

## **Modelado matemático, simulación y control de una mini motocicleta autónoma con rueda de reacción**

J.C. Saavedra, A. Olivares, G. Olivares

Departamento de Arquitectura y Tecnología de Computadores  
Universidad de Granada. Granada, Spain

Juancarlosst97@correo.ugr.es, a.olivares@gmail.com, gonzalo@ugr.es

**Resumen.** Se presenta el modelado, la simulación y el control de una mini motocicleta autónoma, estabilizada mediante rueda de reacción, que se utilizará para la enseñanza del diseño de sistemas de control. El modelo se construye a partir de un diseño CAD para posteriormente ser integrado en Simulink, junto con los módulos de control. Se realiza la modelización del sistema completo, incluyendo elementos mecánicos, sensores, actuadores, así como la dinámica de contacto de las ruedas con el suelo, consiguiéndose un comportamiento muy similar al de la motocicleta física. A partir del modelo matemático (ecuaciones diferenciales, funciones de transferencia y ecuación de estado), se diseñan un controlador PI para la velocidad, y varios controladores para la inclinación (PID, LQR y LQI). Los controladores han sido probados también en la motocicleta física.

**Palabras Clave:** Motocicleta autónoma, Control Digital, Control PID, control LQR, LQI, Simscape, Aprendizaje Experimental.

**Abstract.** In this paper, the modelling and subsequent control of an autonomous mini motorcycle, which will be used to teach about control systems, is presented. The model is initially built from a CAD design and then integrated into Simulink, together with the control modules. The modelling of the complete system, including the mechanical parts, sensors, actuators and wheels behaviour with the ground is carried out, to achieve the same behaviour as the physical motorcycle. From the mathematical model (differential equations, transfer functions and state equation), a PI controller is designed for speed, and several controllers for inclination (PID, LQR and LQI). The controllers have also been tested on the physical motorcycle.

**Keywords:** Autonomous Motorcycle, Digital-Control, PID controller, State space controller, LQR control, Simscape Multibody Simulation, Experimental Learning.

## 1 Introducción

El principal objetivo de este artículo es diseñar un modelo virtual de una motocicleta auto-balanceada que sea lo más fiel posible a una mini motocicleta física, suministrada en el *Arduino Engineering kit*, con el fin de evaluar si este sistema es idóneo para utilizarse en las clases prácticas de las materias de control impartidas en la Universidad de Granada. El uso de modelos virtuales en la enseñanza de sistemas de control tiene varias ventajas (bajo coste, eliminación de coste mantenimiento, posibilidad de trabajar con grupos numerosos de prácticas y de que el alumno pueda trabajar desde casa, etc), pero también el inconveniente de que a veces el resultado se puede alejar de la realidad.

Representaremos el modelo de la motocicleta en un entorno gráfico denominado “*Simscape Multibody*” incluido en las librerías de las últimas versiones de Simulink. Este método de simulación gráfica tiene la ventaja de permitir la verificación del comportamiento del controlador y ver el resultado del movimiento de sistemas mecatrónicos tal y como se produciría en los sistemas reales, incorporando también módulos virtuales de controladores.

Para la creación del modelo mecánico se realiza inicialmente el diseño gráfico con un software CAD (en nuestro caso Solidworks), y posteriormente se exporta a Simulink. A continuación es necesario realizar el diseño del controlador, y las modificaciones necesarias en el modelo para conseguir un sistema que sea fiel a la motocicleta física.

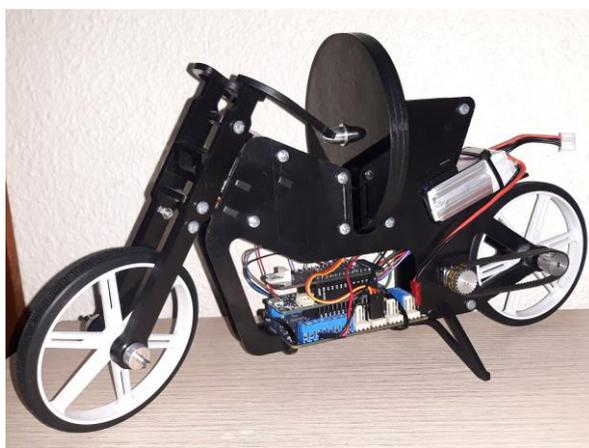
A lo largo de este documento, se muestra cómo tratar las ecuaciones que rigen el comportamiento de la motocicleta para realizar un controlador PI de velocidad y dos controladores diferentes para mantener la inclinación de la motocicleta deseada, siendo el primer controlador de inclinación un PID y el segundo de ellos un controlador LQI en el espacio de estados. Para comprobar los resultados, los controladores son probados también sobre la motocicleta física, lo que nos permitirá sugerir varias mejoras en su diseño.

