

## Manipulación de 7 servomotores con FPGA iCEstick Evaluation Kit

### Handling 7 servo motors with FPGA iCEstick Evaluation Kit

LÓPEZ-TOLEDO, Eliut\*†

ID 1<sup>er</sup> Autor: *Eliut, López-Toledo*

DOI: 10.35429/JCA.2020.13.4.17.23

Recibido Enero 18, 2020; Aceptado Marzo 28, 2020

#### Resumen

Objetivo. Configurar una tarjeta FPGA para la manipulación de 7 servomotores del tipo Futaba s3003 debido a que son estos servomotores son muy utilizados en la enseñanza del control y la robótica. Esta configuración se realiza mediante la creación de diversos módulos que permiten la integración del mismo en diversos procesos de control. Contribución. La interface, en la que se elaboran los bloques paramétricos y no paramétricos que en su conjunto integran el funcionamiento de los 7 servomotores, se realizó en Icestudio que es una plataforma de libre acceso. Una de tantas aplicaciones de este módulo final puede ser aplicado a un robot de 6 grados de libertad con efector final cuyos servomotores sean del tipo Futaba s3003.

**FPGA, Icestudio, Servomotor, iCEstick Evaluation Kit**

#### Abstract

Objective. Configure a FPGA card for the manipulation of 7 servomotors of the Futaba s3003 type because these servomotors are widely used in the teaching of control and robotics. This configuration is done through the creation of various modules that allow the integration of it in various control processes. Contribution. The interface, in which the parametric and non-parametric blocks together integrate the operation of the 7 servomotors are developed, are executed in Icestudio which is a free access platform. One of these applications of this final module can be applied to a 6 degree freedom robot with final effector whose servomotors area of the Futaba s3003 type.

**FPGA, Icestudio, Servomotor, iCEstick Evaluation Kit**

**Citación:** LÓPEZ-TOLEDO, Eliut. Manipulación de 7 servomotores con FPGA iCEstick Evaluation Kit. Revista de Cómputo Aplicado. 2020. 4-13: 17-23.

\* Correspondencia al Autor (Correo Electrónico: eliutl@hotmail.com)

† Investigador contribuyendo como primer autor.

## Introducción

El uso de los servomotores en procesos productivos y educativos es muy común y son controlados en la mayoría de las ocasiones por dispositivos como microcontroladores o PIC's o bien; mediante la tecnología Arduino en sus diversas presentaciones. En esta ocasión se hace uso de una tarjeta FPGA (Field Programmable Gate Array) del tipo ICEstick Evaluation Kit cuyo fabricante es Lattice y el uso del software libre para la configuración de la FPGA llamado Icestudio. Icestudio es una plataforma de acceso libre y cuenta con un entorno amigable para desarrollar bloques con funciones específicas a partir de compuertas básicas, multiplexores y flip-flops; también permite la programación en Verilog. Haciendo uso de estos elementos, se procede a construir una serie de bloques que en su conjunto forman el bloque final paramétrico para manipular a los siete servomotores del tipo Futaba s3003. La configuración propuesta, se puede modificar para manipular tantos servomotores, como terminales lo permitan según las E/S de la tarjeta (ver tabla 1 y tabla 2).

Diligent Pmod Compatible Connector Description					
Connection	Left pins	Row	Right pins	Row	Connection
PIO1_02		1	7		PIO1_06
PIO1_03		2	8		PIO1_07
PIO1_04		3	9		PIO1_08
PIO1_05		4	10		PIO1_09
Ground		5	11		Ground
3.3v		6	12		3.3v

**Tabla 1** Diligent Pmod

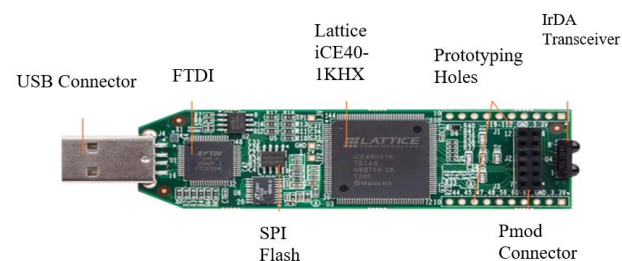
Expansion I/O Connections						
J1 Connector			J3 Connector			
Pin	CPLD Bank 0	I/O	CPLD Pin	Pin	CPLD Bank 2	I/O
1	3.3v		-	1	3.3v	-
2	Ground		-	2	Ground	-
3	PIO0_02		112	3	PIO2_17	62
4	PIO0_03		113	4	PIO2_16	61
5	PIO0_04		114	5	PIO2_15	60
6	PIO0_05		115	6	PIO2_14	56
7	PIO0_06		116	7	PIO2_13	48
8	PIO0_07		117	8	PIO2_12	47
9	PIO0_08		118	9	PIO2_11	45
10	PIO0_09		119	10	PIO2_10	44

