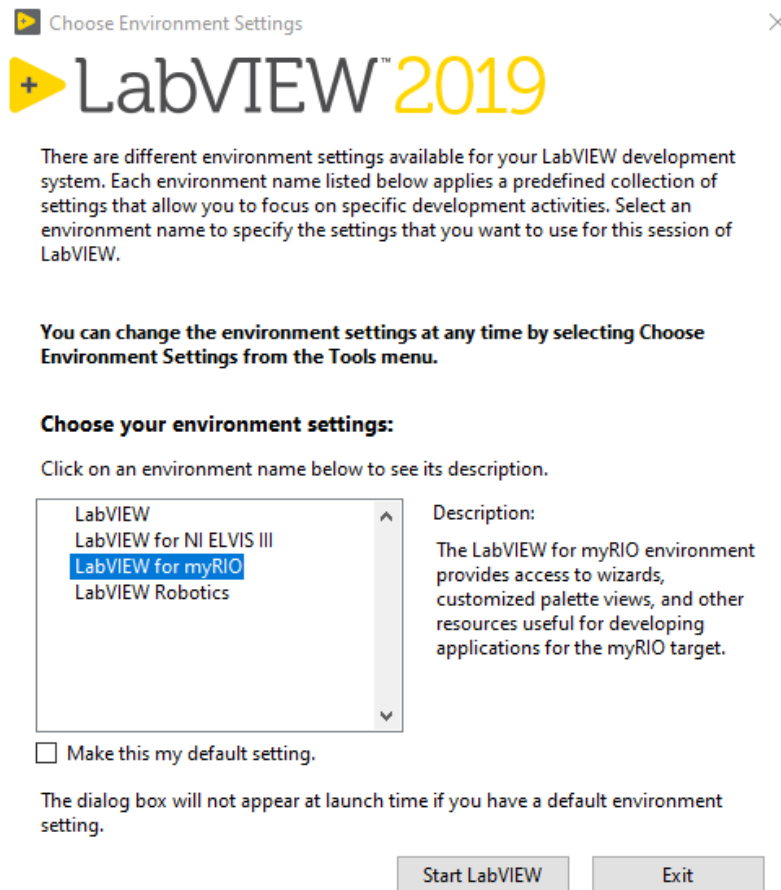


Figura 218. Ejecución de LabView como compatible con ni myRio
práctica 8

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 219.

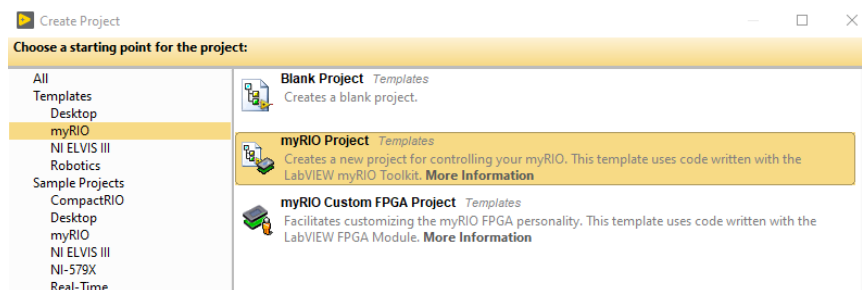


Figura 219. Nuevo Proyecto, práctica 8

Configurar la ruta donde se guarda el archivo (carpeta practica 8), nombre (Robot auto balanceado radio controlado mediante comunicación bluetooth) y el tipo de conexión como wifi y dar en finalizar como se muestra en la imagen 220.

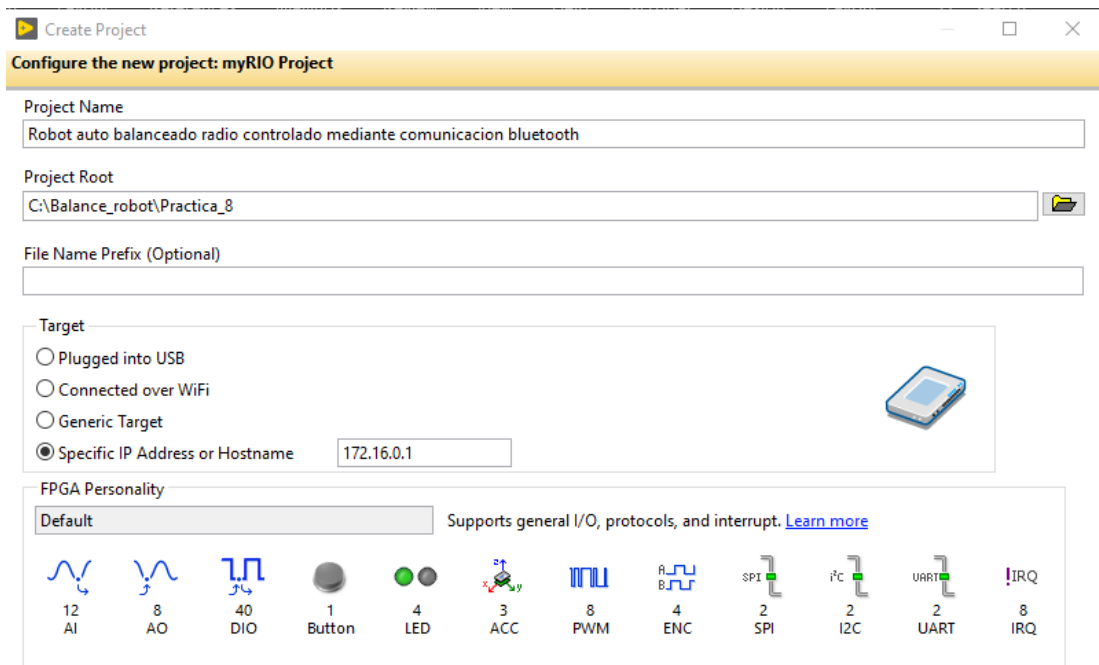


Figura 220. Configuración de ruta, práctica 8

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 221.

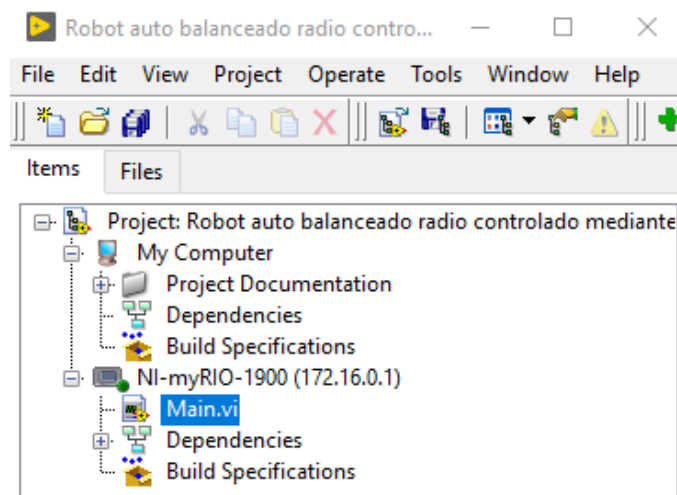


Figura 221. Administrador de proyectos, selección práctica 8.

Agregar tres datos de lectura analógica (bloque de diagrama >> myRio >> accelerometer), las entradas del acelerómetro se activan los ejes “x”, “y”, “z” como se muestra en la imagen 222.

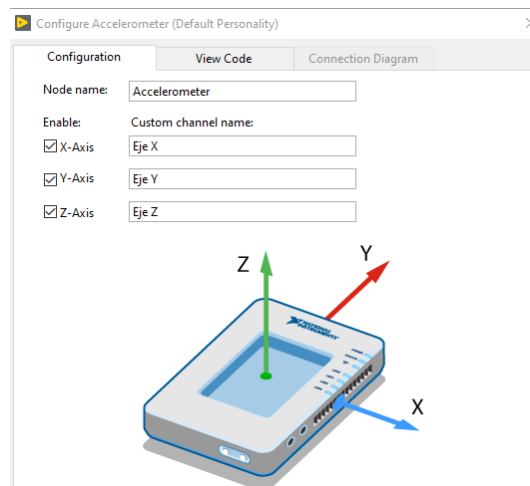


Figura 222. Ingreso de entradas analógicas, práctica 8.

Se procede a conectar las salidas a un indicador gráfico y se crea un indicador numérico para la variable del eje “y” como se muestra en la figura 223



Figura 223. Conexión de salidas analógicas práctica 8.

En la salida del eje y se agrega un filtro para la eliminación de ruido mediante un pasa banda como se muestra en la figura 224.

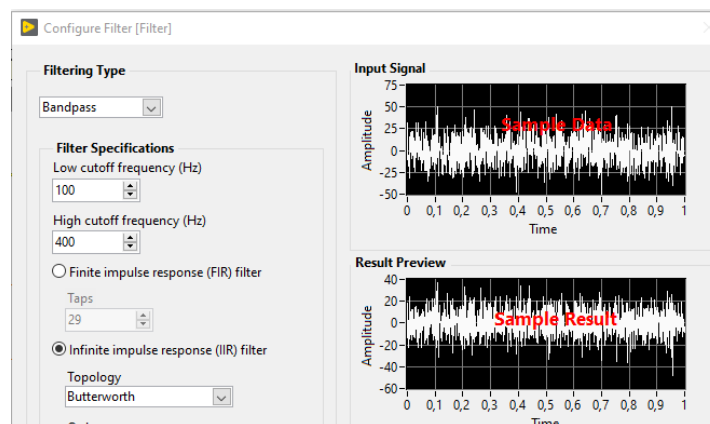


Figura 224. Configuración de filtro, práctica 8.

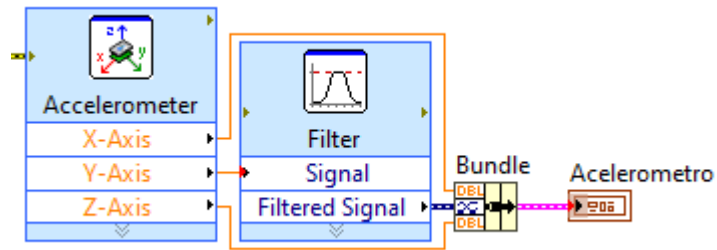


Figura 225. Filtro al eje "Y", práctica 8.

Agregar el bloque de control y simulación para el pid como se muestra en la figura de tal manera que la entrada de la variable del pid sea el valor del eje y la salida se asigne a una variable.

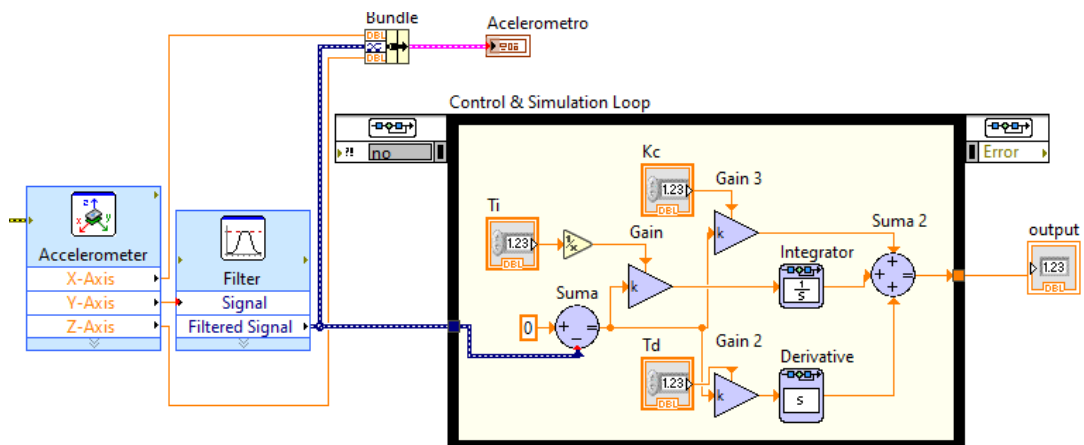


Figura 226. bloque de control y simulación para el pid, práctica 8.

En la salida del controlador se agrega un filtro para la eliminación de ruido mediante un pasa banda como se muestra en la figura 227.