El modelo de la motocicleta puede ser utilizado para aumentar el interés de los estudiantes, ya que es muy llamativo y motivador realizar simulaciones del controlador diseñado y ver su comportamiento en un modelo que representa de manera bastante fiel al modelo real. El trabajo realizado pretende servir de base para la futura construcción de una motocicleta autónoma de tamaño real.

Son muchos los autores que han realizado un estudio de este problema, a partir de un modelado matemático [1-8]. En nuestro caso, pretendemos usar un modelo lo más sencillo posible, para poder usarse en la enseñanza prácticas de asignaturas de control.

## 2 Motocicleta física.

La motocicleta empleada cuenta con todos los componentes necesarios que permiten su conducción de forma autónoma. Dicha motocicleta está compuesta por una estructura de ABS cortada con fresadora. Para conseguir el movimiento hacia adelante y hacia atrás se emplea un motor DC que lleva acoplado un encoder, lo que nos permitirá regular su velocidad. Para conseguir que la moto se equilibre se utiliza un motor DC de alta velocidad acoplado a una rueda de reacción, cuya velocidad angular es medida mediante un tacómetro.



**Figura 1.** Fotografía de la mini-motocicleta real

La tarjeta de control es del tipo *Arduino MKR1000*, que incorpora comunicación WiFi. Para el control de motores y servos se usa la tarjeta *MKR Motor Shield*.

Se emplea además una IMU (Unidad de Medición Inercial) *Bosch BNO055* para medir la inclinación de la motocicleta, que integra acelerómetro ejes, giróscopo y magnetómetro de 3 ejes.

Para mover la rueda de reacción, que va a estabilizar la motocicleta, se emplea un motor DC de alta velocidad. La velocidad de giro de este motor se mide con dos sensores de efecto Hall (*SL353HT de Honeywell*) y unos pequeños imanes que giran solidarios al eje del motor.

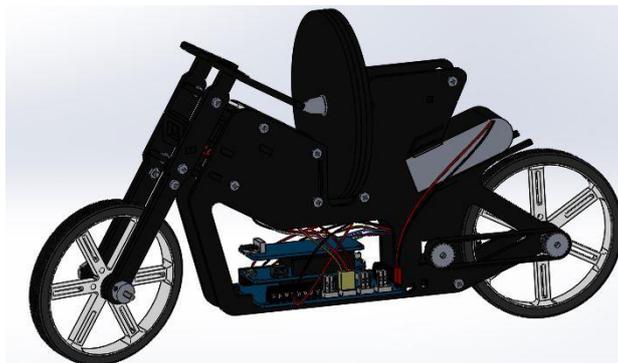
El motor empleado para realizar el movimiento de avance y retroceso de la motocicleta es también un motor DC que lleva acoplado una caja de engranajes para conseguir un mayor par motor y adecuar su velocidad a la requerida para el movimiento de la motocicleta. Este motor lleva acoplado un encoder en cuadratura para medir el ángulo girado y la velocidad angular del motor (derivando el ángulo). Conocida la velocidad angular del motor (y de la rueda) y el radio de la rueda, se obtiene la velocidad lineal de la motocicleta.

### 3 Modelado gráfico

El primer paso es realizar el proceso de modelado gráfico. Para crear un modelo que sea una representación realista del sistema se ha empleado el software de diseño *Solidworks*. Una vez que el modelado ha sido realizado, éste es exportado a un software de simulación que permite simular el comportamiento motriz del modelo creado anteriormente. Dicho software es “*Simscape Multibody*”, una extensión de *Simulink*.

#### 3.1 Diseño CAD

Inicialmente se diseñó un modelo CAD muy realista, pero a la vez demasiado complejo para ser modelado en Simulink, ya que incluía una gran cantidad de elementos mecánicos, con excesivos grados de libertad. Esto provoca que la potencia de cálculo necesaria sea excesiva y lenta de ejecutar en un computador convencional. Es fundamental crear cada parte con las mismas dimensiones, la misma masa, y la misma densidad para que así, las propiedades físicas del modelo sean las mismas que las de la motocicleta física.



