



UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERÍAS INDUSTRIALES

Grado en Ingeniería Electrónica Industrial y Automática

Sistemas para determinar la posición y
orientación de herramientas quirúrgicas en
operaciones de cirugía laparoscópica

Autor:

Vara Rodríguez, David

Tutor:

Juan Carlos Fraile Marinero

Departamento de Ingeniería de Sistemas y Automática

Valladolid, julio 2014



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez

*A mi familia.
Por su amor y apoyo incondicional.*

Per ardua ad astra



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Resumen

El presente proyecto acomete el desarrollo de un sistema de localización de la posición de un cuerpo mediante sensores de bajo coste y el estudio de la viabilidad del mismo en operaciones de cirugía.

El sensor utilizado es un sensor IMU de tecnología MEMS compuesto por un acelerómetro y un giroscopio. Para la señal proporcionada, se analizan distintos métodos de filtrado así como diversas reglas de integración numérica, seleccionándose posteriormente la más óptima.

El microcontrolador utilizado es un Arduino UNO basado en el ATmega 328. La comunicación entre este dispositivo y el PC puede realizarse tanto mediante un cable USB como de forma inalámbrica mediante un módulo bluetooth.

Finalmente se realiza un análisis de los resultados obtenidos así como de las posibles futuras líneas de actuación a seguir en trabajos o proyectos posteriores.



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Palabras clave:

Localización, Posición, Cirugía, Laparoscópica, Acelerómetro



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Índice general

CAPÍTULO 1. INTRODUCCIÓN	23
1. INTRODUCCIÓN AL CAPÍTULO	23
2. DESCRIPCIÓN GENERAL	23
3. CONTENIDOS DE LA MEMORIA.....	28
CAPÍTULO 2. ESTADO DEL ARTE	31
1. INTRODUCCIÓN AL CAPÍTULO	31
2. QUÉ ES UN SENSOR DE POSICIÓN	31
3. TIPOS DE SENSORES DE POSICIÓN.....	32
3.1. SENSORES DE RESISTENCIA VARIABLE	31
3.2. SENSORES CAPACITIVOS.....	33
3.2.1. MODIFICACIÓN DE LA CONSTANTE DIELECTRICA.....	33
3.2.2. MODIFICACIÓN DE LA ZONA DE SOLAPAMIENTO	34
3.3. TRANSFORMADOR DIFERENCIAL DE VARIACIÓN LINEAL.....	34
3.4. SENSORES MAGNETORRESISTIVOS	36
3.5. SENSORES DE ULTRASONIDOS	36
3.6. SENSORES ÓPTICOS	37
3.7. SENSORES INERCIALES (IMU)	38
3.7.1. COMPONENTES DE UNA IMU	39
3.7.2. CARACTERIZACIÓN DE UNA IMU	40
4. MÉTODOS DE LOCALIZACIÓN EN CIRUGÍA LAPAROSCÓPICA	42
4.1. ASPECTOS GENERALES DE LOS SISTEMAS DE LOCALIZACIÓN	42



David Vara Rodríguez

4.2. SISTEMAS PASIVOS DE LOCALIZACIÓN.....	43
4.3. SISTEMAS ACTIVOS DE LOCALIZACIÓN	44
4.4. PROCESAMIENTO DE VÍDEO.....	50
CAPÍTULO 3. ELEMENTOS UTILIZADOS	53
1. INTRODUCCIÓN AL CAPÍTULO	53
2. DESCRIPCIÓN DE LOS ELEMENTOS	54
2.1. IMU MPU6050.....	54
2.2. ARDUINO UNO.....	55
2.3. MÓDULO HC-06.....	56
3. CONEXIONES.....	57
3.1. CONEXIÓN SENSOR – MICROCONTROLADOR.....	57
3.2. CONEXIÓN MICROCONTROLADOR – PC.....	59
3.2.1. CONEXIÓN USB.....	59
3.2.2. CONEXIÓN INALÁMBRICA	60
4. COMUNICACIÓN MICROCONTROLADOR – PC.....	62
4.1. COMUNICACIÓN VÍA PUERTO SERIE	62
4.1.1. SERIE DE PULSOS.....	62
4.2. COMUNICACIÓN ARDUINO – MATLAB VÍA PUERTO SERIE	64
4.2.1. CONFIGURACIÓN DE ARDUINO	64
4.2.1.1. CONFIGURACIÓN DEL MÓDULO BLUETOOTH.....	64
4.2.2. CONFIGURACIÓN DE MATLAB	71



David Vara Rodríguez

CAPÍTULO 4. LOCALIZACIÓN DE LA POSICIÓN A PARTIR DE LOS DATOS DEL ACELERÓMETRO..... 73

1. INTRODUCCIÓN AL CAPÍTULO	73
2. DISPOSITIVOS MEMS	73
3. ANÁLISIS DEL PROBLEMA Y MODELADO	74
4. INTEGRACIÓN NUMÉRICA.....	74
4.1. MÉTODOS DE INTEGRACIÓN NUMÉRICA.....	75
4.1.1. LA REGLA TRAPEZOIDAL.....	75
4.1.2. LA REGLA DE SIMPSON	76
4.2. ERROR DE INTEGRACIÓN.....	77
5. ORIGEN DE LOS ERRORES	78
5.1. ERROR DEL MEMS	78
5.1.1. ERROR DE LA INCLINACIÓN.....	78
5.2. ERRORES DEBIDOS AL MUESTREO DE LA SEÑAL.....	80
5.3. ERRORES DEBIDOS A LA CONVERSIÓN A/D.....	81
5.3.1. ERROR DE NO LINEALIDAD.....	82
5.4. ERROR DE LA RESOLUCIÓN DEL MICROCONTROLADOR.....	83
6. SOLUCIONES PROPUESTAS	84
6.1. REGLA DEL TRAPECIO	84
6.1.1. INTRODUCCIÓN AL ALGORITMO.....	84
6.2. ERRORES Y APROXIMACIÓN TRAPEZOIDAL.....	86
6.3. FILTRO DE KALMAN	88
6.3.1. FILTRO DISCRETO DE KALMAN.....	89
6.3.1.1. MODELO DEL SISTEMA	90
6.3.1.1.1. ALGORITMO	91



David Vara Rodríguez

6.3.1.2. PARÁMETROS DEL FILTRO Y SINTONIZACIÓN	93
7. SOFTWARE DESARROLLADO	94
7.1. INTRODUCCIÓN	94
7.2. CALIBRACIÓN DEL DISPOSITIVO	96
7.3. VENTANA DE FILTRADO	99
7.4. FILTRO DE KALMAN	100
7.4.1. IMPLEMENTACIÓN DEL FILTRO DE KALMAN	102
7.4.1.1. VALORES DE LAS MATRICES DE COVARIANZA Q Y R	104
7.5. FILTRO DE KALMAN VS FILTRO MEDIA MÓVIL	106
7.6. CÁLCULO DE LA VELOCIDAD Y LA POSICIÓN	111
7.7. DISTINCIÓN ENTRE VELOCIDAD NULA Y CONSTANTE	112
CAPÍTULO 5. PRUEBAS REALIZADAS	115
1. INTRODUCCIÓN AL CAPÍTULO	115
2. PRUEBAS REALIZADAS EN UNA DIMENSIÓN (1D)	116
2.1. PRUEBA 35 CM LINEAL	116
2.2. PRUEBA 15 CM LINEAL	119
2.3. PRUEBA 10 CM LINEAL	122
2.3.1. PRUEBA 10 CM LINEAL VELOCIDAD LENTA	125
2.4. PRUEBA 5 CM LINEAL	128
2.5. COMENTARIOS DE LAS PRUEBAS EN 1D	131
3. PRUEBAS REALIZADAS EN DOS DIMENSIONES (2D)	132
3.1. DIAGONAL	132
3.2. RECTÁNGULO	135
3.3. RECTÁNGULO VELOCIDAD LENTA	138



David Vara Rodríguez

3.4. CÍRCULO.....	141
3.5. ROBOT ABB IRB 120.....	144
3.6. COMENTARIOS DE LAS PRUEBAS EN 2D	147
4. PRUEBAS REALIZADAS EN 3 DIMENSIONES (3D)	148
4.1. PARALEPÍPEDO	148
4.2. COMENTARIOS DE LAS PRUEBAS EN 3D	151
5. CONCLUSIONES DE LAS PRUEBAS REALIZADAS.....	152
CAPÍTULO 6. CONCLUSIONES Y LÍNEAS FUTURAS.....	155
1. INTRODUCCIÓN AL CAPÍTULO	155
2. RESULTADOS OBTENIDOS.....	155
3. CONCLUSIONES.....	156
4. FUTURAS LÍNEAS DE ACTUACIÓN	156
BIBLIOGRAFÍA	157



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Índice de figuras

FIGURA 1. VISTA INTERIOR DEL DISPOSITIVO CONSTRUIDO.....	24
FIGURA 2. VISTA EXTERIOR DEL DISPOSITIVO CONSTRUIDO	24
FIGURA 3a. MECANISMO DE GUÍA	25
FIGURA 3b. MECANISMO DE GUÍA + DISPOSITIVO LOCALIZACIÓN	25
FIGURA 4a. ENTORNO DE TRABAJO CON CONEXIÓN USB	25
FIGURA 4b. ENTORNO DE TRABAJO CON CONEXIÓN BLUETOOTH	25
FIGURA 5. ENTORNO DE TRABAJO PRUEBA ROBOT.....	26
FIGURA 6. DETALLE DEL DISPOSITIVO CONSTRUIDO EN EL TCP DEL ROBOT.....	26
FIGURA 7. FUNCIONAMIENTO DE LOS SENSORES RESISTIVOS	32
FIGURA 8. DESPLAZAMIENTO EN FUNCIÓN DE LA CTE DIELECTRICA	33
FIGURA 9. DESPLAZAMIENTO EN FUNCIÓN DE LA ZONA DE SOLAPAMIENTO..	34
FIGURA 10. SENSOR LVDT.....	35
FIGURA 11. DEVANADOS	35
FIGURA 12. SENSOR MAGNETORRESISTIVO.....	36
FIGURA 13. SENSOR DE ULTRASONIDOS	36
FIGURA 14. SENSOR ÓPTICO	37
FIGURA 15. ORIENTACIÓN PROPORCIONADA POR UNA IMU.....	38
FIGURA 16. SIMULADOR PROMIS DE HAPTICA	43
FIGURA 17. SISTEMA CEBRIS DE ULTRASONIDOS	44
FIGURA 18a. LAPAROSCOPIC SURGICAL WORKSTATION	45
FIGURA 18b. VIRTUAL LAPAROSCOPIC INTERFACE	45
FIGURA 19. LAPAROSCOPIC IMPULSE ENGINE.....	45



David Vara Rodríguez

FIGURA 20. DISPOSITIVO CELTS.....	46
FIGURA 21. SISTEMA ADEPT	46
FIGURA 22. SISTEMA SIMENDO.....	47
FIGURA 23. DISPOSITIVO BLUEDRAGON	47
FIGURA 24. DISPOSITIVO PATRIOT	48
FIGURA 25. DISPOSITIVO TREND0	48
FIGURA 26. COMPONENTES UTILIZADOS EN EL DISPOSITIVO	53
FIGURA 27. IMU MPU6050.....	54
FIGURA 28. ARDUINO UNO	55
FIGURA 29. MÓDULO BLUETOOTH HC-06.....	56
FIGURA 30. ESQUEMA DE LA CONEXIÓN SENSOR – MICROCONTROLADOR.....	57
FIGURA 31. CONEXIÓN SENSOR – MICROCONTROLADOR.....	58
FIGURA 32. CABLE USB	59
FIGURA 33. CONEXIÓN DISPOSITIVO CONSTRUIDO – PC VÍA USB.....	59
FIGURA 34. ESQUEMA DE CONEXIÓN MÓDULO BLUETOOTH – ARDUINO	60
FIGURA 35. DISPOSITIVO CONSTRUIDO CON MÓDULO BLUETOOTH.....	61
FIGURA 36. SERIE DE PULSOS	62
FIGURA 37. AGREGAR NUEVA CONEXIÓN BLUETOOTH.....	65
FIGURA 38. ASISTENTE NUEVA CONEXIÓN BLUETOOTH	65
FIGURA 39. SELECCIONAR NUEVO DISPOSITIVO BLUETOOTH	66
FIGURA 40. INSTALACIÓN BLUETOOTH FINALIZADA.....	66
FIGURA 41. CARACTERÍSTICAS DE LA NUEVA CONEXIÓN BLUETOOTH	67
FIGURA 42. MÓDULO BLUETOOTH LED ROJO PARPADEANDO.....	68
FIGURA 43. RESPUESTA ANTE COMANDO AT	70
FIGURA 44. SELECCIÓN DEL PUERTO SERIE	71



David Vara Rodríguez

FIGURA 45. REGLA TRAPEZOIDAL.....	75
FIGURA 46. APROXIMACIÓN NUMÉRICA DE UNA FUNCIÓN F(X)	76
FIGURA 47. ERROR DEBIDO A LA ORIENTACIÓN	79
FIGURA 48. EFECTO DE LA INCLINACIÓN EN LOS DATOS DE SALIDA	79
FIGURA 49a. ERROR DE OFFSET.....	81
FIGURA 49b. ERROR DE GANANCIA.....	81
FIGURA 50a. ERROR DE NO LINEALIDAD INTEGRAL.....	82
FIGURA 50b. ERROR DE NO LINEALIDAD DIFERENCIAL.....	82
FIGURA 51. ACELERACIÓN MUESTREADA	85
FIGURA 52. ERRORES GENERADOS DURANTE LA INTEGRACIÓN	86
FIGURA 53. LOS ERRORES DE INTEGRACIÓN SE REDUCEN MEDIANTE EL MÉTODO TRAPEZOIDAL.....	86
FIGURA 54. CURVA DE ACELERACIÓN IDEAL.....	87
FIGURA 55. RECURSIVIDAD EN EL FILTRO DE KALMAN.....	88
FIGURA 56. CICLO DE KALMAN.....	89
FIGURA 57. CUADRO RESUMEN DEL FILTRO DE KALMAN	92
FIGURA 58. DIAGRAMA DE FLUJO	95
FIGURA 59. EJEMPLO DE EJECUCIÓN DE LA CALIBRACIÓN	98
FIGURA 60. VENTANA DE FILTRADO.....	99
FIGURA 61. RESULTADOS DE LA PRUEBA 1 PARA R=0.05 Y Q=0.001.....	104
FIGURA 62. RESULTADOS DE LA PRUEBA 2 PARA R=0.07 Y Q=0.001.....	105
FIGURA 63. ACELERACIÓN OBTENIDA APLICANDO EL FILTRO DE KALMAN.....	108
FIGURA 64. POSICIÓN OBTENIDA CON EL FILTRO DE KALMAN	108
FIGURA 65. ACELERACIÓN OBTENIDA CON EL FILTRO DE LA MEDIA MÓVIL... ..	109
FIGURA 66. POSICIÓN OBTENIDA CON EL FILTRO DE LA MEDIA MÓVIL.....	109



David Vara Rodríguez

FIGURA 67. GRÁFICOS DE ACELERACIÓN, VELOCIDAD Y POSICIÓN	112
FIGURA 68. ACELERACIÓN PRUEBA 35 CM LINEAL	116
FIGURA 69. VELOCIDAD PRUEBA 35 CM LINEAL	117
FIGURA 70. ACELERACIÓN VS VELOCIDAD PRUEBA 35 CM LINEAL	117
FIGURA 71. POSICIÓN PRUEBA 35 CM LINEAL	118
FIGURA 72. ACELERACIÓN PRUEBA 15 CM LINEAL	119
FIGURA 73. VELOCIDAD PRUEBA 15 CM LINEAL	120
FIGURA 74. ACELERACIÓN VS VELOCIDAD PRUEBA 15 CM LINEAL	120
FIGURA 75. POSICIÓN PRUEBA 15 CM LINEAL	121
FIGURA 76. ACELERACIÓN PRUEBA 10 CM LINEAL	122
FIGURA 77. VELOCIDAD PRUEBA 10 CM LINEAL	123
FIGURA 78. ACELERACIÓN VS VELOCIDAD PRUEBA 10 CM LINEAL	123
FIGURA 79. POSICIÓN PRUEBA 10 CM LINEAL	124
FIGURA 80. ACELERACIÓN PRUEBA 10 CM LINEAL LENTA.....	125
FIGURA 81. ACELERACIÓN VS VELOCIDAD PRUEBA 10 CM LINEAL	126
FIGURA 82. VELOCIDAD PRUEBA 10 CM LINEAL LENTA.....	126
FIGURA 83. POSICIÓN PRUEBA 10 CM LINEAL LENTA.....	127
FIGURA 84. ACELERACIÓN PRUEBA 5 CM LINEAL	128
FIGURA 85. ACELERACIÓN VS VELOCIDAD PRUEBA 5CM LINEAL	129
FIGURA 86. VELOCIDAD PRUEBA 5 CM LINEAL	129
FIGURA 87. POSICIÓN PRUEBA 5 CM LINEAL	130
FIGURA 88. ACELERACIÓN EJE X PRUEBA DIAGONAL 10CM.....	132
FIGURA 89. ACELERACIÓN EJE Y PRUEBA DIAGONAL 10CM.....	133
FIGURA 90. VELOCIDAD PRUEBA DIAGONAL 10CM.....	133
FIGURA 91. POSICIÓN PRUEBA DIAGONAL 10CM.....	134



David Vara Rodríguez

FIGURA 92. ACELERACIÓN EJE X PRUEBA RECTÁNGULO 14X12 CM.....	135
FIGURA 93. ACELERACIÓN EJE Y PRUEBA RECTÁNGULO 14X12CM.....	136
FIGURA 94. VELOCIDAD PRUEBA RECTÁNGULO 14X12CM.....	136
FIGURA 95. POSICIÓN PRUEBA RECTÁNGULO 14X12CM.....	137
FIGURA 96. ACELERACIÓN EJE X PRUEBA RECTÁNGULO V. LENTA.....	138
FIGURA 97. ACELERACIÓN EJE Y PRUEBA RECTÁNGULO V. LENTA.....	139
FIGURA 98. VELOCIDAD PRUEBA RECTÁNGULO V.LENTA.....	139
FIGURA 99. POSICIÓN PRUEBA RECTÁNGULO V.LENTA.....	140
FIGURA 100. ACELERACIÓN EJE X PRUEBA CÍRCULO	141
FIGURA 101. ACELERACIÓN EJE Y PRUEBA CÍRCULO.....	142
FIGURA 102. VELOCIDAD PRUEBA CÍRCULO.....	142
FIGURA 103. POSICIÓN PRUEBA CÍRCULO	143
FIGURA 104. ROBOT ABB IRB 120 CON EL DISPOSITIVO INSTALADO	144
FIGURA 105. DISPOSITIVO CONSTRUIDO EN EL TCP DEL ROBOT.....	144
FIGURA 106. FLEXPENDANT	146
FIGURA 107. ACELERACIÓN EJE X PRUEBA PARALEPÍPEDO	148
FIGURA 108. ACELERACIÓN EJE Y PRUEBA PARALEPÍPEDO	149
FIGURA 109. ACELERACIÓN EJE Z PRUEBA PARALEPÍPEDO	149
FIGURA 110. VELOCIDAD PRUEBA PARALEPÍPEDO	150
FIGURA 111. POSICIÓN PRUEBA PARALEPÍPEDO	150



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Índice de tablas

TABLA 1. SISTEMAS DE SEGUIMIENTO EN CIRUGÍA LAPAROSCÓPICA.....	49
TABLA 2. PINES DE CONEXIÓN IMU MPU6050 – ARDUINO UNO	57
TABLA 3. PINES DE CONEXIÓN MÓDULO BLUETOOTH – ARDUINO.....	60
TABLA 4. PINES DE CONEXIÓN CONFIGURACIÓN BLUETOOTH – ARDUINO	69
TABLA 5. COMANDOS AT.....	69
TABLA 6. ERROR EN LAS REGLAS DE INTEGRACIÓN	77
TABLA 7. FILTRO DE KALMAN VS FILTRO DE LA MEDIA MÓVIL.....	110
TABLA 8. CARACTERÍSTICAS PRUEBA 35 CM LINEAL	116
TABLA 9. RESULTADOS PRUEBA 35 CM LINEAL.....	118
TABLA 10. CARACTERÍSTICAS PRUEBA 15 CM LINEAL	119
TABLA 11. RESULTADOS PRUEBA 15 CM LINEAL.....	121
TABLA 12. CARACTERÍSTICAS PRUEBA 10 CM LINEAL.....	122
TABLA 13. RESULTADOS PRUEBA 10 CM LINEAL.....	124
TABLA 14. CARACTERÍSTICAS 12 CM LINEAL LENTA.....	125
TABLA 15. RESULTADOS PRUEBA 10 CM LINEAL LENTA	127
TABLA 16. CARACTERÍSTICAS PRUEBA 5 CM LINEAL	128
TABLA 17. RESULTADOS PRUEBA 5 CM LINEAL.....	130
TABLA 18. CARACTERÍSTICAS PRUEBA DIAGONAL 10 CM.....	132
TABLA 19. RESULTADOS PRUEBA DIAGONAL 10 CM	134
TABLA 20. CARACTERÍSTICAS PRUEBA RECTÁNGULO 14X12 CM.....	135
TABLA 21. RESULTADOS PRUEBA RECTÁNGULO 14X12 CM	137
TABLA 22. CARACTERÍSTICAS PRUEBA RECTÁNGULO V.LENTA	138



David Vara Rodríguez

TABLA 23. RESULTADOS PRUEBA RECTÁNGULO V.LENTA	140
TABLA 24. CARACTERÍSTICAS PRUEBA CÍRCULO	141
TABLA 25. RESULTADOS PRUEBA CÍRCULO	143
TABLA 26. CARACTERÍSTICAS PRUEBA PARALEPÍPEDO	148
TABLA 27. RESULTADOS PRUEBA PARALEPÍPEDO.....	151



Capítulo 1. Introducción

1. Introducción al capítulo

El presente proyecto, “Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica” se ha desarrollado dentro del marco de Trabajo Fin de Grado para optar al título de Graduado en Ingeniería Electrónica Industrial y Automática.

El periodo de tiempo en el que se ha desarrollado el proyecto ha sido de tres meses, desde marzo hasta junio de 2014. El director del presente proyecto ha sido el Profesor Don Juan Carlos Fraile Marinero, profesor titular del departamento de Ingeniería de Sistemas y Automática de la Escuela de Ingenierías Industriales de la Universidad de Valladolid.

En este capítulo se presenta una descripción general del proyecto realizado, enumerando las tareas que se han llevado a cabo, cada una de las cuales será analizada en profundidad, con un capítulo dedicado.

2. Descripción general

El presente proyecto estudia la posibilidad de determinar la posición y orientación de herramientas quirúrgicas mediante sensores de bajo coste.

Para dicho cometido se ha utilizado un sensor IMU (unidad de medida inercial) MPU6050, compuesto por un giroscopio de 3 ejes y un acelerómetro de 3 ejes, un microcontrolador Arduino UNO y un módulo HC-06 para la comunicación bluetooth con el procesador (PC).

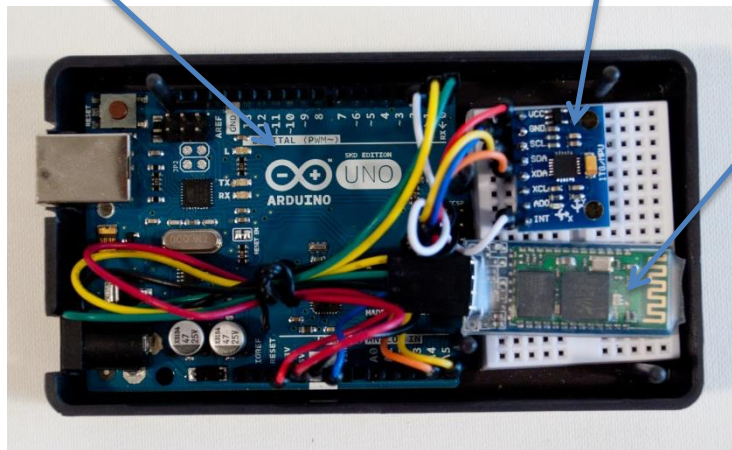
La mayoría de los métodos de localización existentes tienen un coste muy elevado y se basan en el empleo de cámaras por visión artificial. Esto plantea, además del alto importe, una gran complicación: la pérdida de la señal de rastreo si por cualquier circunstancia se pierde el contacto visual, como por ejemplo el sangrado de un paciente o cualquier objeto o persona que dificulte el campo de visión.

David Vara Rodríguez

El trabajo actual plantea una alternativa a estas dos problemáticas: la utilización de un sistema inalámbrico mediante sensores digitales y por un precio inferior a 25 euros.

Arduino UNO

IMU MPU6050



Módulo Bluetooth HC-06

Figura 1. Vista interior del dispositivo construido



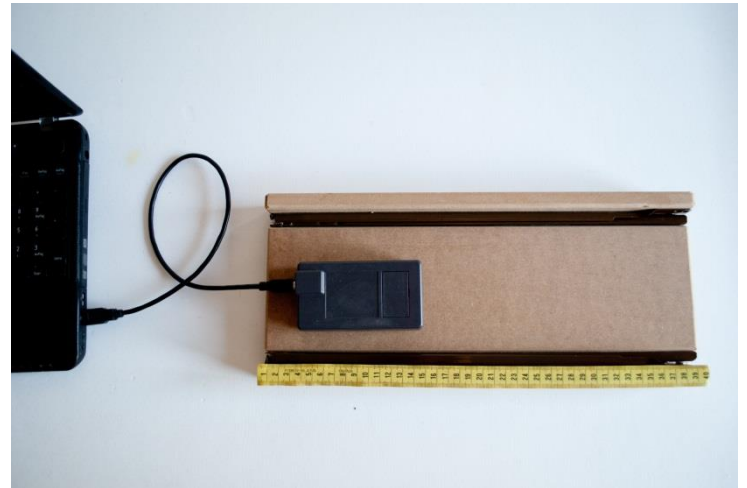
Figura 2. Vista exterior del dispositivo construido

David Vara Rodríguez

Se han realizado distintas pruebas en una, dos y tres dimensiones. Mientras que los ensayos en 2D y 3D se realizaron de forma manual, para las pruebas en una dimensión se construyó el siguiente mecanismo mediante guías de riel y graduada en milímetros:



a)



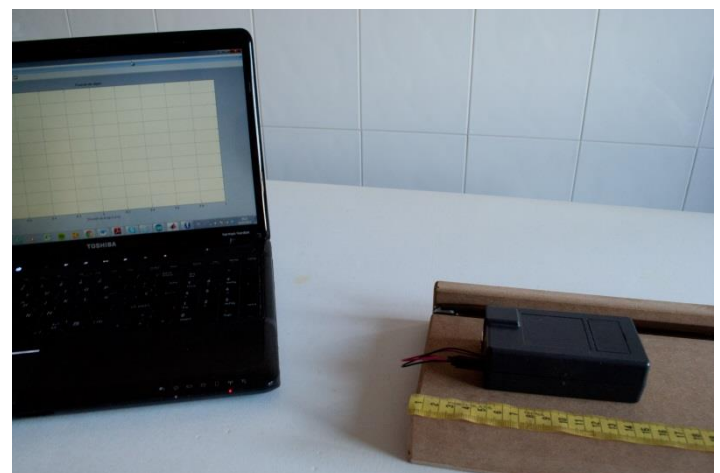
b)

Figura 3. a) Mecanismo de guía b) Mecanismo de guía + dispositivo de localización

El entorno de trabajo está compuesto por el dispositivo de localización construido, el mecanismo de guía y un PC en el que se procesa la señal y se muestran los datos:



a)



b)

Figura 4. Entorno de trabajo con a) conexión USB y b) conexión inalámbrica bluetooth

David Vara Rodríguez

Así mismo se realizó una prueba con el robot ABB IRB 120.

El entorno de trabajo de dicho ensayo estaba compuesto por dicho robot, el dispositivo de localización construido, la consola FlexPendant con la que se controlaba el movimiento del robot y el PC donde se recibía la señal del sensor y se procesaban los datos:



Figura 5. Entorno de trabajo prueba robot



Figura 6. Detalle del dispositivo construido en el TCP del robot



Con esta técnica planteada, se han realizado las siguientes tareas que componen el presente proyecto:

1. Calibración del sensor

En primer lugar se lleva a cabo un sistema para calcular los offsets del acelerómetro y giroscopio a fin de obtener unos resultados lo más exactos posibles.

2. Filtrado de la señal

Se evalúan diversas técnicas de filtrado, tanto de forma teórica como práctica, seleccionándose finalmente uno para la eliminación del ruido de nuestro sensor.

3. Integración numérica

Se estudian diversos métodos de aproximación de integrales de forma que a partir de los datos del acelerómetro, se pueda hallar la posición de nuestro dispositivo.

4. Localización unidimensional

Se comprueban los resultados de las medidas tomadas a lo largo de un único eje.

5. Localización bidimensional

Se evalúa la respuesta del sistema ante movimientos en dos dimensiones y la descripción de trayectorias básicas: rectángulo, circunferencia.

6. Localización tridimensional

Se realizan movimientos en tres dimensiones y se observan las mediciones obtenidas.



3. Contenidos de la memoria

La parte de este documento tomada como memoria se ha dividido en seis capítulos. A modo de índice desarrollado, se resume a continuación el contenido más relevante de cada uno de ellos:

CAPÍTULO 1. INTRODUCCIÓN

Descripción de los aspectos principales del proyecto, motivación del mismo y objetivos planteados inicialmente para su realización, así como una breve descripción del contenido de la memoria.

CAPÍTULO 2. ESTADO DEL ARTE

Se realiza un análisis sobre distintos métodos existentes para determinar la posición y orientación de instrumentos así como los procedimientos utilizados en cirugía laparoscópica.

CAPÍTULO 3. DESCRIPCIÓN DE LOS ELEMENTOS UTILIZADOS

Se analizan los componentes hardware utilizados en este proyecto, realizando una breve explicación sobre las características más importantes de cada uno de ellos y su correspondiente conexionado.

Así mismo se estudia la comunicación entre el microcontrolador y el PC tanto de forma inalámbrica como mediante cable USB.

CAPÍTULO 4. LOCALIZACIÓN DE LA POSICIÓN

En este capítulo exploramos los métodos para hallar la posición de un objeto a partir de los datos proporcionados por el acelerómetro, poniendo particular atención en la evaluación de los errores y la precisión obtenida.



David Vara Rodríguez

Analizamos diferentes soluciones basadas en la integración numérica y el filtro de Kalman.

CAPÍTULO 5. PRUEBAS REALIZADAS

Se presentan diversos ensayos realizados con el dispositivo una vez ha sido construido y programado.

Este capítulo se encuentra subdividido en tres apartados en función de las pruebas realizados en una, dos y tres dimensiones.

CAPÍTULO 6. CONCLUSIONES Y LÍNEAS FUTURAS

Se realiza una reflexión sobre los resultados obtenidos, comprobando la exactitud y precisión de los mismos.

Así mismo se muestran, tanto las principales conclusiones extraídas, como las posibles futuras líneas de actuación a seguir en trabajos o proyectos posteriores.

BIBLIOGRAFÍA

Enumeración de aquellos libros y proyectos, manuales y páginas web que se han empleado a lo largo del proyecto para documentarse acerca del mismo y obtener la información necesaria para poder completarlo satisfactoriamente.

ANEXOS

Se encuentran en el CD adjunto de este proyecto. En ellos se presentan los códigos fuente de los programas realizados tanto en Arduino como en Matlab R2010a, así como los datasheet de los componentes utilizados en la construcción del dispositivo de localización.



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Capítulo 2. Estado del arte

1. Introducción al capítulo

En este apartado se presenta una introducción a los sensores de posición y los distintos tipos que podemos encontrar actualmente. Así mismo se analiza en profundidad el utilizado en este proyecto (IMU).

Finalmente se realiza un repaso sobre las técnicas de localización usadas actualmente en cirugía laparoscópica.

2. Qué es un sensor de posición

Los sensores son los elementos principales de cualquier sistema de medición. Recogen los datos del sistema físico (entrada) y proporcionan una señal eléctrica como respuesta a los mismos (salida).

Los sensores de posición miden la distancia recorrida por el objeto de estudio a partir de una posición de referencia. Estos dispositivos juegan un papel muy importante en diversos campos como en los sistemas de navegación aérea, vehículos terrestres, máquinas de inyección, equipos quirúrgicos, etc.

3. Tipos de sensores de posición

Los sensores de posición utilizan distintos métodos para detectar el desplazamiento de un cuerpo. Dependiendo de la técnica utilizada, se pueden clasificar de la siguiente manera [1]:

1. Sensores de resistencia variable o potenciómetros
2. Sensores capacitivos
3. Transformador diferencial de variación lineal (LVDT)
4. Sensores magnetorresistivos
5. Sensores de ultrasonidos
6. Sensores ópticos
7. Sensores inerciales (IMU)

3.1 SENSORES DE RESISTENCIA VARIABLE O POTENCIÓMETROS

El elemento sensor es simplemente una resistencia sobre una pista conductora. Dicha resistencia es proporcional a la posición del cuerpo cuyo desplazamiento se va a medir.

Se une el objeto en cuestión a un extremo de la pista y se aplica una tensión constante. En función del voltaje de salida obtenido se calcula la distancia recorrida por el objeto.

La pista conductora puede ser lineal o angular dependiendo de los requisitos. Éstas pueden estar hechas de carbono, alambre resistivo o material piezoeléctrico.

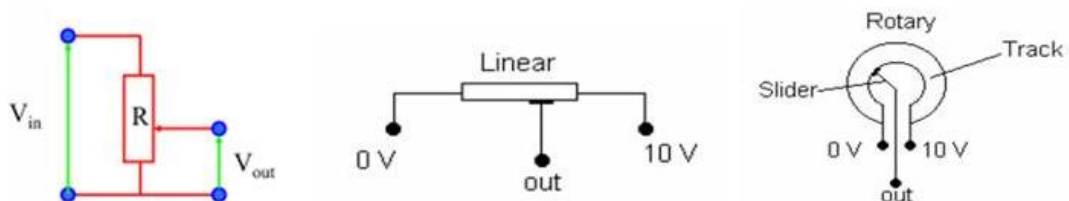


Figura 7. Funcionamiento de los sensores resistivos

3.2 SENSORES CAPACITIVOS

La capacitancia entre dos placas depende de la permitividad del dieléctrico, la zona de solapamiento y la distancia entre las mismas. Cualquiera de estos tres parámetros puede variarse con el fin de diseñar un sensor capacitivo.

Se puede utilizar cualquiera de las dos siguientes configuraciones:

3.2.1 MODIFICACIÓN DE LA CONSTANTE DIELECTRICA

El cuerpo cuyo desplazamiento se desea medir se conecta al material dieléctrico entre las placas. Como el objeto se mueve, la constante dieléctrica efectiva varía, lo cual produce un cambio en la capacidad del condensador y por tanto dicha capacitancia se convierte en una función de la posición del cuerpo.



Figura 8. Desplazamiento en función de la constante dieléctrica

Este método se emplea de forma común en los sensores de nivel, en los que se utilizan cubos concéntricos y el líquido actúa como dieléctrico. La variación de la capacitancia con el nivel de fluido es lineal.

3.2.2 MODIFICACIÓN DE LA ZONA DE SOLAPAMIENTO

En esta configuración, el cuerpo en cuestión se conecta a una de las placas, mientras la otra permanece fija. Con el movimiento del mismo, la zona de solapamiento varía lo cual produce una variación en la capacidad del sensor y por tanto, la capacitancia se convierte en una función de la posición del cuerpo.

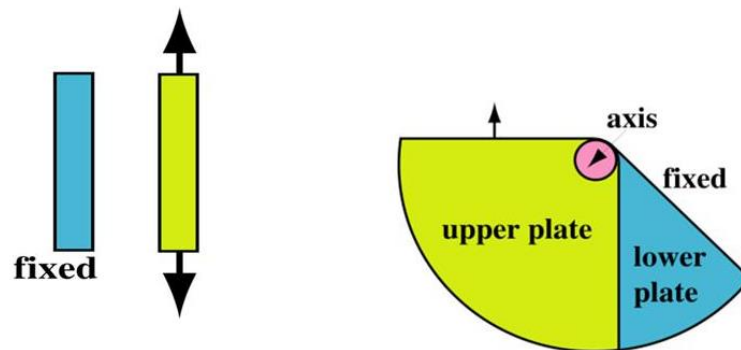


Figura 9. Desplazamiento en función de la zona de solapamiento

Este principio se puede emplear tanto en movimientos lineales como angulares.

3.3 TRANSFORMADOR DIFERENCIAL DE VARIACIÓN LINEAL (LVDT)

Este dispositivo es un transductor electromecánico que convierte el movimiento rectilíneo de un objeto en una señal eléctrica. Se utiliza para la medición de movimientos que van desde micras hasta varias centímetros.

El LVDT consta de un bobinado primario en la parte central y dos secundarios simétricamente espaciados. El elemento móvil de este transformador se llama núcleo y compuesto por material magnético altamente permeable acoplado mecánicamente al objeto cuyo desplazamiento se va a medir.

David Vara Rodríguez

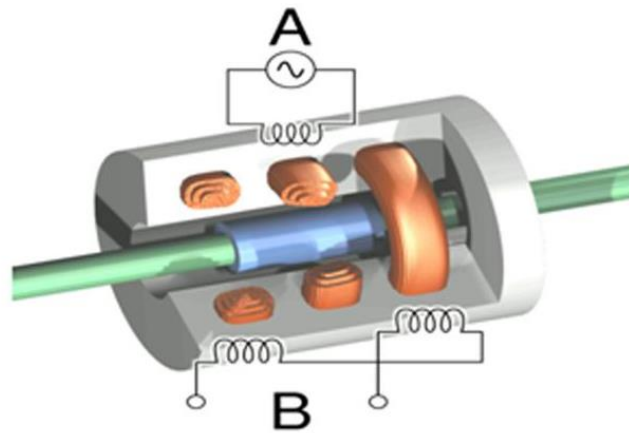


Figura 10. Sensor LVDT

Cuando por el devanado primario circula una corriente alterna, se induce una tensión en el secundario. La salida del LVDT es la tensión diferencial entre los dos devanados secundarios; la cual varía con la posición del núcleo. A menudo, la tensión de salida de CA diferencial se convierte en voltaje de CC para su uso en sistemas de medición.

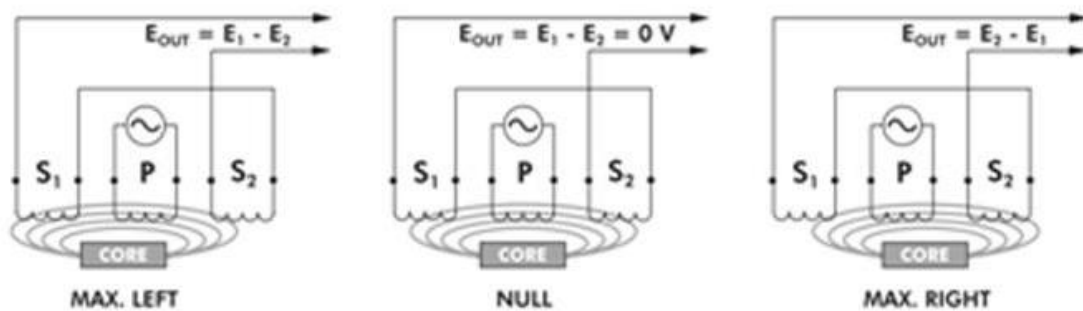


Figura 11. Devanados

3.4 SENSORES MAGNETORRESISTIVOS

Están formados principalmente por un imán de posición, una guía de ondas, un detector de humedad, y un módulo electrónico. Además, una varilla de detección se monta a lo largo del eje de movimiento. El imán de posición es un imán permanente en forma de anillo fijado al objeto cuyo desplazamiento se desea monitorizar.