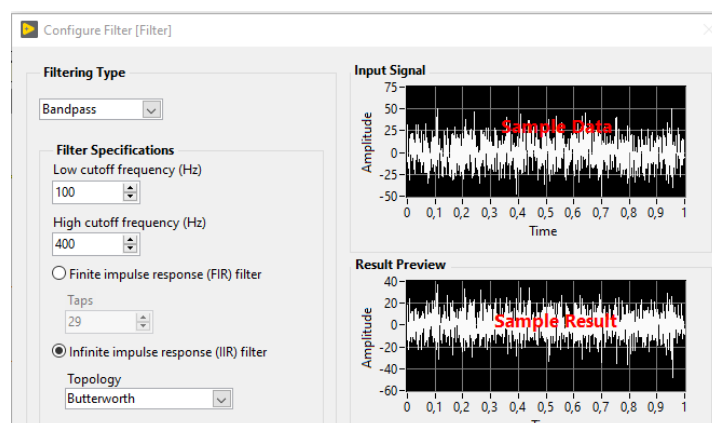


Figura 227. Configuración de filtro, práctica 8.

Luego a la salida del controlador se agrega las salidas digitales para el control de sentido horario y antihorario en cada motor en el puerto A DIO 1 y puerto A DIO2 como se muestra en la figura 228.

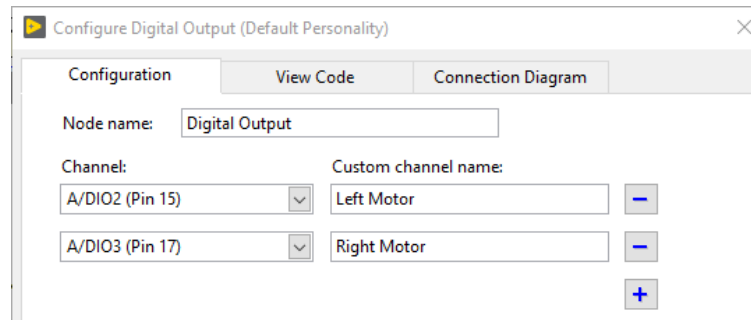


Figura 228. Configuración de salidas digitales, practica 8.

Posteriormente agregar dos bloques de salida analógicas para controlar el voltaje suministrado a los motores como se muestra en la figura 229.

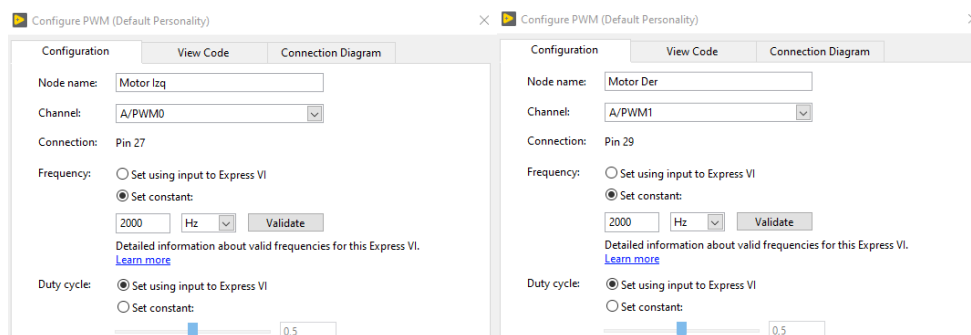


Figura 229. Ingreso de bloques de salidas analógicas, practica 8.

En la salida agregar un condicional cuando sea mayor que 0 enviar un dato a al motor 1 y en el segundo motor enviar la misma señal negada y controlar los voltajes mediante el dato de salida como valor absoluto como se muestra en la figura 230

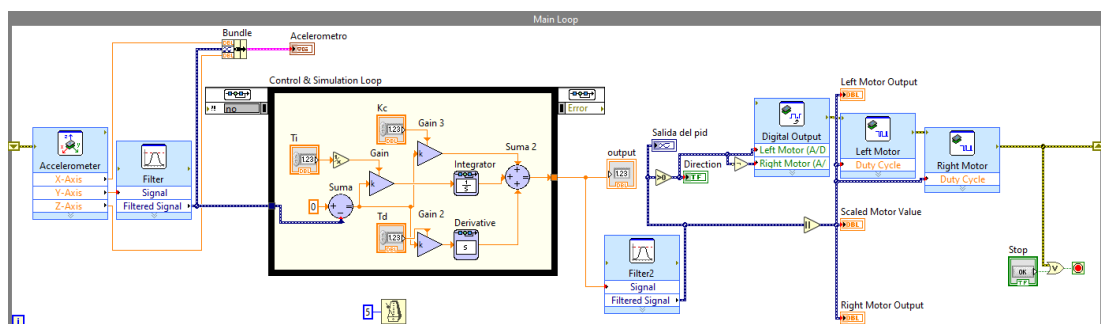


Figura 230. Condiciones de motores, practica 8.

Configurar el valor de K_c en 1.65 , T_i en 0.1 y T_d 0 como se muestra en la gráfica 231.

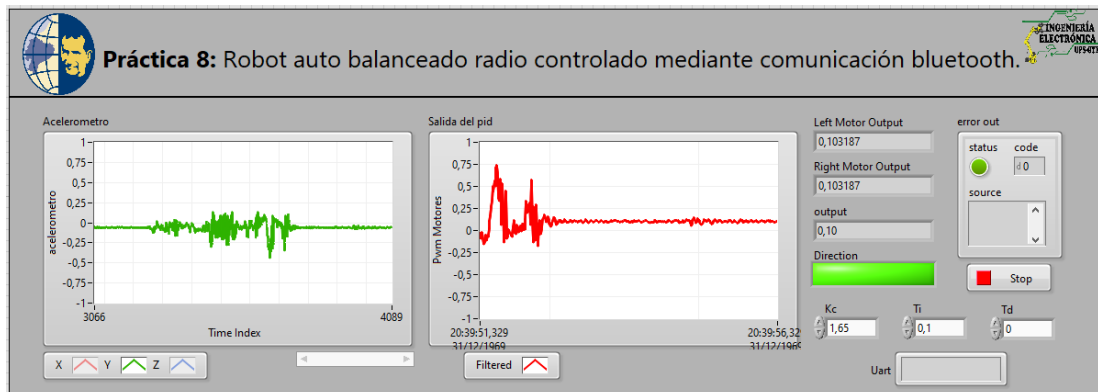


Figura 231. Interfaz practica 8.

Se procede a configurar la adquisición de datos uart mediante un bloque de comunicación serial como se muestra en la figura 232.

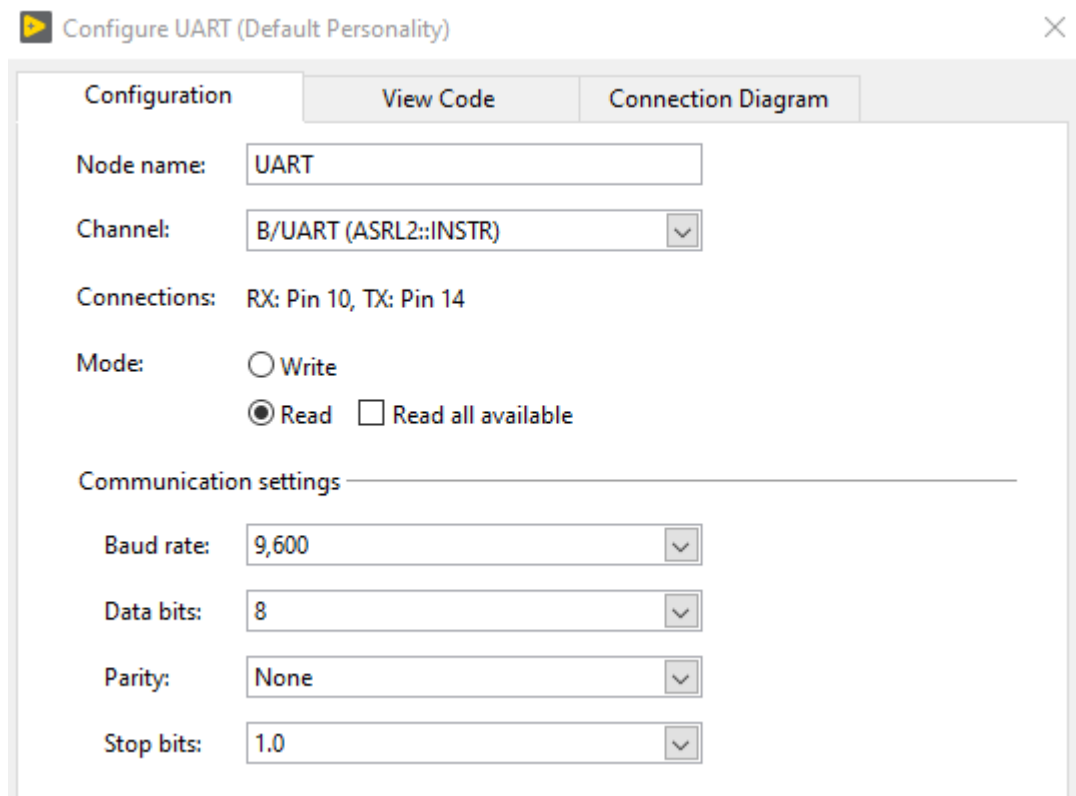


Figura 232. Gráficas de ganancia K_p , practica 8.

Luego se crea una estructura de comparación para el valor en el buffer de tal manera que realice las operaciones de adelante(F), reversa (B), izquierda (L),

derecha (R). El primer condicional es la ejecución por defecto cuando no se tiene ningún dato en el buffer en ese instante debe realizar el control como se muestra en la figura 233.

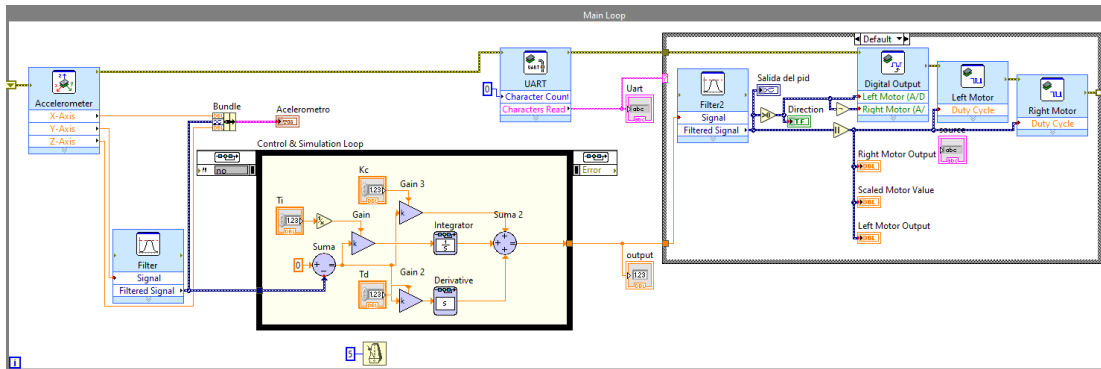


Figura 233. Condicional por defecto, practica 8.

El condicional para avanzar seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 234.

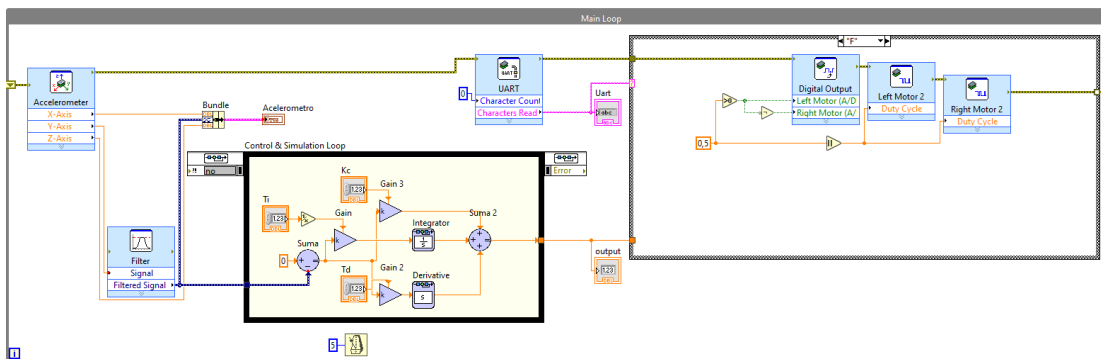


Figura 234. Condicional para avanzar, practica 8.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 235.