**Figura 2.** Modelo inicial

A partir de este primer modelo se construyó otro modelo, que conserva las mismas características físicas que el inicial, pero cuya complejidad se ve reducida. Para conseguir esta reducción de piezas se emplea el comando “unir” disponible en *Solidworks*. Este comando permite, a partir de dos sólidos que estén en contacto, crear un único nuevo sólido, que posee la combinación de las características físicas que poseían los sólidos que lo forman. Gracias a esto, el modelo de la motocicleta queda reducido a siete elementos fundamentales. En la figura 3 se muestra el modelo gráfico simplificado.

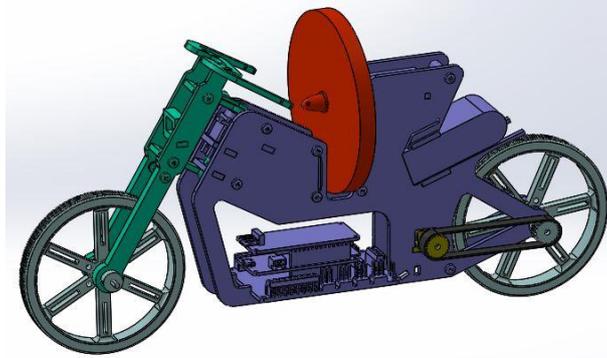


Figura 3. Modelo gráfico simplificado

### 3.2 Exportación del diseño CAD a Simulink

Este modelo puede ser importado a Simscape Multibody, para ello es necesario utilizar el elemento “*Simscape Multibody Link Inventor Plug-In*”. Una vez instalado este complemento, es necesario abrir el modelo en *Solidworks*, ir a *Simscape Multibody Link* y seleccionar la opción de exportar a *Simscape Multibody*. En la figura 4 se puede ver el diagrama de bloques del sistema.

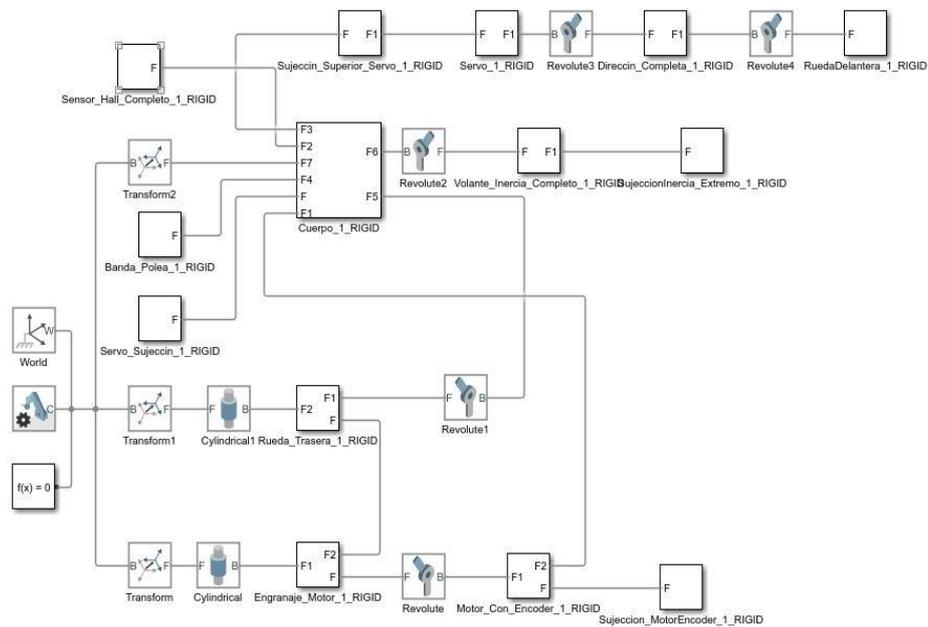
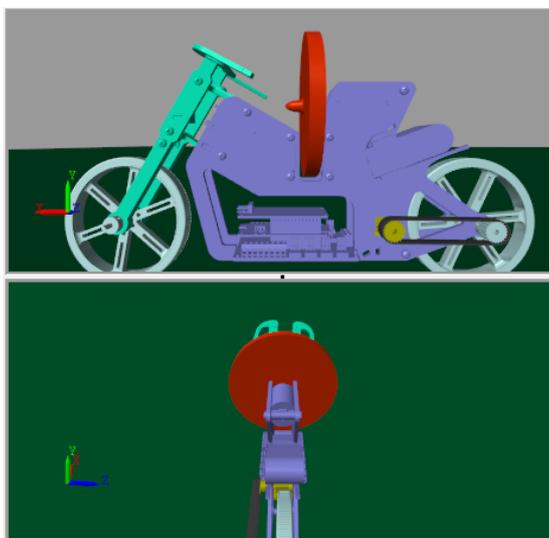