El sensor calcula el recorrido midiendo la distancia entre el imán y el extremo de la cabeza de la varilla de detección.

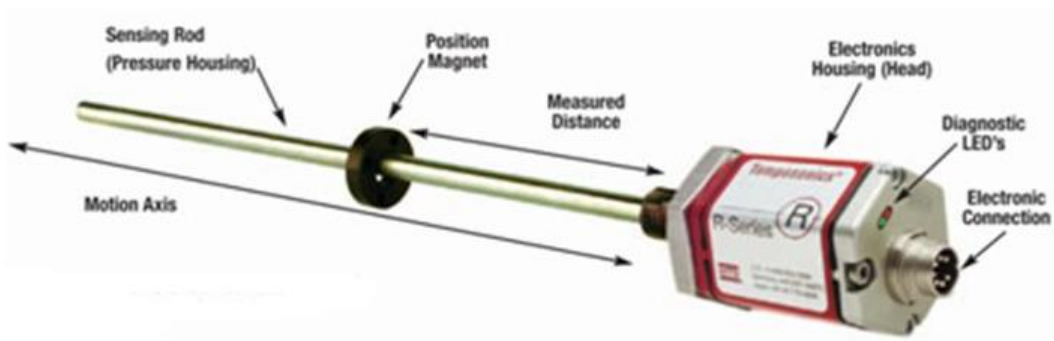


Figura 12. Sensor magnetorresistivo

3.5 SENSORES DE ULTRASONIDOS

Estos dispositivos constan de un emisor y un receptor. Emiten una señal y reciben el eco de la misma midiendo el tiempo entre ambas. Como la velocidad de la onda es conocida, resulta trivial calcular la distancia a la que se encuentra el objeto.



Figura 13. Sensor de ultrasonidos

3.6 SENSORES ÓPTICOS

Los sensores ópticos tienen dos posibles mecanismos:

En el primero, la luz se transmite desde un extremo hasta otro y se monitoriza el cambio en una de sus características: intensidad, longitud de onda, polarización, etc. Se utilizan en codificadores ópticos para proporcionar información acerca de la posición de los actuadores.

Los codificadores ópticos consisten en un disco de vidrio o plástico que gira entre una fuente de luz (LED) y un receptor (fotodetector). El disco está codificado con luz alterna y sectores oscuros de manera que los pulsos son generados a medida que el disco gira. Basándose en el recuento de los impulsos, y la velocidad del disco, la posición angular se calcula. Para identificar la dirección del movimiento, se utilizan dos fotodetectores. Los codificadores ópticos absolutos tienen un código único que puede ser detectado para cada posición angular.

En el segundo tipo, la luz transmitida se refleja en el objeto y ésta es la que se monitoriza.

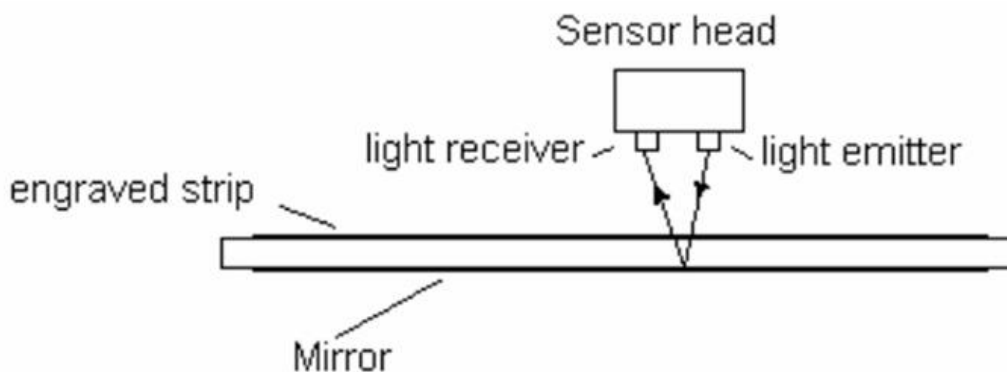


Figura 14. Sensor óptico

3.7 SENSORES INERCIALES (IMU)

Una unidad de medida inercial o IMU es un componente electrónico basado en sensores de aceleración y velocidad angular (acelerómetros y giróscopos respectivamente) la cual nos reporta el movimiento y orientación (Figura 15) que sufre dicha unidad. Es el componente principal de sistemas de guía inercial usados en vehículos aéreos, espaciales, marinos y aplicaciones robóticas.

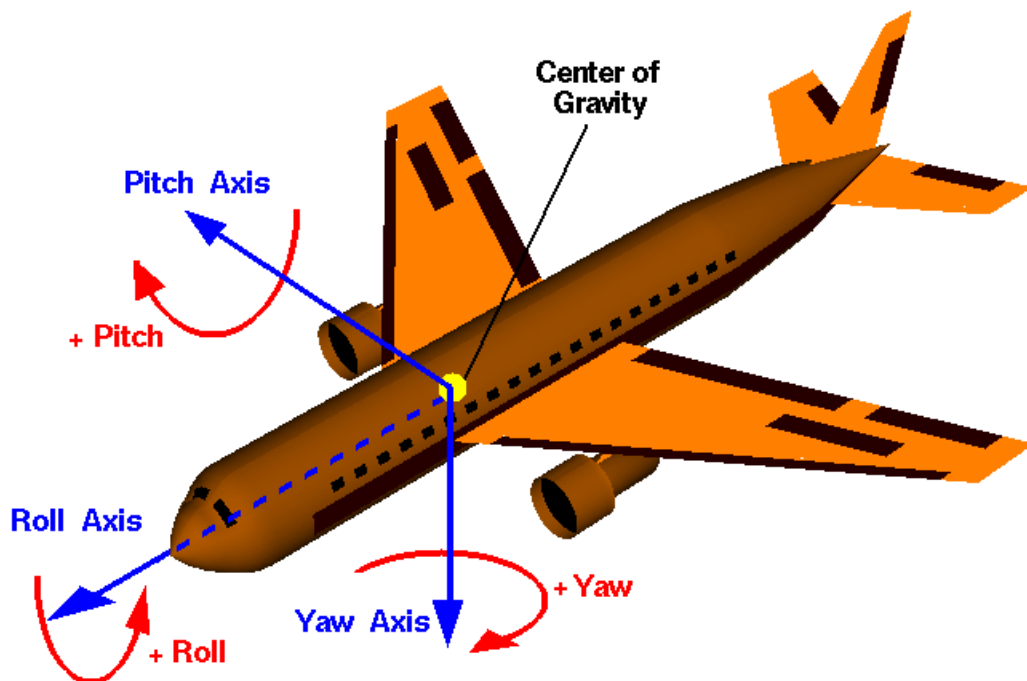


Figura 15. Orientación proporcionada por una IMU.



David Vara Rodríguez

3.7.1. COMPONENTES DE UNA IMU

Cualquier unidad de medida inercial está compuesta como mínimo por un acelerómetro y un giróscopo para captar una aceleración y una velocidad angular en concreto. Generalmente, es interesante que las IMUs capten la aceleración y la velocidad angular en los tres ejes de coordenadas para conocer el movimiento exacto del componente.

También podemos encontrar unidades que incorporan un microprocesador que se encarga de recoger los datos de dichos sensores y enviarlos de forma ordenada al usuario mediante el protocolo de comunicación incorporado en la IMU.

A continuación se detallarán los componentes nombrados:

- **Acelerómetro:** instrumento capaz de medir aceleración en uno, dos o tres ejes.

Existen varios tipos de acelerómetros, dependiendo de su fabricación y funcionamiento. Las IMUs incorporan acelerómetros integrados en silicio, utilizando la tecnología llamada MEMS⁶, debido a la necesidad de reducir el tamaño total de la unidad. La mayoría de éstos son capacitivos, y calculan la aceleración mediante el voltaje obtenido entre dos placas una de las cuales varía su posición dependiendo del movimiento del acelerómetro. Se caracterizan por ser muy precisos en situaciones estables y tener un gran error en situaciones vibratorias o movimientos muy inestables.

- **Giróscopo:** dispositivo que mide la orientación, basándose en los principios de la conservación del momento angular.

Las unidades de medida inercial utilizan giróscopos MEMS, es decir, integrados y de tamaño reducido. La salida de dicho sensor es un voltaje, la variación del cual nos indica en grados por segundo ($V/^\circ/s$) la velocidad angular sufrida por el sensor. Se caracterizan por tener un error constante y lineal llamado bias el cual debemos tener en cuenta.

- **Microprocesador:** algunas unidades de medida inercial, como ya hemos comentado, incorporan un microprocesador. Éste es programable, pero su principal función es recoger los datos entregados por los sensores, procesarlos según desee el usuario y enviarlos. En el microprocesador se define la frecuencia de trabajo de la unidad, que será el tiempo comprendido desde que recoge el dato del primer sensor hasta que envía al usuario el dato procesado del último sensor.



David Vara Rodríguez

La mayoría de los microprocesadores incorporan un conversor analógico-digital, para así convertir el voltaje dado por el sensor en una muestra.

- **Protocolo de comunicación:** los protocolos de comunicación alámbricos típicos en las IMUs son el UART, el RS-232 o el USB7. Algunas unidades incluyen protocolos inalámbricos, siendo los más utilizados ZigBee y Bluetooth.
- **Magnetómetro:** algunas unidades de medida inercial incluyen también sensores magnetómetros. Estos dispositivos miden la fuerza y/o dirección de los campos magnéticos que los afectan respecto al campo magnético terrestre. Aunque cabe la posibilidad de que se vean afectados por variación de otros campos magnéticos en algunas zonas.

3.7.2. CARACTERIZACIÓN DE UNA IMU

Los diferentes tipos de unidades de medida inercial que podemos encontrar en el mercado se caracterizan generalmente por el tipo de sensores de que están compuestas.

Estos sensores, la frecuencia de trabajo, que puede interesar mayor o menor dependiendo de la aplicación para la que esté destinada la unidad, y el protocolo de comunicación son los que definen una IMU mejor que otra.

Los sensores de que se componen las unidades de medida inercial se definen principalmente por su rango de trabajo (máxima medida que soporta el sensor), su sensibilidad (relación entre la variación de la magnitud de salida y la de entrada) y su ancho de banda de respuesta (frecuencia de funcionamiento del sensor). Para el caso de los acelerómetros, el rango de trabajo se mide con la gravedad estándar, aceleración de $g = 9.80665 \text{ m/s}^2$. Podemos encontrar IMUs con un rango de 1.5g, 2g, 4g, etc. Dependiendo de este rango, obtenemos una sensibilidad u otra, ya que el nivel de voltaje máximo que podemos obtener viene definido por la alimentación del sensor.

Entonces, para mayor rango, menor sensibilidad. En los giróscopos, el rango de medida se mide en $^\circ/\text{s}$, ya que obtenemos una velocidad angular, y la sensibilidad en $\text{mV}/^\circ/\text{s}$. Valores típicos de rango que podemos encontrar son $\pm 200 \text{ }^\circ/\text{s}$, $\pm 300 \text{ }^\circ/\text{s}$, $\pm 500 \text{ }^\circ/\text{s}$, etc.

En el caso de que la IMU tenga magnetómetros, el rango se mide en gauss = 1 maxwell / cm^2 .



David Vara Rodríguez

La frecuencia de trabajo de la unidad de medida inercial viene definida por el reloj que use el microprocesador para enviar los datos al usuario y por la frecuencia de conversión del conversor analógico-digital. También influye la velocidad de transmisión, que depende del protocolo que utilicemos y de cómo lo definamos. Ésta velocidad se mide en baudios o bits por segundo y son valores típicos 38400 bps, 57600 bps, 115200 bps, etc.



4. Métodos de localización en cirugía laparoscópica

Actualmente se comercializan varios sistemas para la localización de instrumentos en cirugía laparoscópica. Estos sistemas son diferentes en función del ambiente en el que vayan a ser utilizados: módulos de entrenamiento, realidad virtual o salas de operaciones.

El objetivo de este apartado es dar una visión general y estructurada sobre los sistemas de seguimiento actuales.

4.1 ASPECTOS GENERALES DE LOS SISTEMAS DE LOCALIZACIÓN

Los sistemas de localización de instrumentos quirúrgicos constituyen una interfaz o sistema de comunicación entre el cirujano y el ordenador.

En estos sistemas podemos distinguir tres claros componentes: una fuente generadora de señal, un sensor que recibe la señal y un sistema de adquisición de datos, que procesa la información obtenida y se comunica con el ordenador.

Estos sistemas pueden ser:

- Activos: el sensor que mide el movimiento está unido al propio instrumento quirúrgico .
- Pasivo: el sensor localiza, desde una cierta distancia, unas marcas en el instrumento quirúrgico o bien unas señales transmitidas por este.

Los sensores utilizados para el rastreo del instrumental quirúrgico pueden ser mecánicos, ópticos, acústicos o electromagnéticos:

- Mecánicos: localizan la posición en función de los ángulos formados por cada articulación.
- Ópticos: pueden funcionar de dos formas:



David Vara Rodríguez

- 1) Una o varias cámaras se conectan al objeto junto con unos LED de tal forma que la zona de interés permanezca iluminada.
 - 2) Las cámaras se sitúan en unos puntos fijos mientras que los LED se sitúan en el objeto.
- Acústicos: utilizan transmisores y receptores de ondas de sonido en la banda de 20 kHz. Midiendo el tiempo entre la emisión y recepción de la señal o bien comparando la fase de la señal con una de referencia, se puede calcular la posición del instrumento.
 - Electromagnéticos: se basan en el movimiento de pequeños sensores colocados a lo largo del instrumento quirúrgico. Cada uno consta de tres bobinas de un terminal colocadas ortogonalmente a modo de ejes y una segunda bobina de 3 terminales, la cual genera un campo magnético de baja frecuencia.

4.2 SISTEMAS PASIVOS DE LOCALIZACIÓN

ProMIS: tres cámaras distintas capturan los movimientos del instrumento laparoscópico desde tres ángulos distintos lo cual permite medir los movimientos en los tres ejes (x,y,z). El instrumento quirúrgico posee unas marcas de color amarillo para su seguimiento.

Este sistema solo puede usarse en módulos de entrenamiento donde las cámaras se sitúan en el maniquí a modo de paciente. Así pues, el sistema de seguimiento del instrumental no es móvil. Sin embargo proporciona una realimentación con la que el cirujano puede sentir la fuerza ejercida.



Figura 16 . Simulador ProMIS de Haptica



David Vara Rodríguez

Zebri: utiliza un sistema de medida de ultrasonidos en 3D para el seguimiento del instrumental. El sistema calcula la posición espacial (coordenadas x,y,z y rotación) del instrumento laparoscópico en función de la posición relativa de los transmisores respecto a un sistema fijo formado por tres micrófonos.



Figura 17. Sistema Zebri de ultrasonidos adaptado para la cirugía laparoscópica.

4.3 SISTEMAS ACTIVOS DE LOCALIZACIÓN

Laparoscopic Surgical Workstation, Virtual Laparoscopic Interface y Laparoscopic Impulse Engine son las interfaces más conocidos para la simulación virtual de operaciones quirúrgicas en laparoscopia.

Laparoscopic Surgical Workstation y Virtual Laparoscopic Interface presentan dos herramientas quirúrgicas totalmente equipadas. Los movimientos de estas herramientas son medidos mediante cuatro transductores electromecánicos situados en un mecanismo gimbal.

Laparoscopic Surgical Workstation ofrece una realimentación con la que el cirujano puede sentir la fuerza aplicada. Sus dimensiones son elevadas y por tanto no



David Vara Rodríguez

es móvil. Por su parte la Virtual Laparoscopic Interface tiene dimensiones menores pero no ofrece realimentación y la precisión de sus sensores es menor.



Figura 18. Laparoscopic Surgical Workstation (izq) y Virtual Laparoscopic Interface (der)

El Laparoscopic Impulse Engine utiliza unos servo-motores como actuadores. Permite movimientos en los cuatro grados de libertad. Se pueden utilizar diversas herramientas quirúrgicas en este dispositivo.



Figura 19 . Laparoscopic Impulse Engine

CELTS: (Computer Enhanced Laparoscopic Training System) es un simulador desarrollado por el CIMIT en Boston (USA). Se trata de una modificación del Virtual Laparoscopic Interface en el que los ejes principales han sido sustituidos por un sistema que permite el uso de instrumentos reales utilizados en cirugía laparoscopia



David Vara Rodríguez

así como su posterior seguimiento. Es un sistema móvil con una realimentación háptica.



Figura 20. Dispositivo CELTS

ADEPT: (Advanced Dundee Endoscopic Psychomotor Tester) consiste en un mecanismo gimbal en el que podemos posicionar distintos instrumentos laparoscópicos. El sistema realiza el seguimiento del instrumento en 3D mediante unos potenciómetros situados en el mecanismo gimbal.

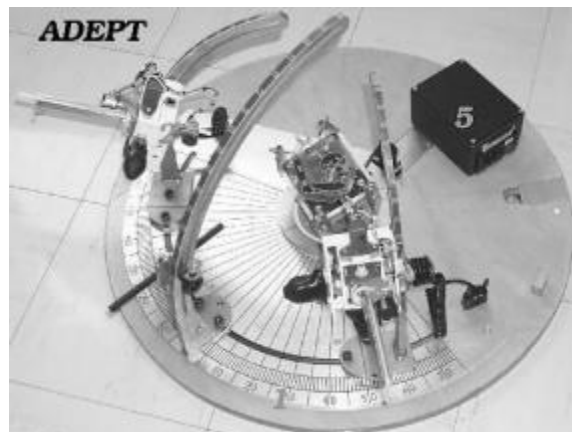


Figura 21. Sistema ADEPT

Simendo: es un simulador de realidad virtual para cirugía mínimamente invasiva. El sistema de seguimiento del instrumental consiste en un mecanismo gimbal. La traslación y rotación son medidas por un sensor óptico mientras que los movimientos de cabeceo y guiñada son medidos por encoders ópticos. La combinación de ambos permite medir los movimientos en los cuatro grados de libertad. No puede utilizarse en operaciones reales de cirugía y no proporciona realimentación de fuerza. Es fácilmente transportable y su peso es reducido.



David Vara Rodríguez



Figura 22. Sistema Simendo

BlueDRAGON: el sistema está formado por barras conectadas al instrumento quirúrgico como puede verse en la figura 8. La medida de la posición y orientación se realiza mediante unos potenciómetros situados en las cuatro articulaciones del mecanismo. Es un sistema bastante grande y fijo.

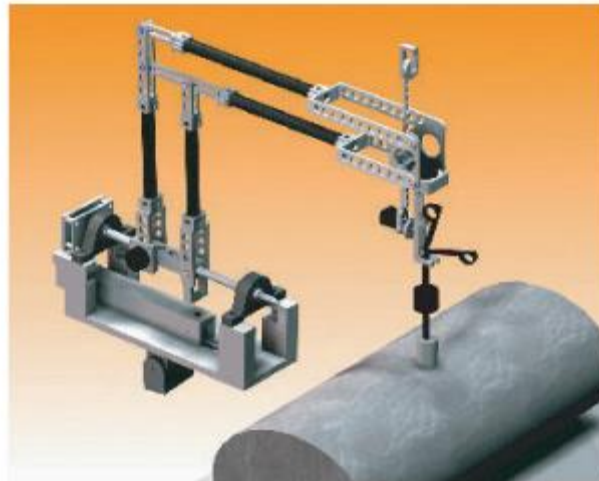


Figura 23. Dispositivo BlueDRAGON

Patriot: es un sistema doble de seguimiento creado por Polhemus. Este sistema consiste en un transmisor electromagnético y un receptor. El transmisor sirve al sistema como marco de referencia para realizar las medidas; el receptor detecta el campo magnético emitido por el transmisor. Este receptor suele posicionarse en el instrumento laparoscópico.

David Vara Rodríguez



Figura 24. Dispositivo Patriot

TrEndo: es un sistema que consiste en un mecanismo gimbal de dos ejes con tres sensores ópticos. El mecanismo gimbal sirve de guía al instrumento mientras los sensores ópticos miden los movimientos del mismo en los cuatro grados de libertad. Se obtiene una realimentación háptica debido al uso de instrumentos reales utilizados en cirugía laparoscópica.

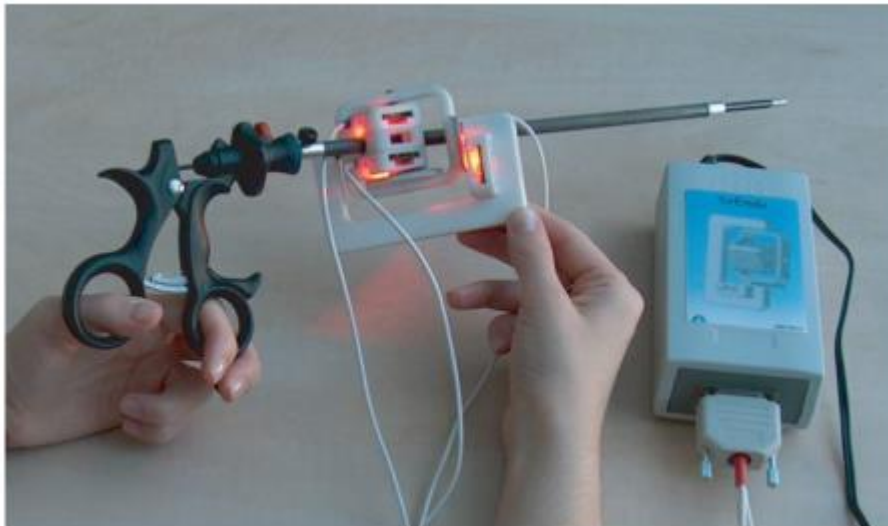


Figura 25. Dispositivo TrEndo



David Vara Rodríguez

Nombre	Sistema	Mecanismo	DOF	Entorno	Portabilidad	Feedback	Precisión	Disponibilidad
ProMIS	P	-	3	Entr, VR	No	Sí		Sí
UMS	P	-	4	Entr, VR, OP	No	Sí		Sí
LSW	A	Gimbal	4	VR	No	Sí		Sí
VLI	A	Gimbal	4	VR	Sí	No		Sí
LIE	A	Joystick	4	VR	Sí	Sí		Sí
CELTS	A	Gimbal	4	Entr, VR	Sí	Sí		No
ADEPT	A	Gimbal	4	Entr		Sí	±0.5mm	Sí
Simendo	A	Gimbal	4	VR	Sí	No		Sí
BlueDRAGON	A	Barras	4	Entr, OP	No	Sí		
Patriot	A		4	Entr, VR	Sí	Sí	2,54mm ; 0,75°	
TrEndo	A	Gimbal	4	Entr, VR	Sí	Sí	>95%	No

Sistema: P – Pasivo, A – Activo.

DOF – Grados de libertad.

Entorno: Entr - módulo entrenamiento, VR – realidad virtual, OP – operación.

UMS – Ultrasound Measurement System; **LSW** – Laparoscopic Surgical Workstation; **VLI** – Virtual Laparoscopic Interface; **LIE** – laparoscopic impulse engine; **ADEPT** – Advanced Dundee Endoscopic Psychomotor Tester; **TrEndo** – Tracking Endoscopy

Tabla 1. Algunos de los principales sistemas de seguimiento utilizados en cirugía laparoscópica



4.4 PROCESAMIENTO DE VÍDEO

Los sistemas basados en sensores ofrecen buenos niveles de precisión, pero son sensibles a distintas fuentes de ruido en el quirófano y contribuyen a la sobrecarga tecnológica del mismo. Una alternativa es analizar la imagen del vídeo endoscópico para llevar a cabo la detección y localización espacial del instrumental. A continuación se presentan algunos estudios sobre este procedimiento:

- *Métodos probabilísticos que usan la diferencia de color entre el tejido anatómico y el instrumental haciendo uso de un filtro bayesiano [2].*

El algoritmo utilizado consta de cuatro pasos: 1) clasificación; según el color para lo cual se emplea un filtro, 2) agrupación y etiquetado; en función si pertenecen a tejido o al instrumento así como su región, 3) análisis del eje; para obtener el centroide y la orientación así como demás información relevante, 4) predicción temporal; se estima la siguiente posición.

Utilizando unos marcadores de color o infrarrojos sobre el instrumental la distinción entre tejido e instrumental puede ser simplificada [3]

Sin embargo estos sistemas presentan el gran inconveniente de que las marcas en el instrumental pueden ser ocultadas por el sangrado u otras causas durante la operación.

- *Métodos basados en renderizar ciertas partes del instrumental junto con un mapa probabilístico [4].*

Conocido el instrumento quirúrgico, se divide en una o varias áreas relevantes. Cada área se representa por un clasificador el cual asigna diferentes probabilidades a cada vector de la imagen.

A continuación se busca esta configuración, en la imagen que queremos analizar. En ella se realiza un render en 3D de los posibles candidatos y se mide su coincidencia con el mapa de probabilidad.

Este sistema presenta la desventaja por su parte, de un elevadísimo coste computacional.



David Vara Rodríguez

En esta línea también encontramos otros trabajos como [5]. Este sistema proporciona una visión en 3D del instrumento laparoscópico partiendo de un conocimiento previo sobre la geometría de la herramienta. Utiliza la información cinemática proporcionada por el robot en cada momento para realizar un render de las posiciones próximas a su trayectoria y con ello va creando un patrón de orientación gradual para una coincidencia más rápida con la imagen real.

- *Métodos basados en obtener la posición en 3D del instrumento quirúrgico a partir de una imagen monocular*[6].

Este sistema utiliza un marco de trabajo basado en niveles para diferenciar de forma sencilla el tejido de la herramienta. Para clasificar los píxeles se utiliza el método de Random Forests.

Además este sistema utiliza la forma del instrumental, ya conocida previamente, para solucionar los problemas de luz y oclusiones de la herramienta quirúrgica debido al sangrado u otras razones y obtener la imagen en 3D a partir de una imagen binarizada.

El trabajo de [8] también merece ser destacado. Este sistema consta de dos etapas:

1) Una primera etapa de tratamiento de la señal de vídeo capturada por la cámara y extracción de la información 2D relevante de la escena.

Esta información 2D está compuesta por los bordes del instrumental y su extremo, los cuales son detectados y segmentados en la imagen. En el procesado de la imagen se utiliza una estrategia de detección de bordes, empleando la continuidad temporal entre imágenes consecutivas y operadores de detección de bordes (Sobel).

2) Una segunda etapa en la que se realiza el análisis de color del instrumental laparoscópico, lo cual permite identificar su extremo, cuyas coordenadas 3D serán estimadas.



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez

Capítulo 3. Elementos utilizados

1. Introducción al capítulo

En este capítulo analizamos los diferentes componentes hardware utilizados en este proyecto.

Los elementos principales son:

- Sensor IMU MPU6050 (acelerómetro + giroscopio)
- Microcontrolador Arduino UNO
- Módulo bluetooth HC-06

Primeramente se realizará un breve repaso a las características más importantes de cada uno de ellos para después dar paso al conexionado de los mismos.

Posteriormente se analiza la comunicación entre el microcontrolador y el pc vía puerto serie tanto por cable como de forma inalámbrica.

Para la comunicación por bluetooth se explica la configuración y programación tanto del módulo HC-06 como de la aplicación de recepción de datos en el PC (Matlab).

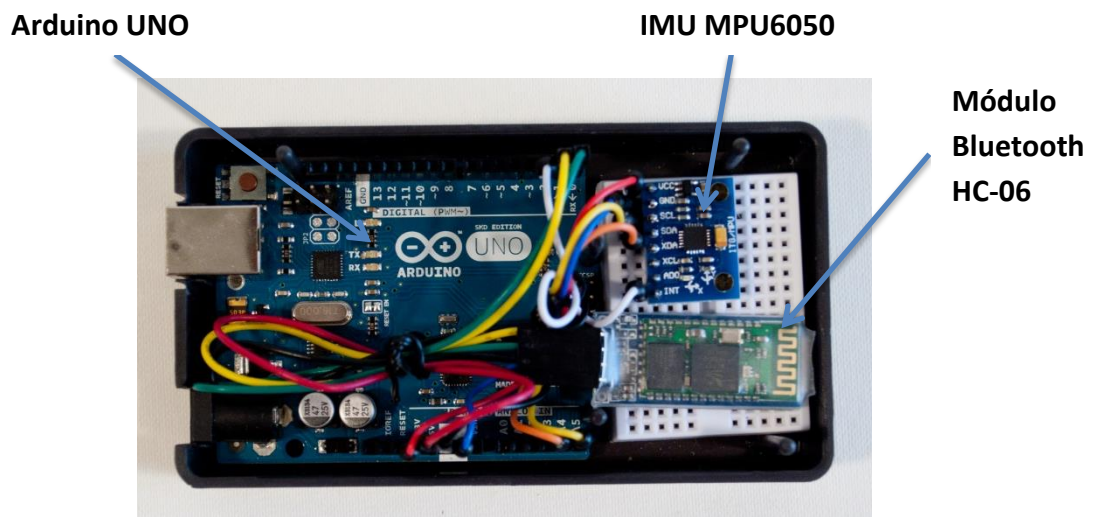


Figura 26. Componentes utilizados en el dispositivo

2. Descripción de los elementos

2.1 IMU MPU6050

Este dispositivo es un sensor inercial que combina un giroscopio de 3 ejes y un acelerómetro de 3 ejes junto con un procesador a bordo de Movimiento Digital (DMP).

El giroscopio cuenta con un rango de escala de ± 250 , ± 500 , ± 1.000 y ± 2.000 $^{\circ}/s$ (dps) y el acelerómetro un rango de $\pm 2g$, $\pm 4g$, $\pm 8g$, y $\pm 16g$.

Resumen de las características principales:

- Salida I2C digital de 6 o 9 ejes. Incorpora MotionFusion para el cálculo de las matrices de rotación, cuaternios, ángulos de Euler, etc
- Voltaje de entrada: 2.3 - 3.4V
- Giroscopio de 3 ejes con una sensibilidad de hasta 131 LSBs/dps and rango de escala de ± 250 , ± 500 , ± 1000 , and ± 2000 dps
- Acelerómetro de 3 ejes programable of $\pm 2g$, $\pm 4g$, $\pm 8g$ and $\pm 16g$
- Procesador a bordo Digital Motion Processing™ (DMP™)
- Algoritmos internos para la calibración del dispositivo
- Incluye un sensor de temperatura digital con una oscilación de +1%.

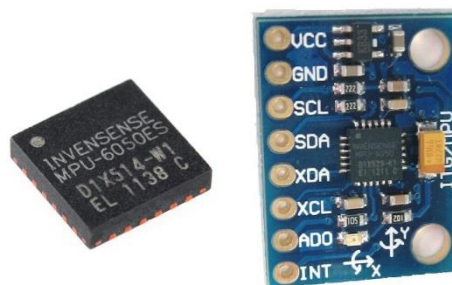


Figura 27. IMU MPU6050

Para una información más detallada consúltese el anexo adjunto en el CD de este proyecto.

David Vara Rodríguez

2.2 ARDUINO UNO

Es un microcontrolador basado en el ATmega328. Tiene 14 pines de entrada/salida digitales, de los cuales 6 pueden ser usados como salidas PWM, 6 entradas analógicas, un oscilador cerámico de 16 MHz, conexión usb, conexión de alimentación vía Jack, un cabezal HCSP y un botón de reset.

Resumen de las características principales:

Microcontrolador	ATmega328
Voltaje de funcionamiento	5V
Voltaje de entrada (Recomend)	7-12V
Voltaje de entrada (límites)	6-20V
Pines digitales de E/S	14 (of which 6 provide PWM output)
Pines analógicos de entrada	6
Corriente DC pines E/S	40 mA
Corriente DC para pines a 3.3V	50 mA
Memoria Flash	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Velocidad de reloj	16 MHz

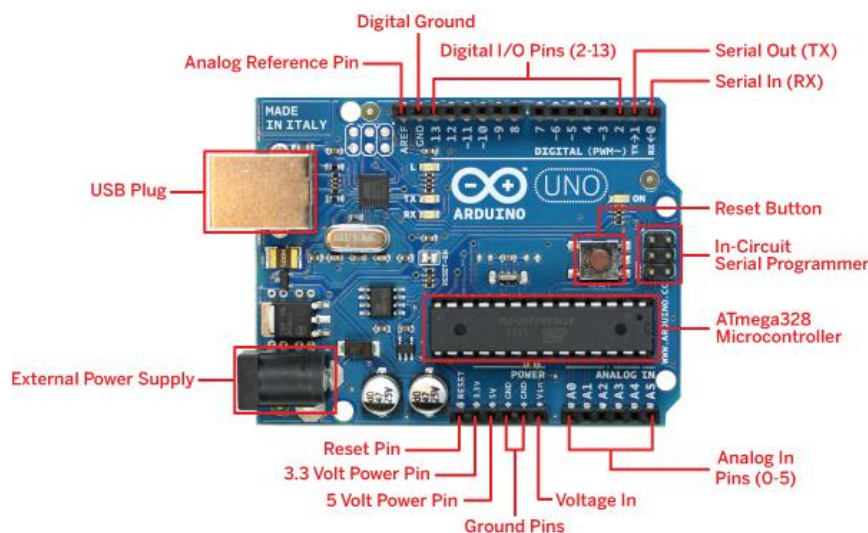


Figura 28. Arduino UNO

Para una información más detallada consúltese el anexo adjunto en el CD de este proyecto.

David Vara Rodríguez

2.3 MÓDULO HC-06

Es un dispositivo bluetooth para la conexión inalámbrica entre el microcontrolador (Arduino UNO en nuestro caso) y PC.

La comunicación se realiza vía serial mediante los pines TX/RX.

Resumen de las características principales:

- Protocolo bluetooth: Bluetooth Specification v2.0+EDR
- Frecuencia: 2.4GHz banda ISM
- Modulación: GFSK(Gaussian Frequency Shift Keying)
- Potencia de emisión: $\leq 4\text{dBm}$, Class 2
- Sensibilidad: $\leq -84\text{dBm}$ a 0.1% BER
- Velocidad: Asíncrona: 2.1Mbps(Max) / 160 kbps, Síncrona: 1Mbps/1Mbps
- Seguridad: Authentication and encryption
- Perfiles: Bluetooth serial port
- Voltaje de alimentación: +3.3VDC 50mA
- Temperatura de trabajo: $-20 \sim +75$ Centigrade
- Dimensiones: 26.9mm x 13mm x 2.2 mm



Figura 29. Módulo Bluetooth HC-06

Para una información más detallada consúltese el anexo adjunto en el CD de este proyecto.

3. Conexiones

3.1 CONEXIÓN SENSOR - MICROCONTROLADOR

La conexión entre el microcontrolador Arduino UNO y nuestro sensor IMU MPU6050 se realiza de la siguiente forma:

IMU MPU6050	ARDUINO UNO
VCC	5V
GND	GND
SDA	A4
SCL	A5
INT	PIN 2

Tabla 2. Pines de conexión IMU MPU6050 y microcontrolador Arduino UNO

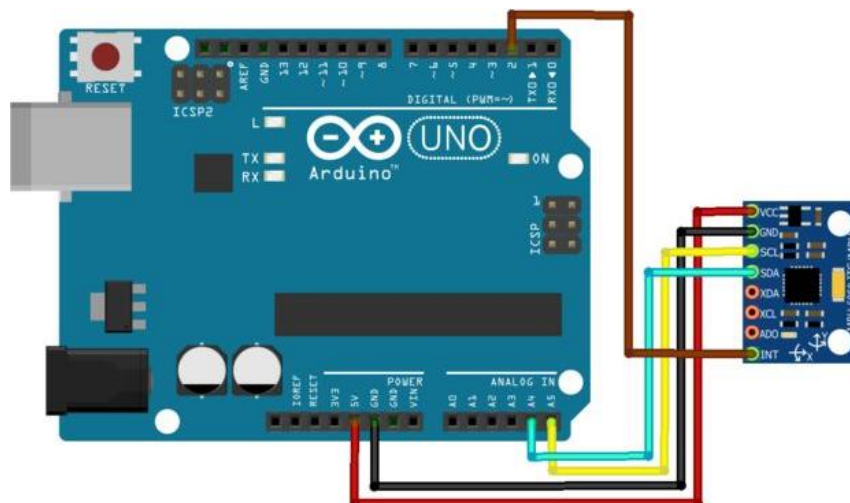


Figura 30. Esquema de la conexión sensor - microcontrolador

David Vara Rodríguez

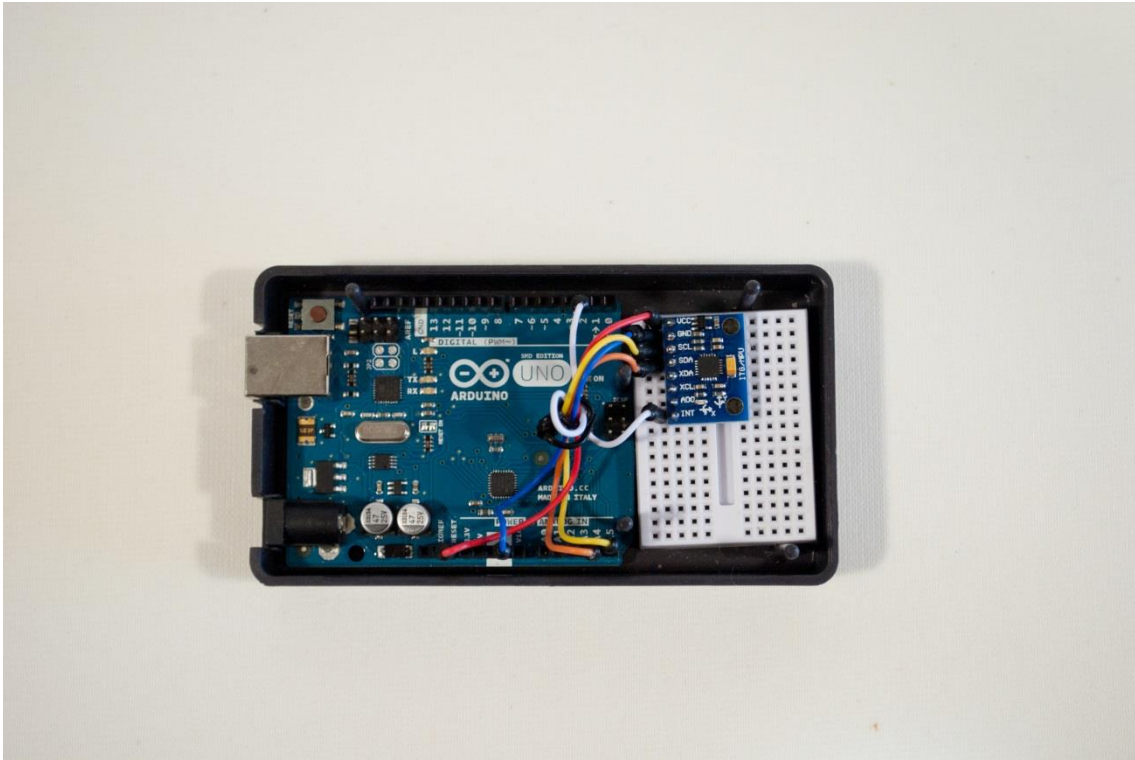


Figura 31. Conexión sensor – microcontrolador en el dispositivo construido

David Vara Rodríguez

3.2 CONEXIÓN MICROCONTROLADOR – PC

Puede realizarse de dos formas: mediante cable USB o inalámbricamente.

3.2.1 CONEXIÓN USB

La placa Arduino UNO consta de un puerto USB en el que podemos conectar un cable para la comunicación con el ordenador. Simplemente se necesita un cable estándar USB de conexión A a conexión B.