**Tabla 2** Expansion I/O connections.

## Metodología y desarrollo

Algunas especificaciones técnicas de la tarjeta FPGA iCEstick Evaluation Kit (ver Fig.1) que se pueden visualizar en la dirección web del fabricante Lattice, se muestran a continuación:

- Procesador iCE40HX-1k.
- Conector Digilent Pmod de 2 x 6 posiciones para otras conexiones periféricas.
- El dispositivo USB FTDI 2232H permite la programación del dispositivo iCE y la interfaz UART a una PC.
- 5 LED de usuario.
- Oscilador Discera MEMS de 1MHz.
- 16 conexiones de E/S digitales LVC MOS/LVTTL (3.3V) en conexiones de orificio pasante de 0.1”.
- Fuente de alimentación por USB.



**Figura 1** Tarjeta FPGA iCEstick Evaluation Kit

Icestudio es de código abierto y permite trabajar con hardware abierto como en este caso; es una interfaz que permite diseñar el hardware real uniendo bloques con líneas y acepta código elaborado en verilog3 para la construcción de los bloques que pueden ser paramétricos o no paramétricos. Se optó por esta interface debido a la existencia de una comunidad que involucra tanto hardware abierto como código abierto y permite fortalecer este entorno de desarrollo.

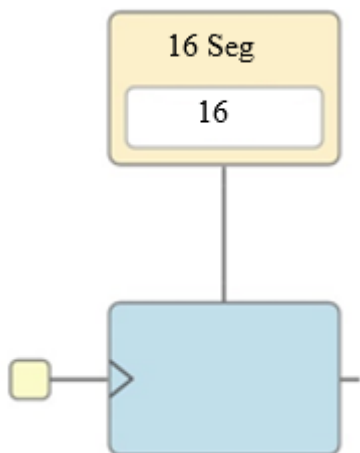


Figura 2 Bloque paramétrico de tiempo

Debido a que los componentes que proporciona inicialmente la plataforma Icestudio son: Bit, multiplexor 2-1, compuertas lógicas, biestables T y D, antirrebote y prescaler; es necesario la creación de algunos bloques para este trabajo.

Los bloques utilizados para este fin son denominados: Paramétrico de tiempo (Fig.2), Multiplexor de 4 a 1 de 8 bits (Fig.3) que cuenta con las 4 entradas habilitadas y un bus de 8 canales cada una de las entradas, módulo integrador de 8 a 1 (Fig.4); un módulo de 8 salidas para los servos que en esta ocasión son 7 únicamente los utilizados (Fig.5), un multiplexor de 8 a 1, de 8 bits (Fig.6), la configuración de cada servomotor (Fig.7) y un módulo de constantes (Fig. 8).