Figura 4. Diagrama de bloques *Simscape* del modelo simplificado de la motocicleta

Para modelar el contacto entre las ruedas y el suelo, se emplea una librería adicional, denominada “*Simscape Multibody Contact Forces Library*” que hay que añadirla manualmente. Esta librería permite simular el comportamiento que se produce entre cualquier tipo de superficie cuando una de ellas está fija y otra de ellas se mueve. En el caso de este modelo, se emplea el tipo *Sphere to Plane Force*. Mediante esta forma de contacto se simula la forma en la que una esfera se mueve a lo largo de un plano.



**Figura 1.** Modelo en Simscape Multibody

#### 4 Modelo matemático

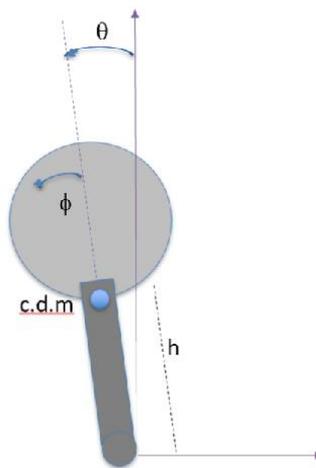
Vamos a considerar que la motocicleta sólo se estabiliza mediante la rueda de reacción. Inicialmente consideramos que la motocicleta se mueve en línea recta, y el ángulo del manillar no influye en la estabilidad, ni en la inclinación. Otro modelo que tienen en cuenta la inclinación producida por la variación de dicho ángulo se puede consultar en [2].

Tal y como se aprecia en la figura 6, la inclinación con respecto a la vertical se mide con el ángulo  $\theta$  y la rueda de reacción gira con un ángulo  $\phi$ . La distancia entre el suelo y el centro de masas, la denominamos  $h$ , y  $m$  es la masa de la motocicleta.

La Energía cinética de rotación viene dada por la expresión:

$$T = \frac{1}{2} I_M \dot{\theta}^2 + \frac{1}{2} J \dot{\phi}^2$$

Donde  $I_M = mh^2$  es el momento de inercia de la motocicleta con respecto a la vertical, y  $J$  es el momento de inercia de la rueda de inercia, gobernada por un motor DC de alta velocidad.



**Figura 6.** Esquema de la motocicleta

La energía potencial es:  $V = mgh \cos \theta$

El lagrangiano será:

$$L = T - V = \frac{1}{2} I_M \dot{\theta}^2 + \frac{1}{2} J \dot{\phi}^2 - mgh \cos \theta = \frac{1}{2} mh^2 \dot{\theta}^2 + \frac{1}{2} J \dot{\phi}^2 - mgh \cos \theta$$

Resolviendo las ecuaciones de Euler-Lagrange, se obtienen las ecuaciones diferenciales del sistema:

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = -\tau \quad \rightarrow \quad mh^2 \ddot{\theta} - mgh \sin \theta = -\tau$$

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = \tau \quad \rightarrow \quad J \ddot{\phi} = \tau$$

donde  $\tau$  es el par del motor que mueve la rueda de reacción.

Linealizando se obtiene la función de transferencia:

$$G(s) = \frac{\theta(s)}{\tau(s)} = -\frac{1/mh^2}{s^2 - \frac{g}{h}}$$

y la ecuación de estado: 
$$\begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{g}{h} & 0 \end{pmatrix} \begin{pmatrix} \theta \\ \dot{\theta} \end{pmatrix} + \begin{pmatrix} 0 \\ -\frac{1}{mh^2} \end{pmatrix} \tau$$

Si incorporamos el motor, de constante K (y consideramos que la constante de par y la constante electromotriz son iguales), tendremos finalmente:

$$\begin{aligned} \ddot{\theta} &= \frac{g}{h} \theta + \frac{K^2}{mh^2 R} \dot{\phi} - \frac{K}{mh^2 R} V \\ \ddot{\phi} &= -\frac{K^2}{RJ} \phi + \frac{K}{RJ} V \end{aligned}$$