Figura 32. Cable USB

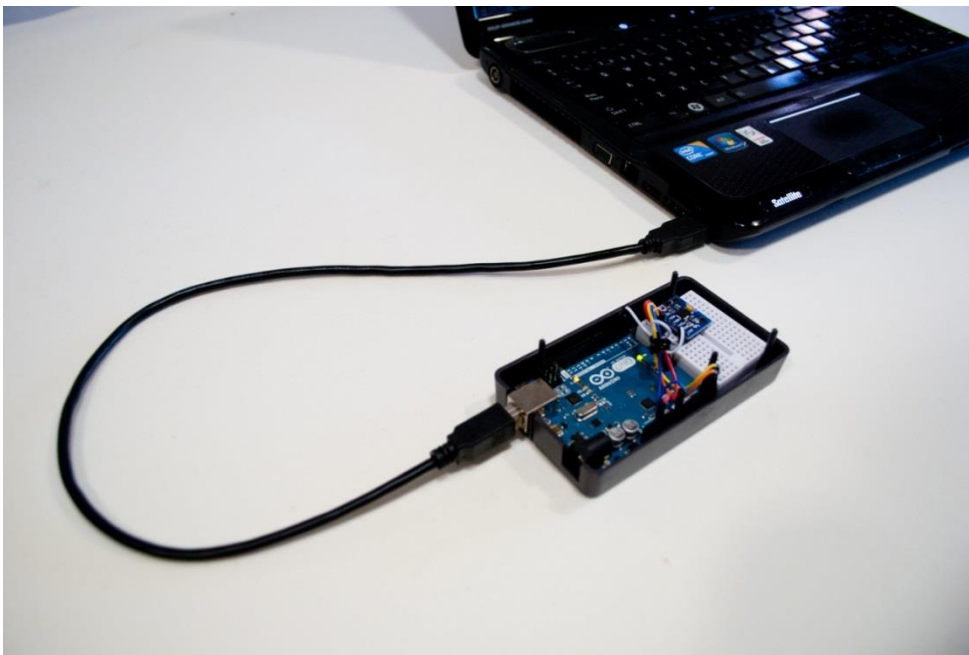


Figura 33. Conexión dispositivo construido – PC vía cable USB

3.2.2 CONEXIÓN INALÁMBRICA

Existen varias formas de conectar nuestro microcontrolador con el PC sin utilizar cables. De todas ellas hemos elegido la comunicación bluetooth debido a su buen funcionamiento en cortas distancias (inferiores a 10 metros) y su bajo coste económico.

Para esta comunicación empleamos el módulo HC-06 cuyas características fueron descritas anteriormente y pueden encontrarse de forma más detallada en el anexo adjunto en el CD de este proyecto

La conexión se realiza tal y como se muestra a continuación:

MÓDULO BLUETOOTH HC-06	ARDUINO UNO
VCC	5V
GND	GND
TX	RX
RX	TX

Tabla 3. Pines de conexión módulo bluetooth – microcontrolador Arduino UNO

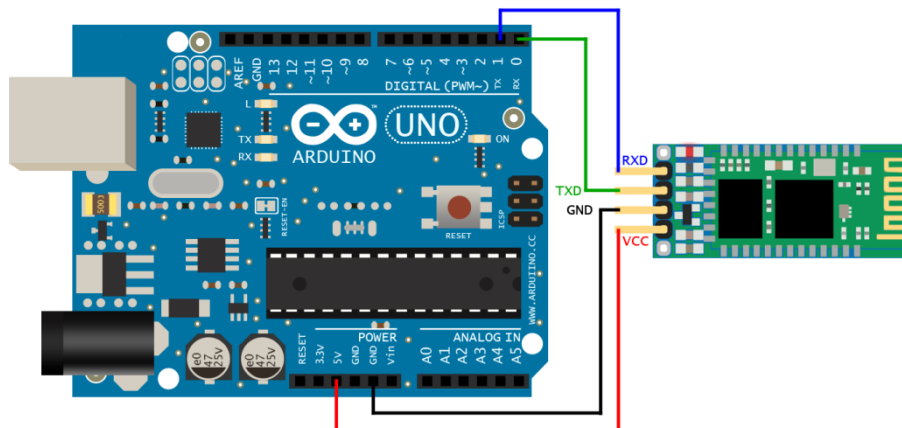


Figura 34. Esquema de conexión entre el módulo bluetooth y Arduino UNO



David Vara Rodríguez

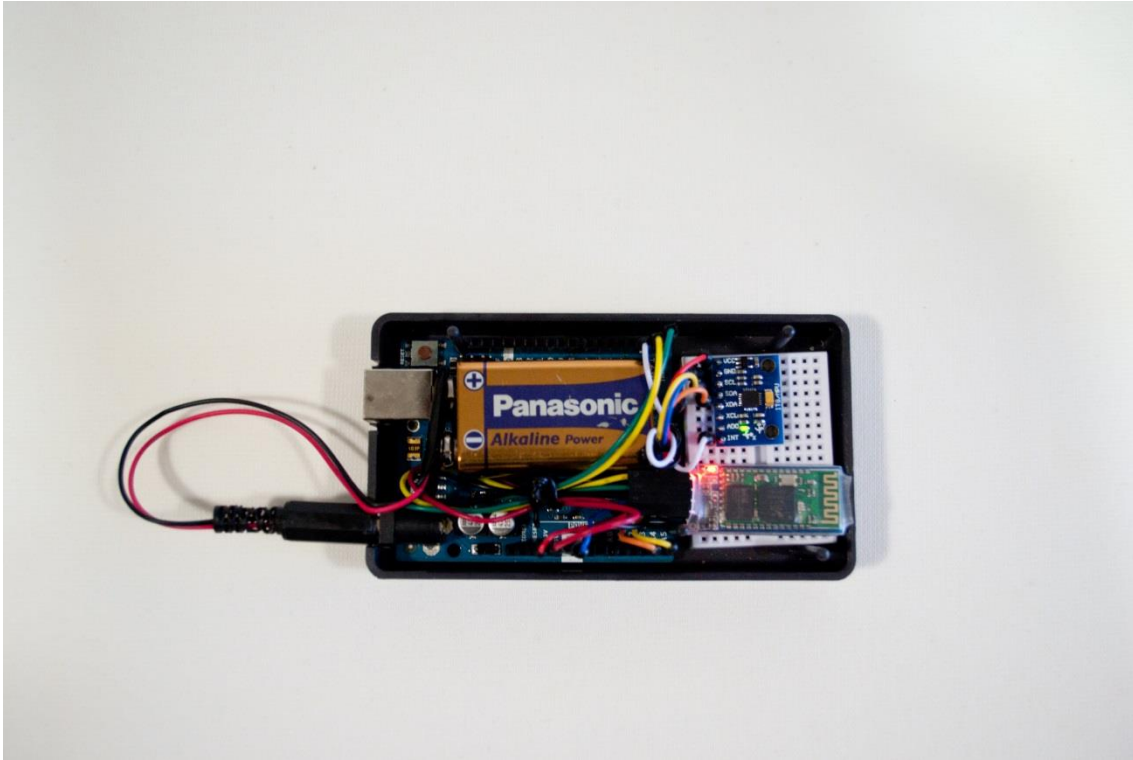


Figura 35. Dispositivo construido con el módulo bluetooth incorporado y alimentación interna



4. COMUNICACIÓN MICROCONTROLADOR – PC

La comunicación entre nuestro dispositivo y el computador o PC, ya sea mediante conexión por cable o inalámbrica, se realiza **vía puerto serie**. A través de este tipo de comunicación podemos enviar y recibir datos desde nuestro Arduino UNO.

4.1 COMUNICACIÓN VÍA PUERTO SERIE

El cable con el que programamos el microcontrolador desde un computador, es también cable de comunicación por puerto serie. Para establecer la comunicación, lo primero es abrir ese puerto serial en la aplicación de Arduino. Para ello utilizamos la función:

```
Serial.begin(9600);
```

El número que va entre paréntesis es la velocidad de transmisión. Todos los dispositivos que van a comunicarse deben tener la misma velocidad para poder entenderse.

4.1.1. SERIE DE PULSOS

En el modo más sencillo y común de comunicación serial (asincrónica, 8 bits, más un bit de parada) siempre se está enviando un byte, es decir un tren de 8 pulsos de voltaje legible por la máquina como una serie de 8 bit (1 ó 0) [15]:

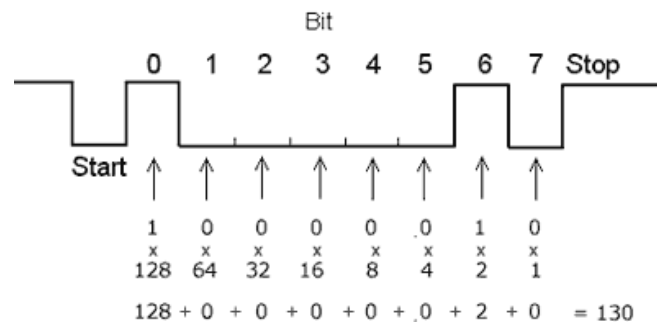


Figura 36. Serie de pulsos



David Vara Rodríguez

Siempre se están enviando bytes sin importar qué modificador usemos. La diferencia está en lo que esos bytes van a representar y sólo hay dos opciones en el caso del Arduino: una serie de caracteres ASCII o un número.

Si Arduino lee en un sensor analógico un valor de 65, equivalente a la serie binaria 01000001 esta será enviada, según el modificador, como:

Dato	Modificador	Envío (pulsos)
65	---DEC---	("6" y "5" ASCIIs 54-55) 000110110-000110111
65	---HEX---	("4" y "1" ASCIIs 52-49) 000110100-000110001
65	---OCT---	("1", "0" y "1" ASCIIs 49-48-49) 000110001-000110000-000110001
65	---BIN---	("0", "1", "0", "0", "0", "0", "0", "0" y "1" ASCIIs 49-48-49-49-49-49-49-48) 000110000...

El modificador BYTE permite el envío de información de manera más económica (menos pulsos para la misma cantidad de información), lo que implica mayor velocidad en la comunicación.



4.2 COMUNICACIÓN ARDUINO – MATLAB VÍA PUERTO SERIE

A continuación procedemos a configurar nuestro microcontrolador Arduino UNO y la aplicación Matlab para la transmisión de datos:

4.2.1 CONFIGURACIÓN DE ARDUINO

Tanto para conexión por cable como inalámbrica, en primer lugar debemos inicializar el puerto serie a través del cual se transmitirán los datos. Esto se realiza mediante el comando *Serial.begin()*.

Ejemplo:

```
Serial.begin(11520); //Abrimos puerto serie
```

Donde 11520 representa la velocidad de transmisión.

Para obtener los datos del sensor y enviarlos utilizamos la función *Serial.print* o *Serial.println*.

Ejemplo:

```
accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz); //Obtenemos datos sensor
```

```
Serial.print(ax) //Enviamos la aceleración del eje x por el puerto serie
```

4.2.1.1 CONFIGURACIÓN DEL MÓDULO BLUETOOTH

Si queremos enviar los datos a través del puerto serie pero sin cables mediante bluetooth, debemos realizar los siguientes pasos:

En primer lugar debemos configurar nuestro PC para que detecte este dispositivo bluetooth. Para ello nos dirigimos a Panel de control → Hardware y sonido → dispositivos e impresoras → agregar dispositivo o bien, configuración bluetooth → nueva conexión.

David Vara Rodríguez

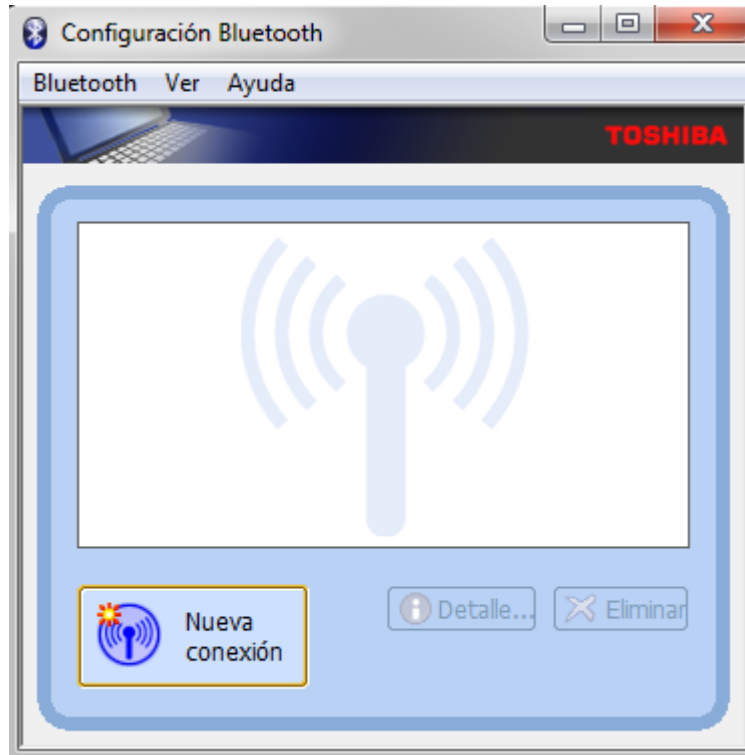


Figura 37. Agregar nueva conexión bluetooth

Una vez aquí, a través del asistente podemos detectar nuevos dispositivos bluetooth e instalarlos.

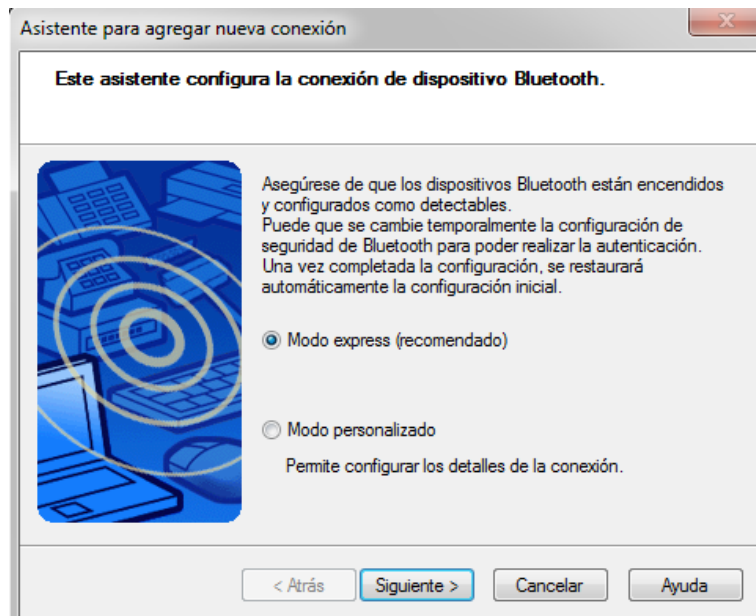


Figura 38. Asistente nueva conexión bluetooth

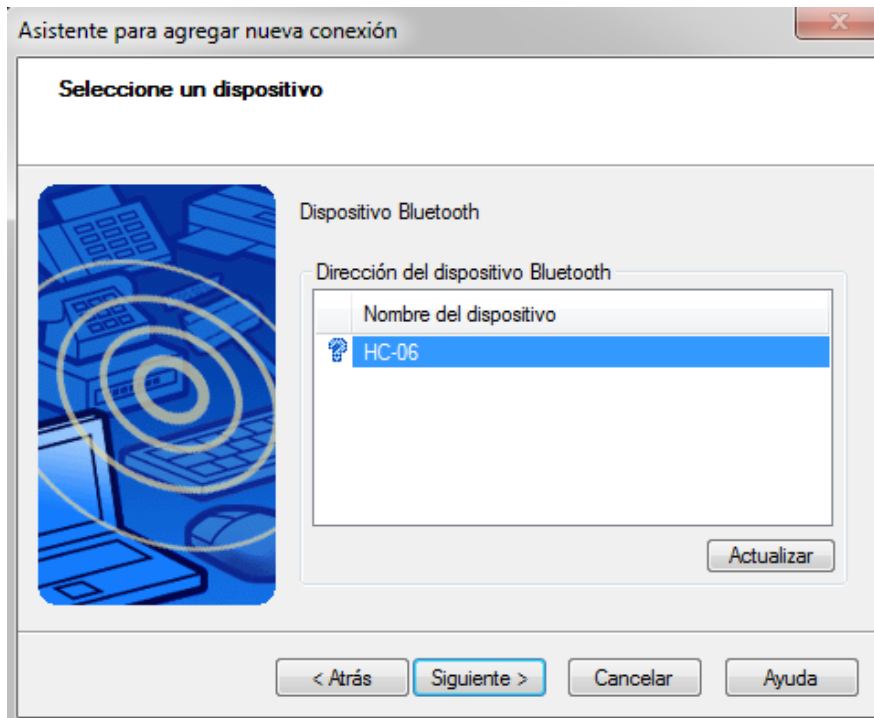


Figura 39. Seleccionar nuevo dispositivo bluetooth

Quando la instalación finalice podemos observar que ahora sí aparece una conexión bluetooth con el nombre de nuestro módulo HC-06:

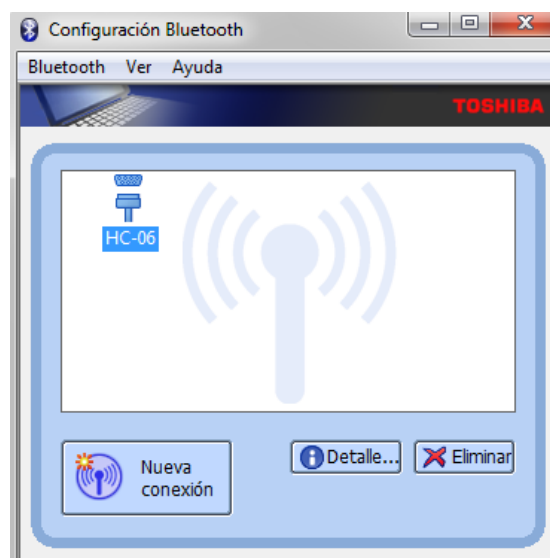


Figura 40. Instalación bluetooth finalizada



David Vara Rodríguez

Una vez instalada podemos configurar la conexión. Los detalles de la misma se muestran a continuación:

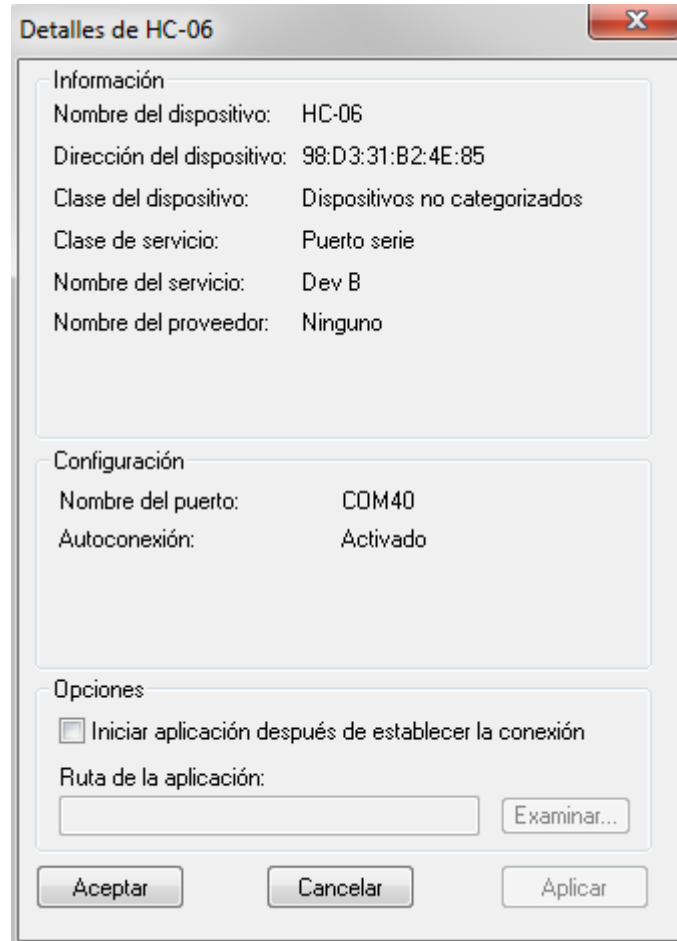


Figura 41. Características de la nueva conexión bluetooth



David Vara Rodríguez

Finalizada la configuración en nuestro PC, procedemos a configurar el módulo bluetooth.

Para tal efecto se ha desarrollado un código en lenguaje Arduino mediante comandos AT el cual puede encontrarse en el CD adjunto de este proyecto.

Los comandos AT sirven para configurar el módulo Bluetooth a través de un ordenador, microcontrolador o cualquier otro dispositivo con comunicación serie (TX / RX). Mediante ellos podemos modificar la velocidad del módulo, la contraseña de conexión o el nombre del dispositivo.

Para utilizar los comandos AT, el módulo Bluetooth no debe estar vinculado a ningún dispositivo (led rojo del módulo parpadeando).

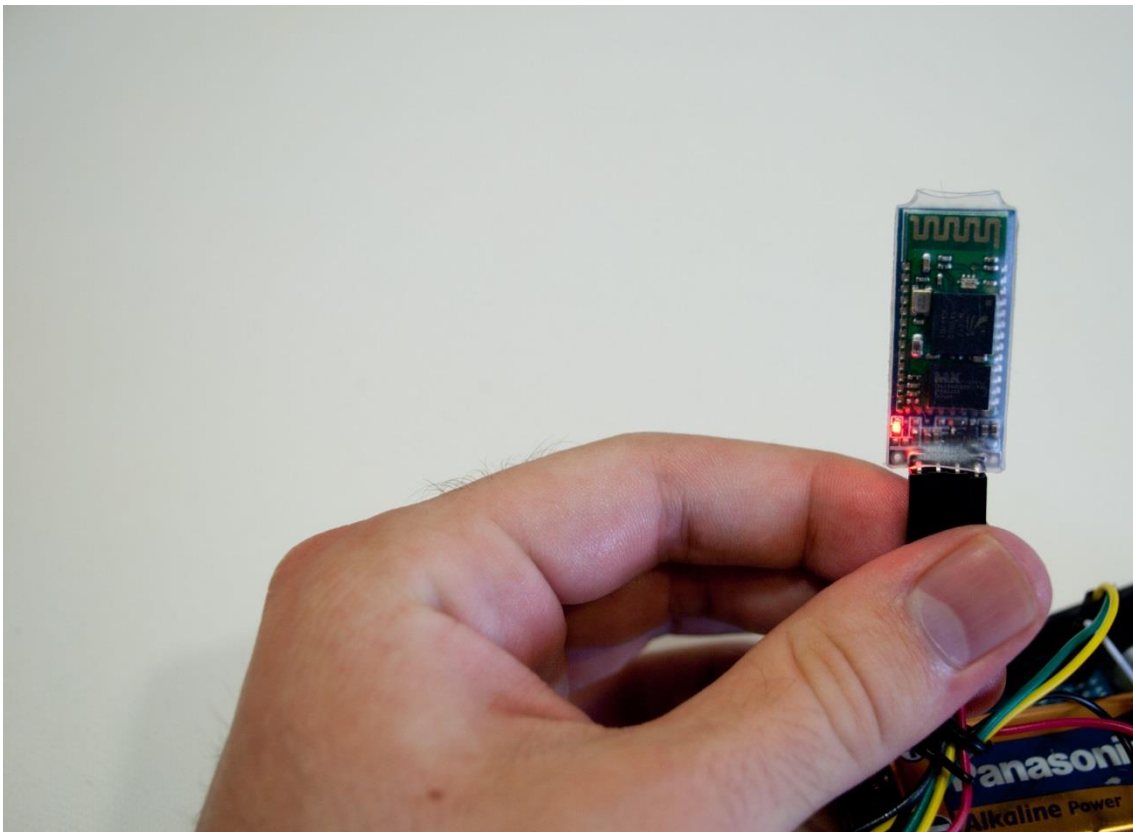


Figura 42. Módulo bluetooth led rojo parpadeando



David Vara Rodríguez

Según las especificaciones del HC-06, entre el envío de un comando AT y otro, ha de pasar al menos 1 segundo, de lo contrario el módulo no devolverá una respuesta.

En cuanto a la conexión, en lugar de hacerlo como se detalla en el apartado 2.2.2, debemos cambiar las conexiones a estas otras:

MÓDULO BLUETOOTH HC-06	ARDUINO UNO
VCC	5V
GND	GND
TX	PIN 10
RX	PIN 11

Tabla 4. Pines de conexión módulo bluetooth – arduino para la configuración del bluetooth

Hecho esto cargamos el programa en el microcontrolador y abrimos el Monitor Serial de Arduino. Los comandos AT que se pueden enviar en este módulo son los siguientes:

Comando AT	Descripción	Respuesta
AT	Test de comunicación.	Responde con un OK
AT+VERSION	Retorna la versión del Modulo	OKlinvorV1.8
AT+BAUDx	Configura la velocidad de transmisión del modulo según el valor de "x"1 = 1200 bps 2 = 2400 bps 3 = 4800 bps 4 = 9600 bps (por defecto) 5 = 19200 bps 6 = 38400 bps 7 = 57600 bps 8 = 115200 bps 9 = 230400 bps A = 460800 bps B = 921600 bps C = 1382400 bps	AT+BAUD4 Configura la velocidad a 9600 baud rate Responde con OK9600
AT+NAMEx	Configura el nombre con el que se visualizará el modulo, soporta hasta 20 caracteres	AT+NAMEDIYMakers Configura el nombre del modulo a DIYMakers Responde con OKsetname
AT+PINxxxx	Configura el Pin de acceso al modulo (password).1234 por defecto.	AT+PIN1122 Configura el pin a 1122 Responde con OKsetPIN

Tabla 5. Comandos AT



David Vara Rodríguez

Si enviamos el comando *AT*, observamos cómo el dispositivo responde que todo es correcto con un *OK*.

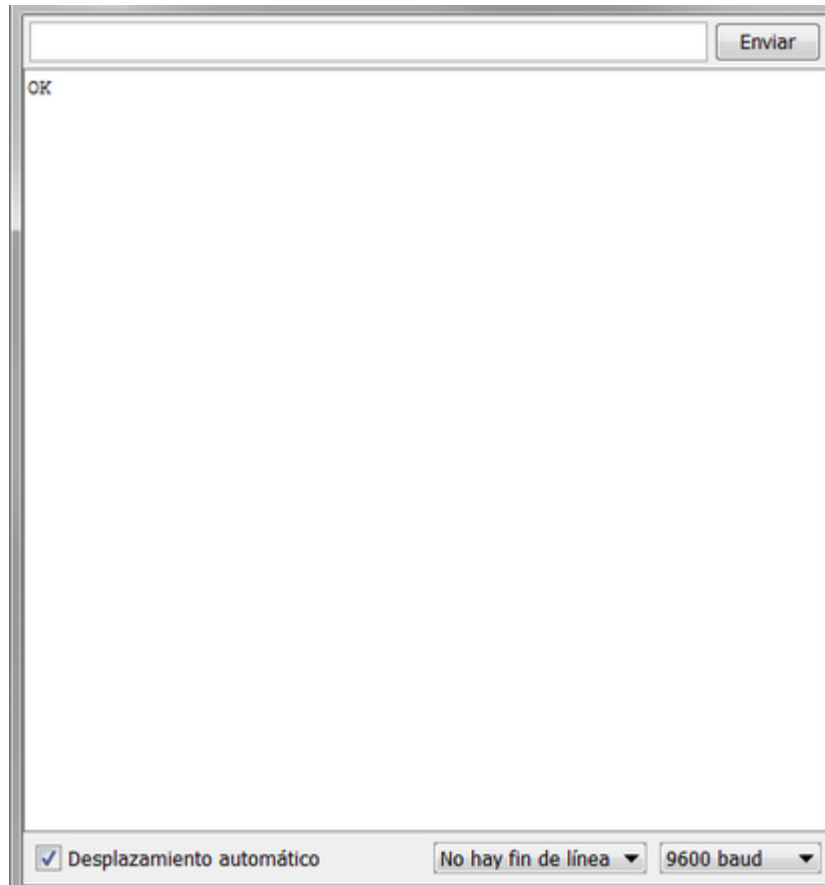


Figura 43. Respuesta ante comando AT

Si deseamos cambiar la velocidad de transmisión por ejemplo a 115200 baudios basta con introducir *AT+BAUD8*.

4.2.2 CONFIGURACIÓN DE MATLAB

Una vez hemos configurado el Arduino, las lecturas del sensor se enviarán a través de la conexión USB pero simulando un puerto serie, el cual debemos conocer con anterioridad. Para ello debemos dirigirnos al apartado herramientas de la interfaz:

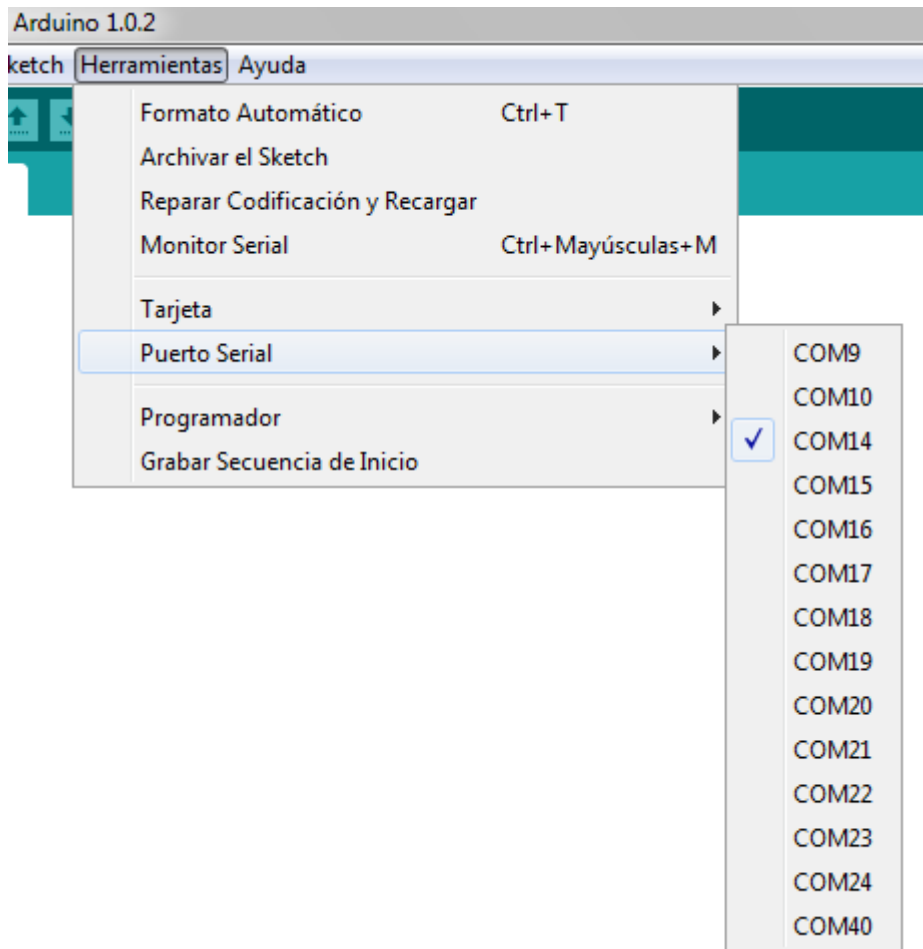


Figura 44. Selección del puerto serie

En nuestro caso, el puerto es el COM14. Una vez conocido, creamos un objeto serie en Matlab y lo abrimos para establecer la conexión:

- Extracto del código en Matlab:

```
%% APERTURA DEL PUERTO SERIE (COM)

%Borrar previos
delete(instrfind({'Port'}, {'COM14'}));
```



David Vara Rodríguez

```
%Crear objeto serie
s = serial('COM14','BaudRate',115200,'Terminator','CR/LF');
warning('off','MATLAB:serial:fscanf:unsuccessfulRead');

%Abrir puerto
fopen(s);
```

Establecida la conexión, procedemos a leer los datos del sensor enviados a través del puerto serie:

```
medida = fscanf(s,'%d %d');
```

Obtenidos los datos, se aplicarán los métodos de filtrado mediante Kalman y se hallará la velocidad y posición mediante integración numérica tal y como se explica en el capítulo 4 “Localización de la posición a partir de los datos del acelerómetro”.

Una vez las lecturas del sensor han sido tratadas y visualizadas procedemos a cerrar el puerto, de forma que otros programas puedan utilizarlo y eliminamos el objeto serie creado anteriormente.

- Extracto del código en Matlab:

```
%% CERRAR PUERTO Y ELIMINAR OBJETO SERIE

fclose(s);
delete(s);
clear s;
```




Capítulo 4. Localización de la posición a partir de los datos del acelerómetro

1. Introducción al capítulo

El objetivo de este proyecto es concebir un sistema de bajo coste para estimar la posición de un cuerpo a partir de los datos proporcionados por un sensor IMU (acelerómetro + giroscopio).

En este capítulo exploramos la factibilidad de esta técnica poniendo particular atención en la evaluación de la precisión obtenida.

Analizamos diferentes soluciones basadas en la integración numérica y el filtro de Kalman.

2. Dispositivos MEMS

El rastreo de la posición de un objeto a partir de sensores inerciales, es un campo de la robótica que está adquiriendo cada vez más importancia hoy en día. Este nuevo interés se debe a la introducción de los dispositivos microelectromecánicos MEMS (MicroElectroMechanic Systems).

Estos pequeños chips están integrados en cantidad de aparatos como móviles como teléfonos o PDAS. Son pequeños, baratos y su campo de aplicación comprende el posicionamiento GPS, localización de peatones, compensación de la deriva y la detección de micro movimientos.

Nuestro trabajo se centra en la identificación de los errores y la aplicación de algoritmos para una mejor precisión.



3. Análisis del problema y modelado

El propósito de nuestro sistema es hallar la posición de un cuerpo a partir de los datos del acelerómetro en un sistema cartesiano de 3 dimensiones. Reduciendo este ámbito a un solo eje, el procedimiento para hallar extraer la posición con respecto al tiempo puede ser visto como una doble integración de la aceleración :

$$\iint_0^t a(t) dt dt$$

donde $a(t)$ es la aceleración.

Para estimar la posición usando un dispositivo digital, es necesario aplicar la integración numérica. En el siguiente apartado se exploran los principios básicos de esta técnica; un estudio más profundo del método del trapecio (el cual es uno de los algoritmos más utilizados en la integración numérica) será presentado y puesto en práctica con posterioridad.

4. Integración numérica

La finalidad de esta técnica es aproximar el cálculo de una integral definida usando técnicas numéricas.

$$\int_a^b f(x) dx$$

La causa de utilizar este procedimiento se debe a que en nuestro caso, $f(x)$ se corresponde con la aceleración proporcionada por nuestro sensor, la cual está dada como un conjunto de valores en determinados instantes de tiempo.

A continuación se realizará un análisis sobre distintas reglas de integración.

David Vara Rodríguez

4.1 Métodos de integración numérica

4.1.1 LA REGLA TRAPEZODIAL

La primera técnica es la regla trapezoidal [16]. La curva a integrar se aproxima con trapezoides como se muestra en la figura 45. Cada trapezoide posee un área proporcional a la multiplicación de la base (espesor) por el promedio de la altura de los lados.

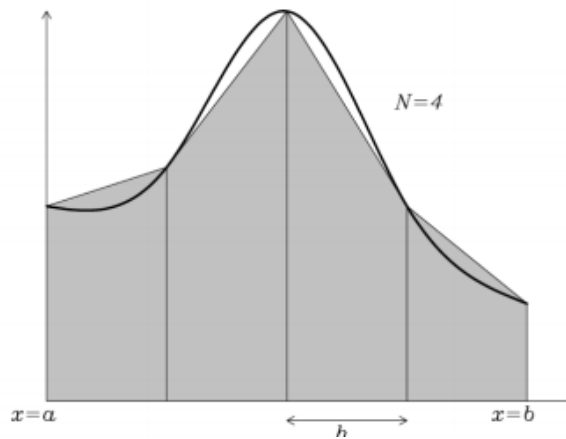


Figura 45. Regla trapezoidal

$$\begin{aligned} \mathbf{T}h[\mathbf{f}; \mathbf{a}, \mathbf{b}] &= \frac{h}{2}(f(a) + f(a + h)) + \frac{h}{2}(f(a + h) + f(a + 2h)) + \dots + \frac{h}{2}(f(b - h) + f(b)) \\ &= \frac{h}{2}(f(a) + f(b)) + h \sum_{j=0}^N f(a + jh) \end{aligned}$$

donde h se corresponde al espaciamiento entre puntos: $h = (b - a)/N$

4.1.2 LA REGLA DE SIMPSON

Esta técnica aproxima, para cada intervalo, la función a integrar $f(x)$ con una parábola, como se observa en la figura 46:

$$f(x) \approx ax^2 + \beta x + \gamma$$

donde el muestreo entre intervalos es homogéneo.

Para el intervalo correspondiente a los puntos $i-1, i, i+1$ (ver figura 46) la función $f(x)$ se puede expresar como la suma ponderada de los valores de la función en esos tres puntos, de la siguiente forma:

$$\int_{i-1}^{i+1} ax^2 + \beta x + \gamma = \frac{f(i-1)}{3} + \frac{4f(i)}{3} + \frac{f(i+1)}{3}$$

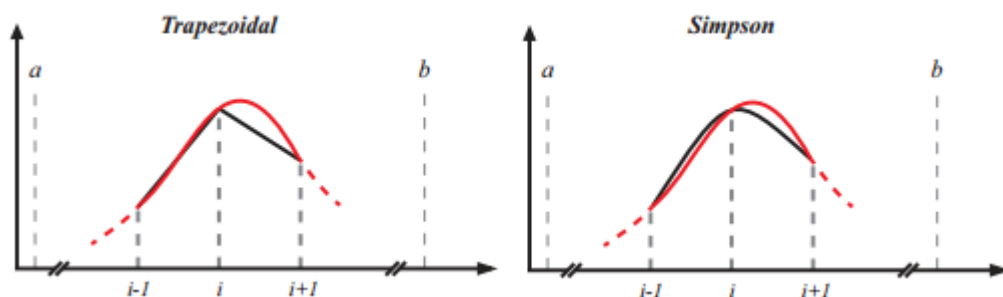


Figura 46. Aproximación numérica de una función $f(x)$ (línea roja) mediante las reglas del trapecio y de Simpson (líneas negras)



David Vara Rodríguez

4.2 Error de integración

La siguiente tabla muestra los errores cometidos mediante las aproximaciones trapezoidal y de Simpson:

Integración	Aproximación ε	Error Rel. ϵ
Trapezoidal	$\varepsilon_t = \left(\frac{[b-a]^3}{N^2}\right) f^{(2)}$	$\epsilon_s = \frac{\varepsilon_t}{f}$
Simpson	$\varepsilon_s = \left(\frac{[b-a]^5}{N^4}\right) f^{(4)}$	$\epsilon_s = \frac{\varepsilon_s}{f}$

Tabla 6. Error en las reglas de integración

Fijándonos en el cuadro anterior, podemos observar que para un menor número N el error disminuye. Reduciendo las evaluaciones del integrando, se reduce el número de operaciones aritméticas involucradas, y por tanto se reduce el error de redondeo total. Además, cada aproximación supone un coste tanto de tiempo como computacional y el integrando puede ser arbitrariamente complicado.

Aunque el error depende de cada caso a tratar, en general, la conclusión fundamental es que la regla de Simpson es una mejora sobre la regla Trapezoidal, pese a una leve adición en el coste computacional del algoritmo. Sin embargo, es importante notar que el número de puntos N TIENE que ser impar para la regla de Simpson, lo cual es un claro inconveniente si se tienen datos de campo o de laboratorio como es nuestro caso. Por esta razón el método de Simpson es descartado.



5. Origen de los errores

Existen diversas fuentes de error que pueden reducir la precisión de nuestros resultados. Algunos son producto de las condiciones ambientales (como la temperatura, campos magnéticos, etc.) mientras que otros se deben al muestreo y procesado de la señal.

5.1 ERROR DEL MEMS

Como ya se ha mencionado en otros capítulos, utilizamos un sensor IMU MPU6050 para medir aceleraciones no gravitacionales en términos de fuerzas g.