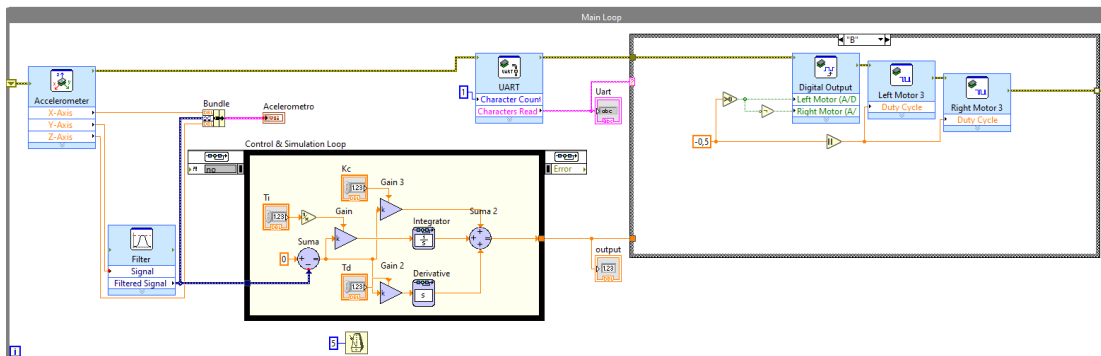


Figura 235. Condicional para retroceder, practica 8.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 236.

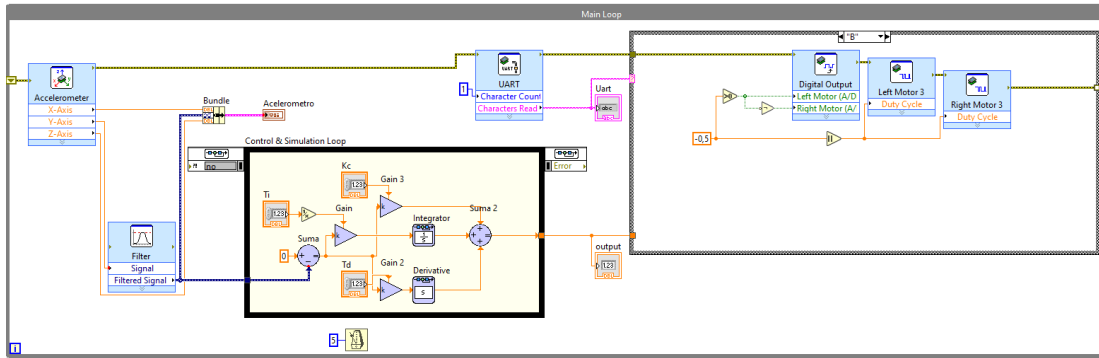


Figura 236. Condicional para girar derecha, practica 8.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 237.

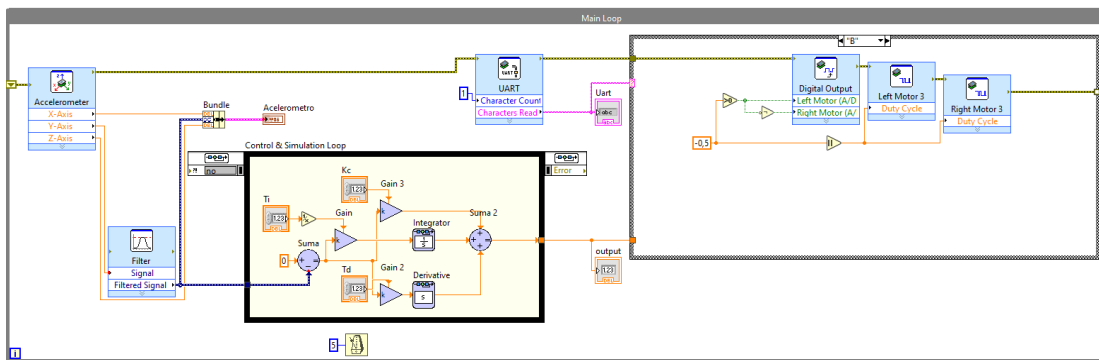


Figura 237. Condicional para girar izquierda, practica 8.

Crear un nuevo vi el cual se encarga de enviar los valores desde otro computador hacia el dispositivo bluetooth mediante el uso de la herramienta de comunicación serial, realizar la configuración por defecto y agregar un bloque de escritura serial como se muestra en la figura 238



Figura 238. Puerto serial con herramienta visa, practica 8.

Posteriormente se agrega 4 botones para el envío de los comando de avanzar, retroceder, giro derecha y giro izquierda como se muestra en la figura 239.



Figura 239.Botoneras de control, practica 8.

Agregar la lógica de tal manera que :

Presionar adelante envíe el comando F ; Presionar reversa envíe el comando B; Presionar izquierda envíe el comando L; Presionar derecha envíe el comando R; como se muestra en la figura 240.

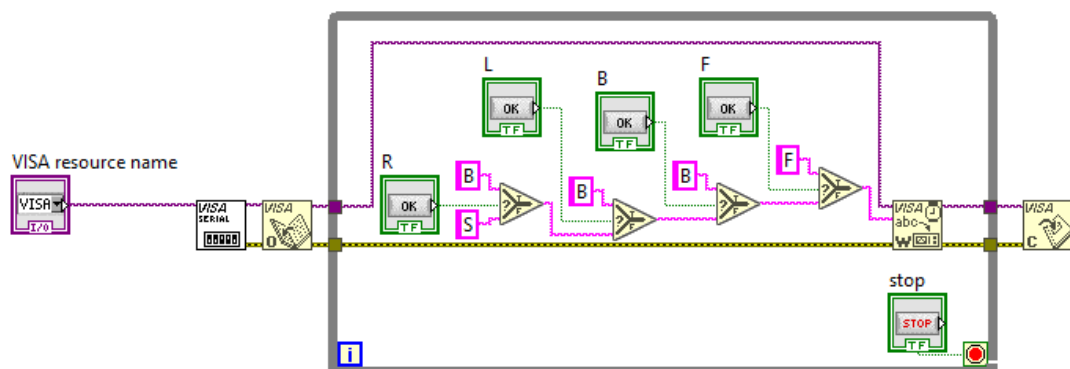


Figura 240.Lógica para el envío de valores al robot, practica 8.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con el dispositivo serial bluetooth
- Revisar el correcto funcionamiento del sensor al momento de la ejecución

Anexo 4.9 Solución de práctica 9

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 241

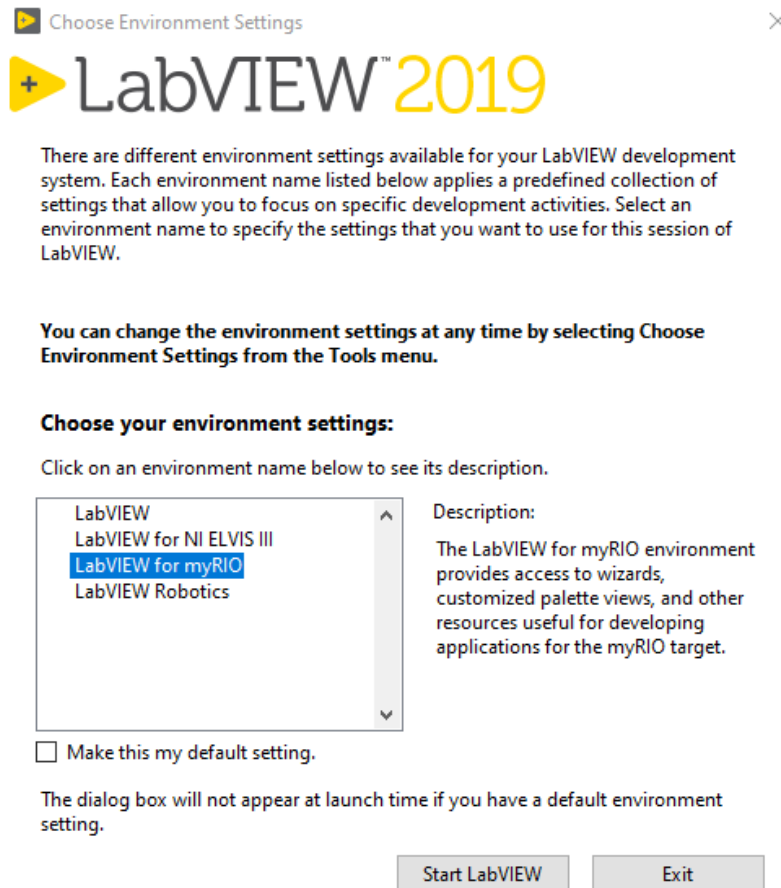


Figura 241. Ejecución de LabView como compatible con ni myRio
práctica 9

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 242.

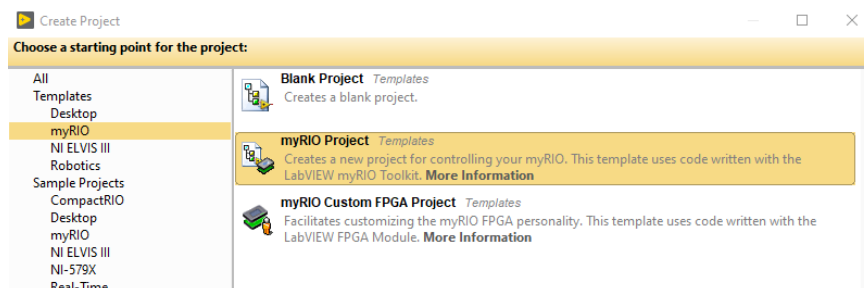


Figura 242. Nuevo Proyecto, práctica 9

Configurar la ruta donde se guarda el archivo (carpeta practica 9), nombre (Robot auto balanceado radio controlado mediante comunicación wifi) y el tipo de conexión como wifi y dar en finalizar como se muestra en la imagen 243.

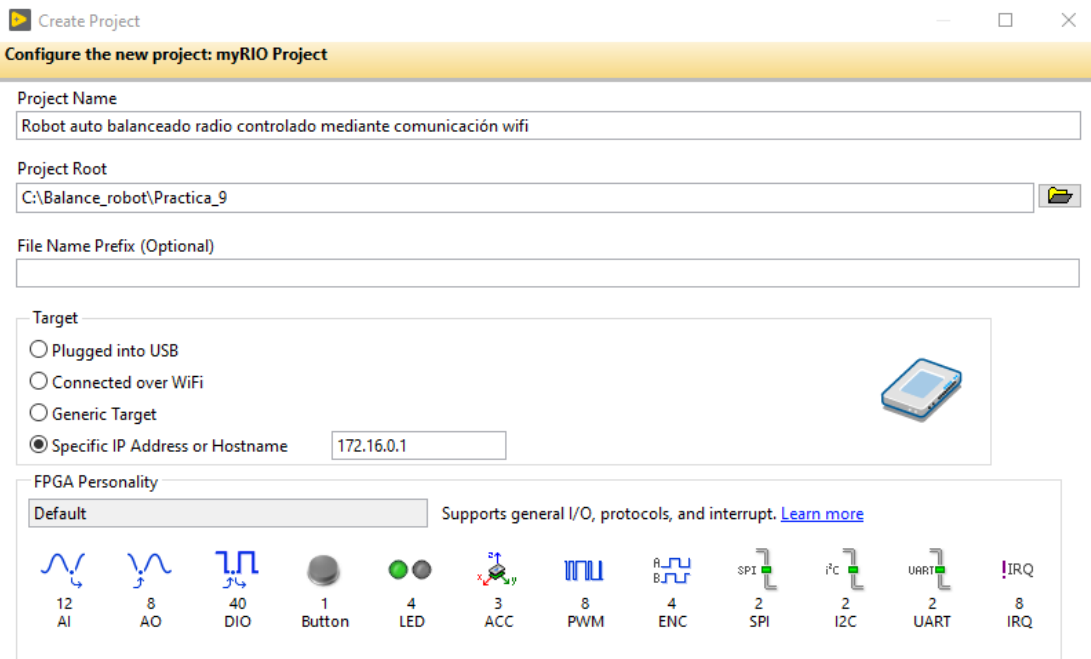


Figura 243. Configuración de ruta, práctica 9

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 244.