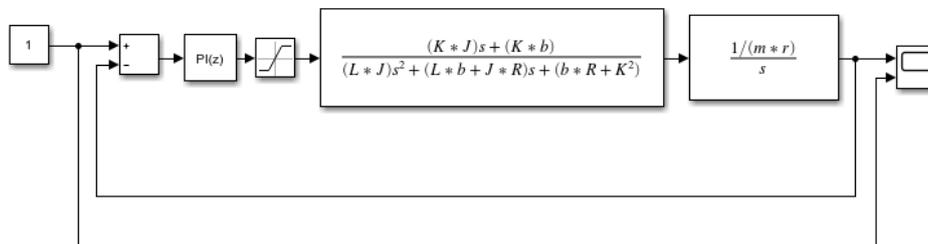
$$\begin{pmatrix} \dot{\theta} \\ \ddot{\theta} \\ \ddot{\phi} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ \frac{g}{h} & 0 & \frac{K^2}{mh^2 R} \\ 0 & 0 & -\frac{K^2}{RJ} \end{pmatrix} \begin{pmatrix} \theta \\ \dot{\theta} \\ \phi \end{pmatrix} + \begin{pmatrix} 0 \\ -\frac{K}{mh^2 R} \\ \frac{K}{RJ} \end{pmatrix} V$$

**Ecuación 1.** Ecuación de estados para la motocicleta.

A partir de la función de transferencia y de la ecuación de estados podremos diseñar varios tipos de controladores (PD, LQR, LQI, etc.), como veremos más adelante.

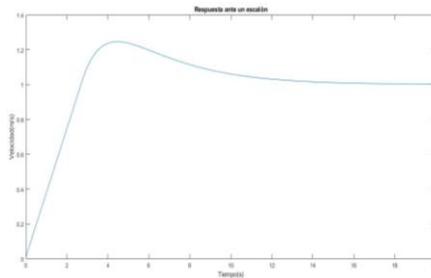
#### 4.2 Controlador PI para el control de la velocidad de avance

Para realizar un buen control de la velocidad de la motocicleta es necesaria una respuesta progresiva y sin oscilaciones, con el fin de evitar grandes desequilibrios en la motocicleta. A partir de los parámetros del motor de avance podemos calcular su función de transferencia y diseñar el controlador Proporcional-Integral discreto correspondiente. El ajuste de las constantes proporcional e integral se ha realizado automáticamente mediante la opción “Auto Tune” del módulo PID de *Simulink*. La figura 7 muestra el lazo del control de velocidad.



**Figura 7.** Diagrama de bloques para el control de velocidad

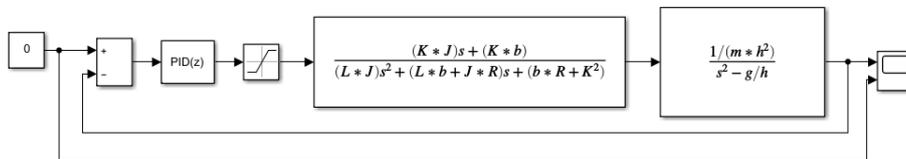
A continuación, el controlador PI de velocidad puede ser testeado. El resultado de introducir una entrada de tipo escalón se presenta en la figura 8. Los parámetros del controlador utilizados son:  $K_p=35.24$  y  $K_i=3$ .



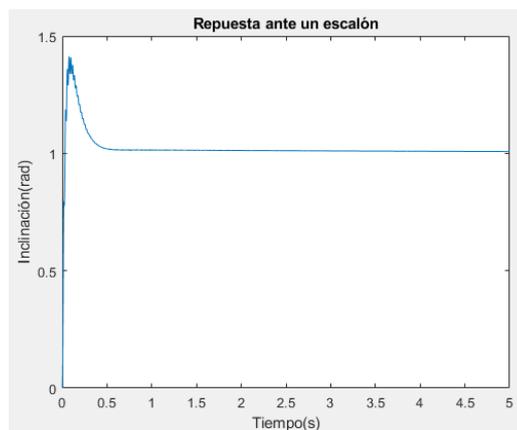
**Figura 8.** Respuesta del controlador PI de velocidad

### 4.3 Controlador PID de inclinación.

A continuación se diseñó un controlador PID de la inclinación de la motocicleta. Para esto, se ha empleado el diagrama de bloques que se presenta en la figura 9. Se ha tenido en cuenta la función de transferencia del motor de la rueda de reacción, junto con la función de transferencia que relaciona el par con la inclinación.