Estos dispositivos presentan múltiples errores:

- El primero de todos es, tal y como puede comprobarse en el datasheet del dispositivo, la variación de la sensibilidad en un $\pm 3\%$. Además esta varía con la temperatura en un $\pm 2\%$ adicional.
- El nivel de cero-g tampoco es constante y puede variar $\pm 0.08g$ para el eje Z y $\pm 0.05g$ para los ejes X e Y.
- La señal proporcionada por el acelerómetro también posee una densidad de error de $400\mu g/\sqrt{Hz}$.

Todo ello afecta a la calidad de las mediciones del sensor. Por ejemplo, un error de un -6% en el nivel de cero-g puede transformar una aceleración nula en una aceleración de $-1.47m/s^2$ lo que puede ocasionar grandes errores en la estimación de la posición después de unos segundos.

5.1.1 ERROR DE LA INCLINACIÓN

Otro error significativo que afecta a este tipo de dispositivos es el error en la inclinación de los mismos. En la posición de $0g$, este puede ser realmente importante. Una inclinación de 1° en dicha posición, produce un error de salida equivalente a una inclinación de 10° en las posiciones de $\pm 1g$.

Esto puede observarse con claridad en la figura de la página siguiente:

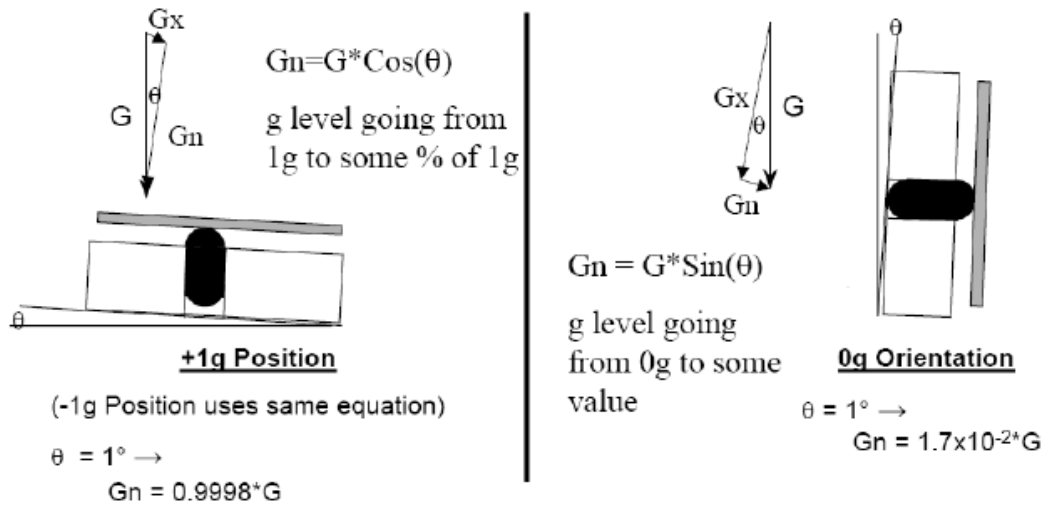


Figura 47. Error debido a la orientación

En la figura 44 podemos contemplar cómo la orientación afecta a la salida del acelerómetro y consecuentemente a la estimación de la posición basadas en esos datos.

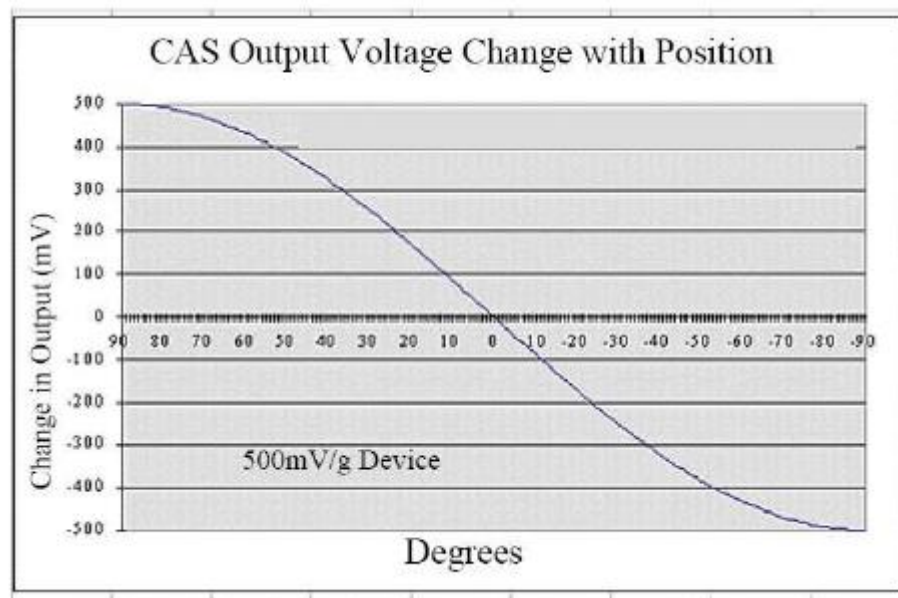


Figura 48. Efecto de la inclinación en los datos de salida



5.2 ERRORES DEBIDOS AL MUESTREO DE LA SEÑAL

Para evitar la distorsión en la reconstrucción de una señal analógica muestreada, es necesario respetar las condiciones del teorema de Nyquist – Shannon, el cual establece que si la banda máxima de una señal es B , entonces la frecuencia de muestreo debe ser al menos el doble ($2B$).

El embotellamiento del muestreo en frecuencia está relacionado con el microcontrolador utilizado, el cual debe realizar una conversión analógica/digital (A/D) antes de tomar una nueva muestra.

Dicha conversión A/D tarda 25 ciclos de reloj en nuestro microcontrolador ATMEGA328. El algoritmo implementado toma una muestra de la aceleración en el eje x , a continuación una del eje y , y posteriormente una del eje z . Esto supone que el tiempo entre dos muestras del mismo canal se corresponde con $3 \cdot 25 = 75$ ciclos de reloj. Como nuestro dispositivo trabaja a una frecuencia de 16MHz, podemos emplear un periodo de muestreo máximo de $\frac{3 \cdot 25}{16MHz} = 4,69\mu s$ (213kHz).

Sin embargo, podemos comprobar que esta suposición es muy optimista en la realidad, ya que no tiene en cuenta todos los ciclos empleados en las llamadas a las rutinas, declaraciones de variables, cálculos de operaciones, etc. Simulando nuestro algoritmo observamos que el tiempo entre dos muestras del mismo eje son 10ms (100 Hz).

Así pues, podemos muestrear la señal analógica de entrada con una frecuencia máxima de 100Hz y por tanto, la máxima frecuencia que podemos aceptar en la aceleración de entrada es $\frac{f_{muestreo}}{2} = \frac{100Hz}{2} = 50Hz$.

Si nuestra aceleración tiene una frecuencia superior, entonces la señal original será distorsionada en el proceso de muestreo. En ese caso habría que filtrar la señal de entrada para asegurar el cumplimiento del teorema de Nyquist.

Esto puede realizarse fácilmente mediante un filtro RC.

David Vara Rodríguez

5.3 ERRORES DEBIDOS A LA CONVERSIÓN A/D

La conversión analógica/digital supone múltiples errores.

El ADC de nuestro microcontrolador (ATMEGA328) usa 10 bits para convertir la señal. Por tanto, nuestra aceleración de entrada se transforma en un valor entre 0 y 1023 ($2^{10} - 1$).

Podemos por tanto cuantificar el error en ± 0.5 LSB, ya que un intervalo de voltajes de entrada de ancho igual a 1 LSB, se transformará en el mismo valor. En nuestro caso, 1 LSB se corresponde con 0.0032V ($3.3V / 1024$ donde 3.3V es el voltaje de referencia).

Un error típico de estos dispositivos es el error de offset, consistente en la desviación de la primera transición (desde 0x000 a 0x001) respecto de la transición ideal (la cual debería ser en +0.5 LSB). Para obtener la precisión adecuada, este error es corregido en el propio algoritmo implementado. Esto puede observarse con detalle en el capítulo 4 - apartado 7.2: "Calibración del dispositivo".

Otro error típico es el error de ganancia. Éste se define como la diferencia entre la ganancia nominal o esperada y la ganancia real.

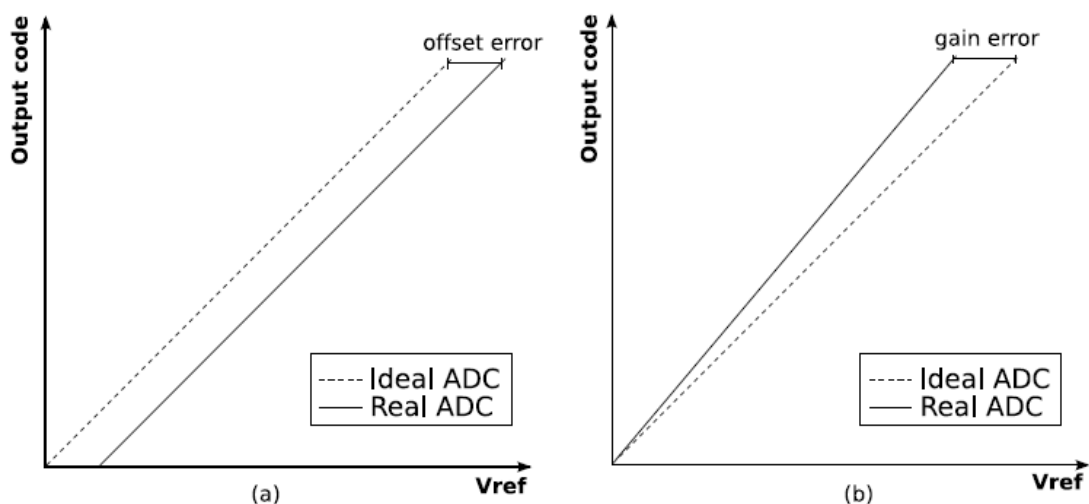


Figura 49. Error de offset (a) y de ganancia (b)

5.3.1 ERROR DE NO LINEALIDAD

También pueden producirse errores debidos a la no linealidad del convertidor.

El error de no linealidad es la máxima diferencia entre la curva de transferencia real y la ideal cuando los errores de cero y de ganancia son nulos.

Es un error que no se puede corregir.

Se expresa como porcentaje del fondo de escala o como fracción de LSB y suele valer $\frac{1}{2}$ LSB. Se denomina también linealidad “integral” porque indica el error que se tiene en una determinada palabra de salida, con independencia de las demás.

La no linealidad ‘diferencial’ hace referencia a la desviación máxima en el intervalo de dos transiciones.

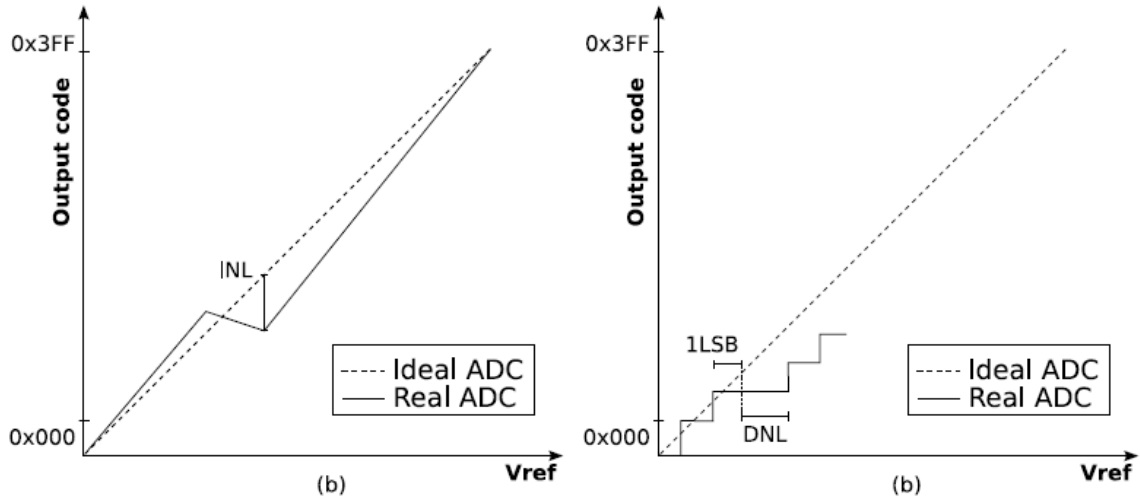


Figura 50. Error de no linealidad integral (a) y diferencial (b)



5.4 ERROR DE LA RESOLUCIÓN DEL MICROCONTROLADOR

En nuestro microcontrolador (ATMEGA328) los números de doble precisión son representados por 32 bits.

En este caso 1 bit es usado para el signo, 8 bits para el exponente y 23 para la mantisa. Como la precisión es finita, necesariamente tenemos algunos errores al operar con números floats. Sin embargo es despreciable respecto de errores introducidos por otros componentes del dispositivo.



6. Soluciones propuestas

En esta sección se proponen dos soluciones basadas en el análisis previo.

La primera consiste en la aplicación de la regla del trapecio. En la segunda, se explora la utilización del filtro de Kalman.

6.1 REGLA DEL TRAPECIO

6.1.1 INTRODUCCIÓN AL ALGORITMO

Como ya se ha explicado en apartados anteriores, nuestro objetivo es hallar la posición de un objeto, a partir de los datos proporcionados nuestro sensor.

La aceleración se define como la variación de la velocidad respecto al tiempo y ésta a su vez, como la variación de la posición respecto del tiempo. Expresado matemáticamente:

$$\vec{a} = \frac{d\vec{v}}{dt}$$

$$\vec{v} = \frac{d\vec{p}}{dt}$$

donde \vec{a} es el vector aceleración, \vec{v} el de velocidad y \vec{p} el de posición.

La integración es el método opuesto a la derivada. Realizando una doble integración sobre las aceleraciones obtenidas, podemos hallar la posición (suponiendo condiciones iniciales nulas):

$$\vec{v} = \int \vec{a}(t)$$

$$\vec{p} = \int \vec{v}(t)$$

David Vara Rodríguez

Una forma de entender las fórmulas anteriores, es definir la integral como el área bajo la curva y su valor, la suma de áreas de anchura muy pequeña [18].

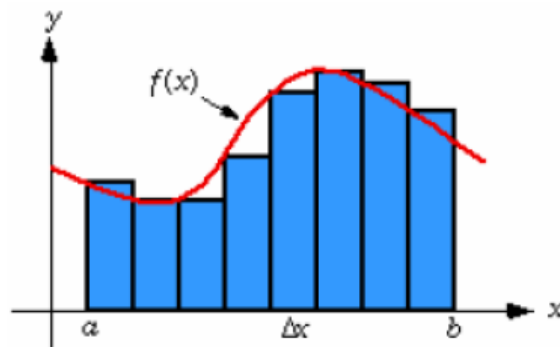


Figura 51. Aceleración muestreada

$$\int_a^b f(x)dx = \lim_{n \rightarrow \infty} \sum_{i=1}^n f(x_i)\Delta x$$

$$\text{donde } \Delta x = \frac{b-a}{n}$$

Con este concepto previo de “áreas bajo la curva” podemos realizar la siguiente deducción: muestrear una señal nos proporciona los valores instantáneos de su magnitud, con lo que se pueden crear pequeñas áreas entre dos muestras y sumándolas obtendríamos el valor de la integral.

David Vara Rodríguez

6.2 ERRORES Y APROXIMACIÓN TRAPEZOIDAL

En la realidad, esta suma da lugar a errores como se puede observar en la siguiente imagen:

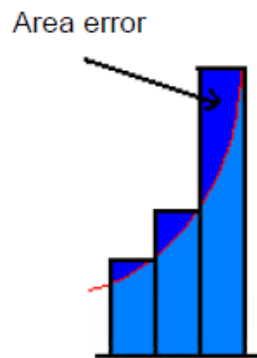


Figura 52. Errores generados durante la integración

Estos errores son conocidos formalmente como errores de muestreo. Para minimizarlos, tenemos que realizar una suposición adicional, por la cual, cada área es la combinación de otras dos más pequeñas:

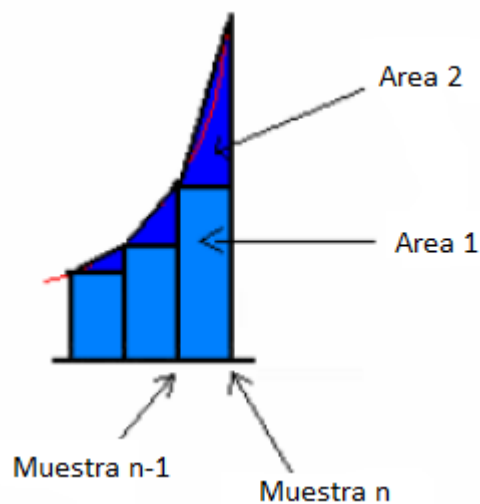


Figura 53. Los errores de integración se reducen mediante el Método Trapezoidal

David Vara Rodríguez

El área 1 es un rectángulo y la 2 un triángulo, ambos formados entre la muestra actual (n) y la anterior (n-1). La combinación de ambas da lugar a una aproximación de primer orden de la señal.

El error se reduce considerablemente respecto de la primera aproximación.

Así pues, empleando dicha técnica y suponiendo una aceleración como la mostrada en la figura:

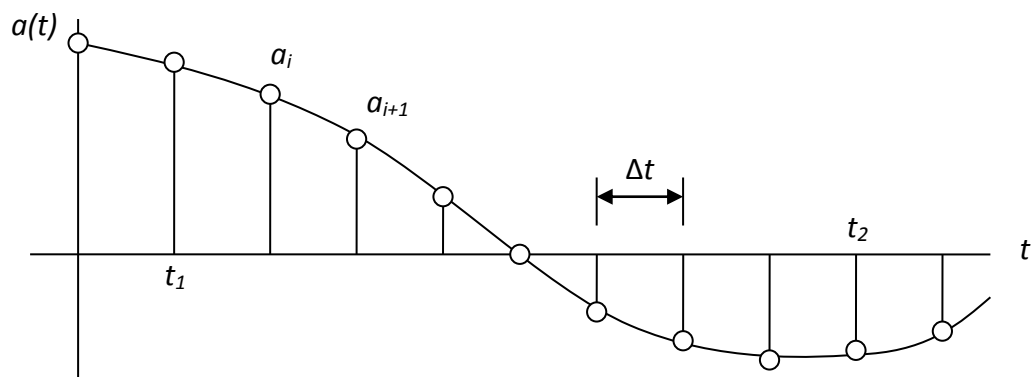


Figura 54. Curva de aceleración ideal

aproximamos la curva como un trapecio de superficie $\frac{1}{2}(a_i + a_{i+1})\Delta t$. Sumando las áreas desde un tiempo t_1 hasta un tiempo t_2 , obtenemos la velocidad de la forma:

$$v(t_2) - v(t_1) = \int_{t_1}^{t_2} a(t)dt \approx \frac{1}{2}(a_i + a_{i+1})\Delta t$$

Una vez hallada, de igual forma volvemos a aplicar el método para hallar la posición del cuerpo:

$$p(t_2) - p(t_1) = \int_{t_1}^{t_2} v(t)dt \approx \frac{1}{2}(v_i + v_{i+1})\Delta t$$

6.3 FILTRO DE KALMAN

El Filtro de Kalman, desarrollado por el ingeniero y matemático Rudolf E. Kalman en la década de 1960, es un algoritmo utilizado en el filtrado de datos discretos [19].

Para su implementación es necesario:

- a) Conocimiento del sistema a tratar así como de las mediciones obtenidas a través de los sensores.
- b) Ruido presente, incertidumbre e información sobre el error
- c) Condiciones iniciales

Es un estimador óptimo y de carácter recursivo. Óptimo ya que a partir de las mediciones obtenidas calcula las variables de interés y recursivo ya que hace uso de los estados anteriores del sistema, junto con las medidas tomadas, para calcular los nuevos resultados.

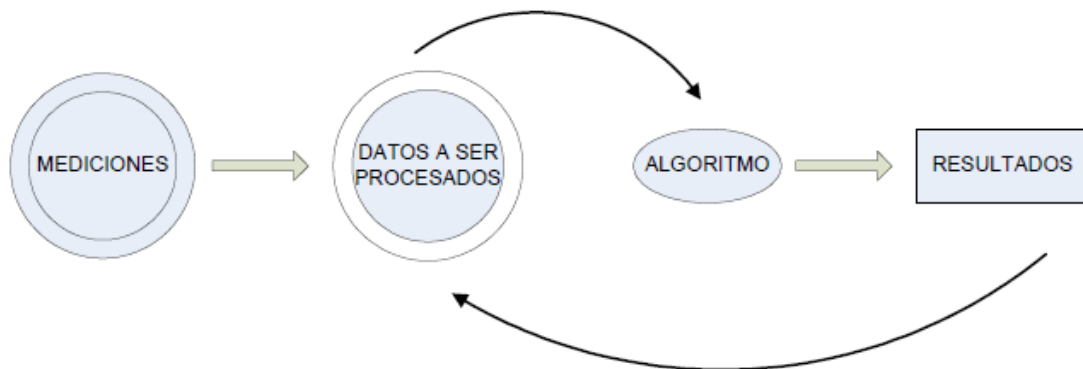


Figura 55. Recursividad en el filtro de Kalman

6.3.1 FILTRO DISCRETO DE KALMAN

El objetivo es estimar un estado $x \in R^n$ de un proceso en tiempo discreto.

El filtro discreto de Kalman consta de dos etapas [20]:

- 1) Un proceso de predicción, donde se tiene el estado actual del sistema y un mapeo de la progresión del sistema en función del tiempo
- 2) Una etapa de corrección, donde se combinan las mediciones y observaciones de las variables a tratar en nuestro sistema, con los valores reales medidos.

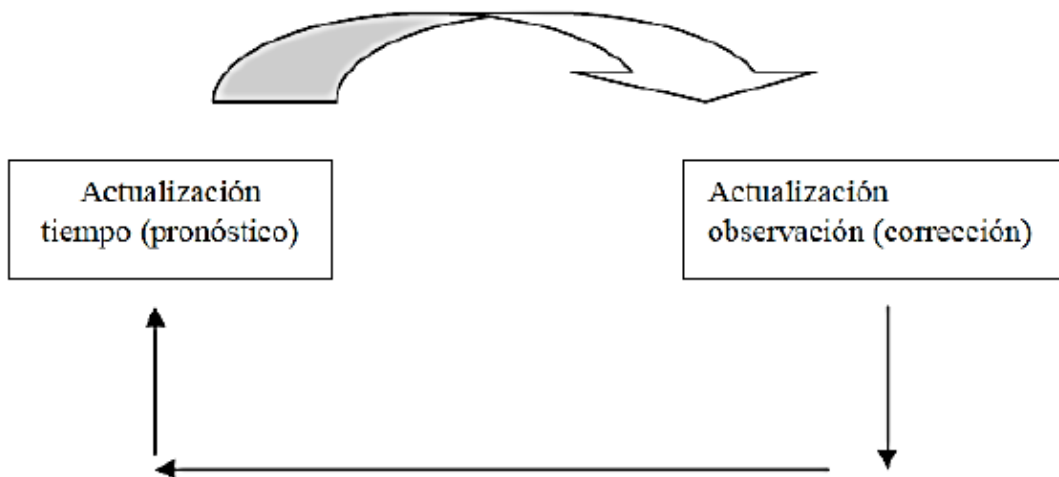


Figura 56. Ciclo de Kalman



David Vara Rodríguez

6.3.1.1. MODELO DEL SISTEMA

El sistema dinámico en el cual se aplica el Filtro de Kalman, ha de expresarse como un modelo lineal en el cual se estima el estado $x \in R^n$. Este sistema puede expresarse mediante la siguiente ecuación:

$$x_k = Ax_{k-1} + Bu_k + w_{k-1}$$

con una medida $z \in R^m$:

$$z_k = Hx_k + v_k$$

Donde:

x = estado del sistema.

k = tiempo.

z = valor observado.

u = entrada del sistema.

v = ruido en la medición.

w = ruido en el proceso.

A, B, H = Matrices determinísticas que definen la dinámica del sistema.

La matriz A , es cuadrada de dimensiones $n \times n$ y expresa la relación entre el estado actual (k) y el anterior ($k-1$). La matriz B con dimensión $m \times l$, relaciona la entrada $u \in R$ con el estado del sistema x . Finalmente la matriz H de dimensión $m \times n$, relaciona el estado x con la medición z_k .

Por su parte, w_k y v_k representan los errores presentes en el proceso. Son independientes entre ellos, con ruido gaussiano blanco y siguen una distribución normal con media cero. Sus matrices de covarianza son Q y R respectivamente:

$$p(w) \sim N(0, Q)$$

$$p(v) \sim N(0, R)$$



6.3.1.1.1 ALGORITMO

Como ya se indicó anteriormente, este filtro se divide en dos etapas: predicción y corrección.

Las ecuaciones de predicción obtienen una estimación de la covarianza del error y el estado actual del sistema en el tiempo k respecto del anterior ($k - 1$).

$$\hat{x}_{k^-} = A\hat{x}_{k-1} + Bu_k$$

$$P_{k^-} = AP_{k-1}A^T + Q$$

Por su parte, las ecuaciones de corrección sirven de retroalimentación. Incorporan nueva información a partir de las mediciones del sensor, con el objetivo de conseguir una mejor estimación final.

En primer lugar, [21] se calcula la ganancia de Kalman K_k de modo que minimice la covarianza del error en la siguiente estimación. Después se toma una medida del proceso Z_k (dato de nuestro sensor) para calcular el estado \hat{x}_k , de forma que se mejore la estimación realizada en la fase anterior de predicción. Finalmente se calcula nuevamente una estimación de la matriz de covarianza del error.

$$K_k = P_{k^-}H^T(HP_{k^-}H^T + R)^{-1}$$

$$\hat{x}_k = \hat{x}_{k^-} + K_k(z_k - H\hat{x}_{k^-})$$

$$P_k = (I - K_kH)P_{k^-}$$

A continuación se presenta un cuadro, a modo resumen, con todas las operaciones llevadas a cabo en el Filtro de Kalman:

David Vara Rodríguez

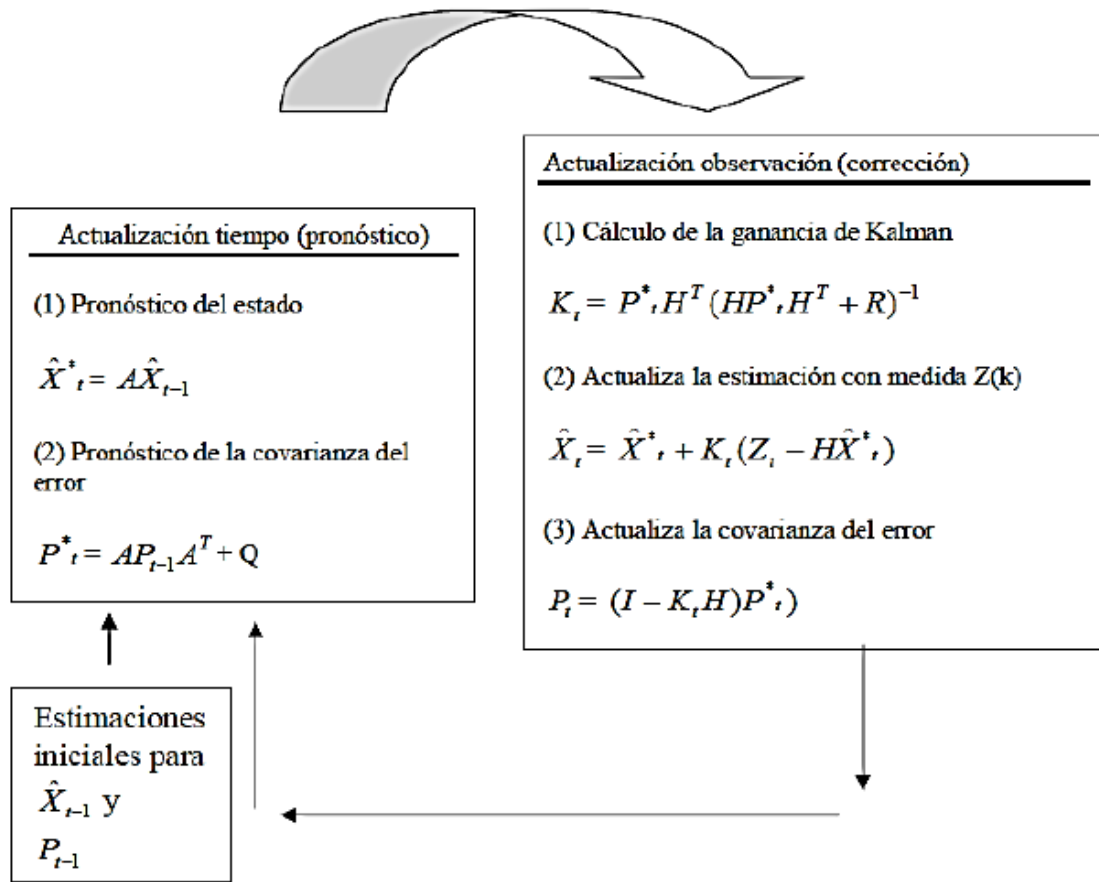


Figura 57. Cuadro resumen del funcionamiento del filtro de Kalman



David Vara Rodríguez

6.3.1.2 PARÁMETROS DEL FILTRO Y SINTONIZACIÓN

Existen dos variables de gran importancia en el comportamiento del Filtro de Kalman. Estos son parámetros son las matrices de covarianza Q y R que representan las perturbaciones en el proceso [22].

En el caso de R , éste puede ser obtenido de forma off-line, bien mediante la toma de mediciones de nuestro sensor y calculando la varianza del ruido presente o bien mediante ensayo y error observando la respuesta de nuestro sistema.

Ambas formas son válidas si bien hay que tener en cuenta ciertas pautas:

- Si la perturbación en el proceso es grande, entonces el parámetro R también ha de serlo. Esto hará que la ganancia de Kalman K sea pequeña y no se dé mucho peso a las mediciones realizadas para la estimación de \hat{x}_k .
- Por el contrario, si R es pequeña, la ganancia de Kalman K será mayor y se dará mayor credibilidad a las mediciones obtenidas para el cálculo de \hat{x}_k .

En cuanto al parámetro Q , su determinación es algo más complicada ya que se necesita observar de forma directa el proceso a estimar. En procesos no muy complejos, una suficiente incertidumbre puede arrojar buenos resultados.

Por otro lado, si los valores hallados de Q y R son correctos y se consideran constantes a lo largo del tiempo, la matriz de ganancia de Kalman K_k y la de covarianza del error P_k pueden estabilizarse rápidamente llegando a permanecer constantes durante el proceso.



7. Software desarrollado

7.1 INTRODUCCIÓN

A continuación se presenta el software implementado en este proyecto con el objetivo final del rastreo de la posición de un cuerpo. Éste ha sido realizado en dos aplicaciones: Arduino Software y Matlab R2010a.

Por un lado, en la aplicación de Arduino se han realizado dos programas:

El primero comprende la calibración del dispositivo a partir del cálculo de los offsets de los sensores.

El segundo programa lleva a cabo las operaciones de:

- Toma de datos de los sensores
- Ventana de filtrado
- Envío de los datos vía puerto serie

Por otro lado, en la aplicación de Matlab se ha realizado otro programa que comprende las operaciones de:

- Recepción de datos por el puerto serie
- Filtrado de datos mediante Kalman
- Cálculo de la velocidad mediante integración numérica
- Cálculo de la posición mediante integración numérica
- Visualización de los resultados

Todos estos pasos pueden visualizarse de forma más esquemática en el diagrama de flujo de la siguiente página.



David Vara Rodríguez

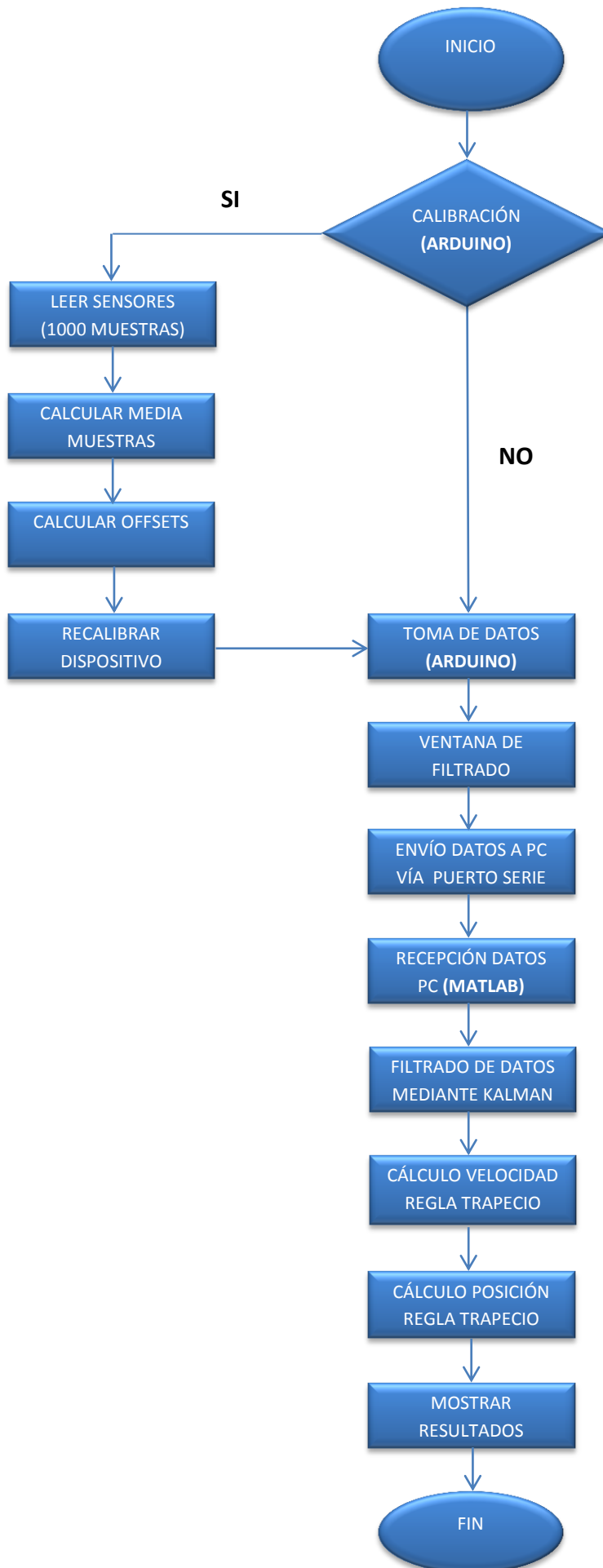


Figura 58. Diagrama de flujo



7.2 CALIBRACIÓN DEL DISPOSITIVO

La señal proporcionada por el acelerómetro es una señal ruidosa que hay que calibrar. Para tal efecto se ha desarrollado un código en lenguaje arduino. Un breve análisis se presenta a continuación:

El programa se inicializa cuando el usuario lo desee. Basta con pulsar una tecla para comenzar la ejecución:

```
while (Serial.available() && Serial.read());  
while (!Serial.available()){  
  Serial.println(F("Presione cualquier tecla para comenzar \n"));  
  delay(1500);  
}
```

A continuación se ejecuta el bucle principal del programa. Dicho bucle contiene tres pasos o etapas principales:

- 1) Paso 1: Leer sensores
- 2) Paso 2: Calibración
- 3) Paso 3: Mostrar resultados

- 1) Paso 1: Leer sensores

Se toma un número de medidas determinadas por el usuario (por defecto 1000) y se calcula la media de dichas lecturas.

```
// Leemos lecturas del dispositivo  
  
  accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);  
  
// Tomamos 1000 medidas. Las 100 primeras son descartadas  
  
if (i>100 && i<=(numero_lecturas+100)){  
  buff_ax=buff_ax+ax;  
  buff_ay=buff_ay+ay;  
  buff_az=buff_az+az;  
  buff_gx=buff_gx+gx;  
  buff_gy=buff_gy+gy;  
  buff_gz=buff_gz+gz;  
}
```




David Vara Rodríguez

```
//Calculamos La media  
  
if (i==(numero_Lecturas+100)){  
ax_media=buff_ax/numero_Lecturas;  
ay_media=buff_ay/numero_Lecturas;  
az_media=buff_az/numero_Lecturas;  
gx_media=buff_gx/numero_Lecturas;  
gy_media=buff_gy/numero_Lecturas;  
gz_media=buff_gz/numero_Lecturas;  
}
```

2) Paso 2: Calibración

A partir de la media de las lecturas del apartado anterior, calculamos los offsets y recalibramos el sensor:

```
//Calculamos offsets  
  
ax_offset=-ax_media/8;  
ay_offset=-ay_media/8;  
az_offset=(16384-az_media)/8;  
  
gx_offset=-gx_media/4;  
gy_offset=-gy_media/4;  
gz_offset=-gz_media/4;  
  
//Recalibramos dispositivo  
  
accelgyro.setYAccelOffset(ay_offset);  
accelgyro.setZAccelOffset(az_offset);  
  
accelgyro.setXGyroOffset(gx_offset);  
accelgyro.setYGyroOffset(gy_offset);  
accelgyro.setZGyroOffset(gz_offset);
```

Posteriormente volvemos a realizar una lectura de las medidas del sensor.

Si se encuentran dentro del margen de error establecido por el usuario, la variable “correcto” se incrementa en 1. De lo contrario recalculamos el offset:

```
if (abs(ax_media)<=error_acelerometro) correcto++;  
else ax_offset=ax_offset-ax_media/error_acelerometro;
```



David Vara Rodríguez

Cuando se llegue a un número establecido de medidas correctas, la calibración habrá terminado y se muestran los resultados por pantalla.

3) Paso 3: Mostrar resultados

```
Serial.println("\nLas lecturas se muestran de la siguiente  
forma: acelX acely acelZ giroX giroY giroZ");
```

```
Serial.println("Comprobar que las lecturas son proximas a  
0 0 16384 0 0 0");
```

```
COM14  
Presione cualquier tecla para comenzar  
  
Calibracion del MPU6050  
  
Poner el MPU6050 en posicion horizontal con las letras hacia arriba.  
No tocar hasta que finalice el programa .  
  
Conexion correcta  
  
Leyendo sensores por primera vez...  
  
Calculando offsets...  
...  
...  
...  
  
FIN DEL PROGRAMA  
  
Lectura de los sensores: 0 0 16378 0 0 0  
Offsets: -139 -468 1427 89 -30 -37  
  
Las lecturas se muestran de la siguiente forma: acelX acely acelZ giroX giroY giroZ  
Comprobar que las lecturas son proximas a 0 0 16384 0 0 0
```

Figura 59. Ejemplo de ejecución de la calibración del dispositivo

7.3 VENTANA DE FILTRADO

Cuando no existe movimiento, pequeños errores en las medidas del acelerómetro pueden ser consideradas como una velocidad constante.

Esa velocidad constante indicia un movimiento continuo y por tanto una variación de la posición que no coincide con la realidad.

Incluso con el calibrado anterior del dispositivo, algunos datos pueden ser erróneos. Por tanto implementamos una “ventana” de discriminación entre los datos “válidos” y “no válidos” para verificar la condición de movimiento.

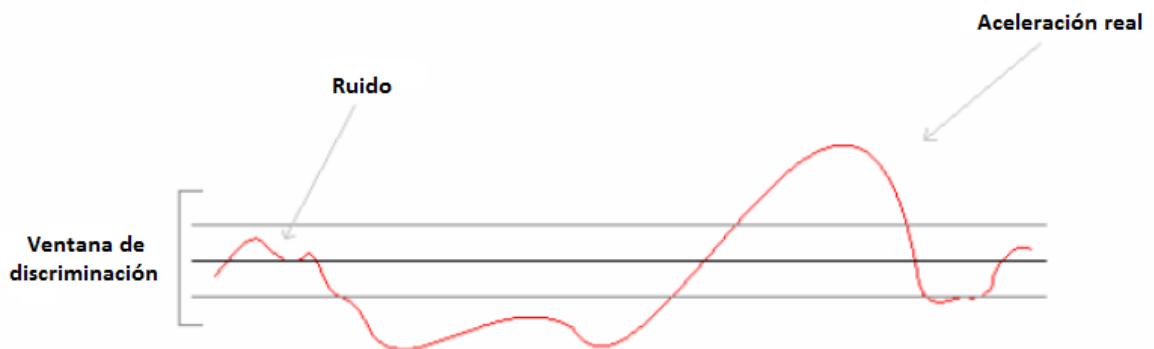


Figura 60. Ventana de filtrado

Extracto de código en Arduino:

```
void ventana(){  
  
  if(ax>(-200) && ax<(200))  
    ax=0;  
  if(ay>(-200) && ay<(200))  
    ay=0;  
  if(az>(-300) && az<(300))  
    az=0;  
  
}
```



David Vara Rodríguez

7.4. FILTRO DE KALMAN

El filtro de Kalman es un algoritmo de estimación recursivo discreto capaz de separar el ruido, de la estimación del estado del sistema.