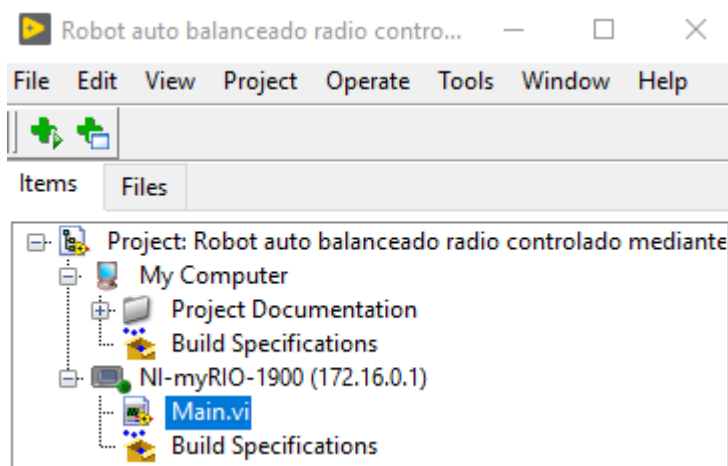


Figura 244. Administrador de proyectos, selección práctica 9.

Agregar tres datos de lectura analógica (bloque de diagrama >> myRio >> acelerometer), las entradas del acelerómetro se activan los ejes “x”, “y”, “z” como se muestra en la imagen 245.

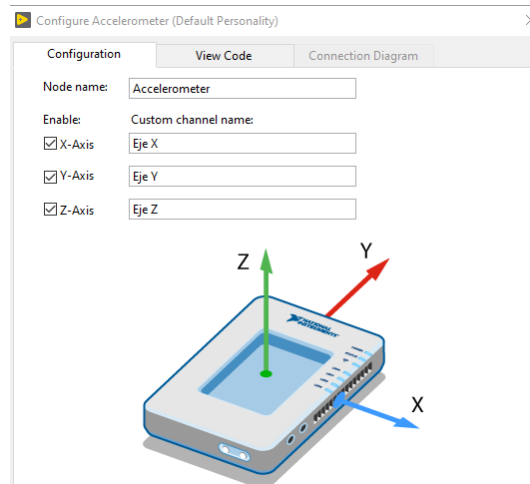


Figura 245. Ingreso de entradas analógicas, práctica 9.

Se procede a conectar las salidas a un indicador gráfico y se crea un indicador numérico para la variable del eje “y” como se muestra en la figura 246.

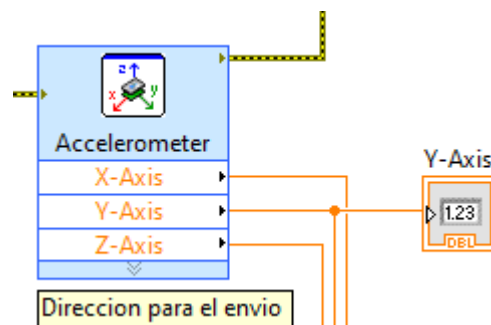


Figura 246. Conexión de salidas analógicas práctica 9.

En la salida del eje y se agrega un filtro para la eliminación de ruido mediante un pasa banda como se muestra en la figura 247.

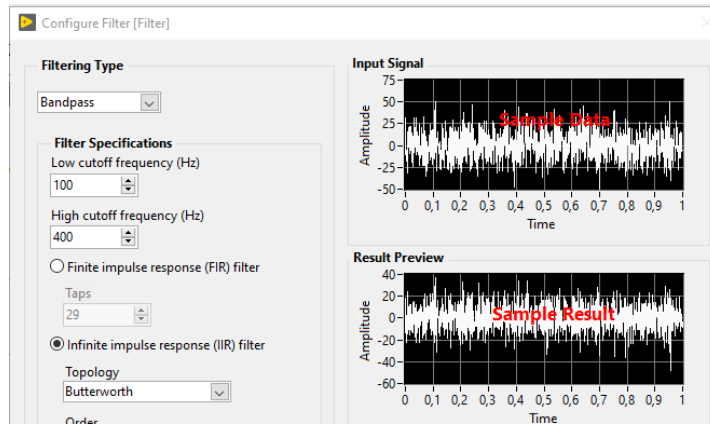


Figura 247. Configuración de filtro, práctica 9.

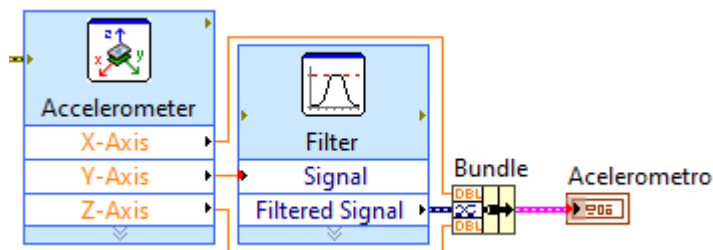


Figura 248. Filtro al eje "Y", práctica 9.

Agregar el bloque de control y simulación para el pid como se muestra en la figura de tal manera que la entrada de la variable del pid sea el valor del eje y la salida se asigne a una variable.

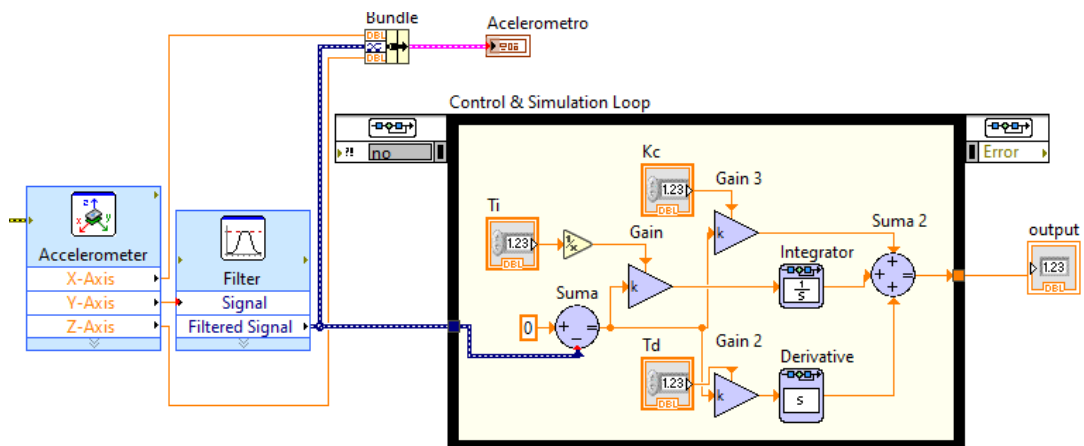


Figura 249. bloque de control y simulación para el pid, práctica 9.

En la salida del controlador se agrega un filtro para la eliminación de ruido mediante un pasa banda como se muestra en la figura 250.

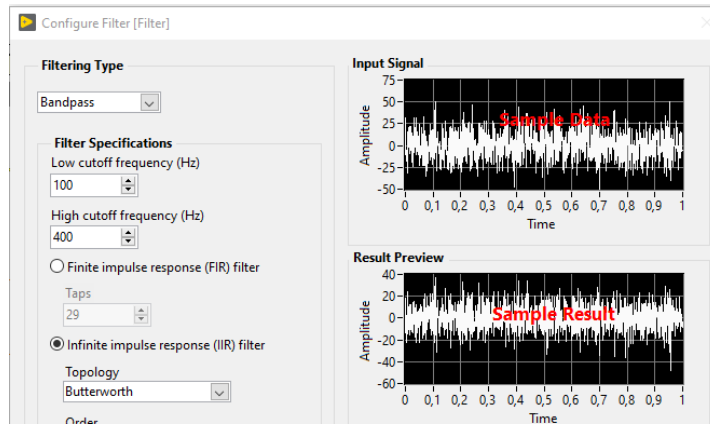


Figura 250. Configuración de filtro, práctica 9.

Luego a la salida del controlador se agrega las salidas digitales para el control de sentido horario y antihorario en cada motor en el puerto A DIO 1 y puerto A DIO2 como se muestra en la figura 251.

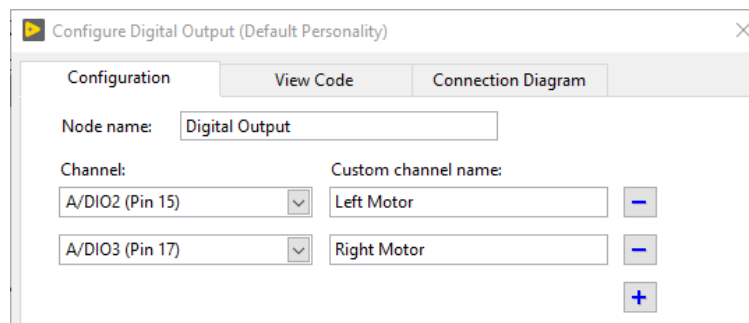


Figura 251. Configuración de salidas digitales, practica 9.

Posteriormente agregar dos bloques de salida analógicas para controlar el voltaje suministrado a los motores como se muestra en la figura 252.

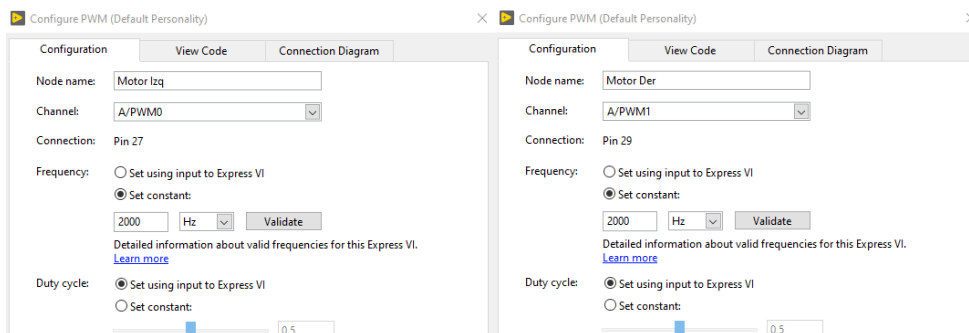


Figura 252. Ingreso de bloques de salidas analógicas, practica 9.

En la salida agregar un condicional cuando sea mayor que 0 enviar un dato a al motor 1 y en el segundo motor enviar la misma señal negada y controlar los voltajes mediante el dato de salida como valor absoluto como se muestra en la figura 253

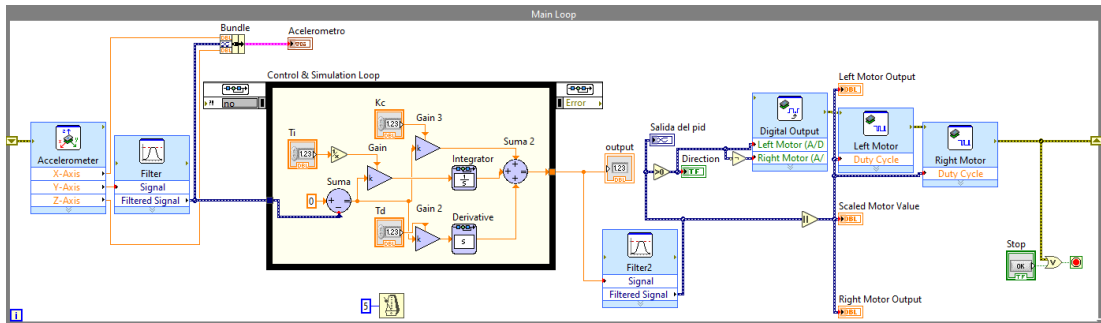


Figura 253. Condiciones de motores, practica 9.

Configurar el valor de kc en 1.65 , ti en 0.1 y Td 0 como se muestra en la gráfica 254.