**Figura 9.** Control de inclinación en bucle cerrado



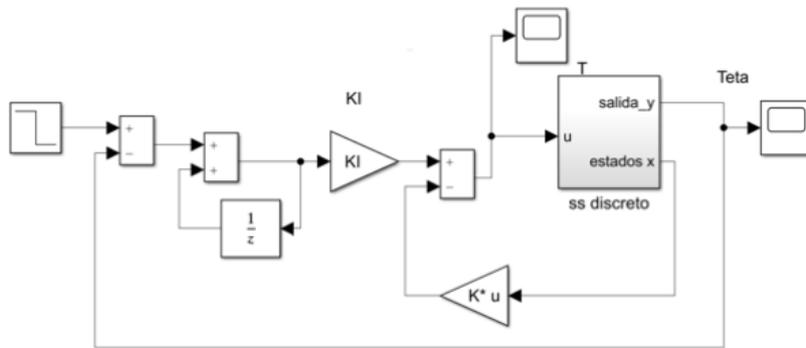
**Figura 10.** Respuesta ante un escalón unitario.

Para este controlador, el valor de las constantes del controlador son  $K_p=307.80$ ,  $K_i=1148.85$ ;  $K_d=19.164$ ;  $N=191.52$ .

Un video de la simulación puede verse en [9].

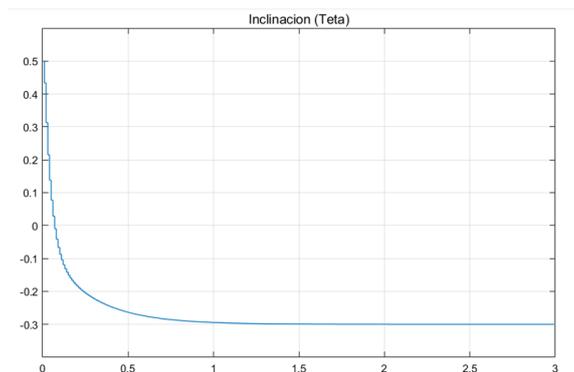
#### 4.4 Controlador en el espacio de estados para la inclinación.

Además del controlador PID de la inclinación se diseñó un controlador LQR en el espacio de estados, a partir de la Ecuación 1 que obtuvimos en el apartado 4.1. La respuesta de este controlador mantiene un error permanente ante un cambio de consigna diferente a la verticalidad. Este problema se ha corregido utilizando la configuración LQI, que incorpora un integrador discreto, como se puede apreciar en la figura 11:



**Figura 11.** Control LQI de la inclinación de la motocicleta

Los resultados obtenidos sobre el modelo Simscape de la motocicleta, ante un cambio de la consigna de inclinación desde 0.5 radianes hasta -0.3 radianes, se pueden ver en la Figura 12.



**Figura 12.** Respuesta del controlador LQI

Un vídeo de la simulación con la motocicleta virtual en movimiento puede verse en [10].

## **5. Control en tiempo real de la motocicleta física.**

Para finalizar, los controladores diseñados se han adaptado en *Simulink* para poder ejecutarse en tiempo real sobre la tarjeta *Arduino MKR1000*. Ello nos permitirá comprobar o no, si el modelo físico es apropiado para ser usado en el laboratorio de prácticas, con los alumnos.

### **5.1 Consideraciones previas**

Es posible programar la tarjeta MKR1000 directamente desde *Simulink*, usando la librería *Simulink Support Package for Arduino Hardware*, lo que permite no tener que realizar grandes cambios en los controladores diseñados previamente. Sin embargo, al diagrama de bloques detallado en apartados anteriores se le deben añadir los elementos necesarios para poder leer los diferentes sensores de los que dispone la motocicleta y los necesarios para manejar los actuadores.

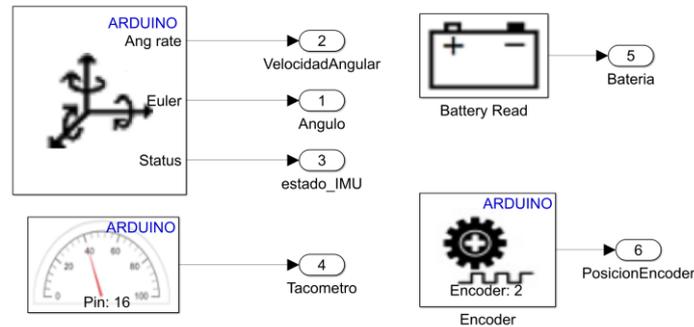
Otra de las ventajas de la tarjeta MKR1000 es que dispone de comunicación WiFi, lo que nos permite visualizar los datos obtenidos por los sensores de la motocicleta, y cambiar las consignas en tiempo real, sin necesidad de usar cable para conexión USB. Así podemos supervisar el control y modificar la velocidad y la trayectoria de la motocicleta a distancia.