Para nuestro caso, el objetivo es filtrar la aceleración dada por nuestro sensor con el fin posterior de obtener una posición con el menor error posible.

Como ya se explicó en capítulos anteriores, tenemos un modelo que puede ser considerado lineal y que registrará el proceso del cálculo de la aceleración:

$$aceleracion_k = aceleracion_{k-1} + u_k \cdot dt - w_{k-1} \cdot dt$$

donde:

u_k = señal de entrada

dt = tiempo de muestreo

$aceleracion_k$ = estado estimado

$aceleracion_{k-1}$ = estado anterior

w_{k-1} = perturbación en el proceso

Como no hay señal de entrada al proceso y consideramos que la perturbación es nula, nuestra ecuación de estado del sistema queda de la siguiente forma:

$$aceleracion_k = aceleracion_{k-1}$$

es decir, la diferencia de tiempos entre dos instantes de muestreo es tan breve, que la aceleración se supone constante.



David Vara Rodríguez

El acelerómetro a su vez, se comporta como un observador del sistema que nos permite determinar el error entre el estado estimado y la observación en un instante de tiempo determinado. El modelo del observador es el siguiente:

$$z_k = Hx_k + v_k$$

Se puede apreciar que la medida entregada por el acelerómetro presenta un margen de error (v_k) con respecto al valor estimado:

$$v_k = z_k - Hx_k$$

Con todo lo anteriormente expuesto, el modelo propuesto para la aplicación del algoritmo del filtro de Kalman es el siguiente:

$$aceleracion_k = aceleracion_{k-1}$$

$$z_k = Hx_k + v_k$$



7.4.1 IMPLEMENTACIÓN DEL FILTRO DE KALMAN

Dadas las ecuaciones del algoritmo de Kalman y el modelo del sistema propuesto en el apartado anterior, se presenta la siguiente implementación del filtro:

a) Etapa de predicción

- Dada la ecuación:

$$\hat{x}_{k-} = A\hat{x}_{k-1} + Bu_k$$

En función del modelo del sistema tenemos:

$$aceleracion_k = aceleracion_{k-1}$$

Donde:

$$A = 1$$

$$B = 0$$

$$\hat{x}_k = aceleracion_k$$



David Vara Rodríguez

- Dada la ecuación:

$$P_{k^-} = AP_{k-1}A^T + Q$$

P_k se inicializa como matriz identidad

$Q = 0.001$. Los valores de las matrices de covarianza Q y R son desarrollados en el apartado 7.3.1.1.

b) Etapas de corrección:

- Dada la ecuación:

$$K_k = P_{k^-}H^T(HP_{k^-}H^T + R)^{-1}$$

H representa la relación entre las medidas y el estado del sistema.
Por tanto: $H = 1$

$$R = 0.07$$

- Dadas las ecuaciones:

$$\hat{x}_k = \hat{x}_{k^-} + K_k(z_k - H\hat{x}_{k^-})$$

$$P_k = (I - K_kH)P_{k^-}$$

z_k representa las medidas del acelerómetro

I = matriz identidad



David Vara Rodríguez

7.4.1.1 VALORES DE LAS MATRICES DE COVARIANZA Q Y R

Los valores de las matrices de covarianza Q y R fueron obtenidos a partir de una búsqueda bibliográfica de los mismos y diversas pruebas realizadas.

Los mejores resultados fueron conseguidos con $Q = 0.001$ y $R = 0.07$.

Un ejemplo de ello se presenta a continuación. En ambas pruebas el movimiento realizado con el dispositivo se corresponde al de un rectángulo:

Prueba 1

$R=0.05$; % covarianza de los ruidos de las señales medidas $z(t)$

$Q=0.001$; % covarianza de los ruidos en el sistema $x(t)$

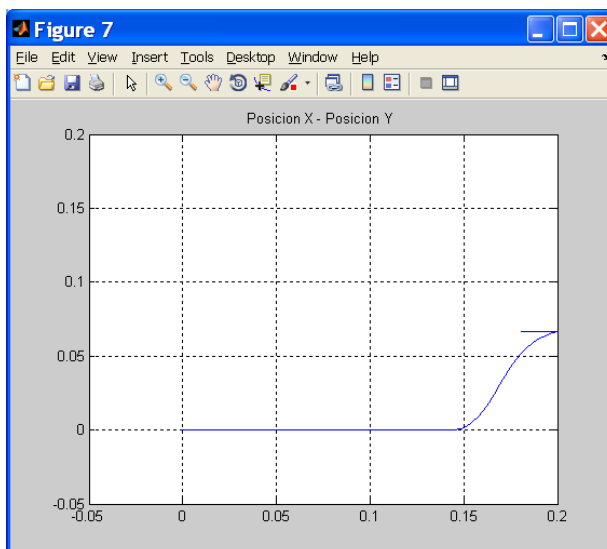
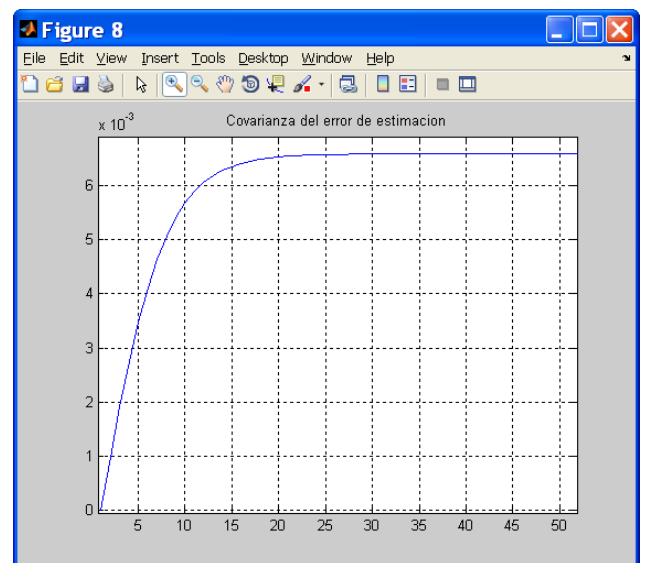
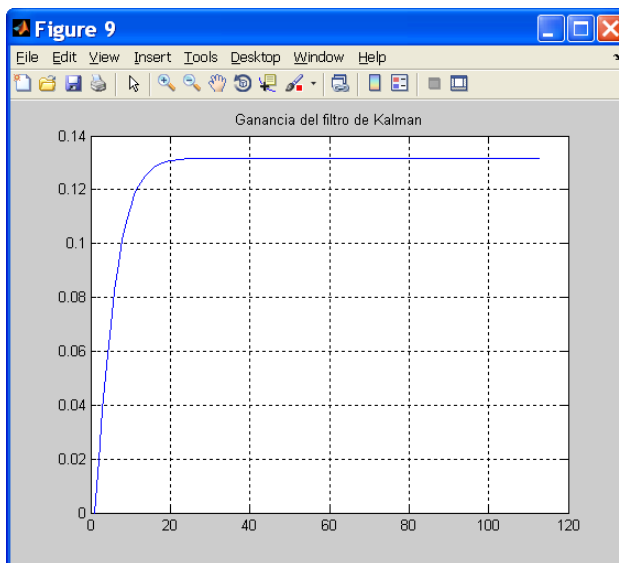


Figura 61. Resultados de la prueba 1 para $R=0.05$ y $Q=0.001$

Se puede observar en la gráfica *Posición X – Posición Y*, como esta muestra un resultado completamente desacorde con el movimiento inicialmente descrito (rectángulo).



David Vara Rodríguez

Prueba 2

$R=0.07$; % covarianza de los ruidos de las señales medidas $z(t)$

$Q=0.001$; % covarianza de los ruidos en el sistema $x(t)$

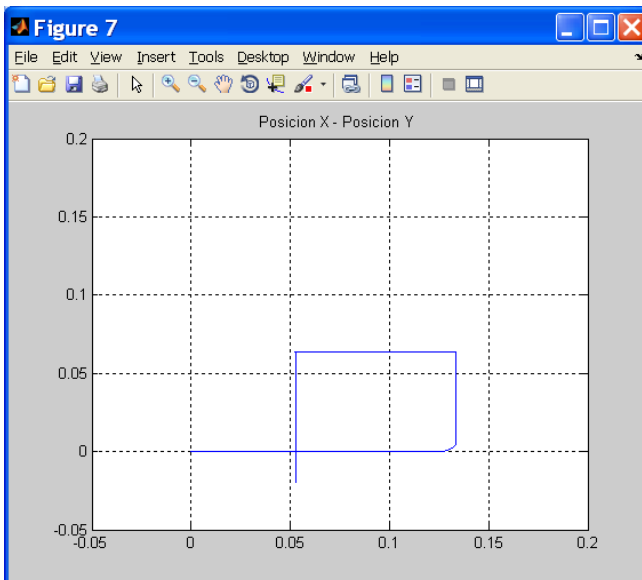
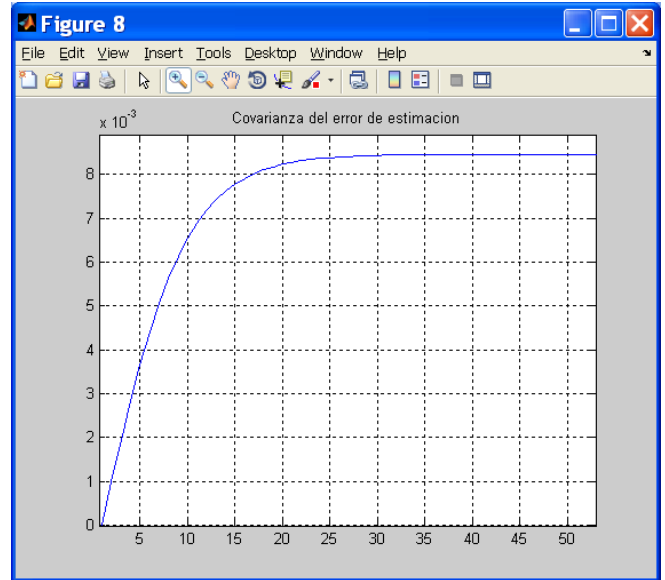
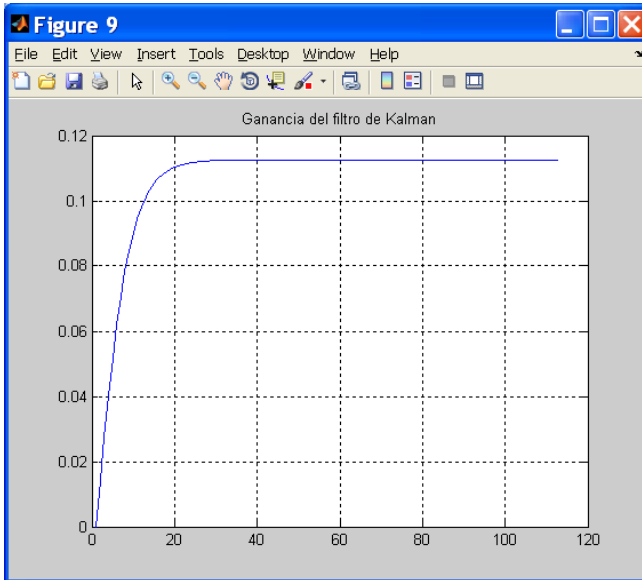


Figura 62. Resultados de la prueba 2 para $R=0.07$ y $Q=0.001$

Se comprueba como para $Q = 0.001$ y $R = 0.07$ el resultado sí refleja el movimiento descrito.



7.5 FILTRO DE KALMAN VS FILTRO MEDIA MÓVIL

En este apartado comparamos el Filtro de Kalman, con otro muy utilizado en la eliminación del ruido de los datos del acelerómetro: el filtro de la media móvil [23].

Éste último tiene como ventaja principal, una mayor simplicidad y menor coste computacional que el de Kalman.

El filtro de la media móvil, efectúa un filtrado de la señal de entrada mediante la realización de la media aritmética entre la muestra actual y las muestras anteriores. El número de muestras anteriores depende del orden del filtro, así para un filtro de orden 4 se escogen las cuatro lecturas previas, para un filtro de orden 5 las cinco lecturas anteriores, etc.

Dicho filtro es fácilmente implementado en la aplicación MATLAB mediante el comando ***filter()***.

Sea $y = \text{filter}(b, a, X)$;

este comando crea un vector y de datos filtrados mediante el procesamiento de las lecturas en el vector X con el filtro descrito por los valores de los vectores a y b .

La función de filtrado puede describirse a partir como una ecuación en diferencias:

$$a(1)y(n) = b(1)x(n) + b(2)x(n-1) + \dots + b(Nb)x(n-Nb+1) - a(2)y(n-1) - \dots - a(Na)y(n-Na+1)$$

donde n es el índice de la muestra actual, Na el orden del polinomio descrito por el vector a y Nb el orden del polinomio descrito por el vector b . La salida $y(n)$ es una combinación lineal de las entradas actuales y anteriores, $x(n) x(n-1) \dots$ y las salidas anteriores, $y(n-1)y(n-2)$.



David Vara Rodríguez

Para un filtro de orden 4 como el utilizado en nuestro proyecto:

$$a = 1;$$

$$b = \left[\frac{1}{4} \frac{1}{4} \frac{1}{4} \frac{1}{4} \right]$$

$$y(n) = \frac{1}{4}x(n) + \frac{1}{4}x(n-1) + \frac{1}{4}x(n-2) + \frac{1}{4}x(n-3)$$

Se han realizado diversas pruebas empíricas para la comparación entre ambos filtros. Una de ellas, así como los resultados obtenidos se muestran en la página siguiente.



David Vara Rodríguez

La prueba consiste en la medición de 10cm recorridos a lo largo del eje x del dispositivo.

Para el **filtro de Kalman**, la aceleración y posición conseguidas fueron:

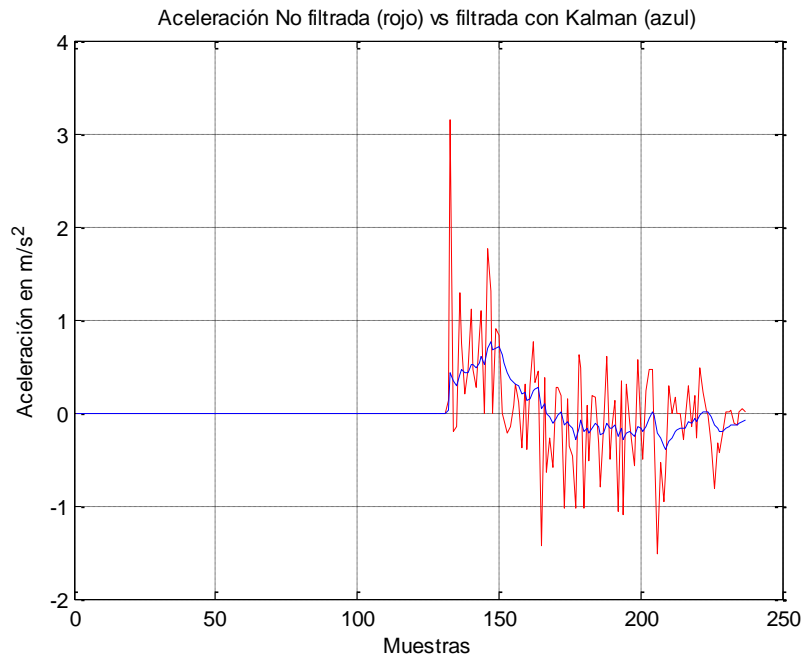


Figura 63. Aceleración obtenida aplicando el filtro de Kalman

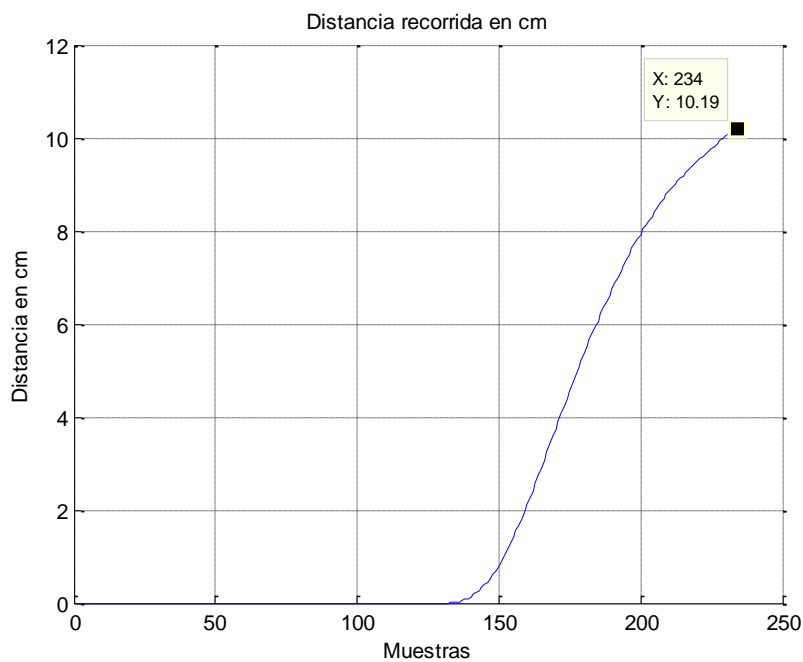


Figura 64. Posición obtenida con el filtro de Kalman



David Vara Rodríguez

Para el **filtro de la media móvil**, la aceleración y posición conseguidas fueron:

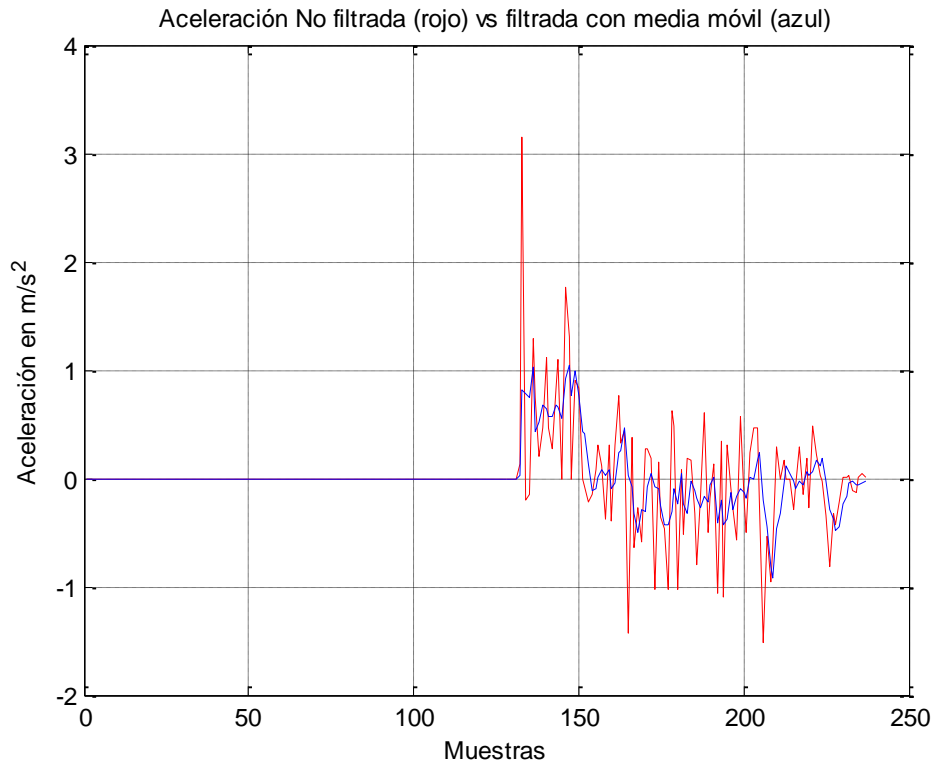


Figura 65. Aceleración obtenida aplicando el filtro de la media móvil

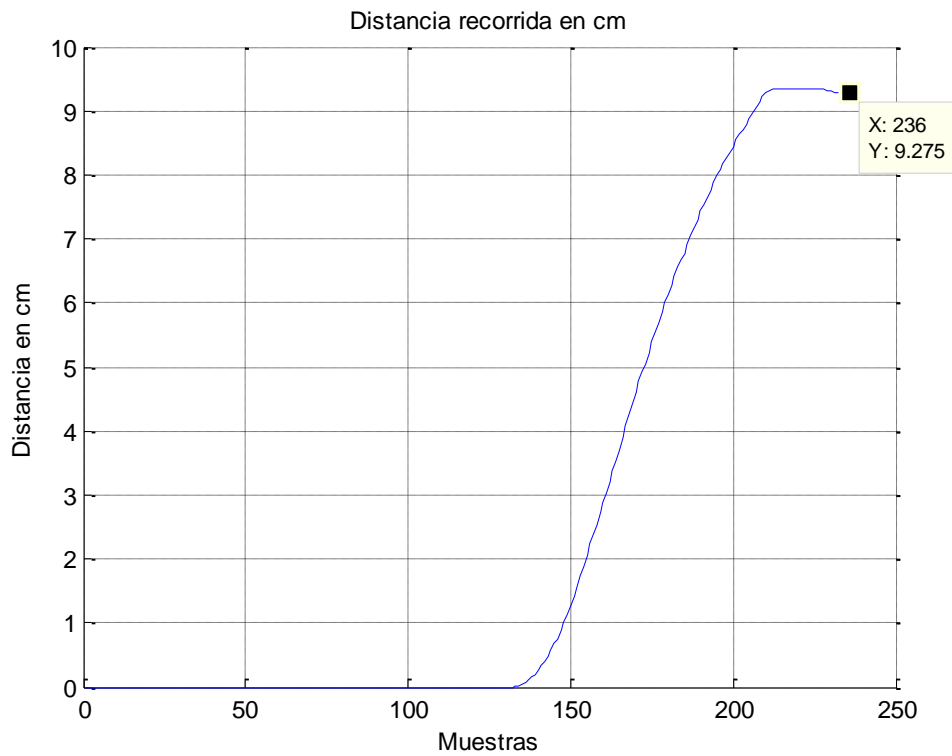


Figura 66. Posición obtenida aplicando el filtro de la media móvil



David Vara Rodríguez

Cuadro resumen:

	Error absoluto posición (cm)	Error relativo posición (%)
<u>Filtro de Kalman</u>	0.19	1.9
<u>Filtro de la media móvil</u>	0.725	7.25

Tabla 7. Filtro de Kalman vs Filtro de la media móvil

Conclusión:

Tras realizar diversas pruebas, se comprueba que el filtro de la media móvil proporciona un error entre dos y tres veces mayor que el filtro de Kalman.

A la vista de estos resultados, el filtro de Kalman es el seleccionado.



7.6 CÁLCULO DE LA VELOCIDAD Y LA POSICIÓN

Una vez filtrados los valores de la aceleración, se procede a la integración de los mismos para hallar la velocidad y la posición.

Dicho cálculo se realiza mediante el método del trapecio cuyos motivos y principios ya fueron explicados con detalle en capítulos anteriores.

Dada la aceleración y el tiempo en el que se produce la lectura en un instante k , ($aceleracion_k$ y $tiempo_k$ respectivamente), la velocidad en ese instante, $velocidad_k$, puede expresarse como:

$$velocidad_k = velocidad_{k-1} + 0.5 \cdot (tiempo_k - tiempo_{k-1}) \cdot (aceleracion_k + aceleracion_{k-1})$$

Una vez calculada la velocidad, del mismo modo la posición en dicho instante k puede expresarse como:

$$posicion_k = posicion_{k-1} + 0.5 \cdot (tiempo_k - tiempo_{k-1}) \cdot (velocidad_k + velocidad_{k-1})$$



David Vara Rodríguez

7.7 DISTINCIÓN ENTRE VELOCIDAD NULA Y CONSTANTE PARA ACELERACIÓN NULA

Cuando la aceleración es nula en un momento determinado, se nos presenta el problema de distinguir entre si la velocidad también es y consecuentemente hallar si existe una variación o no en la posición en dicho instante de tiempo.

Como el movimiento realizado por nuestro dispositivo se va a efectuar de forma manual y el acelerómetro tiene una gran sensibilidad, suponemos que no existe velocidad constante, es decir, que para velocidad nula, el incremento de posición es cero.

Para un movimiento acelerado y decelerado sobre un solo eje, el cual va a efectuar nuestro dispositivo, las gráficas de aceleración, velocidad y posición serían las siguientes:

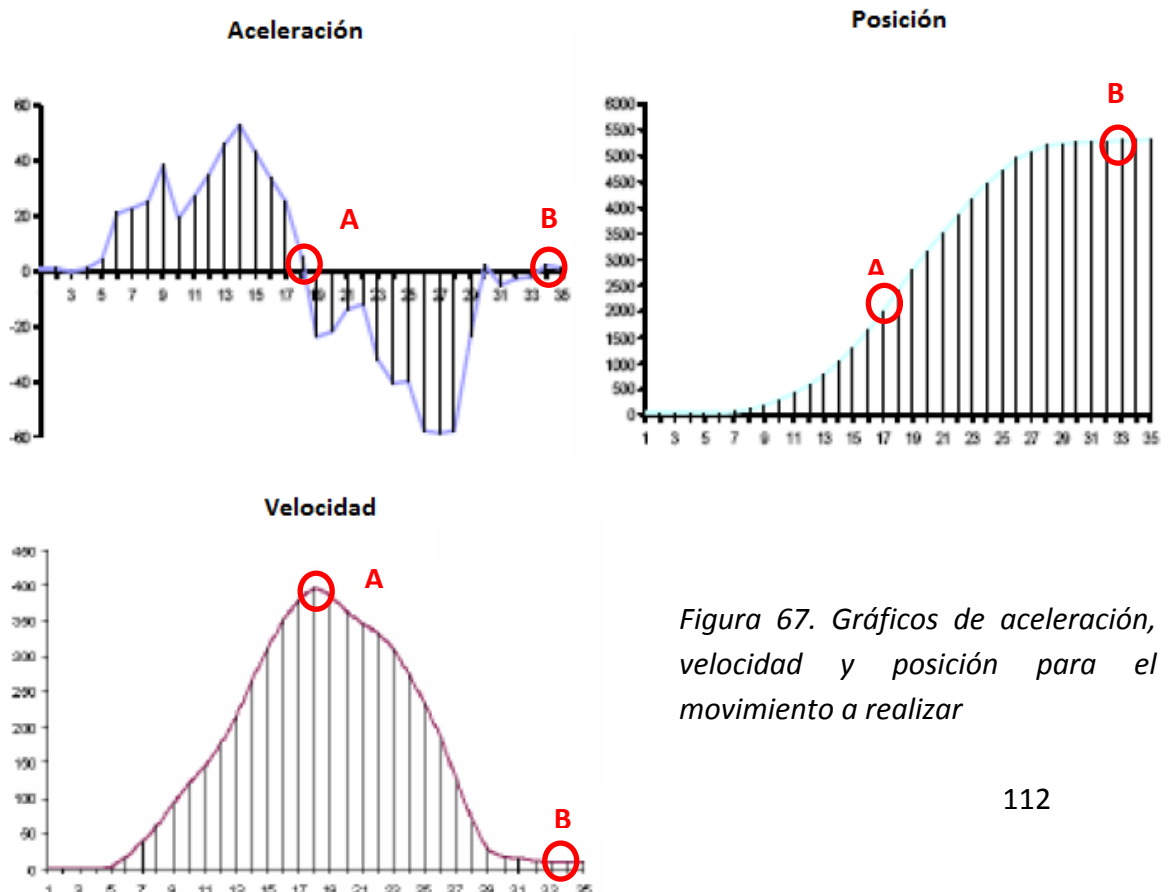


Figura 67. Gráficos de aceleración, velocidad y posición para el movimiento a realizar



David Vara Rodríguez

Podemos observar que cuando la aceleración es nula en el instante A, la velocidad no solamente no es cero, si no que coincide con su valor más alto y existe un incremento en la posición.

En cambio cuando la aceleración vuelve a ser cero en el instante B, la velocidad es prácticamente cero y la posición es constante (no existe variación en la posición).

Así pues podemos concluir que cuando la aceleración es nula, si la velocidad también lo es o se encuentra en valores próximos a cero, no existe variación en la posición del objeto.



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez



Capítulo 5. Pruebas realizadas

1. Introducción al capítulo

En este capítulo se presentan diversos ensayos realizados con el dispositivo una vez ha sido construido y programado.

El objetivo es rastrear distintos movimientos en una, dos y tres dimensiones. Así pues, este capítulo se encuentra subdividido en estos tres apartados en función de los ensayos realizados.

Como resulta imposible presentar la totalidad de los mismos, se mostrarán solamente algunos de ellos de los que podemos extraer más información relevante para el proyecto.

Para cada prueba se muestran:

- Aceleración sin filtrar y con el filtro de Kalman.
- Aceleración filtrada vs velocidad (pruebas en 1D)
- Velocidad del cuerpo
- Posición del cuerpo
- Cuadro resumen con errores absolutos y relativos

Así mismo, al final de cada apartado se presentan las conclusiones principales.

2. Pruebas realizadas en una dimensión (1D)

2.1 PRUEBA 35 CM LINEAL

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	25 cm
Velocidad de ejecución:	Rápida

Tabla 8. Características prueba 35cm lineal

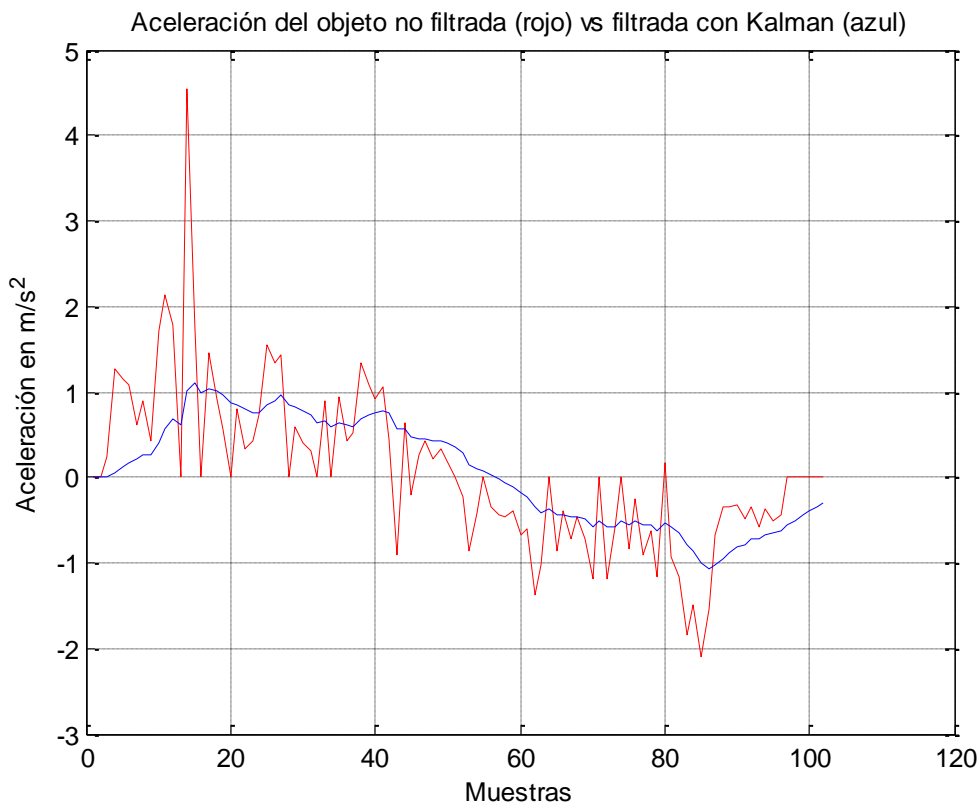


Figura 68. Aceleración prueba 35 cm lineal



David Vara Rodríguez

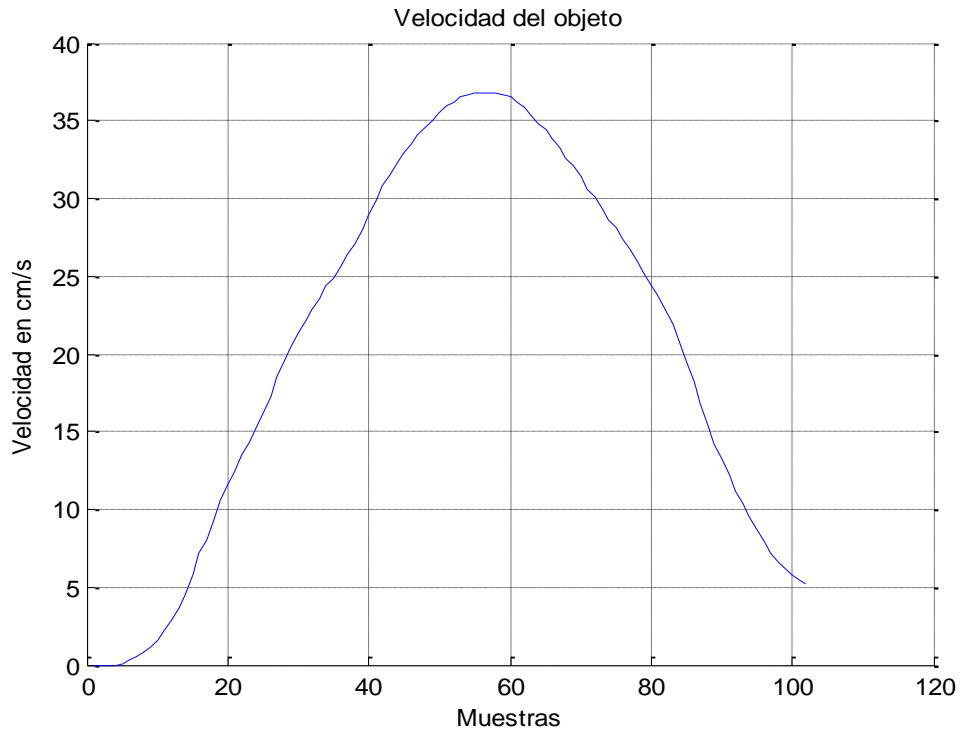


Figura 69. Velocidad prueba 35cm lineal

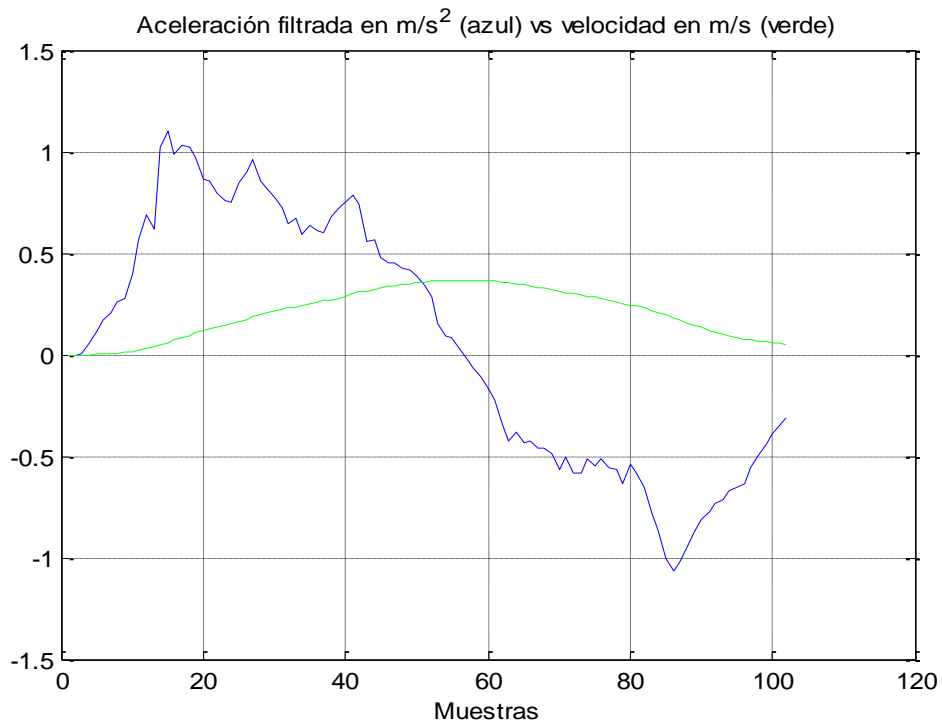


Figura 70. Aceleración vs velocidad prueba 35cm lineal



David Vara Rodríguez

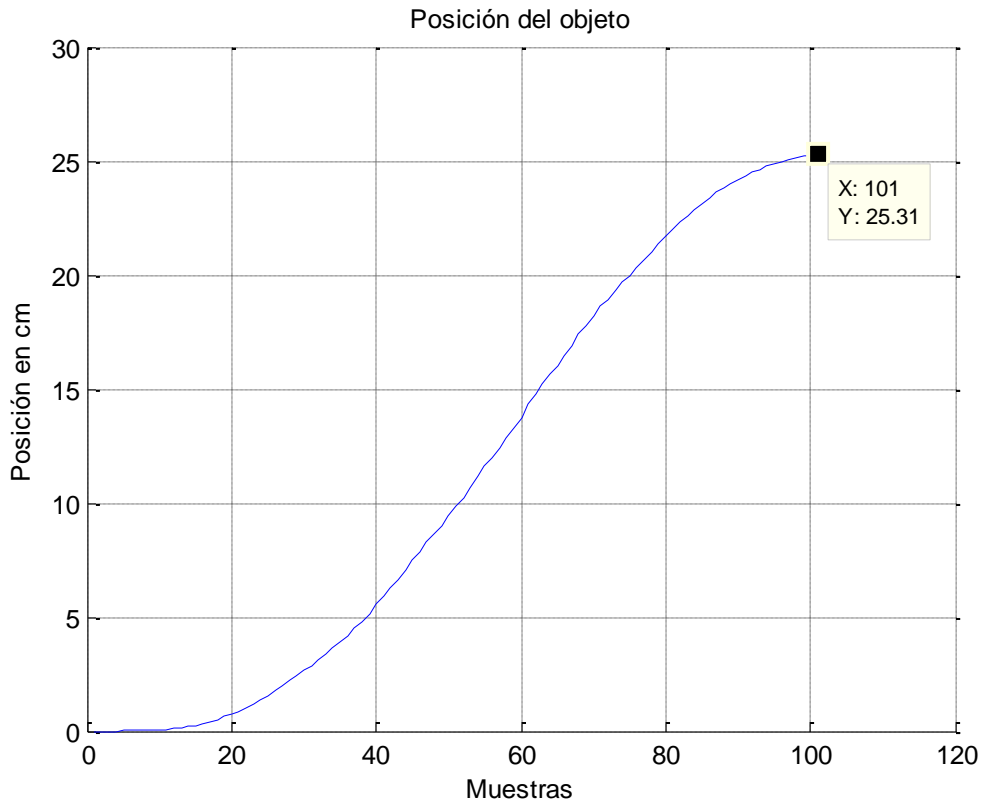


Figura 71. Posición prueba 35 cm lineal

Cuadro resumen	
Distancia teórica	25 cm
Distancia obtenida	25,31 cm
Error absoluto	0,31 cm
Error relativo	1,24%
Velocidad máxima	36,83 cm/s
Aceleración máxima	1,10 m/s^2

Tabla 9. Resultados prueba 35 cm lineal

2.2 PRUEBA 15 CM LINEAL

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	15 cm
Velocidad de ejecución:	Rápida