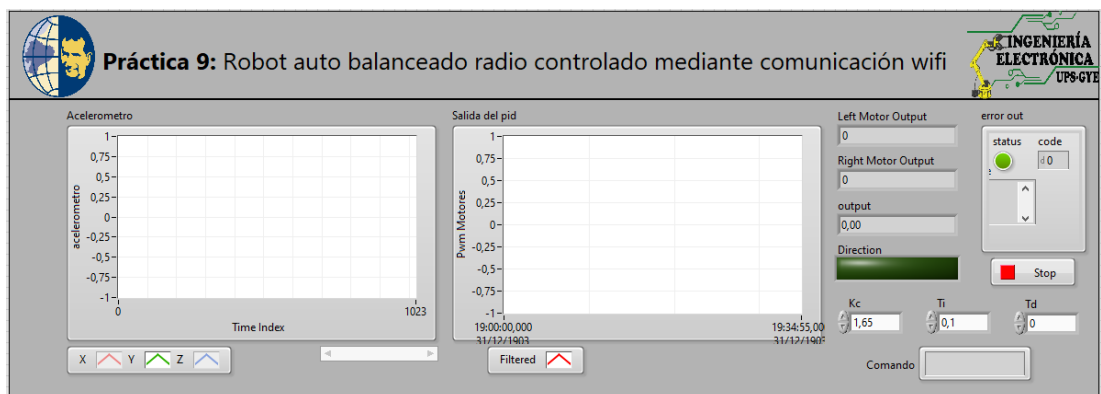


Figura 254. Interfaz practica 9.

Posteriormente se agrega 4 botones para el envío de los comando de avanzar, retroceder, giro derecha y giro izquierda como se muestra en la figura 255.

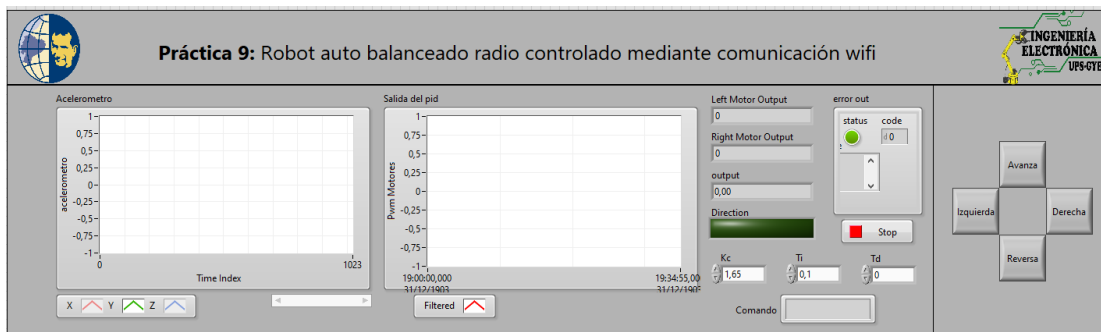


Figura 255. Botoneras de control, practica 9.

Agregar la lógica de tal manera que :

Presionar adelante envíe el comando F ; Presionar reversa envíe el comando B; Presionar izquierda envíe el comando L; Presionar derecha envíe el comando R; como se muestra en la figura 256.

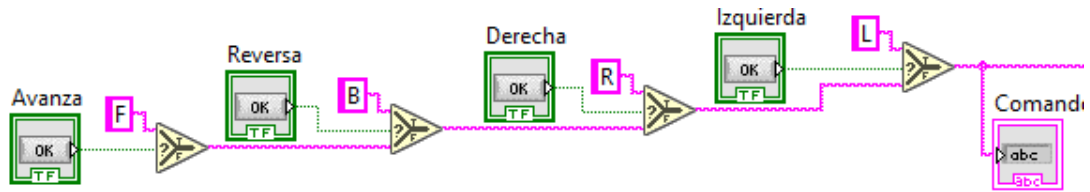


Figura 256.Lógica para el envío de valores al robot, practica 9.

Luego se crea una estructura de comparación para el valor en el buffer de tal manera que realice las operaciones de adelante(F), reversa (B), izquierda (L), derecha (R).El primer condicional es la como se muestra en la figura 257.

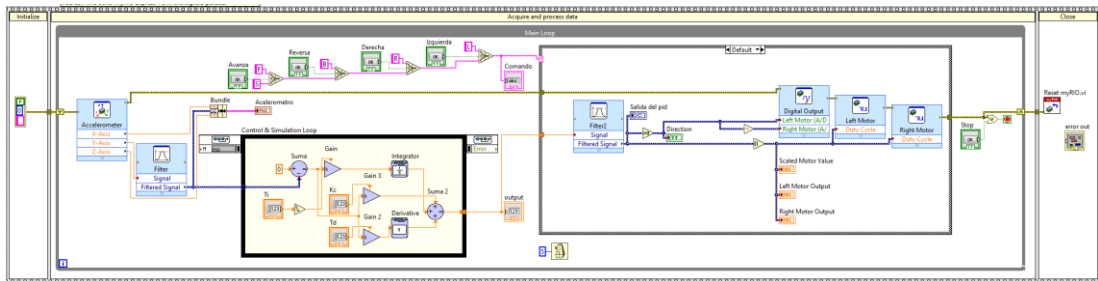


Figura 257.Condicional por defecto, practica 9.

El condicional para avanzar seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 258.

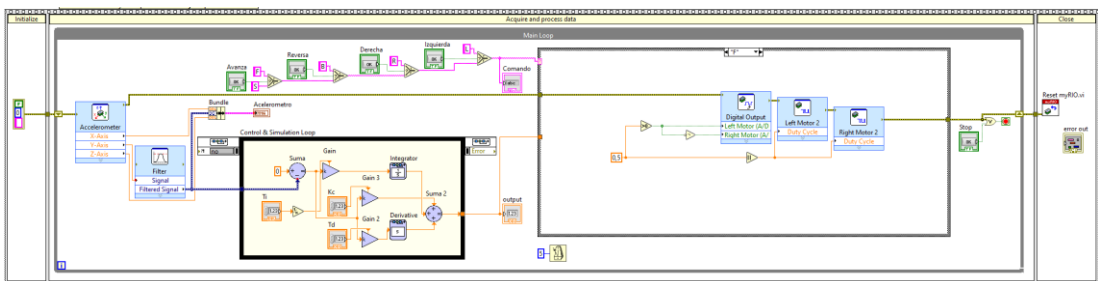


Figura 258.Condicional para avanzar, practica 9.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 259.

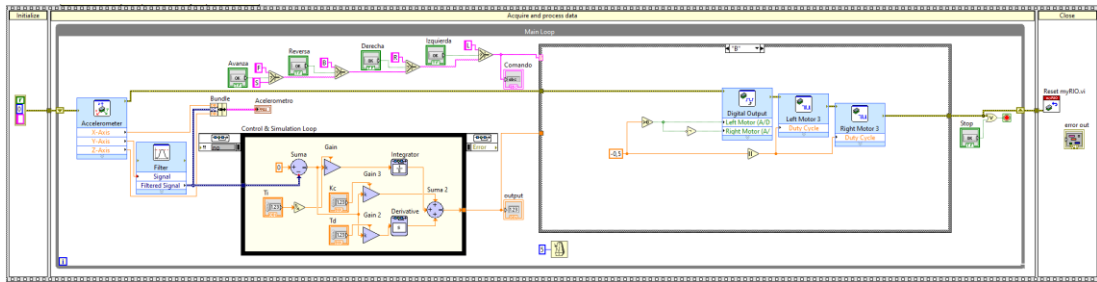


Figura 259. Condicional para retroceder, practica 9.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 260.

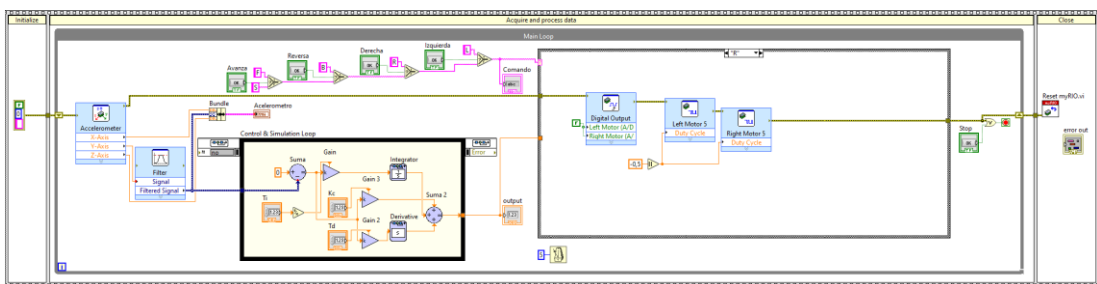


Figura 260. Condicional para girar derecha, practica 9.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 261.

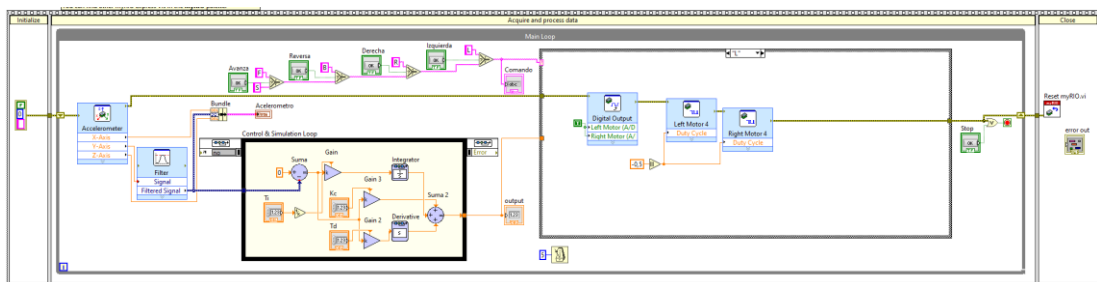


Figura 261. Condicional para girar izquierda, practica 9.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con los dispositivos en el puerto a.
- Revisar el correcto funcionamiento del sensor al momento de la ejecución

Anexo 4.10 Solución de práctica 10

Se procede a abrir la página (<http://ai2.appinventor.mit.edu/>) del aplicativo para la creación de un archivo ejecutable en sistemas Android. Agregar los botones y la herramienta de bluetooth como se muestra en la figura 262.

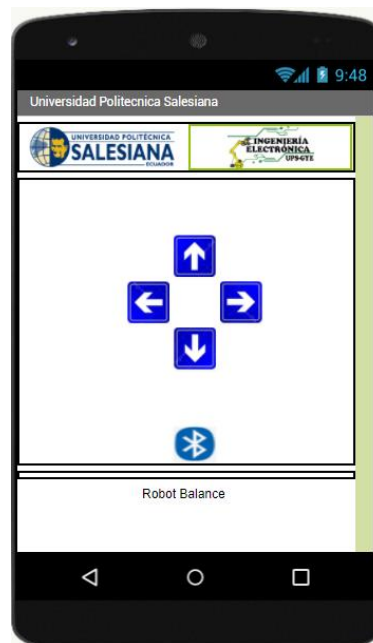


Figura 262. Diseño de aplicativo en Android, práctica 10

Crear en la ventanas de bloques una variable donde se guarda el dato a enviarse, configurar el bluetooth para realizar la búsqueda y conexión de los dispositivos como se muestra en la figura 263.