### **5.2 Lectura de sensores.**

Los sensores empleados en la motocicleta son el tacómetro, que permite medir la velocidad angular de la rueda de reacción, el encoder, que se encarga de determinar la velocidad de giro del motor de avance, y la IMU, que es la unidad encargada de indicar la inclinación de la motocicleta. Todos estos elementos (ver Fig.13) se suministran en el *Arduino Engineering Kit* para ser incorporados en la librerías de *Simulink*.

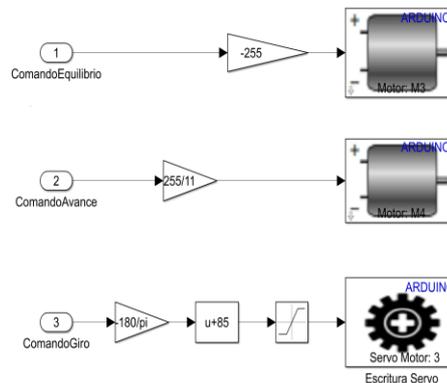
### **5.3 Actuadores.**

Para que el modelo funcione correctamente, es importante también realizar una correcta escritura de las variables calculadas por los controladores. En el caso de los motores de avance y de la rueda de reacción se trata de una señal PWM y en el caso del servomotor de dirección se requiere indicar el ángulo de giro, acotado según las restricciones del servo.



**Figura 13.** Módulos *Simulink* para la lectura de sensores

El bloque para los motores DC que incluye la librería “*Simulink Support for Arduino MKR Motor Carrier*” acepta valores que van entre 255 y -255, siendo estos los máximos valores de giro en un sentido y en el contrario. En el caso del servomotor de dirección, su control se realiza introduciendo directamente el ángulo que se desea girar, teniendo un rango comprendido entre 0 y 180 grados.



**Figura 13.** Módulos *Simulink* para los actuadores

#### 5.4 Control de la motocicleta física.

Los controladores diseñados en las secciones anteriores han sido trasladados a la motocicleta física sin cambiar sus parámetros. En la figura 14 se puede ver el esquema general de control de la motocicleta en tiempo real, realizado mediante *Simulink*.

No se ha realizado el cambio ya que, como se comprobó en el modelo virtual, su comportamiento era el deseado. Sin embargo, al realizar el control en la motocicleta física los resultados cambian eventualmente, y al cabo de unos segundos el sistema se

desestabiliza. Ello se debe a varias razones, relacionadas con el diseño de la motocicleta física, que exponemos a continuación.

La primera razón se debe al comportamiento del motor de la rueda de reacción. Cuando a este motor se le exigen cambios en la dirección de giro (funcionamiento fundamental para mantener la verticalidad) se produce un sobrecalentamiento excesivo del mismo. Esto provoca que sea necesario detener el sistema antes de que el motor pueda dañarse. Además, la medida de la velocidad angular de la rueda de reacción tiene muy poca precisión, ya que incorpora un tacómetro excesivamente simple. Sería conveniente incluir un motor con encoder incorporado.

Otro motivo que influye drásticamente en el control es el continuo cambio del punto de equilibrio del sistema. Es decir, al realizar diferentes pruebas, el punto de equilibrio varía ya que, fundamentalmente, la posición de la batería no es constante, y la estructura de plástico no es suficientemente rígida. Por todo ello es necesario realizar una recalibración de la IMU cada vez que se inicia de nuevo el proceso de control.

Por último, la IMU que el sistema incorpora en el kit no cuenta con las prestaciones óptimas necesarias para obtener una medida fiable de la inclinación. Esta IMU realiza el filtrado de la señal internamente con un procesador poco potente, lo que limita la frecuencia de muestreo, haciendo así que las medidas de la inclinación no sean precisas. En un próximo diseño, para medir la inclinación incorporaremos algoritmos de fusión de datos del acelerómetro y giróscopo que utilicen filtros de Kalman, con más potencia de cálculo, y en los que las matrices de covarianza  $Q$ ,  $R$  puedan ajustarse.