Tabla 10. Características prueba 15cm lineal

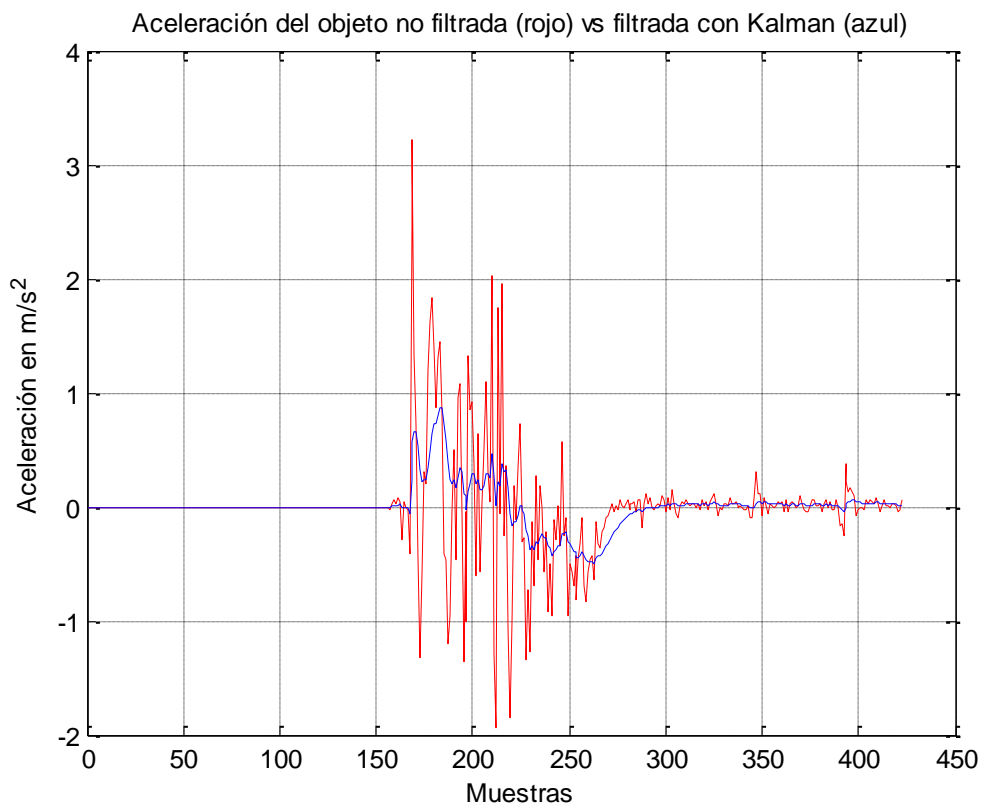


Figura 72. Aceleración prueba 15cm lineal



David Vara Rodríguez

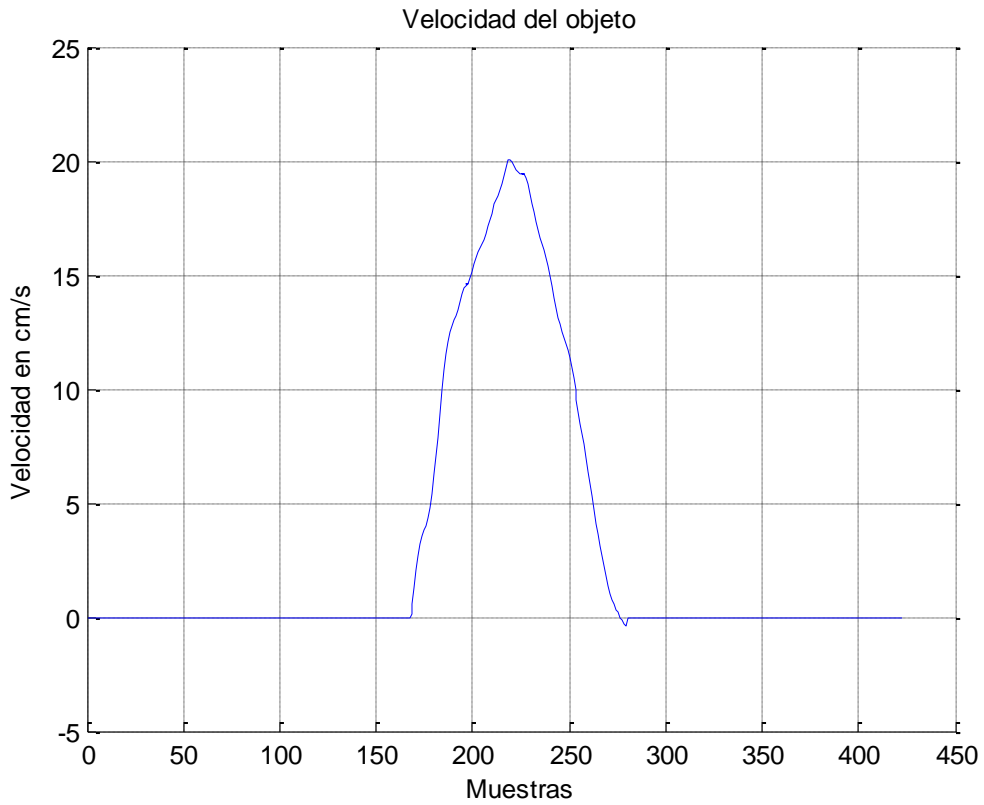


Figura 73. Velocidad prueba 15cm lineal

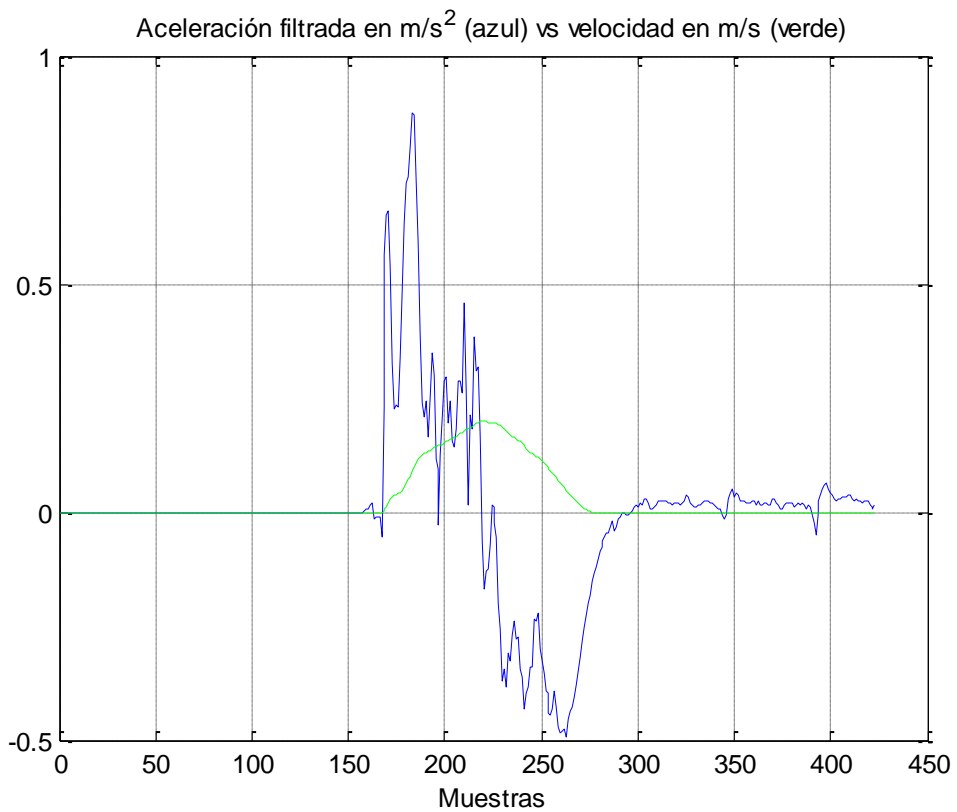


Figura 74. Aceleración vs velocidad prueba 15cm lineal



David Vara Rodríguez

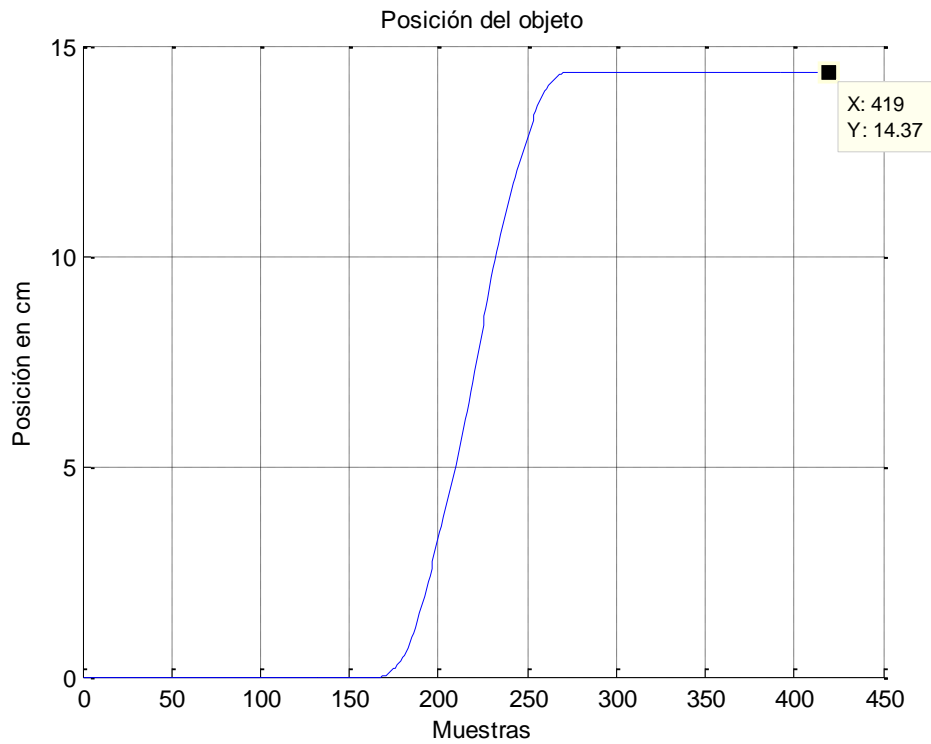


Figura 75. Posición prueba 15cm lineal

<u>Cuadro resumen</u>	
Distancia teórica	15 cm
Distancia obtenida	14,37 cm
Error absoluto	0,63 cm
Error relativo	4,2%
Velocidad máxima	20,06 cm/s
Aceleración máxima	0,87 m/s ²

Tabla 11. Resultados prueba 15cm lineal



David Vara Rodríguez

2.3 PRUEBA 10 CM LINEAL

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	10 cm
Velocidad de ejecución:	Rápida

Tabla 12. Características prueba 10 cm lineal

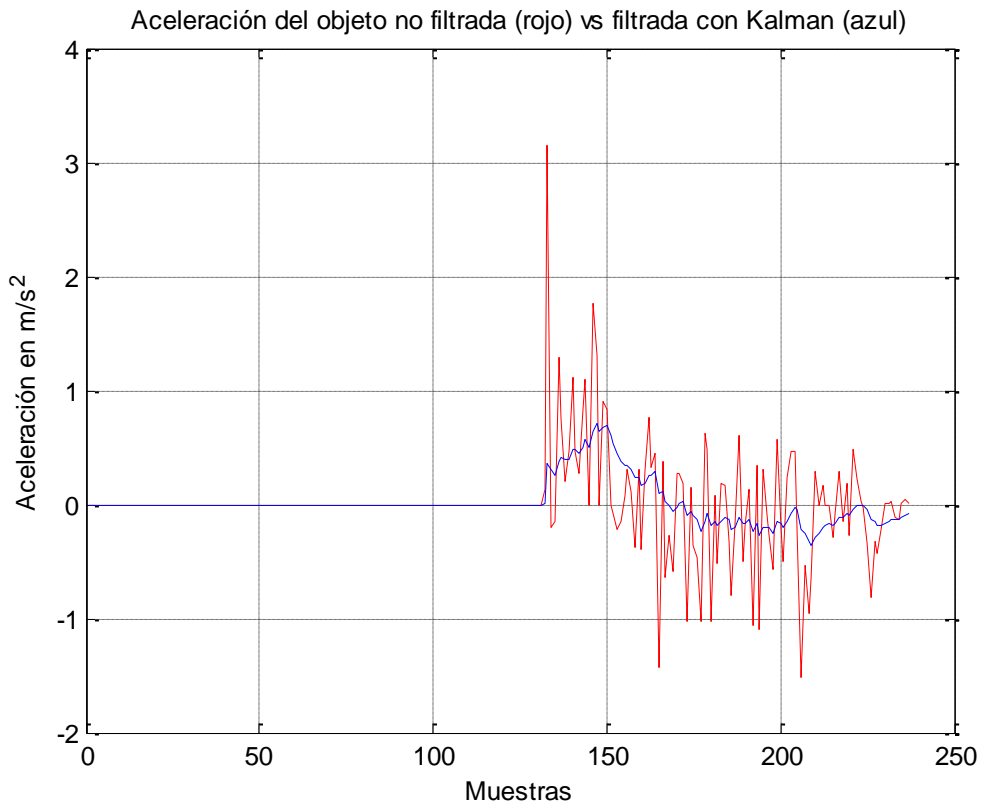


Figura 76. Aceleración prueba 10cm lineal



David Vara Rodríguez

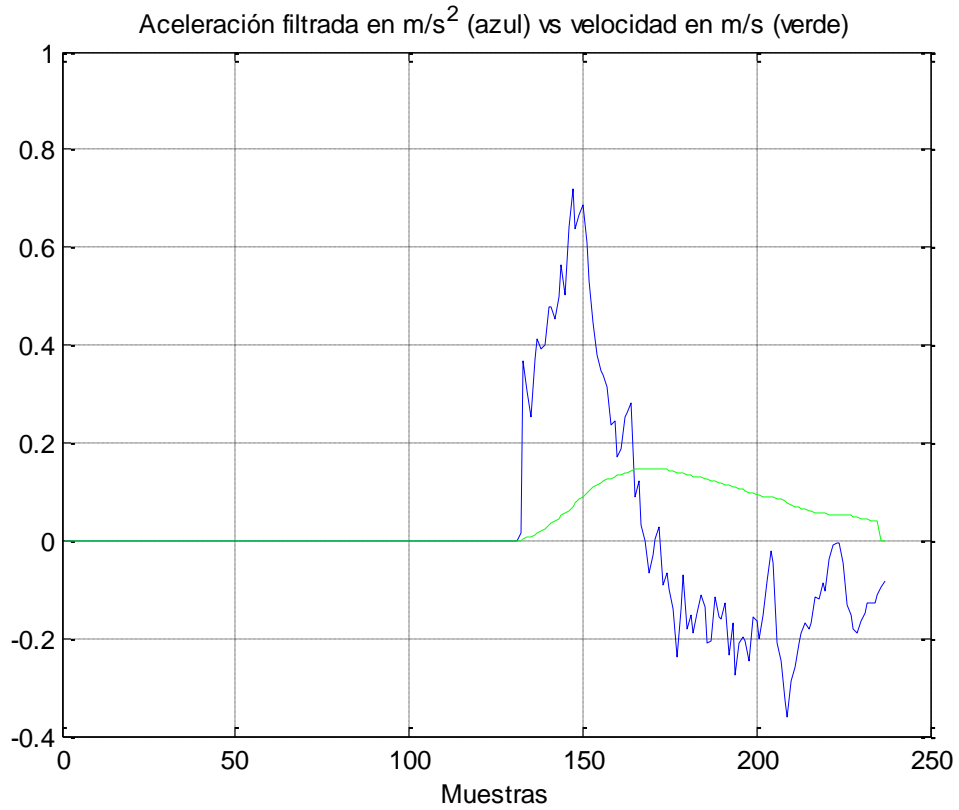


Figura 77. Aceleración vs velocidad prueba 10 cm lineal

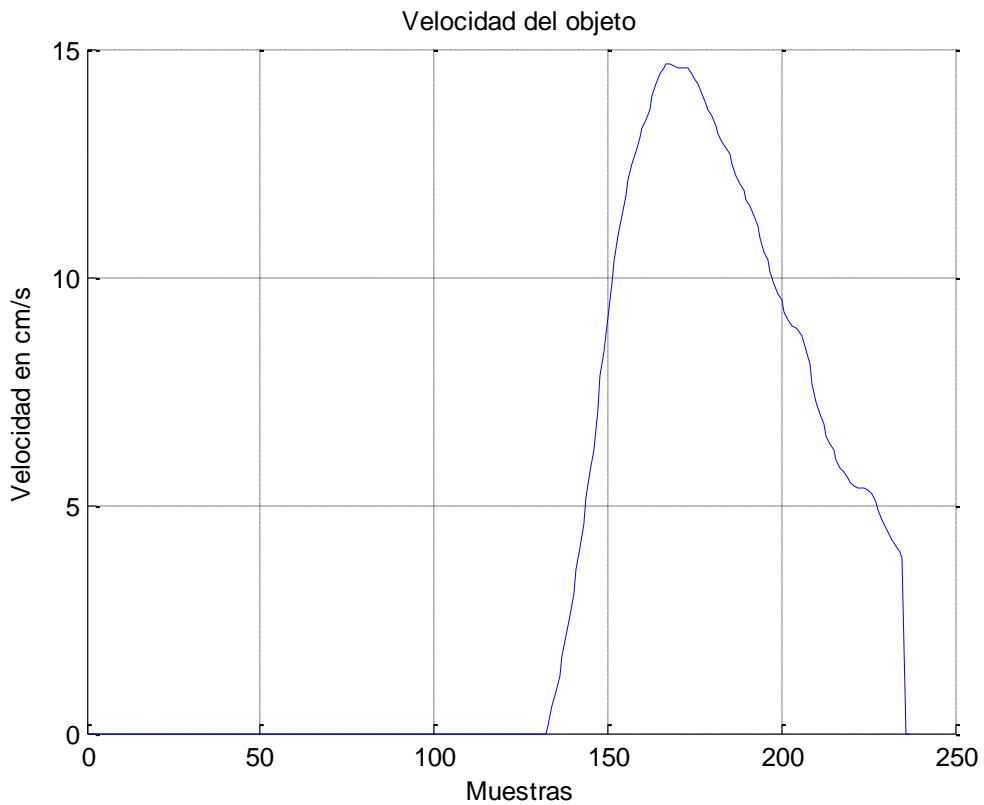


Figura 78. Velocidad prueba 10cm lineal



David Vara Rodríguez

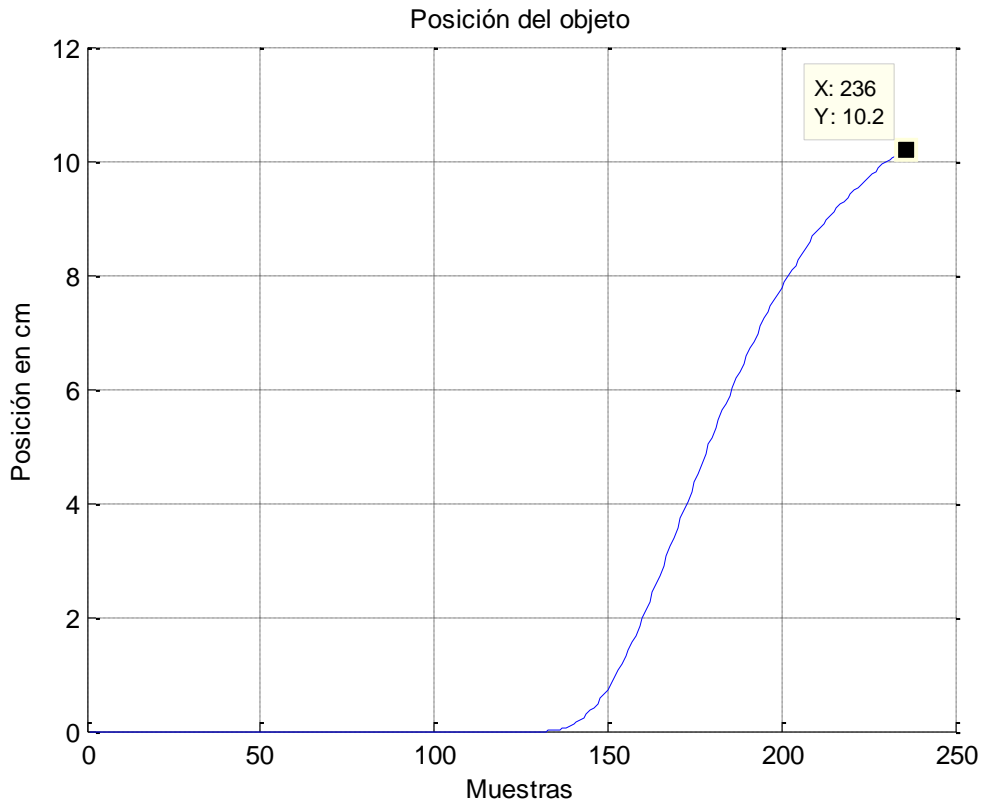


Figura 79. Posición prueba 10cm lineal

<u>Cuadro resumen</u>	
Distancia teórica	10 cm
Distancia obtenida	10,20 cm
Error absoluto	0,20 cm
Error relativo	2 %
Velocidad máxima	14,69 cm/s
Aceleración máxima	0,71 m/s^2

Tabla 13. Resultados prueba 10 cm lineal



David Vara Rodríguez

2.3.1 PRUEBA 10 CM LINEAL VELOCIDAD LENTA

Características del ensayo	
Tipo de movimiento:	Manual
Distancia a medir:	10 cm
Velocidad de ejecución:	Lenta

Tabla 14. Características prueba 10 cm lineal lenta

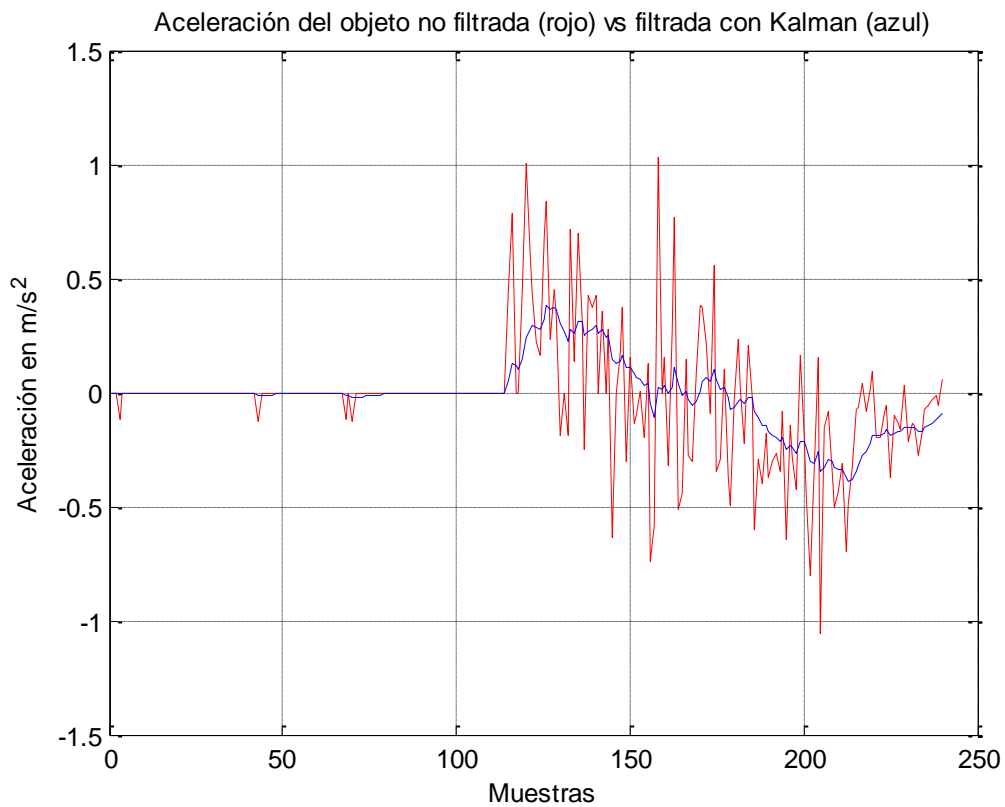


Figura 80. Aceleración prueba 10cm lineal lenta



David Vara Rodríguez

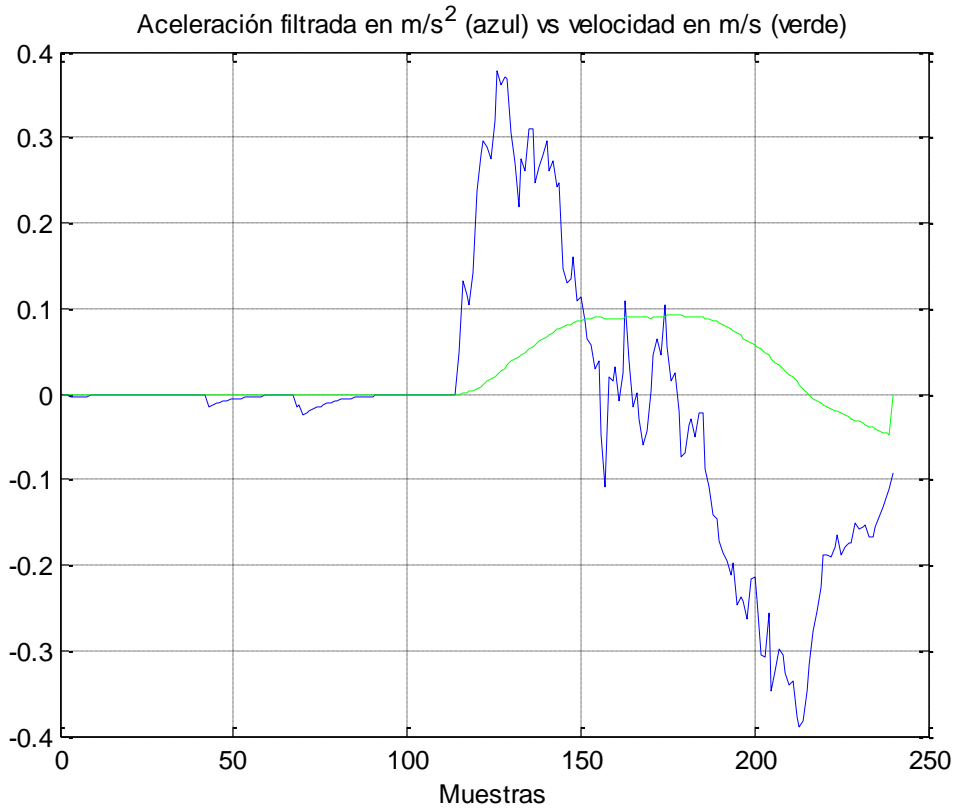


Figura 81. Aceleración vs velocidad prueba 10cm lineal lenta

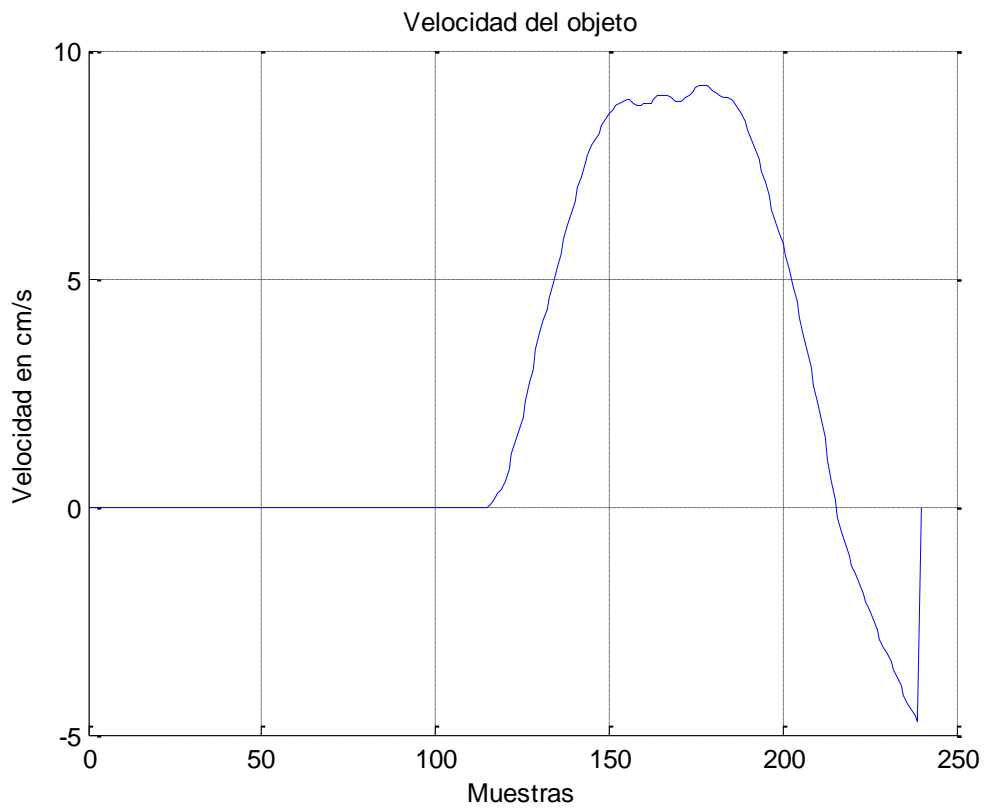


Figura 82. Velocidad prueba 10cm lineal lenta



David Vara Rodríguez

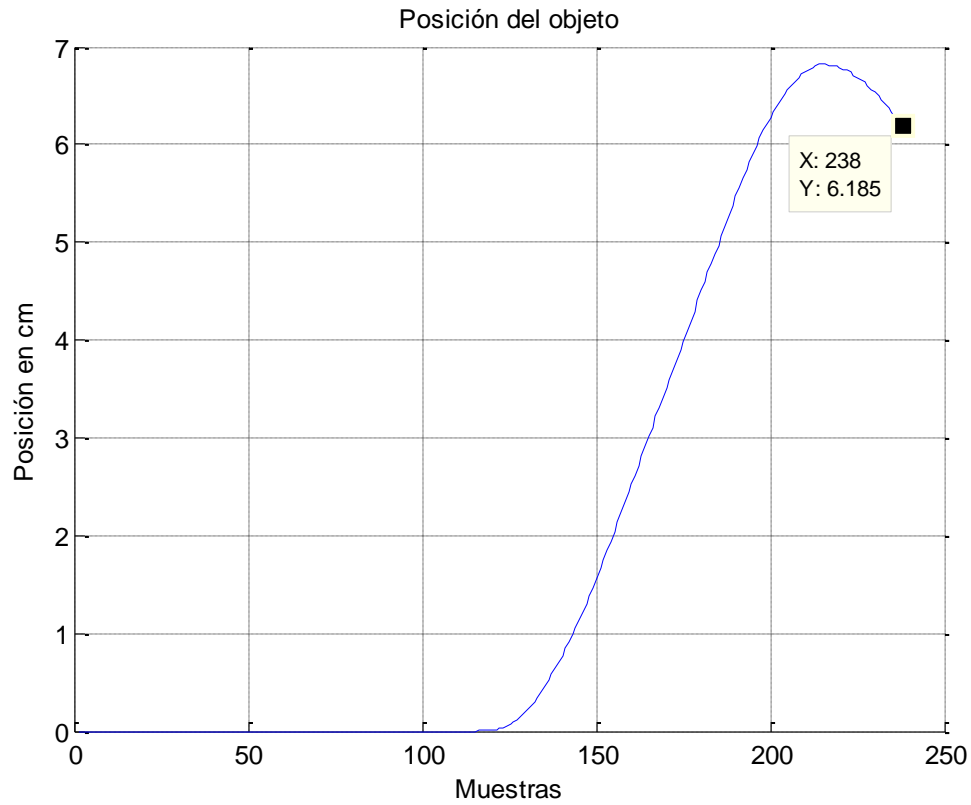


Figura 83. Posición prueba 10 cm lineal lenta

<u>Cuadro resumen</u>	
Distancia teórica	10 cm
Distancia obtenida	6,185 cm
Error absoluto	3,918 cm
Error relativo	38,15 %
Velocidad máxima	9,25 cm/s
Aceleración máxima	0,37 m/s ²

Tabla 15. Resultados prueba 10 cm lineal lenta



David Vara Rodríguez

2.4 PRUEBA 5 CM LINEAL

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	5 cm
Velocidad de ejecución:	Rápida

Tabla 16. Características prueba 5cm lineal

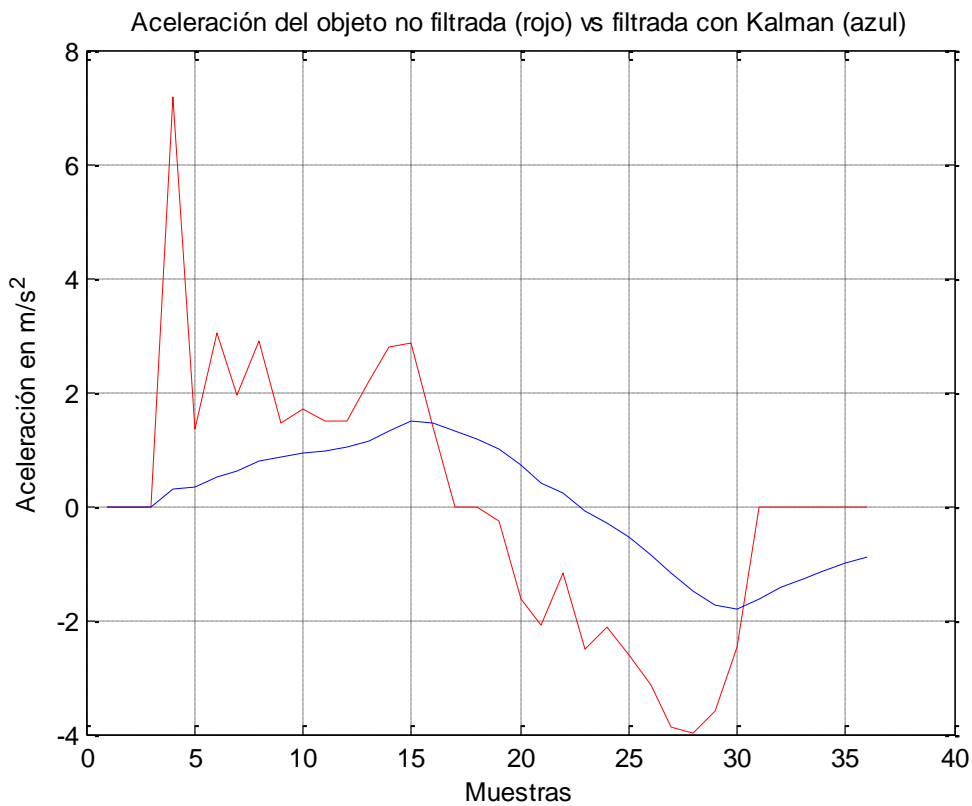


Figura 84. Aceleración prueba 5 cm lineal



David Vara Rodríguez

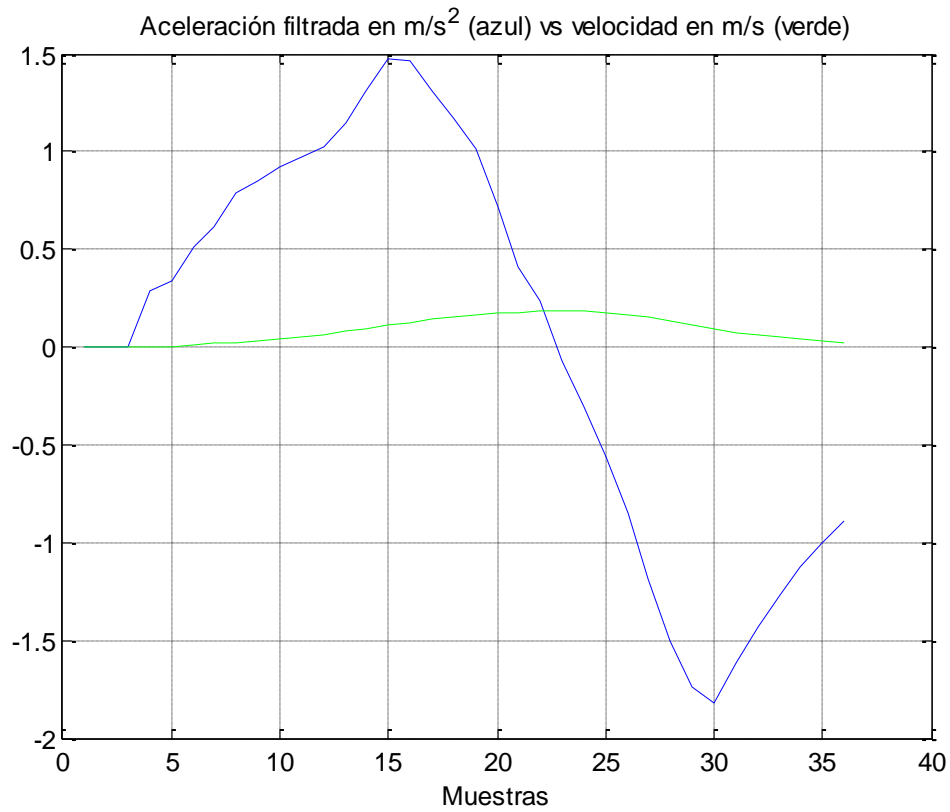


Figura 85. Aceleración vs velocidad prueba 5cm lineal

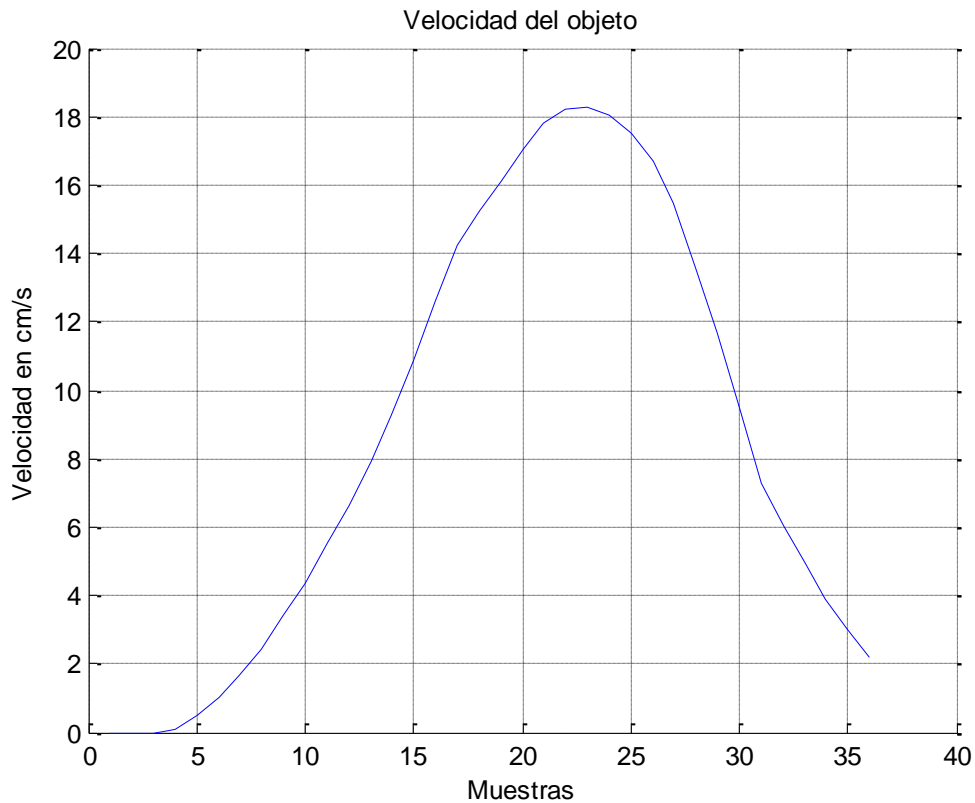


Figura 86. Velocidad prueba 5cm lineal



David Vara Rodríguez

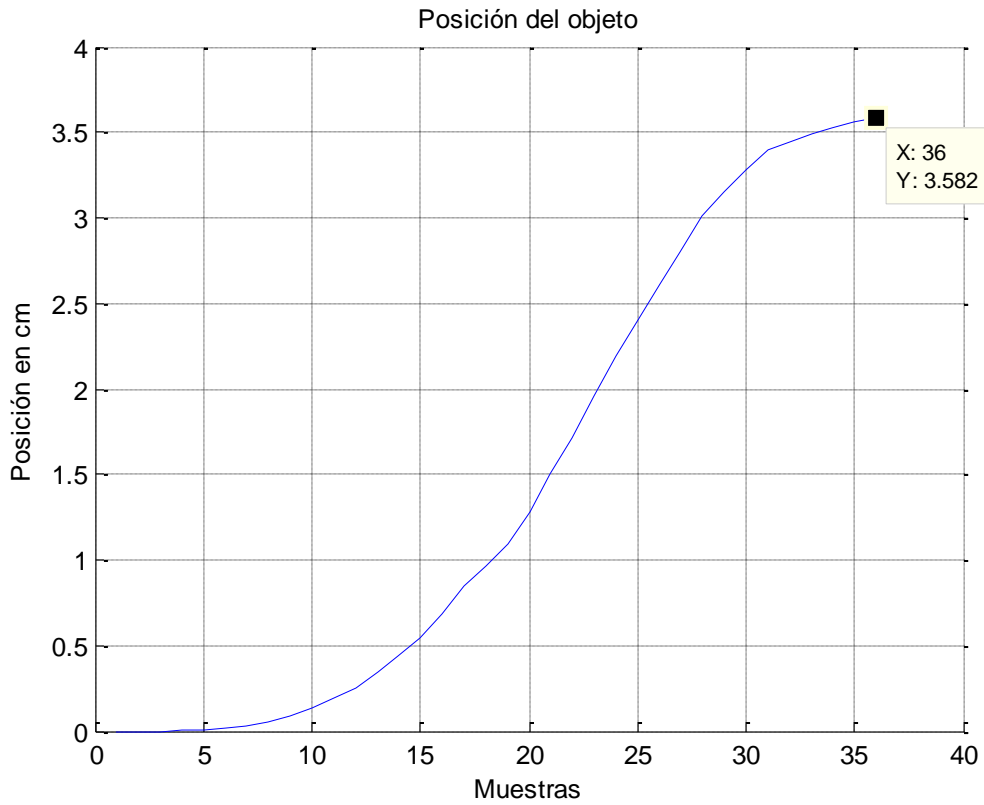


Figura 87. Posición prueba 5cm lineal

Cuadro resumen	
Distancia teórica	5 cm
Distancia obtenida	3,582 cm
Error absoluto	1,418 cm
Error relativo	28,36 %
Velocidad máxima	18,29 cm/s
Aceleración máxima	1,47 m/s^2

Tabla 17. Resultados prueba 5 cm lineal



David Vara Rodríguez

2.5 Comentarios de las pruebas en 1D

- Podemos observar que el error aumenta cuanto menor es la distancia a medir. Así pues, para una medida de 25 cm se obtiene un error del 1,24% mientras que para 5 cm el error alcanza una cota del 28,36%.
- Así mismo, podemos comprobar la dependencia del error con la velocidad del movimiento. Para una medida de 10 cm ejecutada de forma rápida obtenemos un error de un 2% mientras que el mismo movimiento realizado de forma lenta reporta un error del 38,15%.