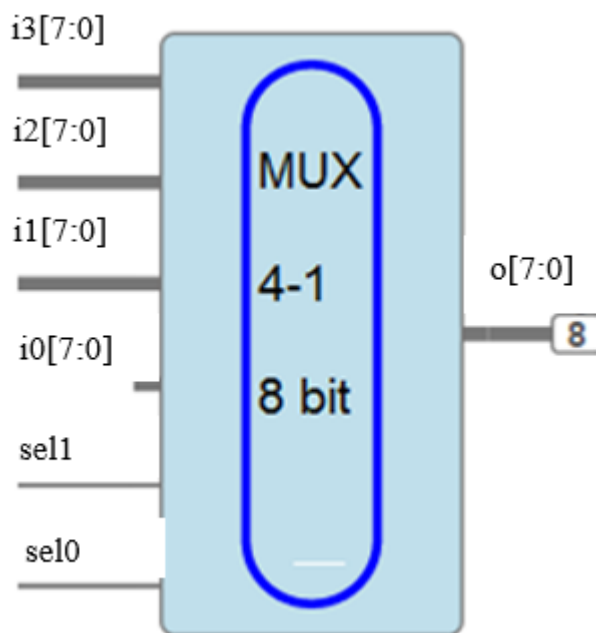


Figura 3 Multiplexor de 4 a 1 de 8 bits

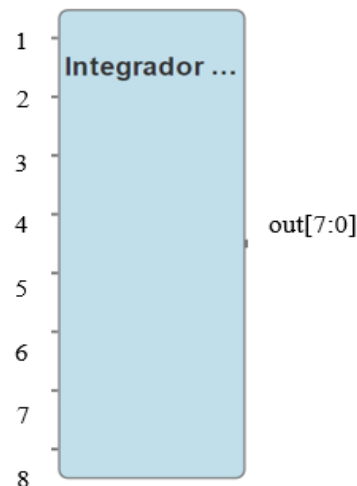


Figura 4 Integrador de 8 a 1



Figura 5 Módulo de 8 salidas

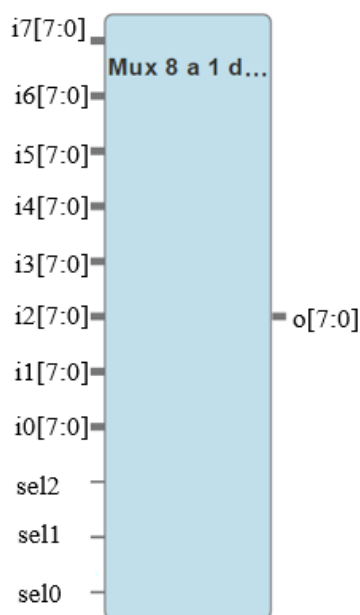


Figura 6 Multiplexor de 8 a 1, de 8 bits

El código para el Multiplexor de 8 a 1, de 8 bits es:

```
// Multiplexor de 8 a 1 de 8 //bits
wire [2:1:0] sel = {s2, s1, s0};
// assign o= (sel == 2'b000) ? i0:
(sel == 2'b001) ? i1:
(sel == 2'b010) ? i2:
(sel == 2'b011) ? i3:
(sel == 2'b100) ? i4:
(sel == 2'b101) ? i5:
(sel == 2'b110) ? i6:
(sel == 2'b111) ? i7: i8;
```

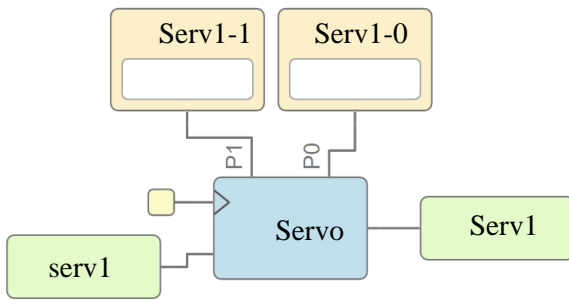


Figura 7 Configuración de Servomotor

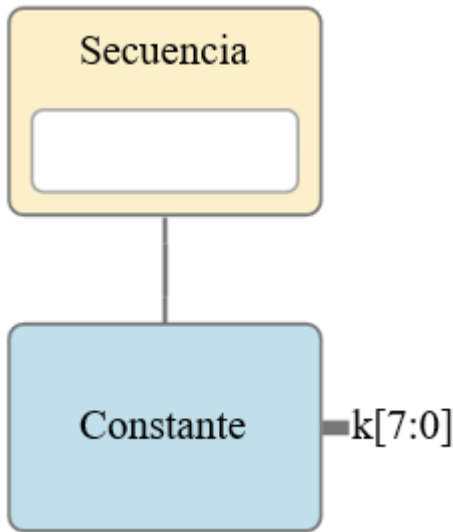


Figura 8 Constante

Considerando los módulos indicados en esta sección, se procede a integrar los para finalmente tener el sistema de avance/retroceso para el grupo de los 7 servomotores. Esta sección se muestra en tres partes (Fig. 9, Fig. 10 y Fig. 11); dónde a la terminarl i7 se le asigna un aconstante de 126 llamada secuencia, a i6 127, a i5 113, a i4 1, a i3 113, a i2 127, a i1 126 y a la entrada i0 se le asigna una constante 0; todas estas entradas se asignan como se muestra en la entrada i7 de la figura 9.