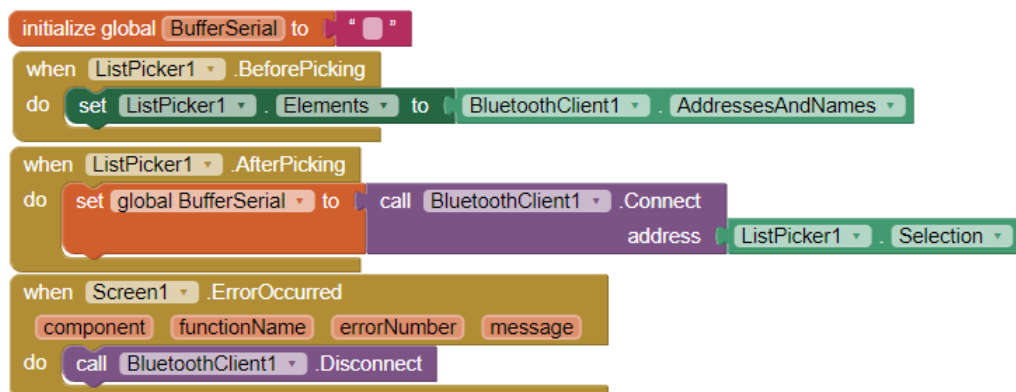


Figura 263. Programación en bloques de conexión, práctica 10

Crear en la ventanas de bloques una variable donde se guarda el dato a enviarse, configurar el bluetooth para realizar la búsqueda y conexión de los dispositivos como se muestra en la figura 264.

```
initialize global BufferSerial to " "
when ListPicker1 .BeforePicking
do set ListPicker1 .Elements to BluetoothClient1 .AddressesAndNames
when ListPicker1 .AfterPicking
do set global BufferSerial to call BluetoothClient1 .Connect
address ListPicker1 .Selection
when Screen1 .ErrorOccurred
component functionName errorNumber message
do call BluetoothClient1 .Disconnect
```

Figura 264. Programación en bloques de conexión, práctica 10

Configurar la lectura del accionamiento de los botones de operación de tal manera de enviar comandos: F – avanzar, B – retroceder, R derecha, L izquierda como se muestra en la figura 265

```
when derecha .Click
do call BluetoothClient1 .SendText
text " R "
when abajo .Click
do call BluetoothClient1 .SendText
text " B "
when arriba .Click
do call BluetoothClient1 .SendText
text " F "
when izquierda .Click
do call BluetoothClient1 .SendText
text " L "
```

Figura 265. Programación en bloques de comandos, práctica 10

Ejecutar LabView y seleccionar el modo de LabView compatible con él ni myRio como se muestra en la figura 266

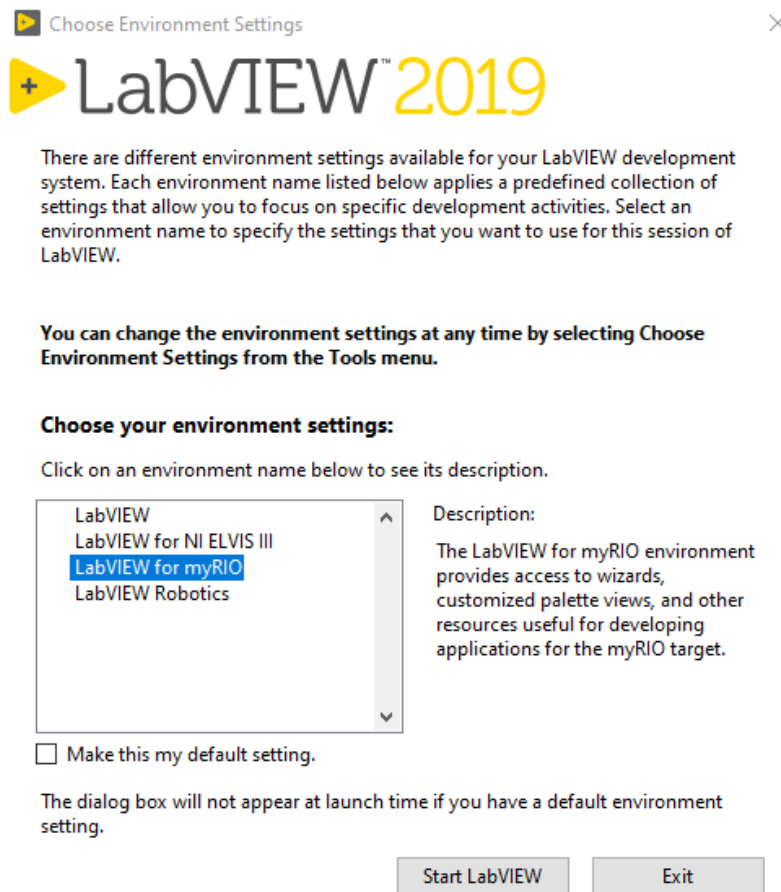


Figura 266. Ejecución de LabView como compatible con ni myRio práctica 10

Seleccionar crear nuevo proyecto myRio como se muestra en la figura 267.

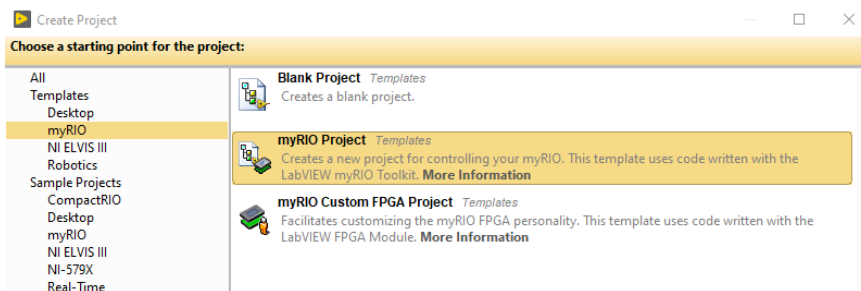


Figura 267. Nuevo Proyecto, práctica 10

Configurar la ruta donde se guarda el archivo (carpeta practica 10), nombre (Robot auto balanceado radio controlado mediante aplicativo en aplicación

App Inventor.) y el tipo de conexión como wifi y dar en finalizar como se muestra en la imagen 268.

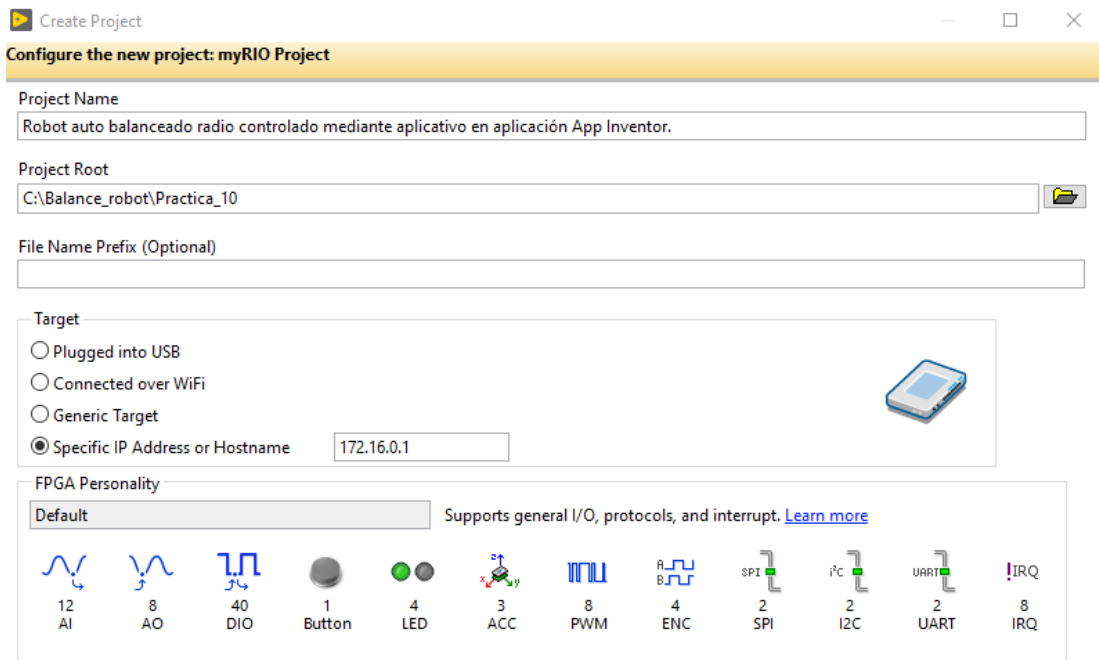


Figura 268. Configuración de ruta, práctica 10

En el administrador de proyectos abrir el vi principal donde se programara el aplicativo como se muestra en la imagen 269.

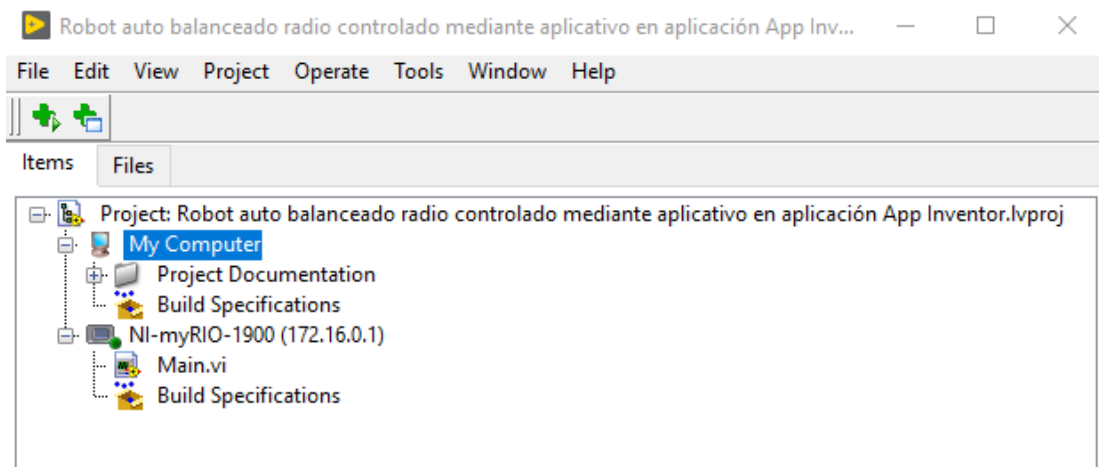


Figura 269. Administrador de proyectos, selección práctica 10.

Agregar tres datos de lectura analógica (bloque de diagrama >>myRio >> acelerometer), las entradas del acelerómetro se activan los ejes “x”, “y”, “z” como se muestra en la imagen 270.

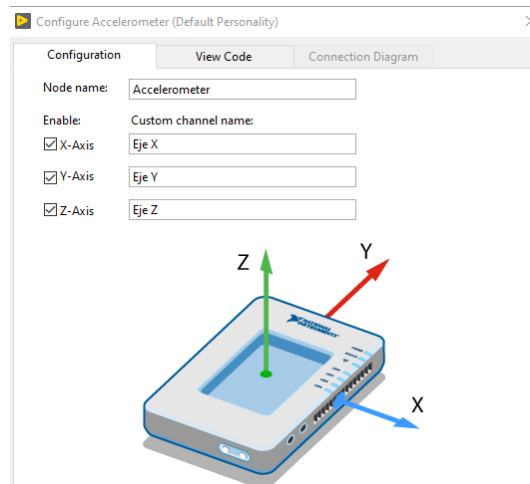


Figura 270. Ingreso de entradas analógicas, práctica 10.

Se procede a conectar las salidas a un indicador gráfico y se crea un indicador numérico para la variable del eje “y” como se muestra en la figura 271

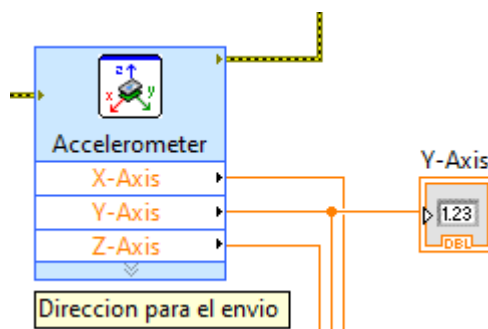


Figura 271. Conexión de salidas analógicas práctica 10.

En la salida del eje y se agrega un filtro para la eliminación de ruido mediante un pasa banda como se muestra en la figura 272.

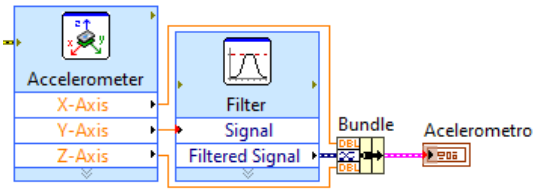


Figura 272. Filtro al eje “Y”, práctica 10.