David Vara Rodríguez

3. Pruebas realizadas en dos dimensiones (2D)

3.1 DIAGONAL

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	10 cm. Ángulo de inclinación: 50º
Velocidad de ejecución:	Rápida

Tabla 18. Características prueba diagonal 10 cm

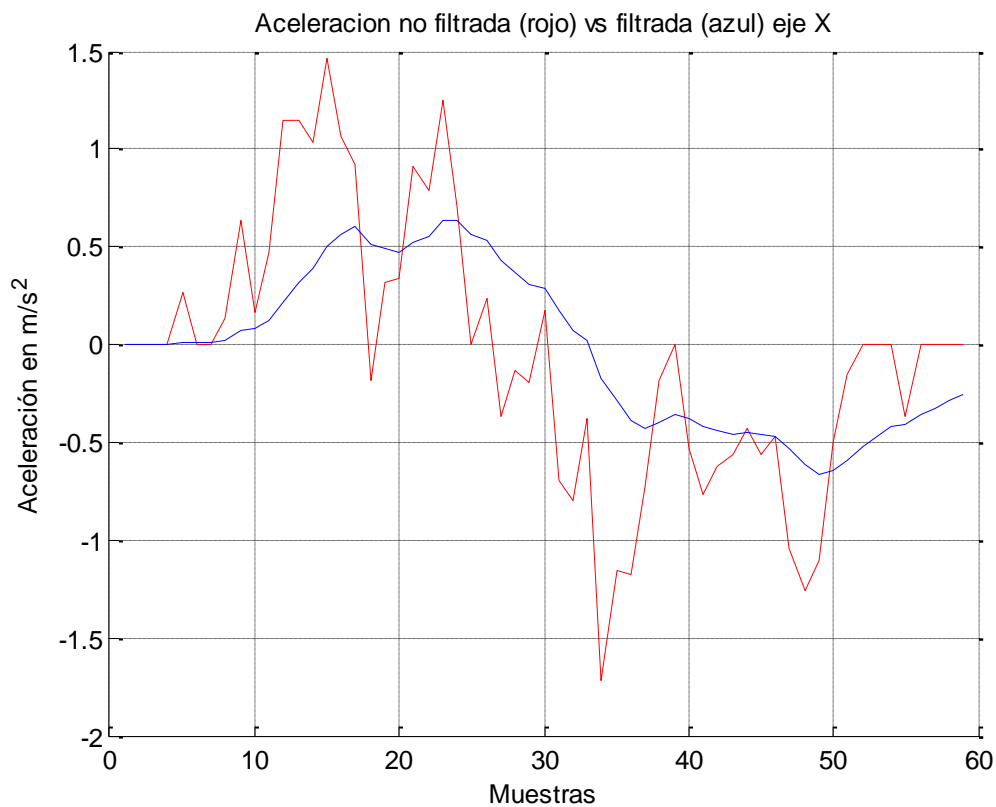


Figura 88. Aceleración eje X prueba diagonal 10 cm

David Vara Rodríguez

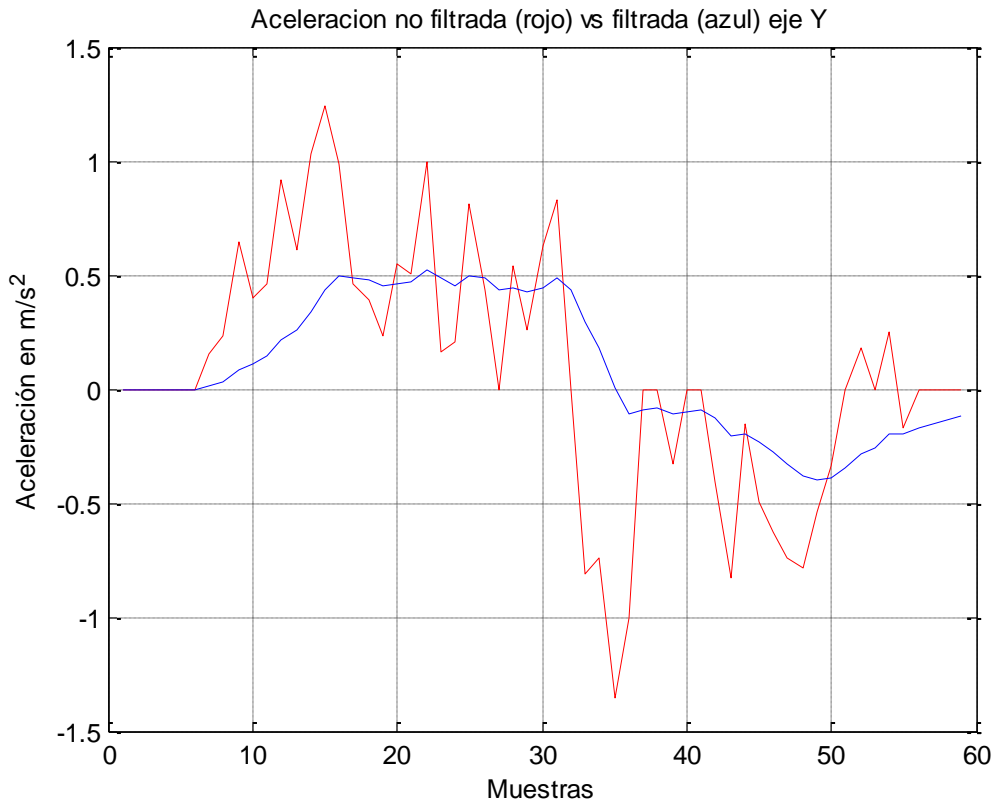


Figura 89. Aceleración eje Y prueba diagonal 10 cm

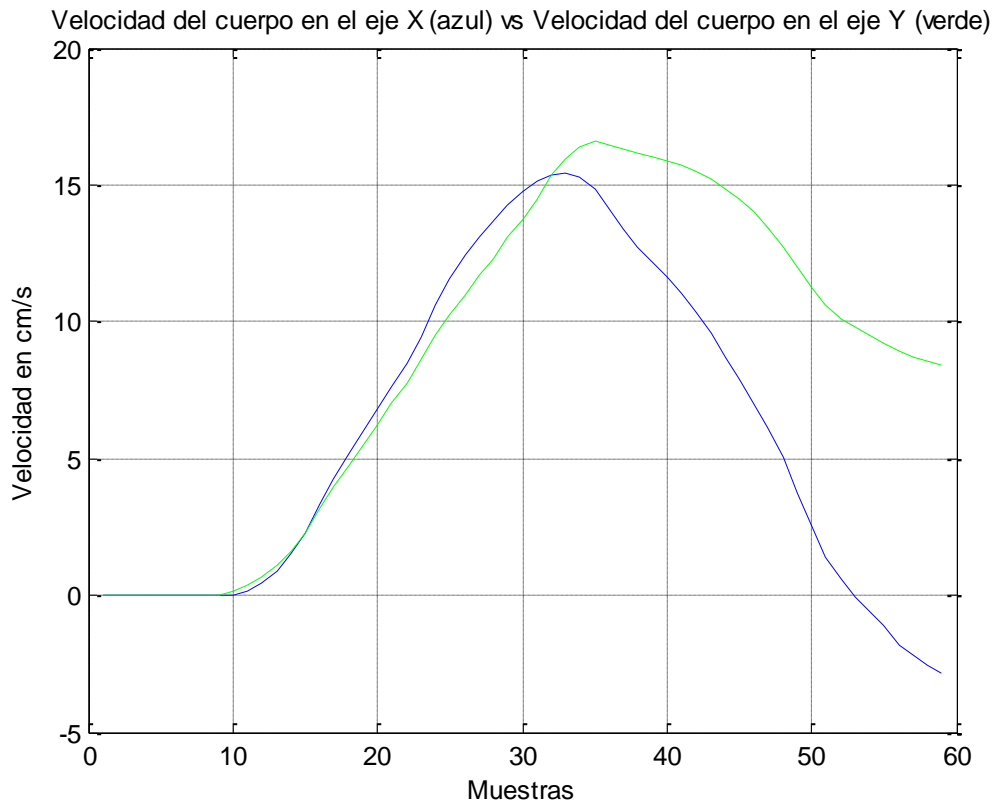


Figura 90. Velocidad prueba diagonal 10 cm



David Vara Rodríguez

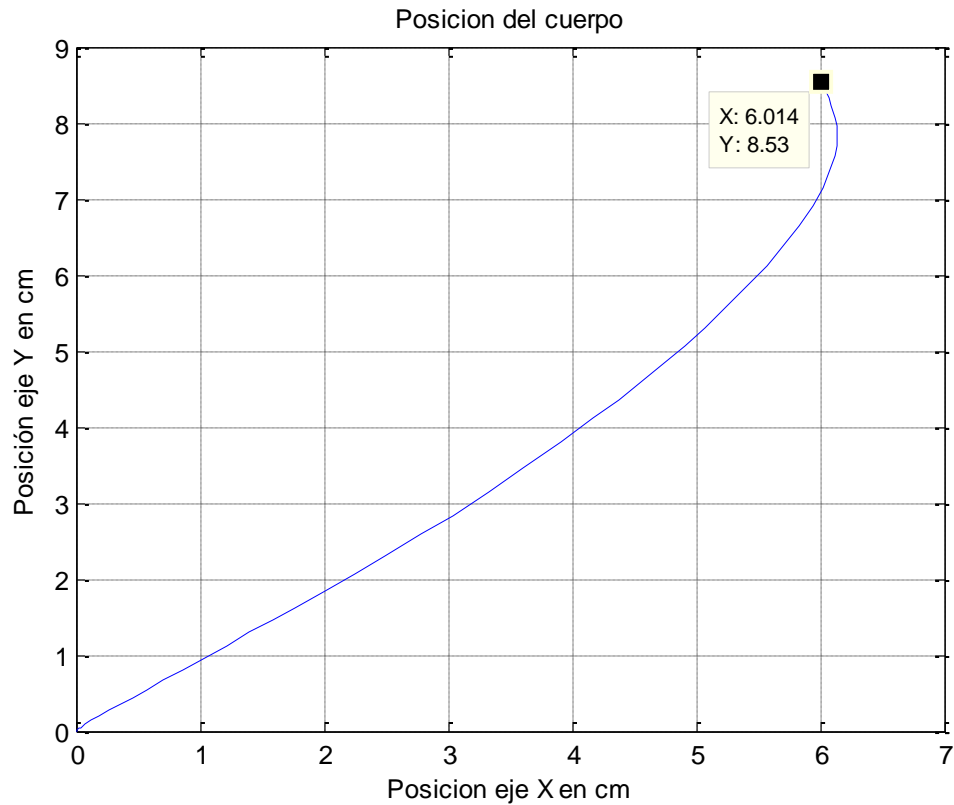


Figura 91. Posición prueba diagonal 10 cm

Cuadro resumen	
Distancia teórica	10 cm - Inclinación 50° Eje x = 6,42 cm Eje y = 7,66 cm
Distancia obtenida	Eje x = 6,014 cm Eje y = 8,53 cm
Error absoluto	Eje x = 0,406 cm Eje y = 0,87 cm
Error relativo	Eje x = 6,32 % Eje y = 11,35 %
Velocidad máxima	Eje x = 15,43 cm/s Eje y = 16,58 cm/s
Aceleración máxima	Eje x = 0,63 m/s^2 Eje y = 0,52 m/s^2

Tabla 19. Resultados prueba diagonal 10 cm

3.2 RECTÁNGULO

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	Rectángulo de dimensiones 14x12 cm
Velocidad de ejecución:	Rápida

Tabla 20. Características prueba rectángulo 14x12 cm

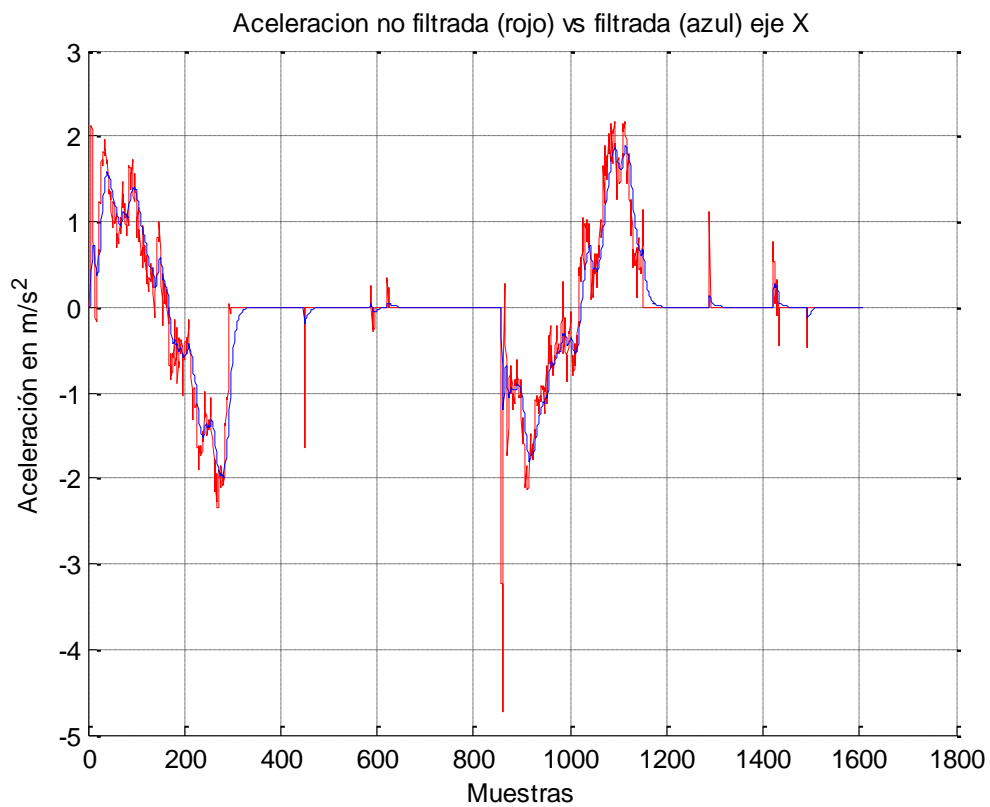


Figura 92. Aceleración eje X prueba rectángulo 14x12 cm

David Vara Rodríguez

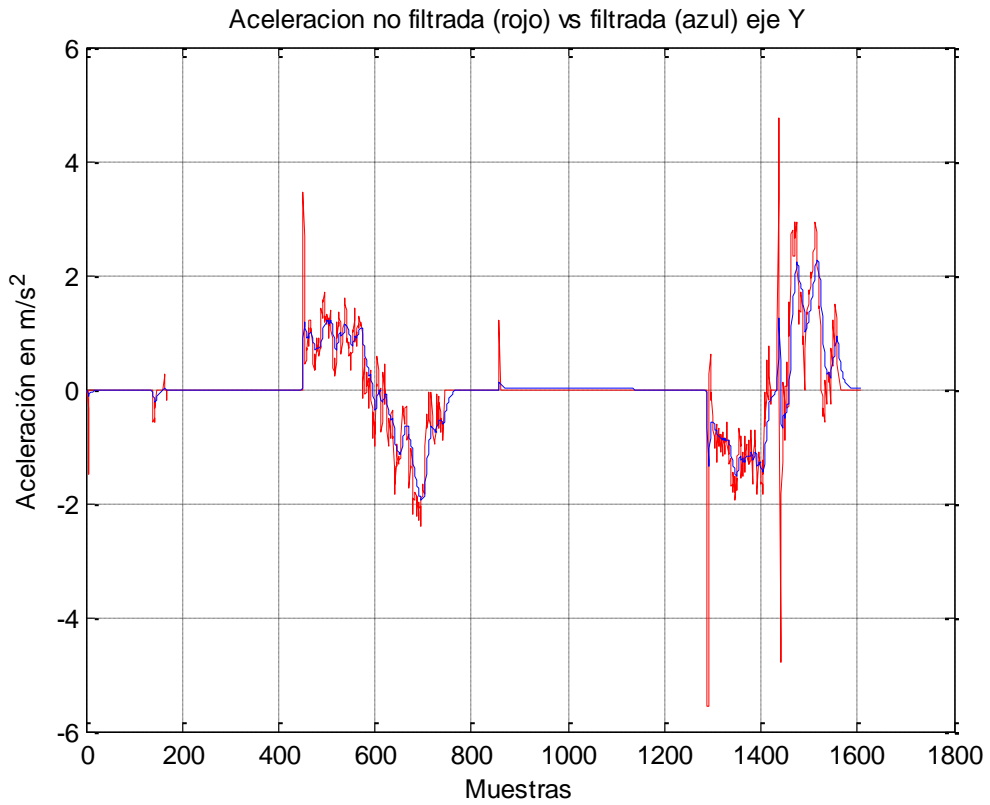


Figura 93. Aceleración eje Y prueba rectángulo 14x12 cm

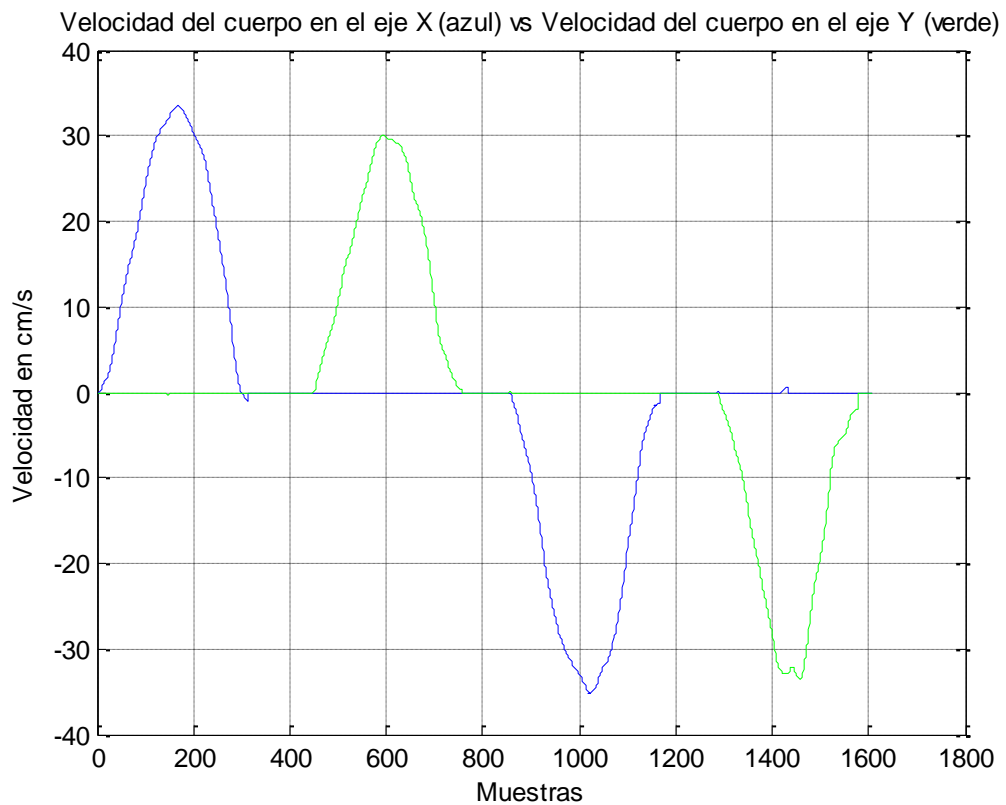


Figura 94. Velocidad prueba rectángulo 14x12 cm



David Vara Rodríguez

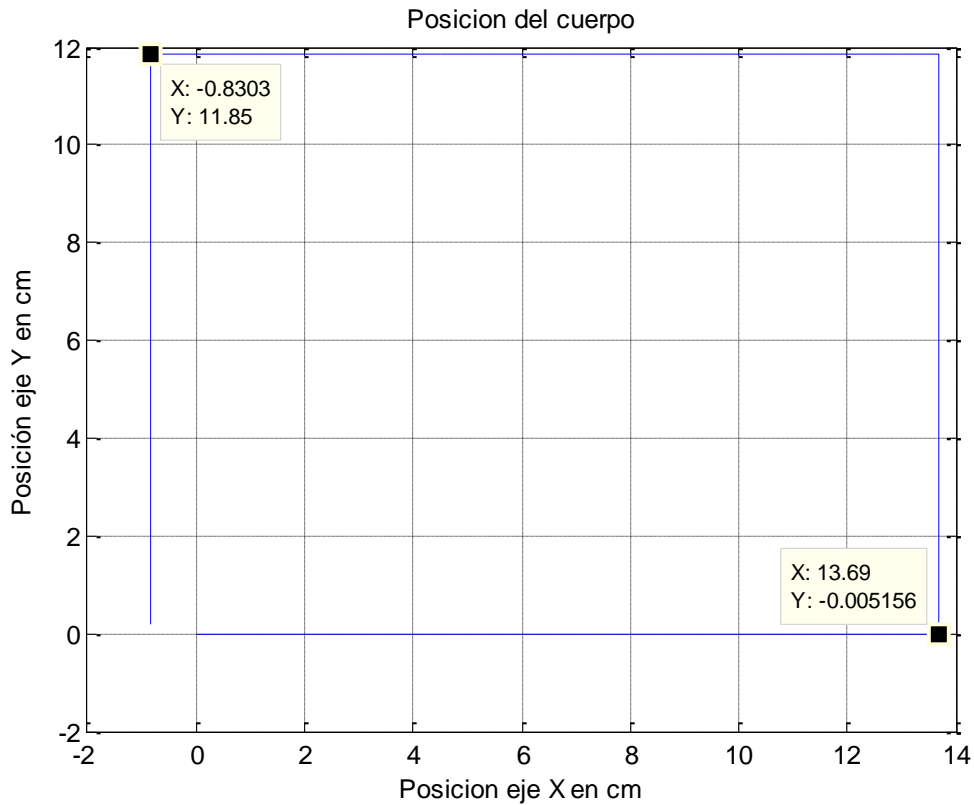


Figura 95. Posición prueba rectángulo 14x12 cm

Cuadro resumen	
Distancia teórica	Rectángulo de dimensiones 14x12 cm Eje x = 14 cm Eje y = 12 cm
Distancia obtenida	Eje x = 13,69 cm Eje y = 11,85 cm
Error absoluto	Eje x = 0,31 cm Eje y = 0,15 cm
Error relativo	Eje x = 2,21 % Eje y = 1,25 %
Velocidad máxima absoluta	Eje x = 35,23 cm/s Eje y = 33,56 cm/s
Aceleración máxima absoluta	Eje x = 1,99 m/s ² Eje y = 2,27 m/s ²

Tabla 21. Resultados prueba rectángulo 14x12 cm

3.3 RECTÁNGULO VELOCIDAD LENTA

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	Rectángulo de dimensiones 14x12 cm
Velocidad de ejecución:	Lenta

Tabla 22. Características prueba rectángulo a velocidad lenta

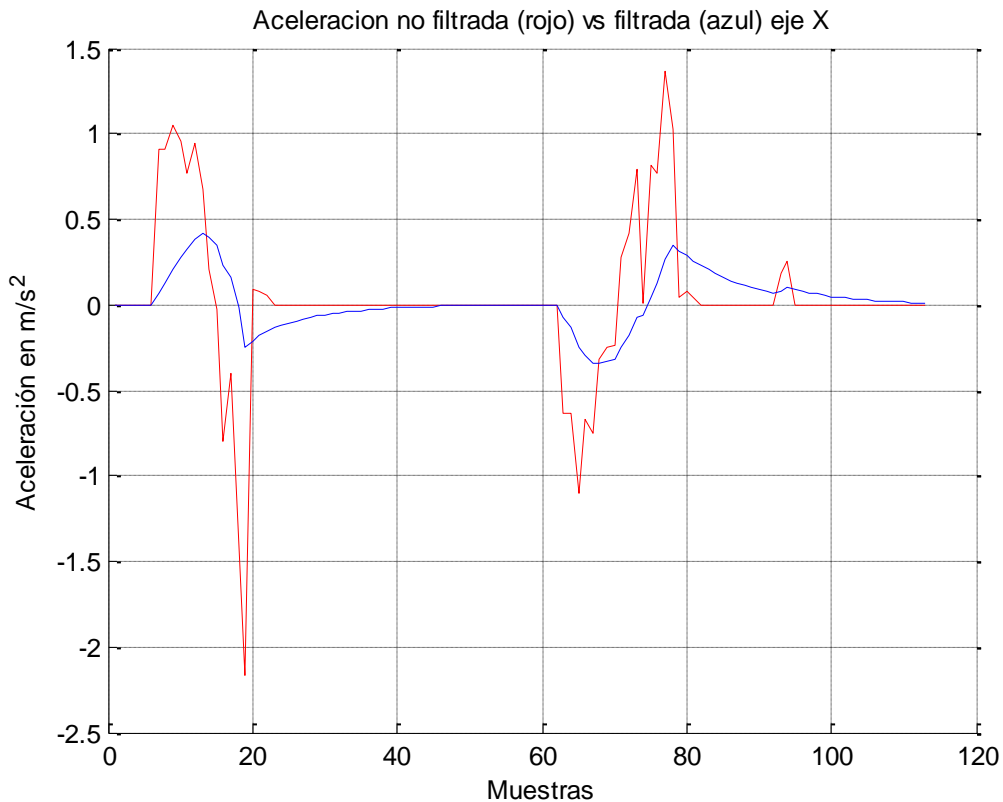


Figura 96. Aceleración eje X prueba rectángulo a velocidad lenta



David Vara Rodríguez

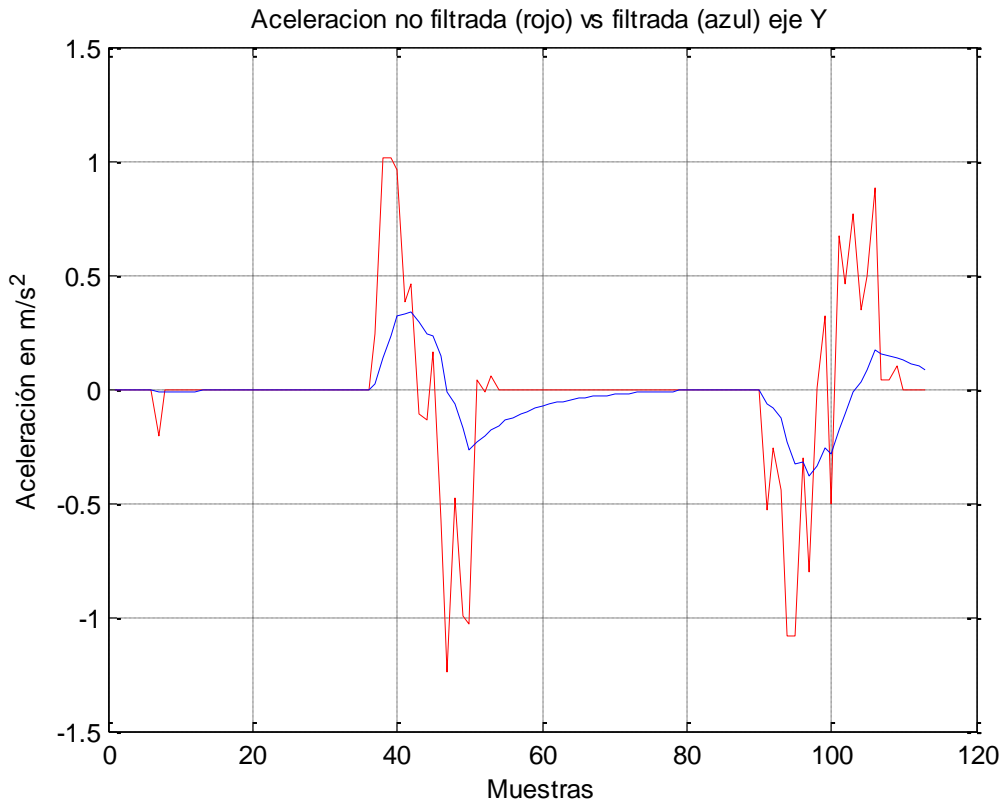


Figura 97. Aceleración eje Y prueba rectángulo a velocidad lenta

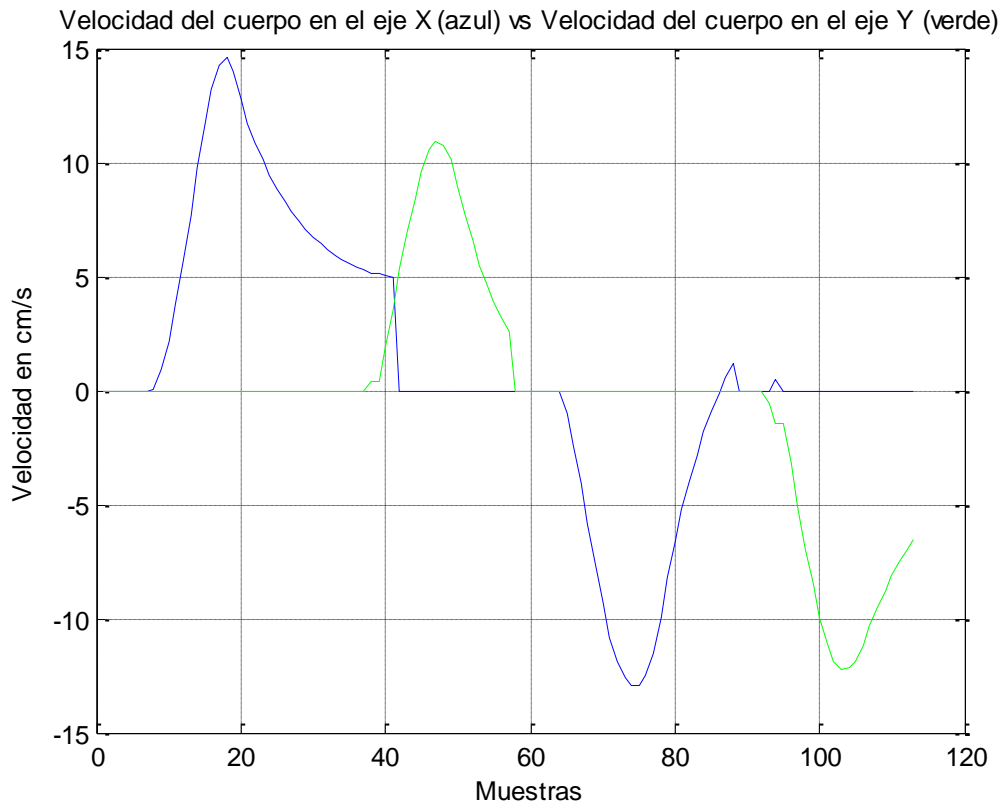


Figura 98. Velocidad prueba rectángulo a velocidad lenta



David Vara Rodríguez

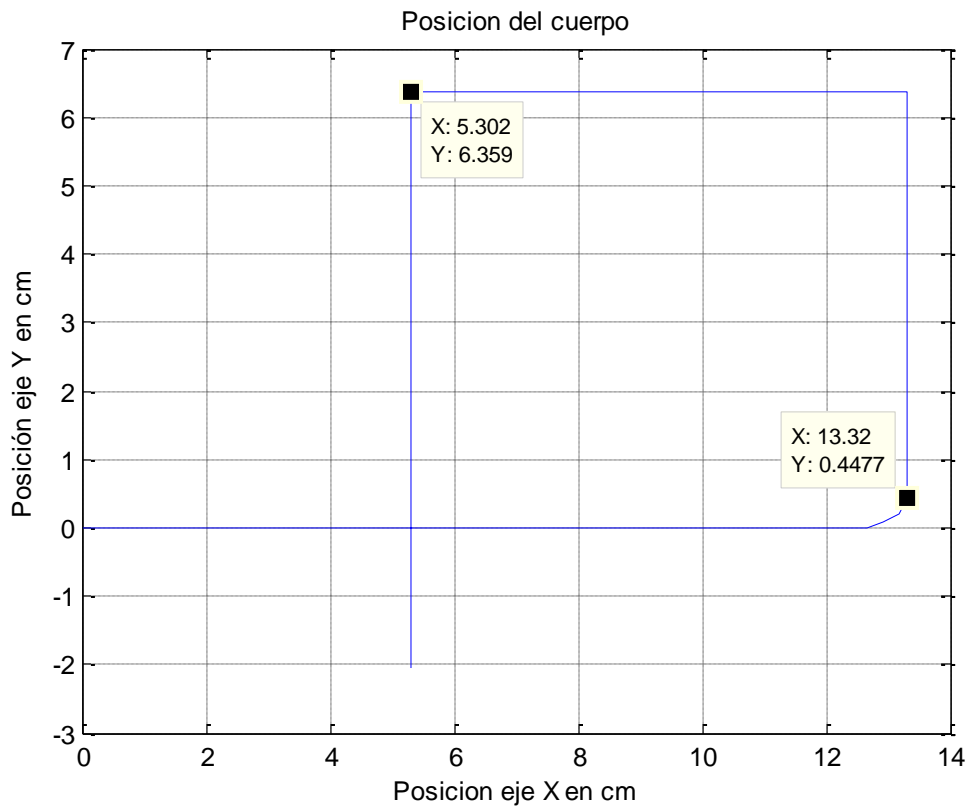


Figura 99. Posición prueba rectángulo a velocidad lenta

Cuadro resumen	
Distancia teórica	Rectángulo de dimensiones 14x12 cm Eje x = 14 cm Eje y = 12 cm
Distancia obtenida	Eje x = 13,32 cm Eje y = 6,35 cm
Error absoluto	Eje x = 0,31 cm Eje y = 0,15 cm
Error relativo	Eje x = 4,85 % Eje y = 47,08 %
Velocidad máxima absoluta	Eje x = 14,62 cm/s Eje y = 12,21 cm/s
Aceleración máxima absoluta	Eje x = 0,41 m/s^2 Eje y = 0,37 m/s^2

Tabla 23. Resultados prueba rectángulo a velocidad lenta

3.4 CÍRCULO

Características del ensayo	
Tipo de movimiento:	Manual
Distancia a medir:	Círculo de 10 cm de diámetro
Velocidad de ejecución:	Rápida