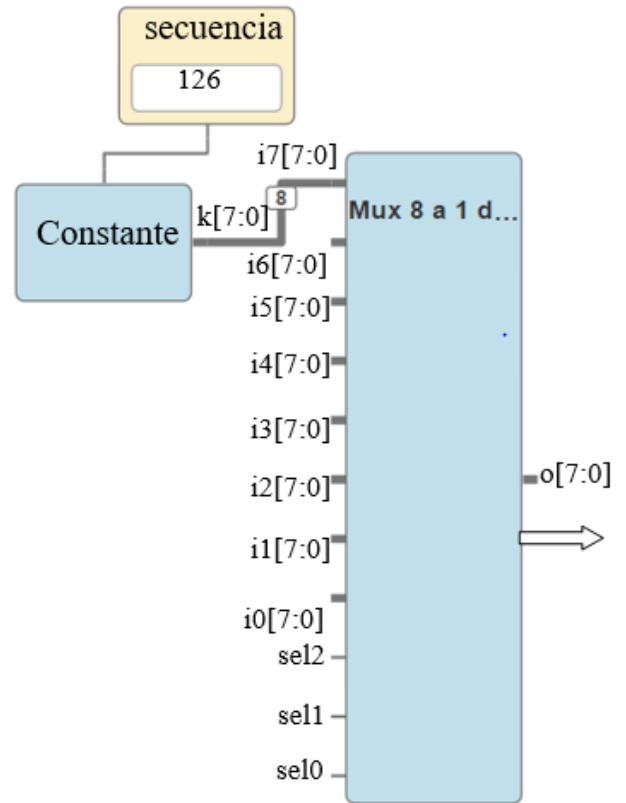


Figura 9 Integración Avance/Retroceso del sistema de 7 servomotores primer parte

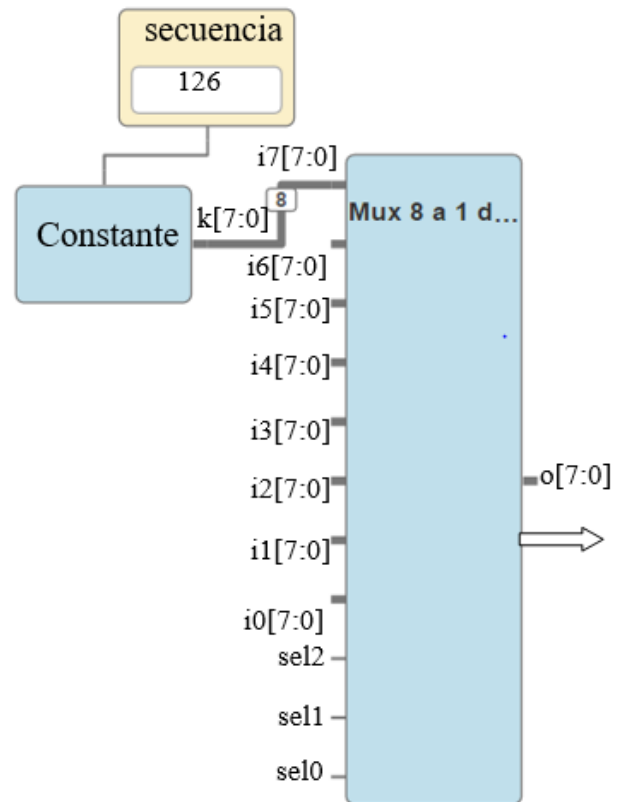


Figura 10 Avance/Retroceso del sistema de 7 servomotores segunda parte

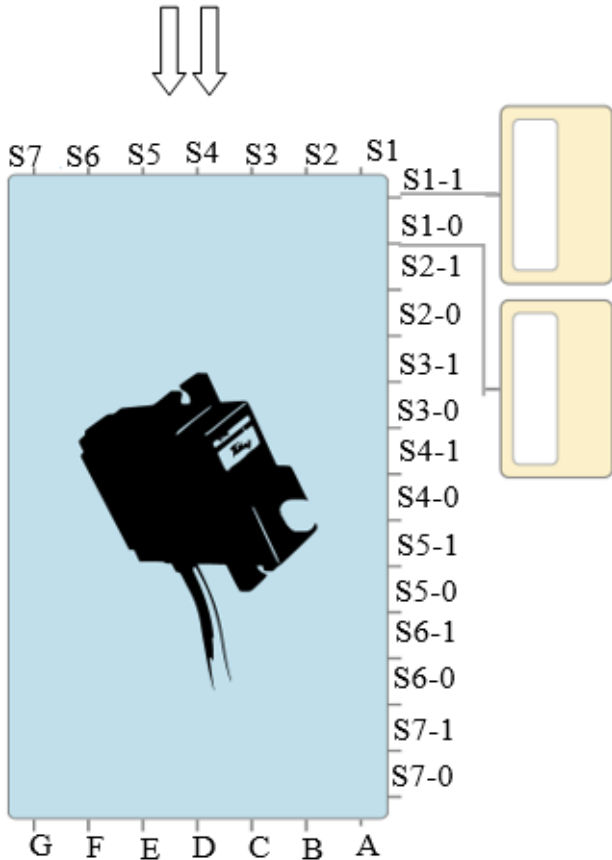


Figura 11 Avance/Retroceso del sistema de 7 servomotores tercera parte

Apartir de la figura 11, la terminal A se conecta al Servo1, la terminal B al Servo2 y así sucesivamente hasta la terminal G, que se conecta al Servo7; los pines utilizados en la FPGA son: PMOD1, PMOD2, PMOD3, PMOD4, PMOD7, PMOD8 y GPIO2\_10. Del mismo modo para las terminales S1-1 y S1-0 se establecen los grados de movimiento cuando recibe el 1 lógico y el 0 lógico respectivamente para el Servo1; es decir se tiene un par de parametros para establecer en cada Servo los grados a moverse y con el 1 lógico y con el 0 lógico; lo cual se indica con las terminales que van desde la S1-1 hasta la S7-0.

La tabla 3 es construida para un robot de 6 grados de libertad con efector final, donde todos los actuadores son servomotores. Realiza 8 acciones o estados que se mencionan en esta tabla 3. Las variables H, I y J, corresponden a los tres primeros grados de libertad para el posicionamiento, las variables K, L y M corresponden a la orientación y la variable N es considerada para el efector final.

m	Valor	H	I	J	K	L	M	N	Acción
0	126	1	1	1	1	1	1	0	Avanzar (tomar pieza)
1	127	1	1	1	1	1	1	1	Toma la pieza
2	113	1	1	1	0	0	0	1	Regresa orientación a inicial