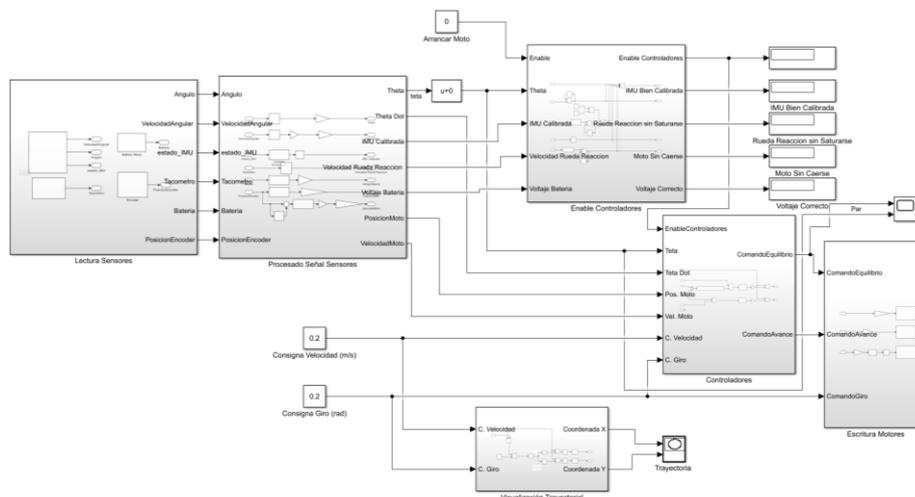


Figura 14. Diagrama Simulink del control en tiempo real de la motocicleta

## 6 Conclusiones

Se ha obtenido el modelo matemático y el modelado gráfico dinámico de una mini-motocicleta autónoma, incluyendo las partes mecánicas, sensores, actuadores y el contacto de las ruedas con el suelo. Esta es una metodología que permite diseñar controladores para sistemas mecatrónicos y visualizar los resultados antes de su implementación física. Este modelo virtual está preparado para poder usarse en el laboratorio de prácticas de mecatrónica, si bien requiere ejecutarse en un computador con cierta potencia de cálculo. Consideramos por otro lado, que aunque es una iniciativa muy interesante, esta primera versión de la motocicleta incluida en el *Arduino Engineering kit* no es idónea para usarse por parte de los alumnos, y requiere ser rediseñada, cambiando el motor de la rueda de reacción y la IMU. Sugerimos también incorporar una estructura metálica para aumentar la rigidez del sistema.

## Referencias

- [1] Adam Owczarkowski, P. K. (2015). Mathematical Modeling of the Bicycle Robot with the Reaction Wheel. *Journal of Automation, Mobile Robotics & Intelligent Systems*, 9(3), 3-8.
- [2] Aström, K. J., Klein, R. E., & Lennartsson, A. (2005). Bicycle Dynamics and Control. *Control Systems Magazine*, 25(4), 26-47.
- [3] Mavros, G. (2010). Enhanced motorcycle roll stability by use of reaction wheel actuator. . *Proceedings of the 10th International Symposium on Advanced Vehicle Control*, 532-537.
- [4] Lychek Keo, K. Y. (2011). Experimental Results for Stabilizing of a Bicycle with a Flywheel Balancer. *IEEE International Conference on Robotics and Automation*, 6150-6155.
- [5] Lei Guo, Q. L. (2009). Design of Linear Quadratic Optimal Controller for Bicycle Robot. *Proceedings of the IEEE International Conference on Automation and Logistics*, 1968-1972.
- [6] Kanjanawanishkul, K. (2015). LQR and MPC controller design and comparison for a stationary self-balancing bicycle robot with a reaction wheel. *Kybernetika*, 51(1), 173-191.
- [7] Bravo M., D. A., Rengifo R., C. F., & Díaz O., J. F. Control of a Robotic Bicycle. (2018).
- [8] C. Tapken, P. Glosekoetter, A.Olivares, G.Olivares “Modelling and Controlling the Kinetic and Dynamic of a Bicycle”. *Enseñanza y Aprendizaje de Ingeniería de Computadores* (2018). ISSN 2173-8688.
- [9] Saavedra Trujillo, J. C. (5 de Junio de 2019). Video de simulación PID. Obtenido de <https://www.youtube.com/watch?v=ecCUvxZ6i-I>
- [10] Saavedra Trujillo, J. C. (6 de Junio de 2019). Video de simulación LQR. Obtenido de <https://www.youtube.com/watch?v=WaIshBh2qtc>