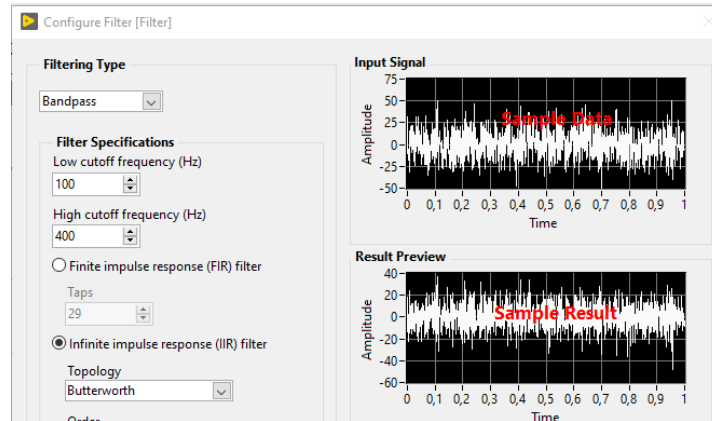


Figura 273. Configuración de filtro, práctica 10.

Agregar el bloque de control y simulación para el pid como se muestra en la figura de tal manera que la entrada de la variable del pid sea el valor del eje y la salida se asigne a una variable.

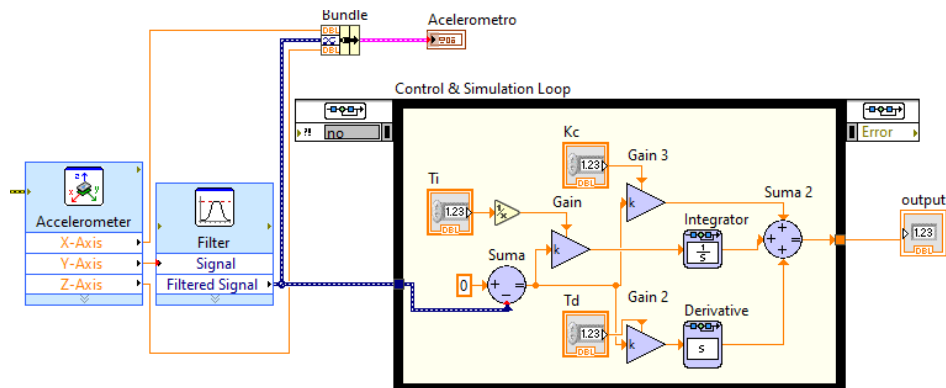


Figura 274. bloque de control y simulación para el pid, práctica 10.

En la salida del controlador se agrega un filtro para la eliminación de ruido mediante un pasa banda como se muestra en la figura 275.

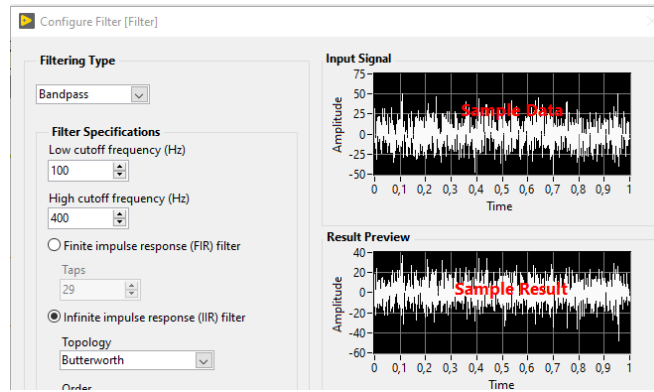


Figura 275. Configuración de filtro, práctica 10.

Luego a la salida del controlador se agrega las salidas digitales para el control de sentido horario y antihorario en cada motor en el puerto A DIO 1 y puerto A DIO2 como se muestra en la figura 276.

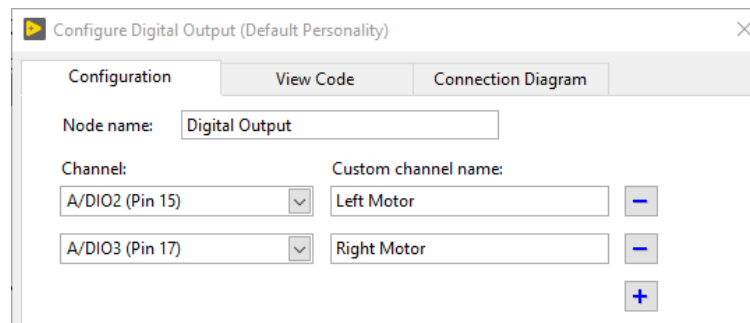


Figura 276. Configuración de salidas digitales, practica 10.

Posteriormente agregar dos bloques de salida analógicas para controlar el voltaje suministrado a los motores como se muestra en la figura 277.

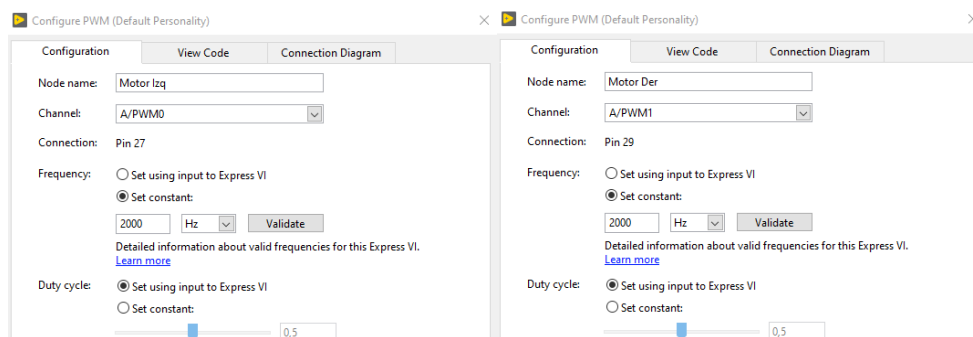


Figura 277. Ingreso de bloques de salidas analógicas, practica 10.

En la salida agregar un condicional cuando sea mayor que 0 enviar un dato a al motor 1 y en el segundo motor enviar la misma señal negada y controlar los

voltajes mediante el dato de salida como valor absoluto como se muestra en la figura 278.

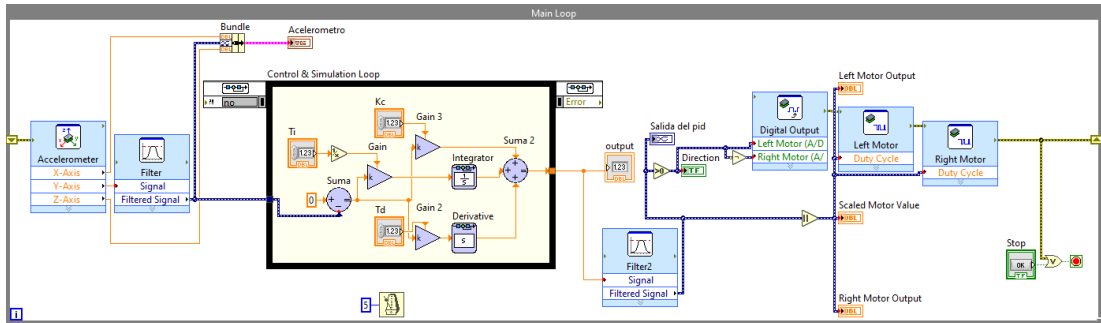


Figura 278. Condiciones de motores, practica 10.

Configurar el valor de k_c en 1.65 , t_i en 0.1 y T_d 0 como se muestra en la gráfica 279.

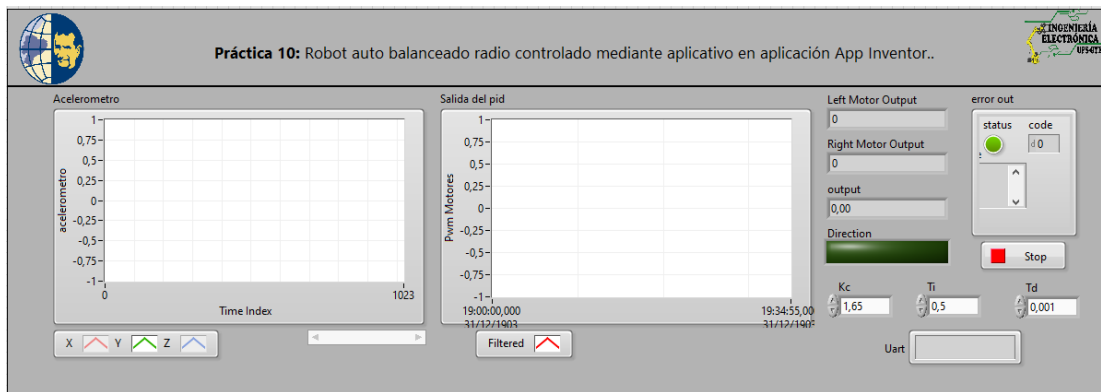


Figura 279. Interfaz practica 10.

Se procede a configurar la adquisición de datos uart mediante un bloque de comunicación serial como se muestra en la figura 280.

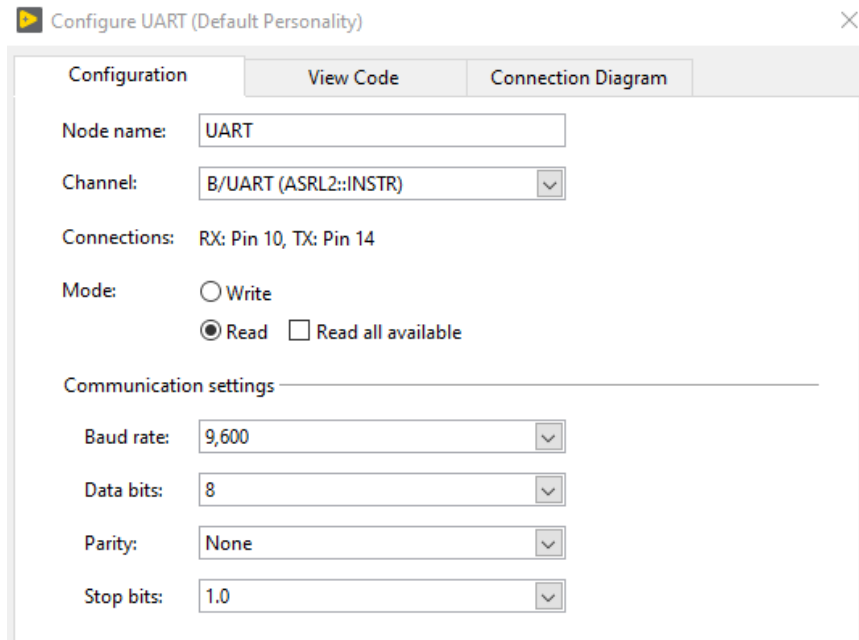


Figura 280. Configurar la adquisición de datos uart, practica 10.

Luego se crea una estructura de comparación para el valor en el buffer de tal manera que realice las operaciones de adelante(F), reversa (B), izquierda (L), derecha (R). El primer condicional es la ejecución por defecto cuando no se tiene ningún dato en el buffer en ese instante debe realizar el control como se muestra en la figura 281.

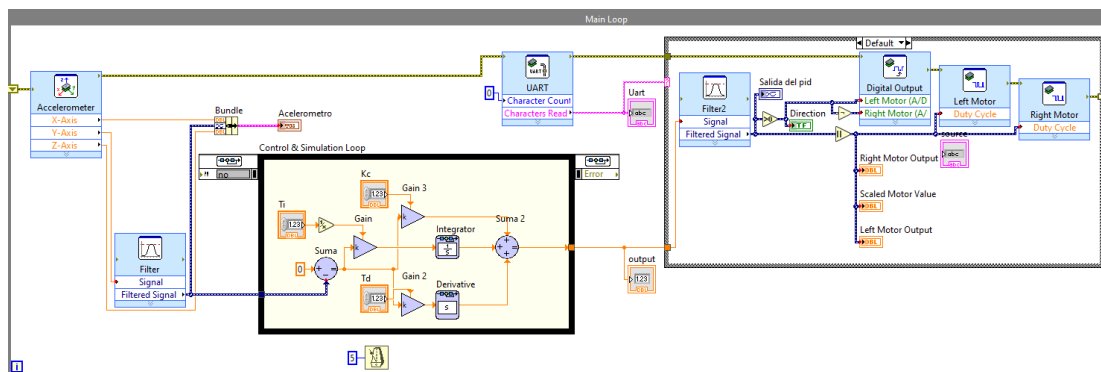


Figura 281. Condicional por defecto, practica 10.

El condicional para avanzar seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 282.