3	1	0	0	0	0	0	0	1	Regresa posicionamiento a inicial
4	113	1	1	1	0	0	0	1	Avanza Orientación para colocar pieza en punto B.
5	127	1	1	1	1	1	1	1	Avanza posicionamiento para colocar pieza en punto B.
6	120	1	1	1	1	1	1	0	Suelta la pieza en punto B.
7	0	0	0	0	0	0	0	0	Regresa a inicio.

Tabla 3 Secuencia de datos para los 7 servomotores.

Debido a que son ocho estados (m) los que se tienen para enviar los datos a los 7 servos (H, I, J, K, L M y N), estos estados se generan por medio de tres frecuencias diferentes que son: 1 Hz, 1/2 Hz, y 1/4 Hz, como se muestra en la figura 12. Es decir, el primer estado es m0 (0,0,0), el siguiente es m1 (1,0,0) y así sucesivamente hasta llegar al estado m7(1,1,1) como se muestra en la tabla 4.

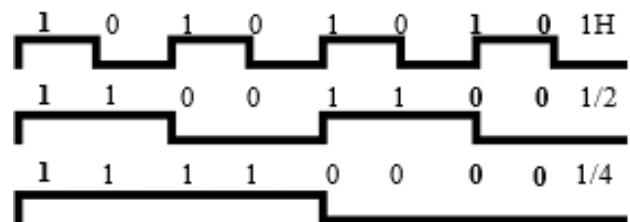
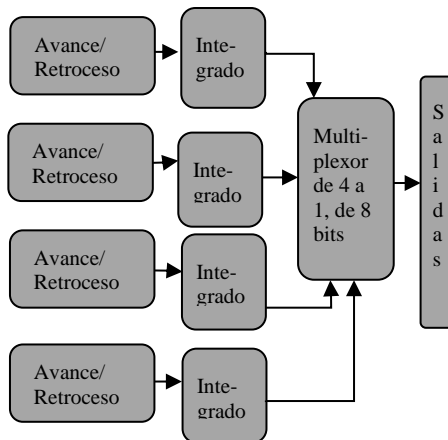


Figura 12 Generación de 8 estados

m	1/4 Hz	1/2 Hz	1 Hz
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Tabla 4 Estados derivados de las tres frecuencias

Cabe mencionar que los elementos que se encuentran en las figuras 9, 10 y 11 se integran a un módulo llamado Módulo integrador Avance/Retroceso, el cual puede ser utilizado para posteriores acciones; es decir, si se requiere que el robot o el sistema al que se le desee aplicar dicho módulo realice 4 tareas diferentes, se pueden añadir cuatro de estos módulos, como se muestra en la figura 13. Por lo tanto, se logra la arquitectura hardware que permite mover al robot o sistema, en cuatro tareas diferentes utilizando ocho secuencias cada módulo integrador.