Tabla 24. Características prueba círculo

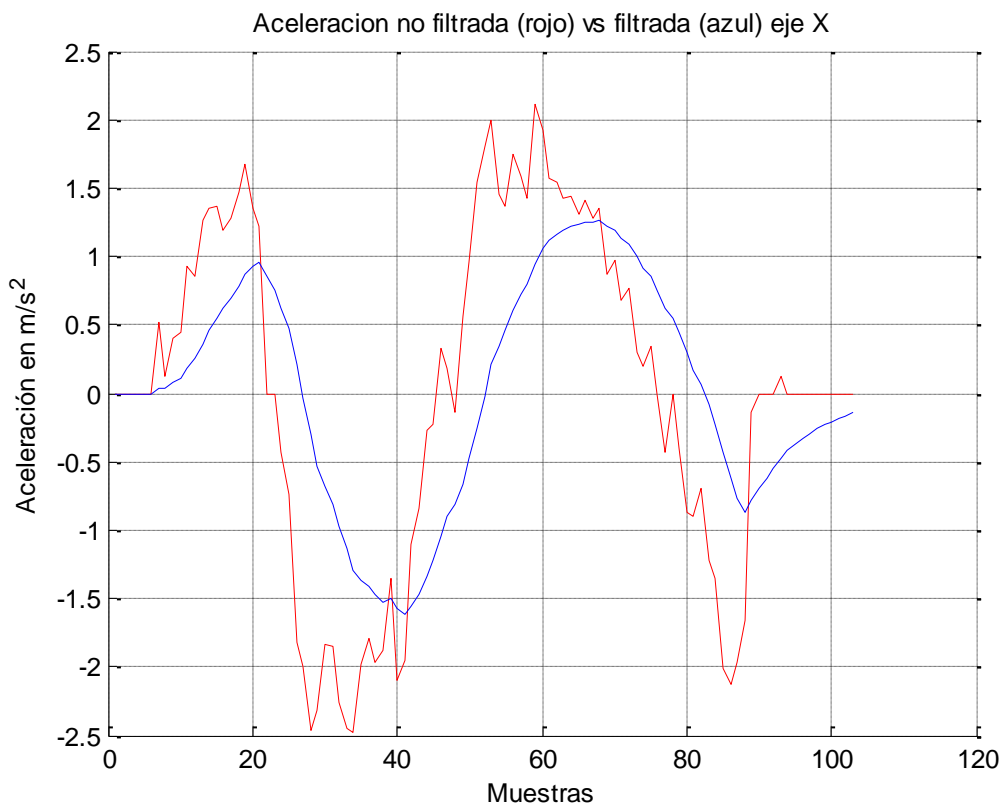


Figura 100. Aceleración eje X prueba círculo



David Vara Rodríguez

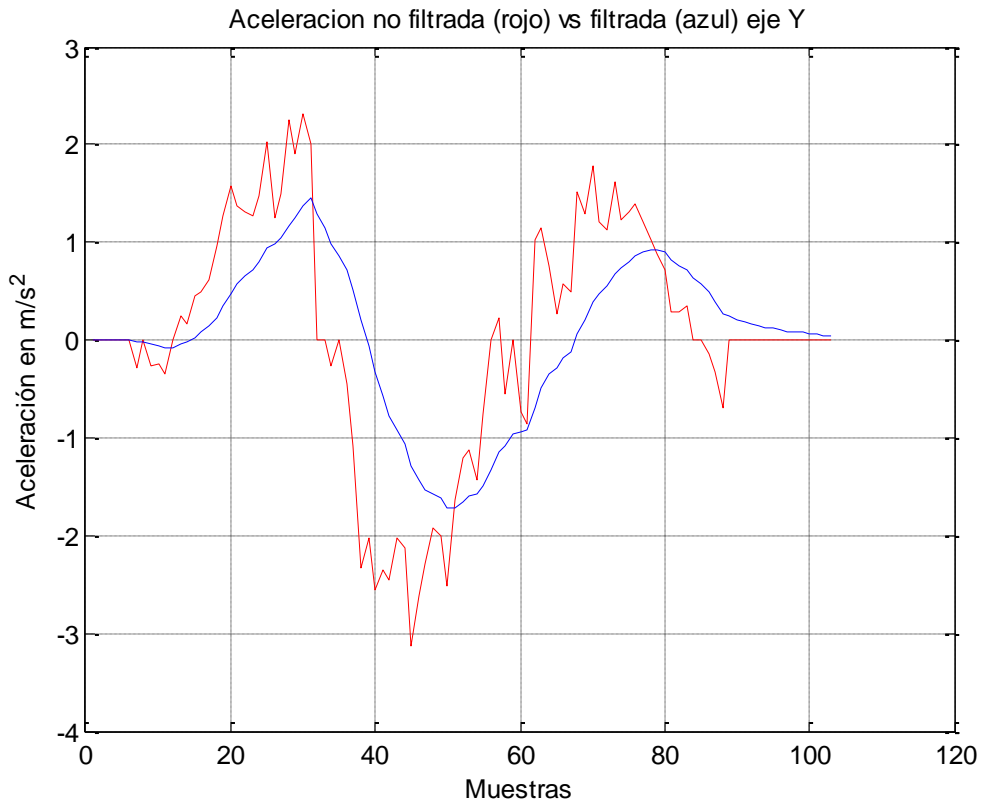


Figura 101. Aceleración eje Y prueba círculo

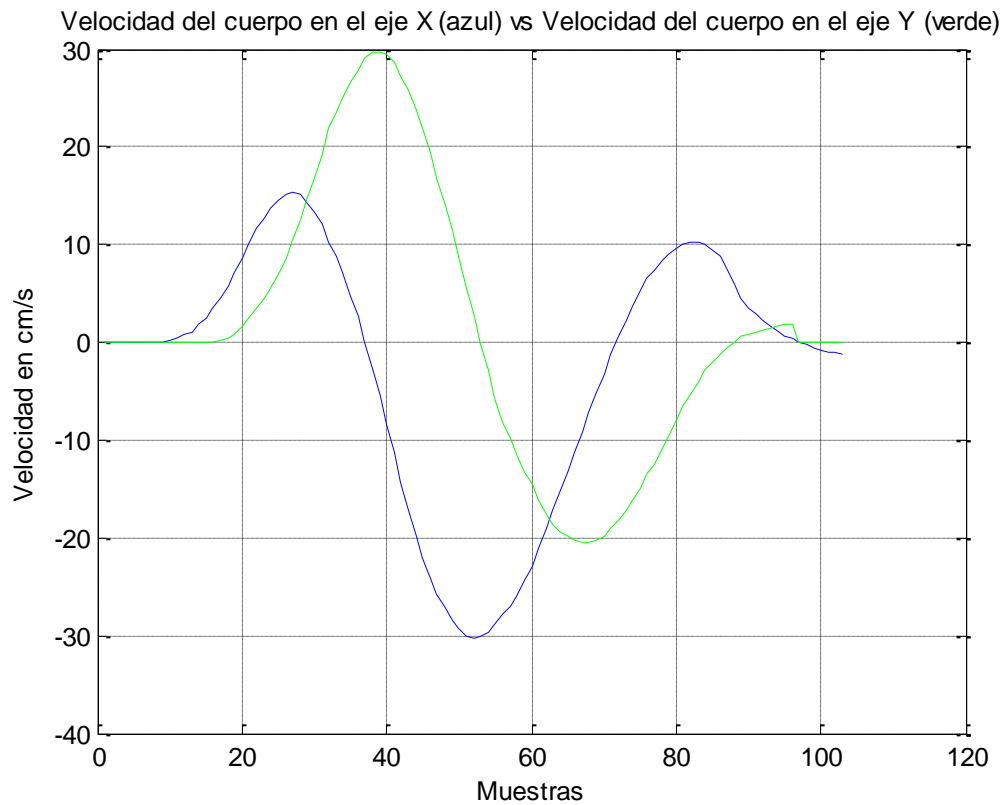


Figura 102. Velocidad prueba círculo



David Vara Rodríguez

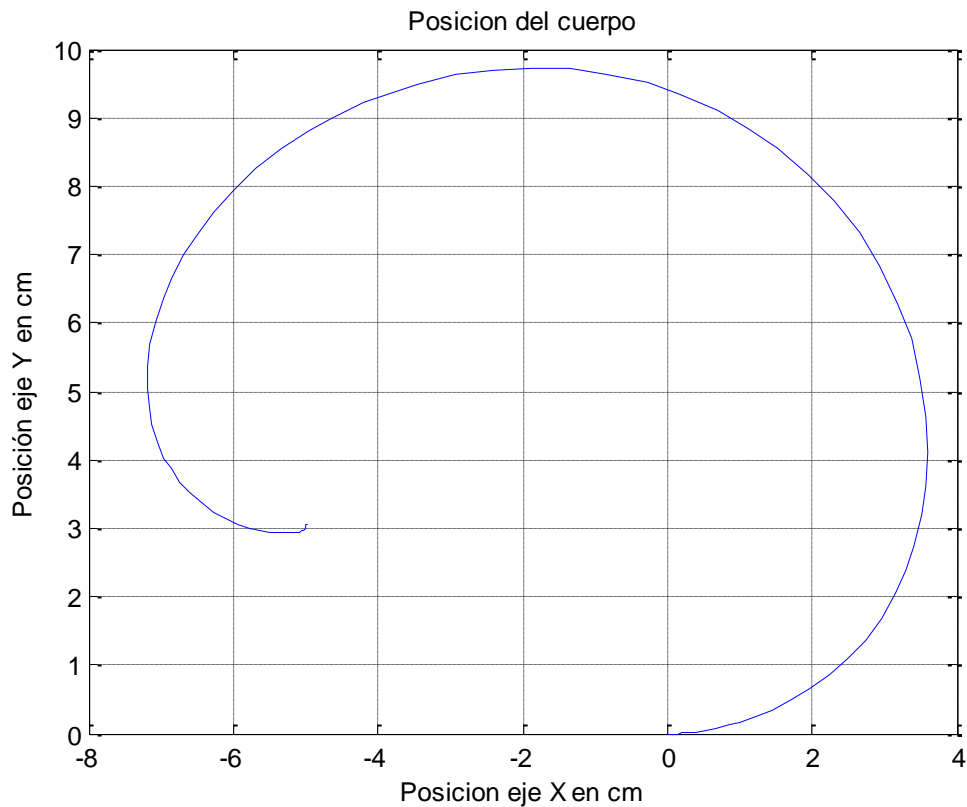


Figura 103. Posición prueba círculo

Cuadro resumen	
Distancia teórica	Círculo de 10 cm de diámetro Eje x = 5 cm Eje y = 5 cm
Distancia obtenida	Centro en (-1,79 , 4,86) cm Distancia x desde centro = 5,39 cm Distancia y desde centro = 4,86 cm
Error absoluto	Eje x = 0,39 cm Eje y = 0,14 cm
Error relativo	Eje x = 7,8 % Eje y = 2,8 %
Velocidad máxima absoluta	Eje x = 30,23 cm/s Eje y = 29,76 cm/s
Aceleración máxima absoluta	Eje x = 1,61 m/s^2 Eje y = 1,72 m/s^2

Tabla 25. Resultados prueba círculo

David Vara Rodríguez

3.5 **ROBOT ABB IRB 120**

Con motivo de evitar los errores debido al movimiento manual, se instaló nuestro dispositivo en el Robot ABB IRB 120.



Figura 104. Robot ABB IRB 120 con el dispositivo instalado

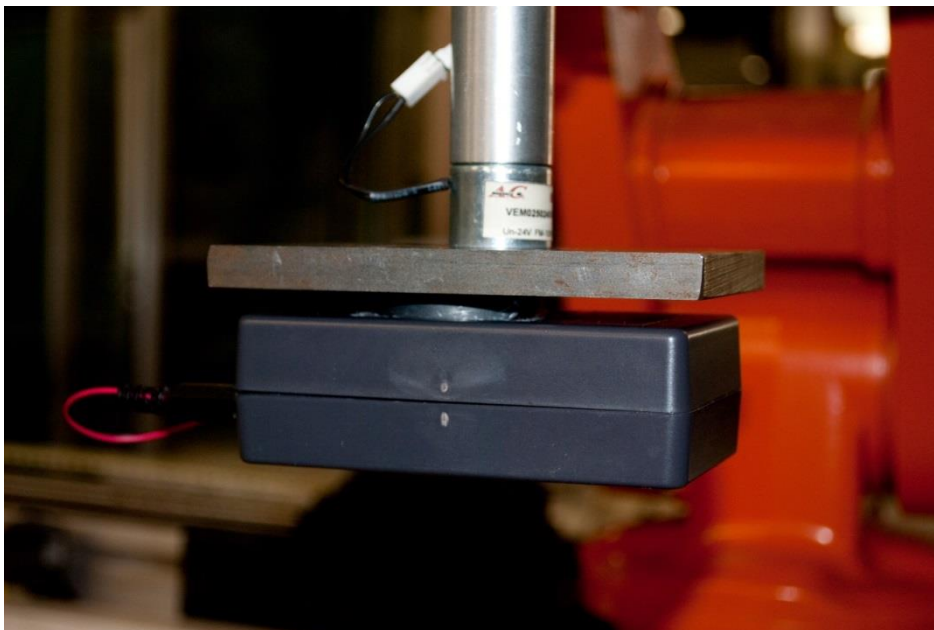


Figura 105. Dispositivo construido en el TCP del robot



David Vara Rodríguez

Para efectuar la prueba se realizó un programa en lenguaje RAPID, el cual puede encontrarse en el CD adjunto de este proyecto, por el cual el robot ejecutaría los movimientos de:

- Recta
- Rectángulo
- Círculo

En todos ellos el usuario puede elegir:

- La longitud o dimensiones de la figura a realizar
- Velocidad de ejecución
- Número de veces que se realiza el movimiento

Extracto del código en RAPID:

```
TPReadFK v1,"Elija una opción","recta","Rectang","Circulo","","Fin";
TEST v1
CASE 1:
    TPReadNum L1_T,"Longitud de la recta ¿";
    TPReadNum N_R,"Nº veces ejecuta la recta? ";
    recta L1_T, N_R;
    MoveABSJ reposo,v800,fine,145ectángulo145\Wobj:=mnegra;

CASE 2:
    TPReadNum L1_R,"Longitud del lado 1 Rectang?";
    TPReadNum L2_R,"Longitud del lado 2 rectang?";
    TPReadNum N_R,"Nº veces ejecuta Rectang?";
    145ectángulo L1_R,L2_R,N_R;
    MoveABSJ reposo,v800,fine,electroiman\Wobj:=mnegra;

CASE 3:
    TPReadNum R,"Radio Circulo?";
    TPReadNum N_C,"Nº veces ejecuta Circulo?";
    circulo1 R,N_C;
    MoveABSJ reposo,v800,fine,electroiman\Wobj:=mnegra;

CASE 5:
    VelSet 100, 5000;
    Stop;

ENDTEST
```

David Vara Rodríguez

El control de robot así como la elección de las distintas opciones se realiza a través de la consola FlexPendant:



Figura 106. FlexPendant

Esta prueba sin embargo, resultó infructuosa debido a la interferencia de los motores del robot con nuestro dispositivo.

Estos motores producían pequeñas vibraciones que eran captadas por nuestros sensores, produciendo resultados erróneos.



3.6 Comentarios de las pruebas en 2D

- No existe independencia entre ejes.
El movimiento en un eje produce pequeños valores en el resto, lo cual se traduce en un incremento del error.

Esto puede observarse en el movimiento de diagonal de 10cm de longitud. Los errores obtenidos son del 6,32% para el eje X y 11,35% para el eje Y, mientras que 10 cm medidos en una sola dimensión reportaron un error del 2%.

Para solucionarlo se ha creado un programa en Arduino, el cual detecta si se está realizando o no un movimiento y en qué eje. En movimientos lineales, se puede comprobar una notable mejora en los resultados obtenidos mediante la utilización de este programa. Para el movimiento de rectángulo, los errores en el eje X y en el eje Y fueron del 2,21% y 1,25% respectivamente, similares a los obtenidos en una sola dimensión.

- Nuevamente comprobamos la dependencia del error con la velocidad de movimiento. Para una un rectángulo de 14x12cm ejecutada de forma rápida obtenemos un error en el eje X del 2,21% y en el eje Y del 1,25% mientras que el mismo movimiento realizado de forma lenta reporta un error del 4.85% en el eje X y 47.08% en el eje Y.
- Al realizarse movimientos circulares se obtienen elevados errores. Esto se debe, entre otras causas y como ya se ha comentado anteriormente, a la interferencia entre ejes al ejecutarse movimientos en más de una dimensión.

4. Pruebas realizadas en tres dimensiones (3D)

4.1 PARALEPÍPEDO

<u>Características del ensayo</u>	
Tipo de movimiento:	Manual
Distancia a medir:	Paralelepípedo de dimensiones 14x9x9 cm
Velocidad de ejecución:	Rápida

Tabla 26. Características prueba paralelepípedo

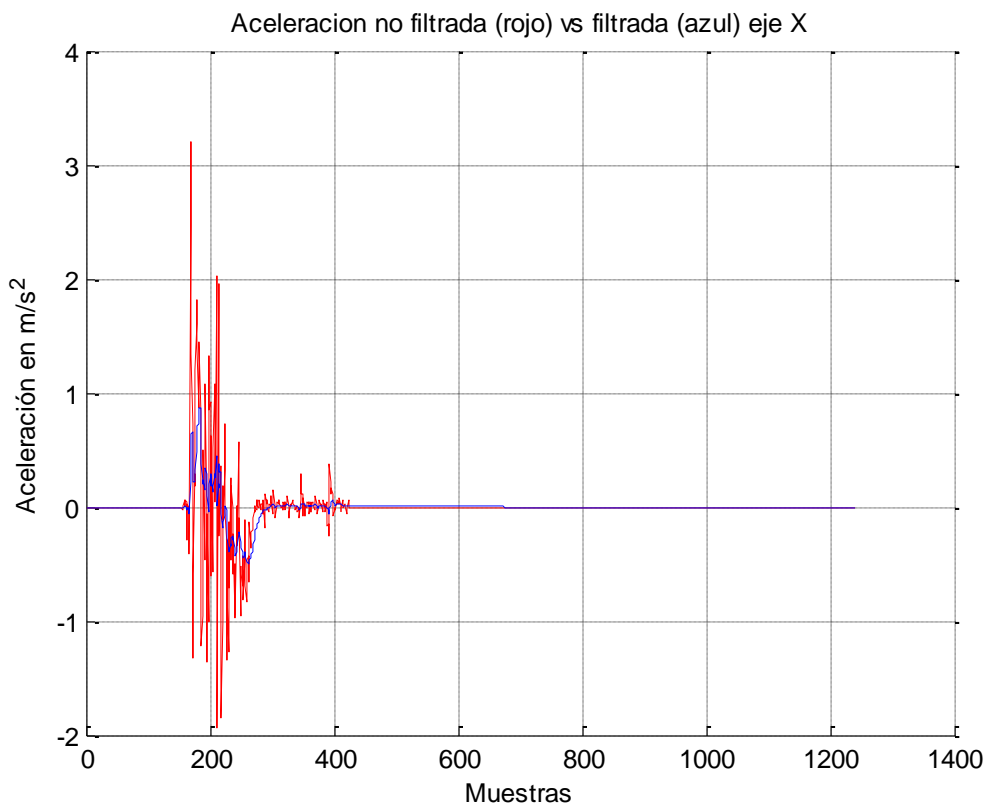


Figura 107. Aceleración eje X prueba paralelepípedo

David Vara Rodríguez

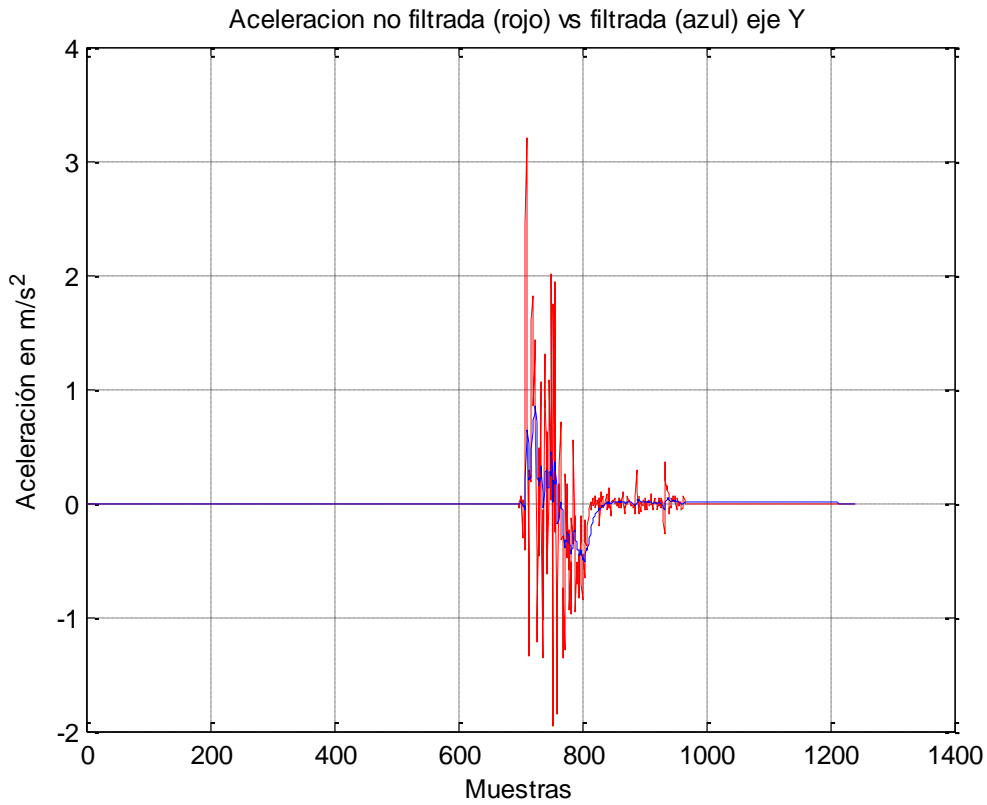


Figura 108. Aceleración eje Y prueba paralepípedo

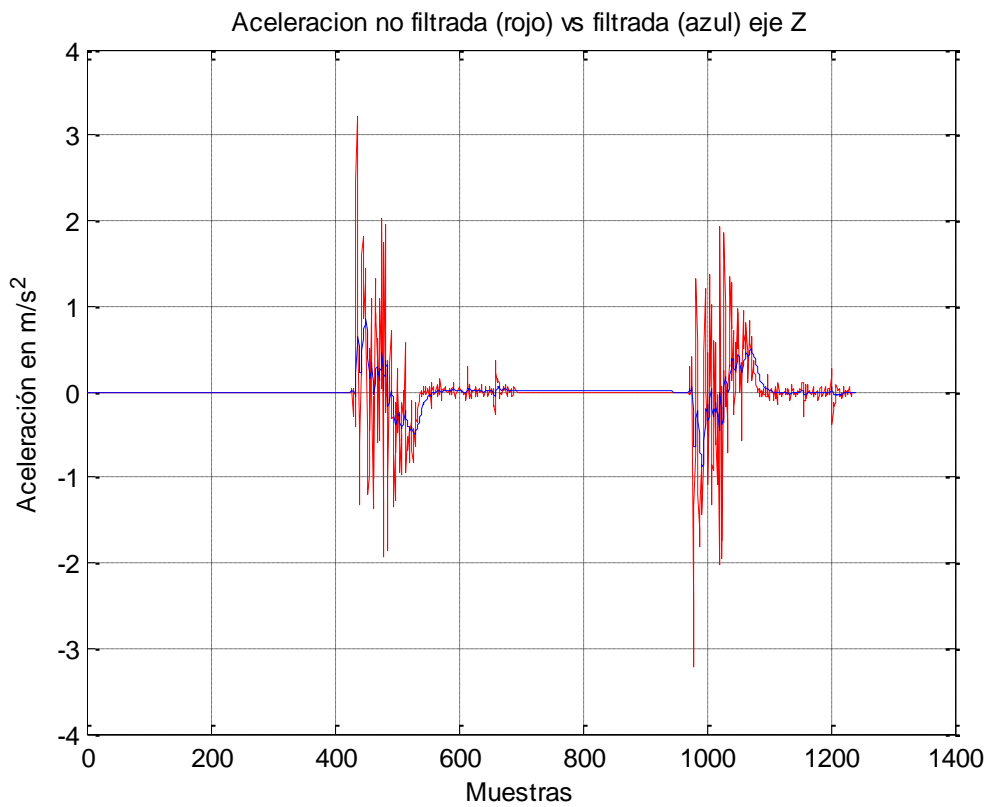


Figura 109. Aceleración eje Z prueba paralepípedo



David Vara Rodríguez

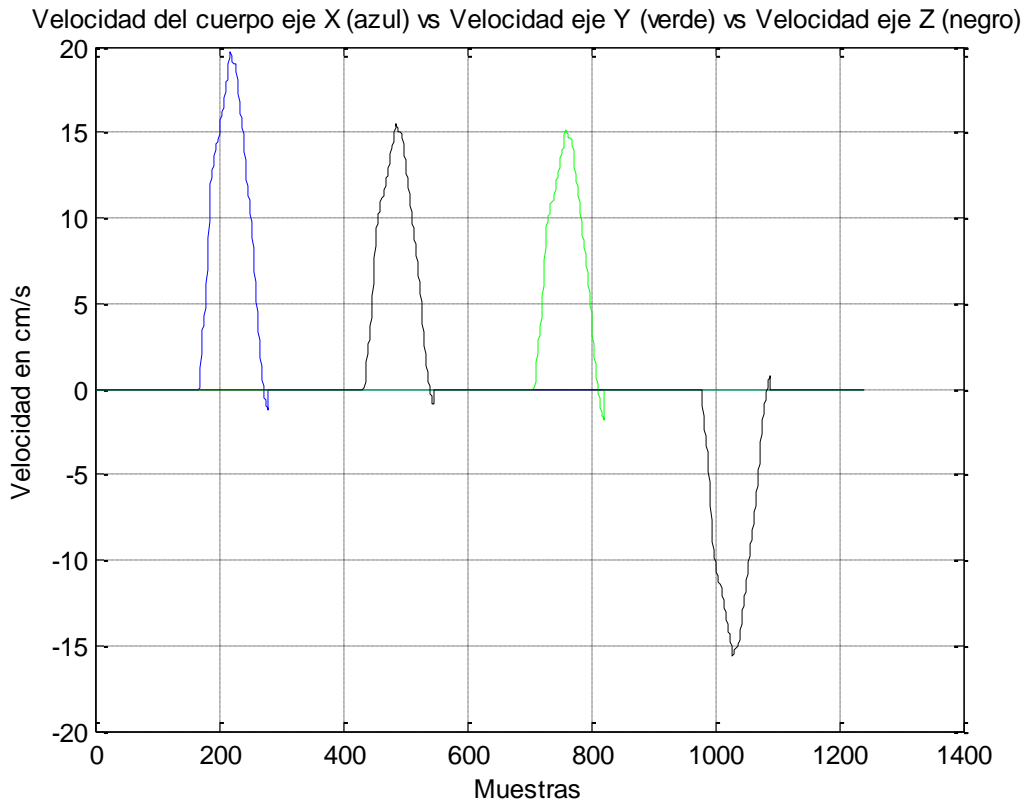


Figura 110. Velocidad prueba paralelepípedo

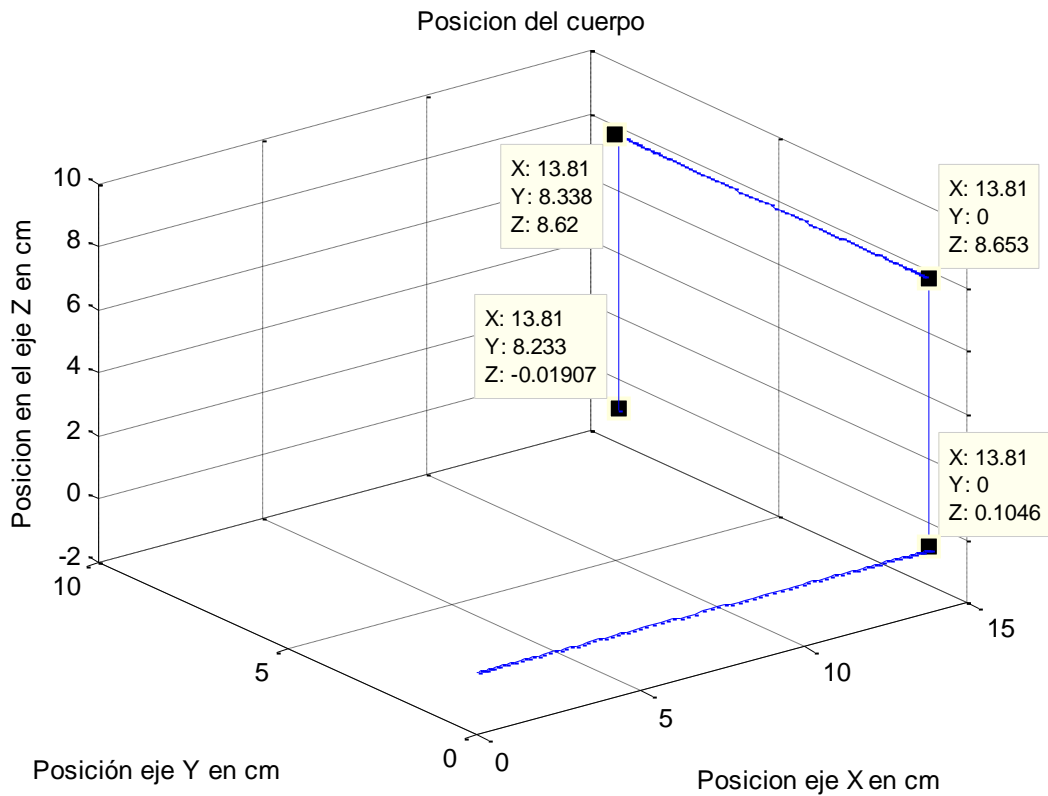


Figura 111. Posición prueba paralelepípedo



David Vara Rodríguez

Cuadro resumen	
Distancia teórica	Paralelepípedo de dimensiones 14x9x9 cm Eje x = 14 cm Eje y = 9 cm Eje z = 9 cm
Distancia obtenida	Eje x = 13,81 cm Eje y = 8,338 cm Eje z = 8,62 cm
Error absoluto	Eje x = 0,19 cm Eje y = 0,662 cm Eje z = 0,38 cm
Error relativo	Eje x = 1,35 % Eje y = 7,35 % Eje z = 4,22 %
Velocidad máxima absoluta	Eje x = 19,65 cm/s Eje y = 15,18 cm/s Eje z = 15,59 cm/s
Aceleración máxima absoluta	Eje x = 0,86 m/s^2 Eje y = 0,85 m/s^2 Eje z = 0,86 m/s^2

Tabla 27. Resultados prueba paralelepípedo

4.2 Comentarios de las pruebas en 3D

- Los errores son acumulativos.
Como nuestro dispositivo mide incrementos en la posición y no la posición absoluta, el error sobre un eje en una primera medida, será tenido en cuenta en movimientos posteriores.

Esto puede observarse en el eje Z. El primer movimiento proporciona una medida de 8,62cm. El segundo movimiento toma esta posición como referencia y no los 9cm reales, con lo que el error cada vez es más grande.



David Vara Rodríguez

5. Conclusiones de las pruebas realizadas

- El valor de la aceleración es extremadamente inestable y en ocasiones deja de corresponderse con la realidad. Esto se debe a que los acelerómetros son muy ruidosos, agravándose en presencia de ruido eléctrico o mecánico, como el de los motores del robot.
- El sensor de bajo coste utilizado, aún con mejores características que otros dispositivos similares existentes en el mercado, presenta múltiples errores entre los que podemos destacar:
 - o La variación de la sensibilidad en un $\pm 3\%$. Además esta varía con la temperatura un 2% adicional.
 - o El nivel de cero-g tampoco es contante y puede variar $\pm 0.08g$ para el eje Z y $\pm 0.05g$ para los ejes X e Y.
 - o Errores de offsets y no linealidad.
- La utilización de una ventana de filtrado con la que para eliminar el ruido en el nivel de cero-g introduce un nuevo error, ya que las aceleraciones entre $-0.12 m/s^2$ y $+0.12 m/s^2$ son consideradas como nulas.

Así mismo, la calibración del dispositivo con la que calcular los valores de offset del sensor, puede introducir otros errores.

- El filtro de Kalman, aún proporcionando mejores resultados que los filtros de señal paso bajo, no es un estimador óptimo para el problema propuesto y la curva de la aceleración proporcionada está aún lejos de la ideal.
- No es posible medir movimientos realizados a velocidades constantes, puesto que la señal proporcionada por el sensor mide variaciones de la misma (aceleraciones). Una consecuencia de esto, es la dependencia del error con la velocidad de movimiento, obteniéndose únicamente resultados aceptables (1 – 4% de error) para aceleraciones cercanas o superiores a $1 m/s^2$.
- Al no existir una independencia entre ejes debido a la gran sensibilidad del sensor y ruido existente, sino que un movimiento uniaxial provoca valores en el resto, se hace necesaria la implementación de un programa con el que distinguir el movimiento realizado.



David Vara Rodríguez

Esta solución elimina la problemática inicial, pero provoca, a su vez, que los movimientos a realizar tengan que ser uniaxiales, con el perjuicio en el tiempo de ejecución y el incremento en la dificultad de maniobra en situaciones reales.

- Otro error significativo que afecta a este tipo de dispositivo es el error en la inclinación del mismo. Una inclinación de 1° en la posición de $0g$, produce un error de 8.6m transcurridos 6 segundos de lecturas del sensor y de 308m tras 10 minutos.
- La localización del objeto mediante la integración de la señal proporcionada por el acelerómetro, introduce necesariamente errores en la medición, al realizarse aproximaciones numéricas. Este error, se verá incrementado cuantas más aproximaciones haya que realizar y menor sea la frecuencia de muestreo.
- Al medir incrementos en la posición y no la posición absoluta, los errores son acumulativos. Esto hace que el error en una medida inicial, se propague en otras posteriores.



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez

-



Capítulo 6. Conclusiones y líneas futuras

1. Introducción al capítulo

En este capítulo se realiza una reflexión sobre los resultados obtenidos, comprobando la exactitud y precisión de los mismos.

También se muestran las principales conclusiones extraídas así como las posibles futuras líneas de actuación que se pueden seguir en trabajos o proyectos posteriores.

2. Resultados obtenidos

A partir de las pruebas realizadas obtenemos las siguientes evidencias:

Distancia medida y errores

- Para movimientos lineales y distancias entre 10 y 40 cm se obtienen errores comprendidos entre el 1 y el 5%.
- Para distancias entre 5 y 10 cm, el error alcanza una cota de hasta el 30%.
- Para distancias inferiores o milímetros, la medición resulta imposible debido al ruido de la señal proporcionada por el sensor y a los errores acumulados al efectuar las aproximaciones para el cálculo de la posición.

Velocidad de movimiento y errores

- Existe una dependencia entre la velocidad de movimiento y el error obtenido. Los mismos movimientos en las mismas condiciones, realizados a velocidades (o con aceleraciones) más grandes, reflejan mejores resultados.

Precisión de las mediciones

- En ocasiones, mismos experimentos realizados en las mismas condiciones proporcionaron resultados diferentes. Entre otras posibles causas, esto se debe al empleo de microcontroladores y sensores de bajo coste.



3. Conclusiones

El objetivo de este proyecto era estudiar la viabilidad de sensores de bajo coste en la localización de herramientas en operaciones de cirugía.

Los errores analizados en el capítulo 4 - apartado 5 de este proyecto (pág.74) así como las conclusiones de las pruebas realizadas (pág.147), muestran la imposibilidad del uso de este tipo de dispositivos para la localización de instrumentos quirúrgicos.

No obstante, estudios bibliográficos como [24] han demostrado que pueden ser útiles en aplicaciones para la localización de personas o vehículos como sensores complementarios al GPS.

4. Futuras líneas de actuación

El filtro de Kalman ha demostrado proporcionar buenos resultados en sistemas lineales con una distribución de error gaussiana. No obstante, si el error posee otra distribución, los resultados empobrecen. Una posible línea de actuación es realizar ensayos para obtener datos de las aceleraciones de coches o viandantes con las que extraer las curvas de la aceleración y la distribución del error. Esto llevaría a una investigación sobre algoritmos de integración.

Otra línea de actuación pasa por el estudio de la eliminación de la gravedad de los datos proporcionados por el acelerómetro. Es decir, para cada instante de muestreo, calcular la matriz de rotación con la que hallar la aceleración física del objeto en movimiento.

Como ya se mencionó en el apartado anterior, un posible uso de estos dispositivos es su empleo como sensores complementarios en sistemas de localización de personas o vehículos. Estos instrumentos además, podrían ser recalibrados en cualquier instante y no solo al comienzo, mediante la señal proporcionada por el GPS, con lo se evitaría la propagación del error y una mejora en el modelo matemático propuesto inicialmente.

La formalización del problema como un sistema lineal, abre la puerta a la aplicación de teorías sobre el control de señales. En este campo, existen métodos para filtrar el ruido de las señales y reconstruirlas a partir de datos muestreados, con lo que puede ser una línea de estudio a explotar.

Como investigación más lejana, la búsqueda de un estimador para remplazar al filtro de Kalman podría proporcionar resultados más precisos.



Bibliografía

[1] Preeti Jain, “Position sensor: Types of position sensor” [Web en línea] 2012. Disponible en: <http://www.engineersgarage.com/articles/position-sensors>.

[2] O. Tonet, T. U. Ramesh, G. Megali, and P. Dario, “Tracking endoscopic instruments without localizer: Image analysis-based approach,” *Stud. Health Technol. Informat.*, vol. 119, pp. 544–549, 2006.

[3] A. Krupa, J. Gangloff, C. Doignon, M. de Mathelin, G. Morel, J. Leroy, L. Soler, and J. Marescaux, “Autonomous 3-D positioning of surgical instruments in robotized laparoscopic surgery using visual servoing,” *IEEE Trans. Robot. Autom.*, vol. 19, no. 5, pp. 842–853, Oct. 2003.

[4] Z. Pezzementi, S. Voros, and G. D. Hager, “Articulated object tracking by rendering consistent appearance parts,” in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 3940–3947.

[5] A. Reiter, P. K. Allen, and T. Zhao, “Marker-less articulated surgical tool detection,” in *Proc. Comput. Assist. Radiol. Surg.*, 2012.

[6] Max Allan, Sebastien Ourselin, Steve Thompson, David J. Hawkes, John Kelly, and Danail Stoyanov, “Toward Detection and Localization of Instruments in Minimally Invasive Surgery” in *IEEE-INST.* Apr. 2013

[7] Magdalena K. Chmarra, Niels H. Bakker, Cornelis A. Grimbergen, Jenny Dankelman, “TrEndo, a device for tracking minimally invasive surgical instruments in training setups”. Man–Machine Systems Group, Faculty of Mechanical, Maritime, and Materials Engineering, Delft University of Technology, Delft, The Netherlands. Oct. 2005



David Vara Rodríguez

[8] A.M. Cano, F. Gayá, P. Lamata, F. del Pozo, F. Sánchez-Margallo, E.J. Gómez, “Método de seguimiento 3D del instrumental quirúrgico mediante análisis de vídeo laparoscópico”. Grupo de bioingeniería y telemedicina, Universidad Politécnica de Madrid. Oct.2008.

[9] Lisa Nilsson, “Accuracy of Laparoscopic 3D Ultrasound”. Royal Institute of Technology. School of Computer Science and Communication. Sweden. 2008

[10] “Automatic Localization of the da Vinci Surgical Instrument Tips in 3D Transrectal Ultrasound”. Omid Mohareri, Student member, IEEE, Mahdi Ramezani, Student member, IEEE, Troy K. Adebar, Student member, IEEE, Purang Abolmaesumi, Senior member, IEEE, and Septimiu E. Salcudean, Fellow, IEEE. 2013

[11] “Laparoscopic Instrument Localization using 3-D Time of Flight/ RGB Endoscope”. Sven Haase, Jakob Wasza, Thomas Kilgus, Joachim Hornegger. School in Advanced Optical Technologies (SAOT).

[12] José Blas Pagador Carrasco, “Técnicas de robótica y visión artificial para el entrenamiento quirúrgico de mínima invasión”. Universidad de Extremadura , Departamento UNEX: Ingeniería de Sistemas Informáticos y Telemáticos, Unidad CCMIJU: Bioingeniería y Tecnologías Sanitarias. 2012.

[13] Zambrano L., Amilcar, Piñango G., Alfredo J. “Diseño y construcción de un simulador para cirugía laparoscópica”. Universidad de Caracas. 2003

[14] M. K. Chmarra, C. A. Grimbergen, and J. Dankelman, “Systems for tracking minimally invasive surgical instruments,” *Minim Invasive Ther Allied Technol*, vol. 16, no. 6, pp. 328-40, 2007.



David Vara Rodríguez

[15] “Comunicando Arduino con otros sistemas” [Web en línea] 2014. Disponible en: <http://playground.arduino.cc/ArduinoNotebookTraduccion/Appendix4>

[16] Germán A. Prieto, “Integración numérica”. Universidad de los Andes.

[17] “Numerical Differentiation and Integration Tutorial”. Universidad de Massachusetts Lowell.

[18] Kurt Seifert, Oscar Camacho, “Implementing positioning algorithms using accelerometers”, 2007.

[19] Maybeck, P., “Introduction from Stochastic Models, estimation, and control, volumen 1”. Department of electrical engineering air force institute of technology Wright-Patterson air force base, OHIO, 1979.

[20] WELCH, BISHOP, “An Introduction to the Kalman Filter”. EEUU: Department of Computer Science University of North Carolina, July 2006.

[21] Solera, A., “El Filtro de Kalman”, Banco Central de Costa Rica, División Económica, Departamento de Investigaciones económicas, julio 2003.

[22] Simon, D., “Kalman Filtering”, junio 2001.

[21] Tompyckebe, “Kalman filtering of IMU data”, mayo 2006.

[23] “Moving average filter” [Web en línea]. Mathworks, 2014. Disponible en: http://www.mathworks.es/es/help/matlab/data_analysis/filtering-data.html



Sistemas para determinar la posición y orientación de herramientas quirúrgicas en operaciones de cirugía laparoscópica



David Vara Rodríguez

[24] "Evaluation of a Low Cost Solid-State Accelerometer as a Distance Measuring Sensor for Vehicle Positioning System", Hugh Liu, Grantham Pang. Dept. of Electrical and Electronic Engineering. The University of Hong Kong.