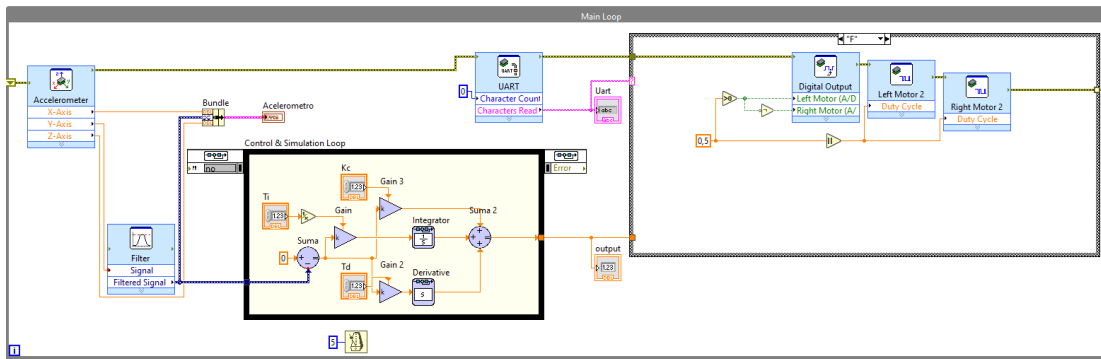


Figura 282. Condicional para avanzar, practica 10.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 283.

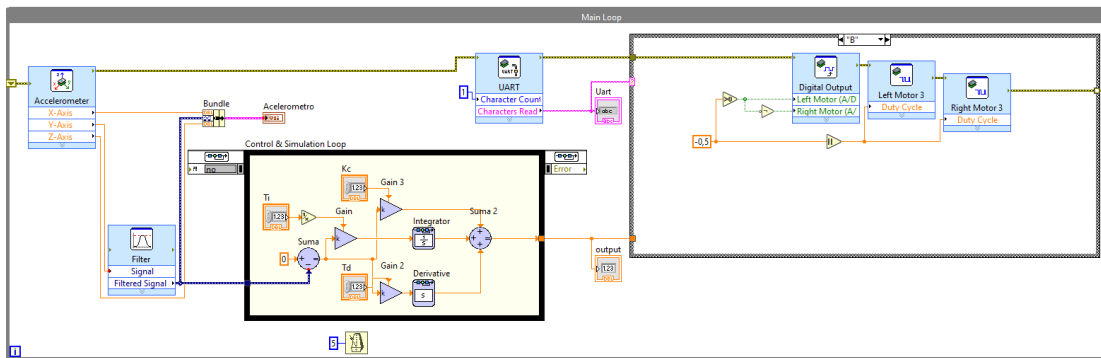


Figura 283. Condicional para retroceder, practica 10.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 284.

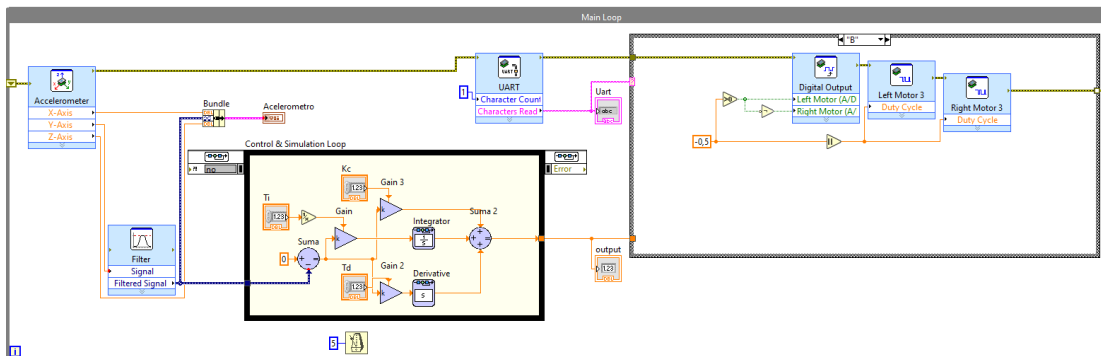


Figura 284. Condicional para girar derecha, practica 10.

El condicional para retroceder seteo un valor a los motores durante el tiempo que dure el pulso y realizar el control como se muestra en la figura 285.

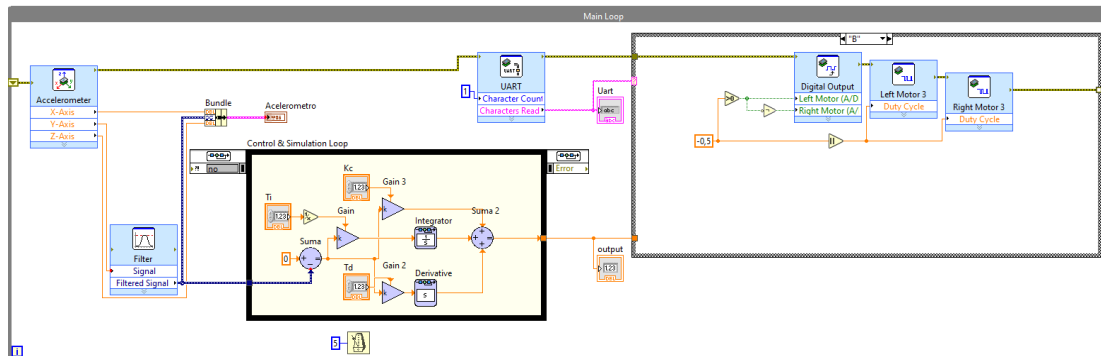


Figura 285. Condicional para girar izquierda, practica 10.

RECOMENDACIONES:

- Comprobar la carga o el estado de la batería.
- Verificar la correcta conexión del ni myRio con el dispositivo serial bluetooth
- Revisar el correcto funcionamiento del sensor al momento de la ejecución

Anexo 5 Configuración de red inalámbrica

Ejecutar el programa ni Max de National Instrument para la configuración del ni myRio con una red inalámbrica



Figura 286. Programa ni Max de National Instrument

Seleccionar el dispositivo ni myRio en sistemas remotos como se muestra en la imagen.

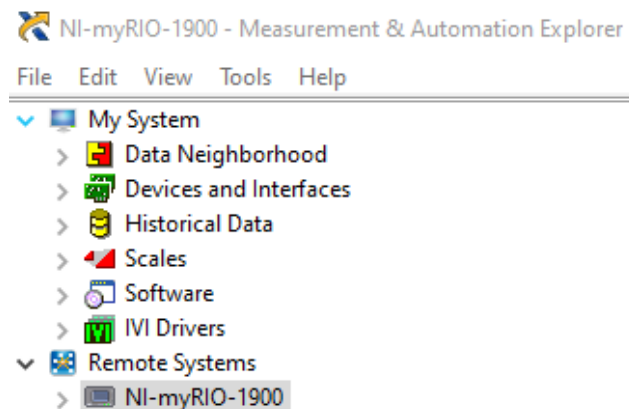


Figura 287. Dispositivo ni myRio en sistemas remotos

Seleccionar configuración de red inalámbrica y conectarse a una red ya existente o crear una red , la dirección ip se configura como estática con la configuración que se muestra en la figura

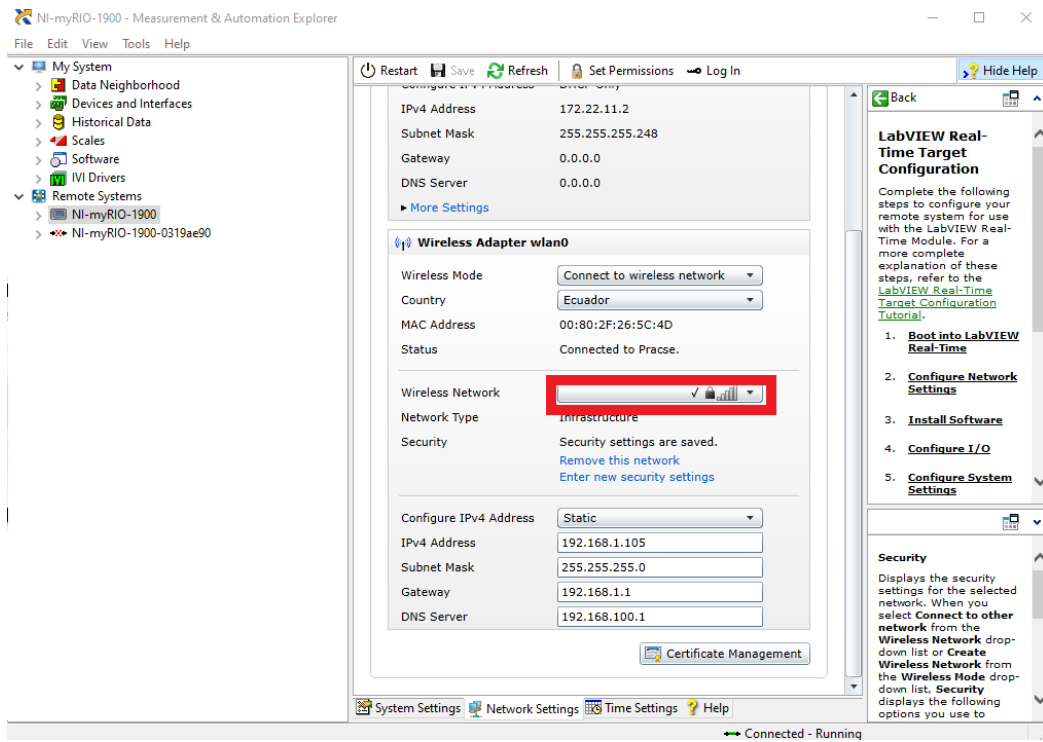


Figura 288. Configuración de red inalámbrica

Al finalizar dar click en guardar y se finalizara la configuración como se muestra en la figura

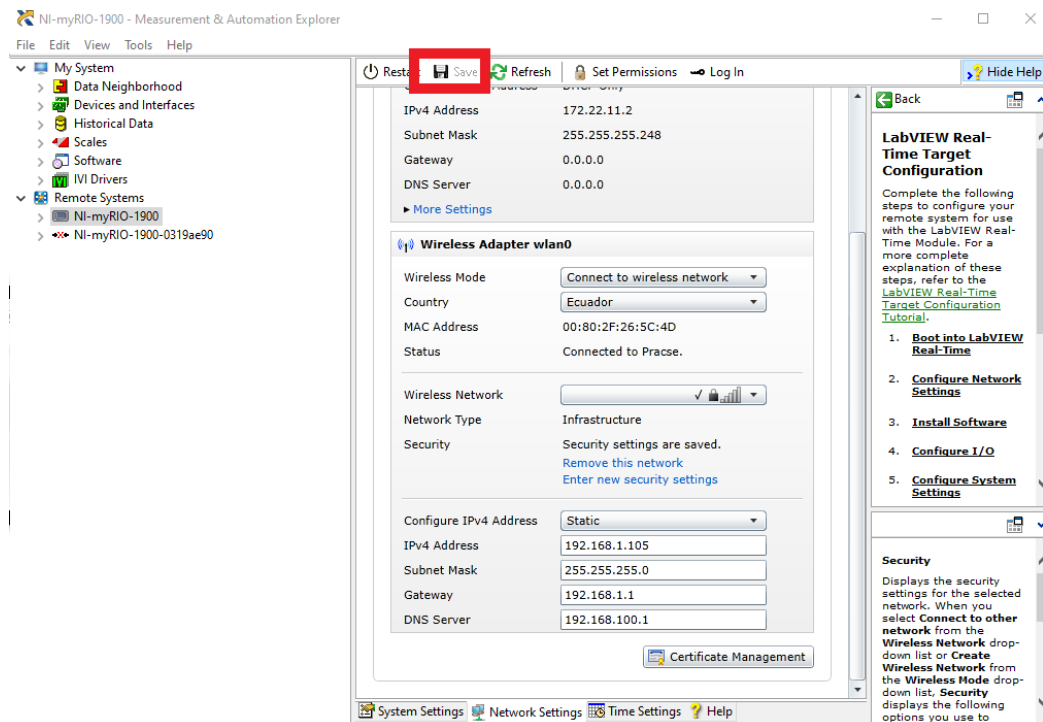


Figura 289. Finalizar la configuración