**Figura 13** Sistema integral para manipulación de 7 servomotores, en 4 tareas diferentes.

## Conclusiones

La manipulación de los servomotores utilizando las FPGA's entre otras ventajas, ofrece la ejecución de las sentencias en forma paralela reduciendo de esta manera el tiempo de procesamiento; otro de los factores importantes es el uso de software libre que permite el desarrollo de este tipo de programación. La aplicación en este caso es a un robot de 6 grados de libertad y un efector final, sin embargo se integró un último bloque que nos permite poder aplicar varias tareas al mismo robot; aunque como se mencionó anteriormente no está limitado al uso del robot, puede ser aplicado a cualquier sistema que haga uso de 7 servomotores. Sin embargo, es importante considerar el tiempo de ejecución de movimientos de los servomotores cuando estos se ponen al estado 1 y cuando se ponen al estado 0; para establecer estos tiempos solo se requiere escribirlos en el módulo paramétrico Avance/Retroceso. Finalmente, debido a las especificaciones técnicas de la tarjeta utilizada iCEstick Evaluation Kit, se utilizan como salida: PMOD1, PMOD2, PMOD3, PMOD4, PMOD5, PMOD6 y GPIO2\_10.

## Referencias

Ted, H., & Cynthia, I. (2010). Handbook of FPGA Design Security. U.S.A: Springer.

Sánchez, E., (2004). Fundamentos y Electrónica de las Comunicaciones. España: PUV.

Cayssials. (2014). Sistemas embebidos en FPGA. Argentina: Marcombo.

Simpson., & Philip, A. (2015). FPGA Design. Inglaterra: Springer.

Ríos, F. (2017). Servidor Web empotrado en un FPGA para configurar un Controlador Maestro del Sistema Inteligente de Tráfico Cubano. Revista Cubana de Ciencias Informáticas, 11, 16 - 28.

Garcés, L., Romero, D., Cabrera & A.(2017). Model-based Implementation of self-configurable intellectual property modules for image histogram calculation in FPGAs . bdigital PORTAL DE REVISTAS UN., 37, 74-81.

Criado, D., Cabot, J. & Escartín, V. (2009). Diseño de módulos I2C en FPGA de la familia Altera. Revista Ingeniería Electrónica, Automática y Comunicaciones., 30, 8-14.

Suárez, A., Matos, D., Rivero, R. & López, R.(2009). Interfaz programable de comunicación serie, asíncrona en FPGA. Revista Ingeniería Electrónica, Automática y Comunicaciones, 30, 22-29.

Sosa, J., García, V., Salinas, E., Ortega, R., & Hernández, R.(2018). Procesador de números complejos enteros de alta velocidad implementada en un FPGA. Revista Ingeniería Investigación y Tecnología., 19, 77-88.

Abarca, G.(2018). Diseño digital con aplicaciones. México: Grupo editorial Patria.

Gonzalez, J.(2018). Visual editor for open FPGA boards. Octubre 2019, de GitHub Sitio web: <https://github.com/FPGAwards/icestudio/>

iCEstick Evaluation Kit. Recuperado el 10 de Abril de 2020 de [www.latticesemi.com/icestick](http://www.latticesemi.com/icestick)

León, R., Alvarado, E., Arevalo, K., Maldonado, A., & Polonio, A. (2020). Detección y extracción de muestras falladas usando visión artificial y un brazo robótico. *Revista CIENCIA Y TECNOLOGÍA*, 16(1), 21-32.

Herrera Jiménez, A., Miranda Castillo, J. R., & Martínez Villa, F. (2020). Automatización del proceso de maceración y germinación de la cebada para la elaboración de malta utilizada en la cerveza artesanal.

Velasco-Sánchez, E., Zapata-Impata, B. S., Gil, P., & Torres, F. (2020). Clasificación de objetos usando percepción bimodal de palpación única en acciones de agarre robótico.

Velasco, E., Zapata-Impata, B. S., Gil, P., & Torres, F. (2020). Clasificación de objetos usando percepción bimodal de palpación única en acciones de agarre robótico. *Revista Iberoamericana de Automática e Informática Industrial*, 17(1), 44-55.